UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
16/997,471	**11/22/2024 08:29:23 PM Z ET**	**2020058 / RMS-02C1C1C14**

Title of Invention

NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	11544183
CONFIRMATION #	1070	FILED BY	Marc Schuyler
PATENT CENTER #	68096708	FILING DATE	08/19/2020
CUSTOMER #	73091	FIRST NAMED INVENTOR	Andrey V. Kuzmin
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	-

Documents

TOTAL DOCUMENTS: 1

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
sb0123_fill-204-Z-US16997471.pdf	2	Change of Address	68 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
sb0123_fill-204-Z-US16997471.pdf	3CD0BA8A41D083AD5B244EF2A5C93F6C9C788EACB343EF5AF7A21E3563A11DDE6D90C6C34BAB3858ED7C88F33981029A18DBE076C2F6322D2BC60329B178FE19

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/123 (04-15)
Approved for use through 03/31/2021. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

CHANGE OF CORRESPONDENCE ADDRESS
Patent

Address to:
Mail Stop Post Issue
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Patent Number	11544183
Issue Date	01/03 /2023
Application Number	16/997471
Filing Date	08/19/2020
First Named Inventor	Andrey V. KUZMIN
Attorney Docket Number	RMS-02C1C1C14 / 204Z

Please change the Correspondence Address for the above-identified patent to:

☐ The address associated with Customer Number: ____

OR

☒ **Firm *or* Individual Name** Radian Memory Systems, LLC

6275 West Plano Parkway, Suite 500

Address

City Plano | **State** Texas | **ZIP** 75093

Country

Telephone | **Email**

This form cannot be used to change the data associated with a Customer Number. To change the data associated with an existing Customer Number use "Request for Customer Number Data Change" (PTO/SB/124).

This form will not affect any "fee address" provided for the above-identified patent. To change a "fee address" use the "Fee Address Indication Form" (PTO/SB/47).

I am the:

☐ Patentee.

☐ Assignee of record of the entire interest. See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96).

☒ Attorney or agent of record. Registration Number 35675.

Signature /Marc P. Schuyler/

Typed or Printed Name Marc P. Schuyler

Date 11/22/2024 | Telephone

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☐ *Total of ____ forms are submitted.

This collection of information is required by 37 CFR 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Mail Stop Post Issue, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

ACKNOWLEDGEMENT OF LOSS OF ENTITLEMENT TO ENTITY STATUS DISCOUNT

APPLICATION #	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET #	REQUEST ID
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	215709

The entity status change request below filed through Patent Center on 05/16/2024 has been accepted.

Certifications

APPLICANT CHANGING TO REGULAR UNDISCOUNTED FEE STATUS

Signature

I certify, in accordance with 37 CFR 1.4(d)(4), that I am one of the signatories making the entity status change.

Signature	Name	Registration #
/Marc P. Schuyler/	Marc Schuyler	35675

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	01/03/2023	11544183	2020058 / RMS-02C1C1C14	1070

73091 7590 12/14/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 0 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Application Assistance Unit (AAU) of the Office of Patents Stakeholder Experience (OPSE), Stakeholder Support Division (SSD) at (571)-272-4200.

INVENTOR(s) (Please see PAIR WEB site http://pair.uspto.gov for additional inventors):

Andrey V. Kuzmin, Moscow, RUSSIAN FEDERATION;
Mike Jadon, Manhattan Beach, CA;
Richard M. Mathews, Chatsworth, CA;

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Radian Memory Systems, Inc., Calabasas, CA;

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The USA offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to encourage and facilitate business investment. To learn more about why the USA is the best country in the world to develop technology, manufacture products, and grow your business, visit SelectUSA.gov.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	37	20050144413	A1	2005-06-30	Kuo et al.	
	38	20070283428	A1	2007-12-06	Ma et al.	
	39	20090172257	A1	2009-07-02	Prins et al.	
	40	20100287217	A1	2010-11-11	Borchers et al.	
Change(s) applied to document, /P.P./ 9/15/2022	41	20120059972	A1	~~2013-01-10~~ 03/2012	Chen	
	42	20130013852	A1	2013-01-10	Hou et al.	
	43	20130297852	A1	2013-11-07	Fai et al.	
	44	20140101371	A1	2014-04-10	Nguyen et al.	
	45	20150113203	A1	2015-04-23	Dancho et al.	
	46	20150261456	A1	2015-09-17	Alcantara et al.	
	47	20140122781	A1	2014-05-01	Smith et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

PART B – FEE(S) TRANSMITTAL

Complete and send this form, together with the applicable fee(s), by mail or fax, or via EFS-Web.

By mail, send to: Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

By fax, send to: (571) 273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being transmitted to the USPTO via EFS-Web or by facsimile to (571) 273-2885, on the date below.

(Typed or printed name)
(Signature)
(Date)

APPLICATION NO	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO	CONFIRMATION NO
16997471	2020-08-19	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

TITLE OF INVENTION:

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	600	0	0	600	12/09/2022

EXAMINER	ART UNIT	CLASS-SUBCLASS

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363)

☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-09 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) The names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) The name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1. Marc P. Schuyler
2.
3.

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document must have been previously recorded, or filed for recordation, as set forth in 37 CFR 3.11 and 37 CFR 3.81(a). Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE: Radian Memory Systems, Inc.

(B) RESIDENCE: (CITY and STATE or COUNTRY): Manhattan Beach, California (USA)

Please check the appropriate assignee category or categories (will not be printed on the patent): ☐ Individual ☑ Corporation or other private group entity ☐ Government

4a. Fees Submitted: ☑ Issue Fee ☐ Publication Fee (if required) ☐ Advance Order - # of Copies _____

4b. Method of Payment (*Please first reapply any previously paid fee shown above*):

☑ Electronic Payment via EFS-Web ☐ Enclosed check ☐ Non-electronic payment by credit card (Attach form PTO-2038)

☐ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment to Deposit Account No. _____

5. Change of Entity Status (from status indicated above)

☐ Applicant certifying micro entity status. See 37 CFR 1.29. NOTE: Absent a valid Certification of Micro Entity Status (see forms PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.

☐ Applicant asserting small entity status. See 37 CFR 1.27. NOTE: If the application was previously under micro entity status, checking this box will be taken as a notification of loss of entitlement to micro entity status.

☐ Applicant changing to regular undiscounted fee status. NOTE: Checking this box will be taken as a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: This form must be signed in accordance with 37 CFR 1.31 and 1.33. See 37 CFR 1.4 for signature requirements and certifications.

Authorized Signature /Marc P. Schuyler/ Date 9 December 2022

Typed or printed name Marc P. Schuyler Registration No. 35675

OMB Clearance and PRA Burden Statement for PTOL-85 Part B

The Paperwork Reduction Act (PRA) of 1995 requires Federal agencies to obtain Office of Management and Budget approval before requesting most types of information from the public. When OMB approves an agency request to collect information from the public, OMB (i) provides a valid OMB Control Number and expiration date for the agency to display on the instrument that will be used to collect the information and (ii) requires the agency to inform the public about the OMB Control Number's legal significance in accordance with 5 CFR 1320.5(b).

The information collected by PTOL-85 Part B is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450. Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Patent Application Fee Transmittal

Application Number:	16997471
Filing Date:	19-Aug-2020
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Filer:	Marc Philip Schuyler
Attorney Docket Number:	2020058 / RMS-02C1C1C14

Filed as Small Entity

Filing Fees for Utility under 35 USC 111(a)

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
UTILITY APPL ISSUE FEE	2501	1	600	600

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**600**

Electronic Acknowledgement Receipt	
EFS ID:	47129873
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	01-DEC-2022
Filing Date:	19-AUG-2020
Time Stamp:	20:35:19
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	CARD
Payment was successfully received in RAM	$600
RAM confirmation Number	E2022B1K36204289
Deposit Account	
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Issue Fee Payment (PTO-85B)	00-ptol85b-RMS-02C1C1C14.pdf	272335 accce455b15c4c81d347af230e1804ecc37e226a	no	2
Warnings:					
Information:					
2	Fee Worksheet (SB06)	fee-info.pdf	38204 b2dee638455b2ded1e69a0c772fc9a205ac0185e	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	310539		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.
National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.
New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
16/997,471	08/19/2020	2136	785	2020058 / RMS-02C1C1C14	20	2

CONFIRMATION NO. 1070

CORRECTED FILING RECEIPT

73091
Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

OC000000136423801

Date Mailed: 09/28/2022

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Andrey V. Kuzmin, Moscow, RUSSIAN FEDERATION;
Mike Jadon, Manhattan Beach, CA;
Richard M. Mathews, Chatsworth, CA;

Applicant(s)

Radian Memory Systems, Inc., Calabasas, CA;

Power of Attorney: The patent practitioners associated with Customer Number 73091

Domestic Priority data as claimed by applicant

This application is a CON of 15/625,931 06/16/2017 PAT 10838853
which is a CON of 14/466,167 08/22/2014 PAT 9727454
which is a CON of 13/767,723 02/14/2013 PAT 9652376
which claims benefit of 61/757,464 01/28/2013

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

Permission to Access Application via Priority Document Exchange: No

Permission to Access Search Results: No

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 08/25/2020
The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 16/997,471**
Projected Publication Date: Request for Non-Publication Acknowledged
Non-Publication Request: Yes
Early Publication Request: No
**** SMALL ENTITY ****
Title

NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

Preliminary Class

711

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific

countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop

technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

Document Description: Application Data Sheet to update/correct info
Doc Code: ADS.CORR

CORRECTED ADS FORM

Application Number	16997471
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

Inventor Information

****If no data is shown, no data has been corrected****

	Data of Record	Updated Data
Order Number	1	
Name	Andrey V. Kuzmin	
Residence Information		
Residency	non-us-residency	
City	Moscow	
State		
Country of Residence	RU	
Mailing Address of Inventor		
Address 1	19 Avantgarde St. #35	
Address 2		
City,State/Province, Postal Code	Moscow 125493	
Country	RU	

Document Description: Application Data Sheet to update/correct info
Doc Code: ADS.CORR

	Data of Record	Updated Data
Order Number	2	
Name	Mike Jadon	
Residence Information		
Residency	us-residency	
City	Manhattan Beach	
State	CA	
Country of Residence	US	
Mailing Address of Inventor		
Address 1	~~26025 Mureau Rd.~~	1590 Rosecrans D417
Address 2		
City,State/Province, Postal Code	~~Calabasas CA 91302~~	Manhattan Beach CA 90266
Country	US	

	Data of Record	Updated Data
Order Number	3	
Name	Richard M. Mathews	
Residence Information		
Residency	us-residency	
City	~~Porter Ranch~~	Chatsworth
State	CA	
Country of Residence	US	
Mailing Address of Inventor		
Address 1	~~18810 San Fernando Mission Blvd~~	10027 Casaba Ave.
Address 2		
City,State/Province, Postal Code	~~Porter Ranch CA 91326~~	Chatsworth CA 91311
Country	US	

Application Information

	Data of Record	Updated Data
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	
Attorney Docket Number	2020058 / RMS-02C1C1C14	
Entity Type	Small	

Domestic Benefit/National Stage Information

****If no data is shown, no data has been corrected****

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121,365(c), or 386(c) or indicate National Stage entry from a PCT application. Providing this information in the application data sheet constitutes the specific reference required by 35 U.S. C. 119(e) or 120, and 37 CFR 1.78(a).

Corrected ADS 1.0

		Data of Record	Updated Data
	Prior Application Status		
	Application Number		
	Continuity Type		
	Prior Application Number		
	Filing Date (YYYY-MM-DD)		
	Patent Number		
	Issue Date (YYYY-MM-DD)		

Foreign Priority Information

****If no data is shown, no data has been corrected****

This section allows for the applicant to claim priority to a foreign application. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55. When priority is claimed to a foreign application that is eligible for retrieval under the priority document exchange program (PDX) the information will be used by the Office to automatically attempt retrieval pursuant to 37 CFR 1.55(i)(1) and (2). Under the PDX program, applicant bears the ultimate responsibility for ensuring that a copy of the foreign application is received by the Office from the participating foreign intellectual property office, or a certified copy of the foreign priority application is filed, within the time period specified in 37 CFR 1.55(g)(1).

		Data of Record	Updated Data
	Application Number		
	Country		
	Filing Date		
	Access Code		

Applicant Information

****If no data is shown, no data has been corrected****
Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

	Data of Record	Updated Data
Applicant Type		

If applicant is the legal representative, indicate the authority to file the patent application, the inventor is		
Name of the Deceased or Legally Incapacitated Inventor		
Applicant is an Organization		
Name		
Organization Name		
Address 1		
Address 2		
City,State/Province,Postal Code		
Country		
Phone Number		
Fax Number		
Email Address		

Assignee Information including Non-Applicant Assignee Information

****If no data is shown, no data has been corrected****
Providing this information in the application data sheet does not substitute for compliance with any requirement of part 3 of Title 37 of the CFR to have an assignment recorded in the Office

	Data of Record	Updated Data
Order		
Applicant is an Organization		
Name		

Organization Name		
Mailing Address		
Address 1		
Address 2		
City,State/Province,Postal Code		
Country		
Phone Number		
Fax Number		
Email Address		

Signature

NOTE: This Application Data Sheet must be signed in accordance with 37 CFR 1.33(b).

This Application Data Sheet **must** be signed by a patent practitioner if one or more of the applicants is a **juristic entity** (e.g., corporation or association). If the applicant is two or more joint inventors, this form must be signed by a patent practitioner, **all** joint inventors who are the applicant, or one or more joint inventor-applicants who have been given power of attorney (e.g., see USPTO Form PTO/AIA/81) on behalf of **all** joint inventor-applicants.

See 37 CFR 1.4(d) for the manner of making signatures and certifications.

Signature	/Marc P. Schuyler/	Registration Number	35675
First Name	Marc	Last Name	Schuyler

Electronic Acknowledgement Receipt	
EFS ID:	46695951
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	26-SEP-2022
Filing Date:	19-AUG-2020
Time Stamp:	18:22:12
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Application Data Sheet to update/ correct info	CorrectedADS.pdf	128438 dabfb192fdce1a092a0602c326a361bd7008eb3b	no	6

Warnings:

Information:	
Total Files Size (in bytes):	128438

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

73091 7590 09/09/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	
ART UNIT	PAPER NUMBER
2136	

DATE MAILED: 09/09/2022

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

TITLE OF INVENTION: NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$600	$0.00	$0.00	$600	12/09/2022

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Maintenance fees are due in utility patents issuing on applications filed on or after Dec. 12, 1980. It is patentee's responsibility to ensure timely payment of maintenance fees when due. More information is available at www.uspto.gov/PatentMaintenanceFees.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), by mail or fax, or via EFS-Web.

By mail, send to: Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

By fax, send to: (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

73091 7590 09/09/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being transmitted to the USPTO via EFS-Web or by facsimile to (571) 273-2885, on the date below.

(Typed or printed name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

TITLE OF INVENTION: NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$600	$0.00	$0.00	$600	12/09/2022

EXAMINER	ART UNIT	CLASS-SUBCLASS
BIRKHIMER, CHRISTOPHER D	2136	711-103000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

❑ Change of correspondence address (or Change of Correspondence Address form PTO/AIA/122 or PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/AIA/47 or PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) The names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) The name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1_____
2_____
3_____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document must have been previously recorded, or filed for recordation, as set forth in 37 CFR 3.11 and 37 CFR 3.81(a). Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ❑ Individual ❑ Corporation or other private group entity ❑ Government

4a. Fees submitted: ❑Issue Fee ❑Publication Fee (if required) ❑Advance Order - # of Copies _____

4b. Method of Payment: *(Please first reapply any previously paid fee shown above)*

❑ Electronic Payment via EFS-Web ❑ Enclosed check ❑ Non-electronic payment by credit card (Attach form PTO-2038)

❑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment to Deposit Account No. _____

5. **Change in Entity Status** (from status indicated above)

❑ Applicant certifying micro entity status. See 37 CFR 1.29

❑ Applicant asserting small entity status. See 37 CFR 1.27

❑ Applicant changing to regular undiscounted fee status.

NOTE: Absent a valid certification of Micro Entity Status (see forms PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.
NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.
NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: This form must be signed in accordance with 37 CFR 1.31 and 1.33. See 37 CFR 1.4 for signature requirements and certifications.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

73091 7590 09/09/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	
ART UNIT	**PAPER NUMBER**
2136	

DATE MAILED: 09/09/2022

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(Applications filed on or after May 29, 2000)

The Office has discontinued providing a Patent Term Adjustment (PTA) calculation with the Notice of Allowance.

Section 1(h)(2) of the AIA Technical Corrections Act amended 35 U.S.C. 154(b)(3)(B)(i) to eliminate the requirement that the Office provide a patent term adjustment determination with the notice of allowance. See Revisions to Patent Term Adjustment, 78 Fed. Reg. 19416, 19417 (Apr. 1, 2013). Therefore, the Office is no longer providing an initial patent term adjustment determination with the notice of allowance. The Office will continue to provide a patent term adjustment determination with the Issue Notification Letter that is mailed to applicant approximately three weeks prior to the issue date of the patent, and will include the patent term adjustment on the patent. Any request for reconsideration of the patent term adjustment determination (or reinstatement of patent term adjustment) should follow the process outlined in 37 CFR 1.705.

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

PTOL-85 (Rev. 02/11)

OMB Clearance and PRA Burden Statement for PTOL-85 Part B

The Paperwork Reduction Act (PRA) of 1995 requires Federal agencies to obtain Office of Management and Budget approval before requesting most types of information from the public. When OMB approves an agency request to collect information from the public, OMB (i) provides a valid OMB Control Number and expiration date for the agency to display on the instrument that will be used to collect the information and (ii) requires the agency to inform the public about the OMB Control Number's legal significance in accordance with 5 CFR 1320.5(b).

The information collected by PTOL-85 Part B is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450. Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Notice of Allowability	Application No. 16/997,471	Applicant(s) Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	AIA (FITF) Status No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☑ This communication is responsive to 08/22/2022.
   ☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2. ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

3. ☑ The allowed claim(s) is/are See Continuation Sheet . As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see **http://www.uspto.gov/patents/init_events/pph/index.jsp** or send an inquiry to **PPHfeedback@uspto.gov.**

4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   Certified copies:

   a) ☐ All b) ☐ Some* c) ☐ None of the:

   1. ☐ Certified copies of the priority documents have been received.
   2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
   3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____ .

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____ .

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☑ Notice of References Cited (PTO-892)
2. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____.
3. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material _____.
4. ☐ Interview Summary (PTO-413), Paper No./Mail Date. _____.
5. ☐ Examiner's Amendment/Comment
6. ☑ Examiner's Statement of Reasons for Allowance
7. ☐ Other _____.

/Christopher D Birkhimer/
Primary Examiner, Art Unit 2136

Continuation of 3. The allowed claim(s) is/are: 1 – 4, 11 – 14, 21 – 27, and 30 – 34 renumbered as 1 - 20

DETAILED ACTION

The current Office Action is in response to the papers submitted 08/22/2022. Claims 1 – 4, 11 – 14, 21 – 27, and 30 – 34 are pending.

Notice of Pre-AIA or AIA Status

1. The present application is being examined under the pre-AIA first to invent provisions.

Allowable Subject Matter

2. Claims 1 – 4, 11 – 14, 21 – 27, and 30 – 34 renumbered as 1 - 20 are allowed.

The following is an examiner's statement of reasons for allowance:

Applicant's arguments, see pages 12 - 17, filed 08/22/2022, with respect to claims 1 – 4, 11 – 14, 21 – 27, and 30 – 34 have been fully considered and are persuasive. The rejections and objections of claims 1 – 4, 11 – 14, 21 – 27, and 30 – 34 have been withdrawn.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

3. Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER D BIRKHIMER whose telephone number is (571)270-1178. The examiner can normally be reached 8-5 Hoteling.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on 571-272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Christopher D Birkhimer/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
	A					
	B					
	C					
	D					
	E					
	F					
	G					
	H					
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N	CN-101241474-A	08-2008	CN	CHEON W	G06F12/0246
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

CPC							
Symbol						**Type**	**Version**
G06F	/	12	/	0246		F	2013-01-01
G06F	/	2212	/	7208		A	2013-01-01
G06F	/	2212	/	2022		A	2013-01-01

CPC Combination Sets								
Symbol					**Type**	**Set**	**Ranking**	**Version**
	/		/					

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	20	
/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	30 August 2022	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	1

U.S. Patent and Trademark Office — Part of Paper No.: 20220830

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

INTERNATIONAL CLASSIFICATION					
CLAIMED					
G06F12/02	/	12	/	02	

NON-CLAIMED					
	/		/		

US ORIGINAL CLASSIFICATION	
CLASS	**SUBCLASS**

CROSS REFERENCES(S)						
CLASS	**SUBCLASS (ONE SUBCLASS PER BLOCK)**					

NONE (Assistant Examiner)	(Date)	**Total Claims Allowed:** 20	
/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136 (Primary Examiner)	30 August 2022 (Date)	O.G. Print Claim(s) 1	O.G. Print Figure 1

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIMS

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
1	1	10	26	19	31										
2	2	11	33	20	32										
3	3	12	34												
4	4	13	11												
5	23	14	12												
6	21	15	13												
7	22	16	14												
8	24	17	27												
9	25	18	30												

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	20	
/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	30 August 2022	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	1

U.S. Patent and Trademark Office — Part of Paper No.: 20220830

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

CPC - Searched*		
Symbol	**Date**	**Examiner**
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	09/30/2021	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	02/10/2022	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	06/23/2022	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	08/30/2022	CDB

CPC Combination Sets - Searched*		
Symbol	**Date**	**Examiner**

US Classification - Searched*			
Class	**Subclass**	**Date**	**Examiner**

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	**Date**	**Examiner**
EAST search performed	09/30/2021	CDB
Assignee/Inventor search performed in EAST	09/30/2021	CDB
Prior Art/Assignee/Inventor/Google Internet database search	02/10/2022	CDB
Prior Art/Assignee/Inventor search performed	06/23/2022	CDB
Prior Art/Assignee/Inventor search performed	08/30/2022	CDB

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner
	Search performed	08/30/2022	CDB

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L181	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"2	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM

		0150324264"|"2015034 7041"|"20150347296"|" 20160018998"|"201600 26564"|"20160342509"| "20160357462"|"20160 364179"|"20170031699 "|"5568423"|"5652857"|" 5860082"|"6134631"|"7 096378"|"7339823"|"74 04031"|"7406563"|"758 1078"|"7702846"|"7710 777"|"7752381"|"78015 61"|"7818489"|"783624 4"|"7861122"|"7941692" |"7970983"|"7991944"|" 8001318"|"8024545"|"8 055833"|"8065471"|"80 65473"|"8068365"|"807 2463"|"8074022"|"8082 389"|"8086790"|"83470 42"|"8572331"|"864563 4"|"9229854"|"9335939" |"9400749"|"9519578"|" 9542118"|"9588904"|"9 652376"|"9696917"|"97 10377"|"9727454"|"978 5572"|"9846541").PN.					
L182	55	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L183	133	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L184	21099	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L185	1	(L22 L23) and (constituent near2 storage near2 location$1) and cop$4 and (erase near2 request)	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L186	21099	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L187	63	L26 and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L188	55	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L189	133	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM

		Richard)).inv.					
L190	1	(L28 L29) and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L191	1467	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L192	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L193	17	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L194	69	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L195	33	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L196	329	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L197	111	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L198	56	((controller$1) near10 (transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM

		moved or moving))					
L199	25	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L200	2421	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L201	1408	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L202	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L203	17	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L204	67	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L205	32	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L206	328	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L207	111	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L208	56	((controller$1) near10	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30

		(transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or moved or moving))					01:38 PM
L209	25	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L210	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM

		"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"20150324264"|"20150347041"|"20150347296"|"20160018998"|"20160026564"|"20160342509"|"20160357462"|"20160364179"|"20170031699"|"5568423"|"5652857"|"5860082"|"6134631"|"7096378"|"7339823"|"7404031"|"7406563"|"7581078"|"7702846"|"7710777"|"7752381"|"7801561"|"7818489"|"7836244"|"7861122"|"7941692"|"7970983"|"7991944"|"8001318"|"8024545"|"8055833"|"8065471"|"8065473"|"8068365"|"8072463"|"8074022"|"8082389"|"8086790"|"8347042"|"8572331"|"8645634"|"9229854"|"9335939"|"9400749"|"9519578"|"9542118"|"9588904"|"9652376"|"9696917"|"9710377"|"9727454"|"9785572"|"9846541").PN.				
L211	0	"store the storage location release information"	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L212	0	"store the storage location release information"	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L213	55	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L214	133	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM
L215	6	(L54 L53) and ((error$1 defect$1)).clm.	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L216	26	("11347638" OR "11347639" OR "11354234" OR	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 01:38 PM

		"11354235" OR "20050160227" OR "20070136555" OR "20080034153" OR "20090144497" OR "20090172499" OR "20090216992" OR "20100287327" OR "20110231623" OR "20120072645" OR "20130275682" OR "20130297880" OR "20140122781" OR "20150339187" OR "20160070496" OR "20170075620" OR "20170109078" OR "20170192901" OR "20170199703" OR "8626989" OR "8898410" OR "9134918" OR "9383926").pn.					
L217	6	L94 AND (updat$4 near10 full near5 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L218	68	((track$4 updat$4) near10 full near5 status near10 block)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L219	7	((track$4 updat$4) near10 full near5 status	(US-PGPUB; USPAT; USOCR; FIT (AU, AP,	OR	ON	ON	2022/08/30 01:38 PM

		near10 (block page) near10 host)	AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L220	183	((track$4 updat$4) near10 full near10 (block page) near10 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L221	9	"20130024423"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L222	10	"20080082596"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO,	OR	ON	ON	2022/08/30 01:38 PM

			RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L223	72	((track$4 updat$4) near10 full near10 (page) near10 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L224	20	((track$4 updat$4) near10 full near10 (page) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L225	0	((track$4 updat$4) near10 full near10 (die) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L226	0	((track$4 updat$4) near10 full near10 (plane) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU,	OR	ON	ON	2022/08/30 01:38 PM

			CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L227	26	((track$4 updat$4) near10 full near10 (cell) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L228	38	(controller near10 write near5 address near5 back near5 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L229	6477	((vm (virtual ADJ machine)) near5 live near5 migrat$4)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA,	OR	ON	ON	2022/08/30 01:38 PM

			VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L230	5328	(((virtual ADJ machine)) near5 live near5 migrat$4)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L231	57	2011/0145473	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L232	5	"20110145473"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L233	73	L94 AND ((hot cold) near5 data near5 host) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB,	OR	ON	ON	2022/08/30 01:38 PM

			HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L234	341	((hot cold) near5 data near5 time) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L235	87	((hot cold) near5 data near5 time) SAME wear SAME host	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L236	10	"20140006688"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT;	OR	ON	ON	2022/08/30 01:38 PM

			IBM_TDB)				
L237	341	((hot cold) near5 data near5 time) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L238	194	((hot cold) near5 data near5 time) near10 cop$3	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L239	5	((hot cold) near5 data near5 time) near10 cop$3 near5 host	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L240	194	((hot cold) near5 data near5 time) near10 cop$3	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV,	OR	ON	ON	2022/08/30 01:38 PM

			MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L241	3955	copy near5 old near5 data	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L242	413	L94 AND ((copy migrate) near5 old near5 data)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L243	3	"20090276588"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L244	7	"20060004957"	(US-PGPUB; USPAT;	OR	ON	ON	2022/08/30

			USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				01:38 PM
L245	0	L94 AND ((copy migrate) near5 timestamp near5 threshold)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L246	0	L94 AND ((copy migrate) near10 timestamp near5 threshold)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L247	5	L94 AND ((copy migrate) SAME (timestamp near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO,	OR	ON	ON	2022/08/30 01:38 PM

			NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L248	1931	L94 AND ((copy migrate) SAME (timenear5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L249	148	L94 AND ((copy migrate) SAME (time near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L250	6	L94 AND ((copy migrate) SAME (age near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L251	1854	((copy migrate) SAME (age near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY,	OR	ON	ON	2022/08/30 01:38 PM

			CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L252	13531	((copy migrate) SAME (old near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L253	980	L94 AND ((copy migrate) SAME (old near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L254	3	L94 AND ((copy migrate) SAME (old near5 data near10 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK,	OR	ON	ON	2022/08/30 01:38 PM

			TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L255	81	L94 AND ((copy migrate) SAME (age near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L256	0	L94 AND (send near5 host near5 erase near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L257	5	(send near5 host near5 erase near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L258	0	(send near5 host near5 P/E near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA,	OR	ON	ON	2022/08/30 01:38 PM

			EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L259	0	(send near5 host near5 P/E)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L260	9	(send near5 host near5 (program erase) near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L261	0	(send near5 host near5 (P/E))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO;	OR	ON	ON	2022/08/30 01:38 PM

			JPO; DERWENT; IBM_TDB)				
L262	540	(host near5 (program erase) near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L263	3	"20110161784"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L264	10	"20080082596"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:38 PM
L265	65	L184 AND (write SAME host AND address AND (page ADJ utilization))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS,	OR	ON	ON	2022/08/30 01:42 PM

			IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L266	28	L184 AND (write SAME host SAME address SAME (page ADJ utilization))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:43 PM
L267	31	(write SAME host SAME address SAME (page ADJ utilization))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:45 PM
L268	30	L183 AND (page ADJ utilization)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:45 PM

L269	3	L183 AND ((page ADJ utilization)).clm.	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/08/30 01:46 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	0	(write SAME host SAME address SAME (page ADJ utilization)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 02:05 PM
N2	3	(write AND host AND address AND (page ADJ utilization)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 02:05 PM
N3	0	(write AND host AND address AND (page ADJ utilization) AND eras$4 AND subdivision AND (erase ADJ (block unit))).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 02:07 PM
N4	0	(write AND host AND address AND (page ADJ utilization) AND subdivision AND (erase ADJ (block unit))).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 02:07 PM
N5	1	(write AND host AND address AND (page ADJ utilization) AND flash AND (erase ADJ (block unit))).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/30 02:07 PM

System and method for memory mapping

DOCUMENT ID	DATE PUBLISHED
CN-101241474-A	2008-08-13

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
CHEON, WON-MUN	N/A	N/A	N/A	N/A
LEE, YANG-SUP	N/A	N/A	N/A	N/A

DATE FILED
2008-02-13

CPC CURRENT

TYPE	CPC	DATE
CPCI	G06F12/0246	2013-01-01
CPCI	G06F12/06	2013-01-01
CPCI	G06F12/00	2013-01-01
CPCA	G06F2212/7201	2013-01-01

Abstract

The invention claims a system and method for memory mapping, and the system comprises a mapping table of logical unit and physical unit, data unit groups in signal communication with the mapping table, and log unit groups, each log unit group is associated with a corresponding one of the data unit groups; wherein updated data for any data unit within one of the data unit groups is stored in any log unit within the corresponding one of the log unit groups, and the method including receiving write data for a logical unit number from a host, determining which of a plurality of data block groups comprises the logical unit number, and storing the write data in any unfilled log unit of a log block group corresponding to the determined data block group.

Description

INVENTION-TITLE
TECHNICAL-FIELD
the technical field
the size of the mapping table of the mapping table of the size ratio of the block mapping method needs much greater. In the block mapping method, generally determines the size of the mapping table by the block unit. Here, it is necessary to sequentially write the page in the block. Disadvantage is, when using such a method, in order to create a free block (free block), need a lot of combination operation (merge operation). the invention claims a system and method for memory mapping to solve these and other problems. example embodiments are provided. An exemplary method of memory mapping comprises: writing data received from the host for a logical unit number, determining which one of a plurality of data block groups comprises the

logical unit number, storing the write data in any unfilled log unit to the determined data block group of the corresponding log block group. Another exemplary method further comprises receiving a second write request for logical unit number from the host, a second logical unit number corresponding to the second physical unit number stored in a log block in the log block group. An exemplary memory mapping system comprising: an input device for writing data received from the host for a logical unit number, a mapping device, for determining which one of plurality of data block groups comprises the logical unit number, a storage device for storing the write data in any unfilled log unit in the determined data block group of the corresponding log block group. An exemplary memory mapping system comprising: at least one logic unit and the mapping table of the physical unit, a plurality of data unit group is in signal communication with the at least one mapping table, a plurality of log unit groups, each in a data unit group unit group and the plurality of data unit groups, wherein the associated corresponding data for use in updating any data unit of a plurality of data unit data unit group in the group is stored in any log unit corresponding to one log unit group in the plurality of log unit in the group. according to the description of the exemplary embodiment of the lower facing the to further understanding of the present disclosure, wherein, to read the description of an exemplary embodiment with drawings. Description of the disclosure provides according to a system and method for memory mapping of exemplary drawings, wherein: FIG. 1 shows a schematic block diagram of memory available mapping system of the disclosed exemplary embodiment of the present invention; FIG. 2 a schematic frame of available flash memory card system of the exemplary embodiment of the claims FIG. FIG. 3 shows a schematic diagram of a hybrid mapping system of the disclosed exemplary embodiment, schematic block diagram, FIG. 6 shows a schematic block diagram of flash memory conversion layer group mapping according to the disclosed exemplary embodiments. FIG. 7 shows a logical flash according to the disclosed exemplary embodiment schematic diagram of the storage structure, FIG. 8 shows a page mapping according to the disclosed exemplary embodiment structure schematic diagram of FIG. 9 shows a schematic diagram of the block mapping structure according to the disclosed exemplary embodiments. FIG. 10 shows a hybrid mapping according to the disclosed exemplary embodiment structure schematic diagram, the schematic diagram of FIG. 11 The exemplary embodiment of the claims combining method; FIG. 12 shows a copy-merging method of the invention an exemplary embodiment of a schematic block diagram, schematic block diagram FIG. 13 The exchange-combination method of the disclosed exemplary embodiment of the present invention; FIG. 14 a schematic diagram of the mapping structure 1:1 of the disclosed exemplary embodiment; FIG. 15 a schematic diagram of mapping structure 1:2 of the disclosed exemplary embodiment, FIG. 16 shows the exemplary embodiment according to the claims 1: the schematic diagram of N mapping structure, FIG. 17 shows the N according to the disclosed exemplary embodiment schematic diagram of M + K mapping structure. Specific implementation examples shown in Figure, memory mapping system generally represented by reference numeral 100. memory mapping system 100 a processor 116 in signal communication with the processor for flash memory 110, a random access memory (RAM) read only memory (ROM) in signal communication with the processor 112 and is in signal communication with the processor 114. For example, ROM 112 may include the processor 116 program steps can be executed for providing a read command and a write command, to read data or the data written into the flash memory 110 or the RAM 114 from the flash memory 110 or RAM 114. 4. The memory mapping embodiment of the present disclosure, the flash memory 110 is performed in response to a read command and a write command of a read operation and a write operation. Furthermore, the ROM 112 and the RAM 114 may store data related to structure and/or the processor 116 step is capable of executing the application program. turning to FIG. 2, the flash memory system is represented generally by reference numeral 200. flash memory card system 200 may be a portable electronic device, such as a computer, a digital camera, a digital music player, a cellular phone, and a personal digital assistant (PDA) and so on. flash memory system 200 comprises a flash memory 220 for signal
communication with a main computer 210. flash memory 220 may be a solid state disk (SSD), a SD card, a MMC, a memory stick, an embedded card (such as moviNAND, GBNAND and iNAND, etc.). flash memory card 220 comprises a controller in signal communication with the flash memory 250 230. host 210 using a flash translation layer (FTL) 250 to communicate with the flash memory, the FTL may include for effectively managing the flash memory card 220 of the firmware and/or logic. FTL may be stored in the memory 250 of the controller 230 or flash, or a memory 250 is implemented in the controller 230 or flash. controller 230 a signal communication with the controller bus 232 to host interface 231, is in signal communication with the

controller bus 232 flash memory interface 233, in signal communication with the controller bus 232 of buffer memory 235 in signal communication with the controller bus 232 of the CPU 237 and in signal communication with the control bus 232 of the ROM 239. In operation, the address mapping operation is one function of FTL address mapping operation from the host 210 receives a logical address (LA), then the received LA is converted into a physical address (PA). PA is used for storing the data in address 250 of the flash memory. address mapping table may be used to help address the conversion of LA to PA. in the address mapping table, the logical address and the corresponding physical address is associated. The address mapping table may be stored in the flash memory 250, and loaded to the controller 230 in the buffer memory 235. determined by the mapping unit is limited by the size of the address mapping table. Now turning to FIG. 3, hybrid mapping system generally represented by reference numeral 300. hybrid mapping system 300 includes a host 310, a host 310 in signal communication with the FTL 320. FTL 320 then memory 330 in signal communication with the flash. logical address (LA) transferred between host 310 and FTL 320. the physical address (PA) transfer between the FTL 320 and the flash memory 330. FTL 320 includes a block mapping table 321 and a page mapping table 322. For example, the log block (log) application page mapping table, and can be applied to the data block the block mapping table. flash memory 330 comprises data area 331. Here, the data area 331 includes a physical block number (PBN) 100, 101, 102, 103, 104, 105, 206, 303. .., 0,901,902 and 903 of the data block. flash memory 330 further comprises a log region 332, where the log area 332 a PBN300, 400 and 500 of the log block. flash memory 330 further comprises a free zone 333, where free 333 comprises free blocks 600, 601, 602, and 603. Further, the flash memory 330 includes a meta (meta) 334. In operation, for example, the hybrid mapping method can be used for mapping address in a flash memory system. In the hybrid mapping method, whether the page unit or the block unit, a mapping table can be a logical address associated with a physical address. when the hybrid mapping method for page mapping and data block block mapping, log block mapping and combining operation of the table will reduce the number.

For example, when a write operation is performed, the page data to be stored in the designated data block is firstly stored in the specified log block. If there is no free block as a log block, the FTL is performing the combination operation. in the combining operation, the log block in the page data and the page data in the data block corresponding to the log block is stored or copied to the new data block. change the mapping information is stored in a meta-block during the operation. Once the content is assigned to the other block, the log block and the old data block can be safely erased. As shown in FIG. 4, merging order generally represented by reference numeral 400. combining the sequence 400 received data block 100 and log block 300, and provides the combined data block 101. Here, when data block 100 has been stored in the physical page number (PPN) 2 and logical page number (LPN) 1 to the data and the host according to LPN 2, 3 and 0 order to request a write operation, log block 300 PPN 1, 2, and 3 in order is written. when the FTL such as to perform merging operation of the log block 300 due to the lack of free block, log block 300 of LPN 2, 3 and 0 and the corresponding LPN 1 of the data block 100 are copied to the new data block 101 PPN 1, 2, 3 and 4 to the LPN 0, 1, 2, and 3 that order. When such designated to the new log block or a free block, the log block 300 and data block 100 is erased. The method uses a number of page copy operation and the block erase operation. if one block is composed of 4 pages, then each merge operation requires four page copy operation and two block erase operations. In addition, the method has low page utilization rate in the . log block 300 before converted into free block by combining, using only 4 pages in the page 3. for frequent merge operation of many copy operation and erasing operation and low page utilization, resulting in overall performance of the flash memory system is reduced. turning to FIG. 5, a group mapping table of the hybrid mapping system generally represented by reference numeral 500. hybrid mapping system 500 comprises host computer 510 in signal communication with FTL 520. FTL 520 then stores 530 in signal communication with the flash. logical address (LA) transferred between host 510 and FTL 520. physical address (PA) transfer between the FTL 520 and the flash memory 530. FTL 520 includes a data block mapping table 521, mapping table 522 and a page mapping table 523. for example, can be applied to the log block page mapping table, and can be applied to the data block the block mapping table. flash memory 530 comprises data area 531. Here, the data area 531 includes a physical block number (PBN) 100, 101, 102, 103, 104, 105, 206, 303. ..., 900, 901, 902, and 903 of data block. the data blocks distributed in the data block group. further comprises a journal area 530 in the flash memory 532. Here, log region 532 a with the PBN 300,400 500 of log block. the log block is distributed in the log block group. flash memory 530 further comprises a

free zone 533, where free 533 comprises free blocks 600, 601, 602, and 603. Further, the flash memory 530 includes a meta area 534. In operation, the group mapping method, the log block group (LBG) is assigned to data block group (DBG).
LBG comprising a plurality of log blocks and DBG comprises a plurality of data blocks. Here, the LBG 1 composed of log 300 and 400, and is assigned to DBG 1. DBG 1 of data blocks 100, 101, 102, and 103. Thus, programming to the data block 100-103, the data is programmed into the first log block 300 or 400. each log block in the LBG can be designated in the designated DBG of any data or all of the data blocks. the number of blocks in each DBG or LGB can be changed. the group mapping method, which can reduce the number of combination operation, and an average page usage log block will be improved. Now turning to FIG. 6, the set of mapped flash translation layer (FTL) is generally represented by reference numeral 600. In this example, the FTL 600 includes a block mapping table (BMT) 610, the group map table (GMT) 620 and a page mapping table (PMT) 630. Here, the block mapping table 610 the logical block number (LBN) 0 to 7 and physical block number (PBN) is associated with 100 to 105,206 and 303. a group mapping table 620 the DBG 1 and 400 associated with the log block PBN 300, and the DBG 2 associated with the log block PBN 500. For DBG 1, the page mapping table 630 the logical page number (LPN) 3, 4, 11, 12, 13, and 14 respectively to a physical page number (PPN) 1200, 1201, 1202, 1203, 1600 and 1601 are associated. Furthermore, for DBG 2, the page mapping table 630 the LPN 17 to 20 are associated with PPN 2000 to 2003. Therefore, the LBG 1 PBN300 with and LPN3, respectively 4, 11 and 12 associated with PPN 1200 to 1203. LBG 1 PBN 400 associated with the LPN 13 and 14, respectively of the PPN 1600 to 1601. LBG 1 associated with the DBG 1. DBG 1 to 4 1 PBN 100 PPN 0 to 3 is associated with the LPN. DBG 1 to 4 1 PBN 101 PPN 4 to 7 is associated with the LPN. DBG 1 to 4 1 PBN 102 PPN 8 to 11 is associated with the LPN. DBG 1 to 4 1 PBN 103 PPN 12 to 15 is associated with the LPN. Furthermore, the LBG 2 PBN 500 comprises PPN associated with the LPN 17 to 20 2000~2003. LBG 2 associated with the DBG 2. PPN PBN DBG2 104 16 to 19 associated with the LPN 1 to 4. DBG 1 to 4 2 PBN 105 PPN 20 to 23 is associated with the LPN. DBG 1 to 4 2 PBN 206 PPN 24 to 27 is associated with the LPN. DBG PBN 303 2 associated with the LPN28~31 PPN 1 to 4. Therefore, the block mapping table 610 the logical block number (LBN) into a physical block number (PBN). mapping information of mapping table 620 with each DBG corresponding to DBG of the log block PBN. page mapping table 630 has mapping information of each of DBG, LPN and the corresponding PPN. In operation of the exemplary embodiment of the group mapping received from host LPN write requests 3 and 4. FTL in the specified log block from a free block 300, and generate the LBG 1 corresponding to the DBG 1. the daily FTL
300 registered in the group mapping table 620. log block 300 may be assigned to DBG all the data block 1. Here, LPN0~15 may be stored in the log block 300. LPN3 and 4 is stored in the PPN 1200 and 300 daily 1201. the mapping information is stored in the page mapping table 630. Next, the receiving LPN 11 to 14 from the host write request. LPN 11 and 12 are previously stored in the specified log block 300 PPN in the 1202 and 1203. 4. The new mapping information to update the table 630. in addition, the new log block 400 assigned to the LBG 1, memory LPN 13 and 14. FTL 400 registered in the log block group mapping table 620. LPN 13 and 14 are stored in the log block 400 PPN 1600 1601, and PMT 630 is updated. the writing request from the host arrives at the LPN 17~20. Because LPN 17 to 20 does not belong to DBG 1, so they can not be stored in the log block 400. FTL specified new journal block 500 from the free block and generating LBG 2 corresponding to the DBG 2. FTL 500 registered in the log block group mapping table 620. LPN 16 to 31 may be stored in the log block 500. LPN 17 to 20 is stored in the PPN 2000 to 2003 of the log block 500, and PMT 630 is updated. by using the group mapping method, a page utilization rate of a log block is increased. the example log block 300 will only use one page (such as photooriented 1200) of the traditional method of forming contrast. In this embodiment, since one log block is assigned to all the data blocks in the data block group, then the log block 300 uses all of the pages (such as PPN 1200 to 1203). In addition, corresponding to the page utilization rate of a log block, the number of combination operation is reduced. Furthermore, the copy operation and reduces the number of erase operations, the performance of the system is increased. . As shown in FIG. 7, the logical flash memory structure generally represented by reference numeral 700. flash memory 700 comprises metadata area 710 and user data area 720, wherein the metadata region 710 is not visible, the user data area 720 is visible to the user to the user. metadata area 710 comprises a firmware block 712, retaining block 714, the MAP block 716 and write buffer block 718, wherein the retention block 714 for replacing the bad block. MAP block 716 includes a block mapping

table 717. The block mapping table 717 the logical block number 0, claim. ..., 31 are respectively mapped to the physical block numbers 3, 15. N, 0. a plurality of write buffer block 718 comprises a write buffer block 719. Here, write buffer block 719 stores the logical page number 1 and 3 replace the data (i.e. ' and 3 '). a user data area 720 comprises a data block 722 to 726. data block 725 includes logical page numbers 0, 1, 2, and 3. In operation, the merge operation 725 to write the data block buffer block 719 combined to form a new data block 726. Here, the new data block 726 comprises a logic page number 0 ', 2 and 3 '. turning to FIG. 8, a page mapping structure generally represented by reference numeral 800. page mapping structure 800 includes the page mapping table 810 and the flash memory 870. a page mapping table 810 comprises the logical page number 830 and the corresponding object
physical page number 850. 832 for inquiring the physical page number corresponding to the logic page number. In operation, the logical page number (LPN) 1 initially corresponds to a physical page number (PPN) 2. new PPN 5 is written in the flash memory 870, and the page mapping table 810 is updated to the LPN 1 associated with the LPN 5. This may be called "any position (out-of-place) maps". the updated page can be written in different positions in the new block. page mapping table updating using the relatively large size of the mapping table described above. For example, mapping table of 128KB for 128 MB of the NAND flash memory. Now turning to FIG. 9, the block mapping structure generally represented by reference label. block mapping structure 900 includes a block mapping table 920 and a flash memory 970. the block mapping table 920 includes logical block number 940 corresponding to the physical block number 960. logical block number 942 for inquiring the corresponding physical block number. physical page number 962 is added to the offset of the physical block number. In operation, the flash memory 970, a physical page number (PPN) 2 is updated to the PPN 2 '. This may be referred to as the " sequence position (in-place). Here the same bit page of the update is written into the new block, and the block mapping table is updated. When any position update block copy operation page when writing operation is present during the replication. As shown in FIG. 10, the hybrid mapping structure generally represented by reference numeral 1000. hybrid mapping mechanism 1000 a page mapping table 1010, block mapping table 1020, log block 1070 and data block 1080. page mapping table 1010 comprises a logical page number 1030 and the corresponding physical page number 1050. the block mapping table 1020 includes logical block number 1040 corresponding to the physical block number 1060. logical block number 1032 for inquiring the corresponding physical block number. In operation, the logical block number 1032 for inquiring the corresponding physical block numbers in block mapping table 1020, or the logical page number 1052 for inquiring the physical page corresponding to coding, for example in the page mapping table 1010, page mapping for write cache or log block, and the block mapping data block can be used. when the logical page number 1 is initially corresponding to 1070 physical page numbers in the log block 2, the page mapping table 1010 can be updated so that the logical page number 1 corresponding to the physical page number 4. When the log block 1070 for writing the data block 1080, the block mapping table 1020 can be updated. to map, merging method generally represented by reference numeral 1100. Here, the log entry (entry) information 1110 for updating the log block 1120. Subsequently, the log block 1120 and data block 1130 are combined. after combining the data is written to the free block 1140. Thereby, the free block 1140 into a new data block. Therefore, combining the method of specifying the free block, and the valid page is copied to the free block is designated, the assigned free block set is data block to update the mapping page, and the log block and the old data block set
is an erased state. Now turning to FIG. 12, copy-combination method generally represented by reference numeral 1200. Here, the log entry information 1210 for updating pages of the log block 1220. data block 1230 does not change the page re-converted to the log block 1220. Therefore, the copy-merging method 1200 the effective page in the data block copied to the daily , the old data block to the log block is set as updated mapping page, and the old data block is set to be the erasing state. As shown in FIG. 13, exchange-merging method generally represented by reference numeral 1300. Here, the log entry information 1310 is used to update the log block in all the pages 1320. by the old data block to the log block is set as updated mapping page, and the old data block is set to be the erasing state. mapping the total structure to FIG. 14,1:1 is represented by reference numeral 1400. 1:1 mapping mechanism 1400 a unit mapping table 1410, the unit mapping table 1410 has the part or column 1430 corresponding to logical unit number section or column 1420 with the physical unit number. physical unit number corresponding to the flash memory comprises physical unit 1440 of transmitting unit 1450. For example, the unit can be block. That is, each

mapping table entry corresponding to one physical unit. Now turning to FIG. 15,1:2 mapping structure is generally represented by reference numeral 1500. 1:2 mapping structure 1500 includes the page mapping table 1510 and block mapping table 1520. a page mapping table 1510 has with the logic page number or column corresponding to the physical page number or column 1550 1530. the block mapping table 1520 with logical block number or column corresponding to the physical block number or column 1560 1540. physical block number and the physical page number corresponding to the flash memory or physical storage device in physical block 1570 to 1573 and pages thereof. Here, from the block mapping table 1520 of physical block numbers (e.g. 101) pointing to physical block 1570 from the physical page number of the page mapping table 1510 (such as 1, 0, 2, and 3) pointing to the block 1570 of the physical page. direct the identified block and page in the mapping table is original original units, each unit may have an additional log unit associated therewith for recording the updating. Therefore, the original block 1570 and 1571 can be respectively and log block 1572 and 1573 are associated. That is, each mapping table entry corresponding to one or two physical units. As shown in FIG. 16, the 1: N mapping structure generally represented by reference numeral 1600. 1: N structure 1600 a page mapping table 1610 and block mapping table 1620. a page mapping table 1610 has with the logic page number or column corresponding to the physical page number or column 1650 1630. the block mapping table 1620 with logical block number or column corresponding to the physical block number part or column 1660 1640. physical block number and the physical page number corresponding to the physical storage device or physical block 1670 to 1675 in the flash memory and page them. Here, from the block mapping table 1620 of physical block numbers (e.g. 101)
points to physical block 1670 from the physical page number of the page mapping table 1610 (such as 1, 0, 2, and 3) pointing to the physical page in block 1670. direct the identified block and page in the mapping table is the original unit, each original unit can have N additional log unit associated therewith for recording the updating. Therefore, the original block 1671 and log block 1673, 1675. .. 4 related. That is, each mapping table entry can be physical units corresponding to 1 N. 1: N mapping structure 1600 using the data block and the log block of the combined delay. turning to FIG. 17, N: M + K mapping structure generally represented by reference numeral 1700. N: M + K mapping structure 1700 includes the page mapping table 1710 and block mapping table 1720. a page mapping table 1710 has with the logic page number or column corresponding to the physical page number or column 1750 1730. the block mapping table 1720 with logical block number or column corresponding to the physical block number or column 1760 1740. physical block number and the physical page number corresponding to the flash memory or physical storage device in physical block 1769 to 1775 and pages thereof. Here, from the block mapping table 1720 the physical block number (such as 101) pointing to physical block 1770, the physical page number of the page mapping table 1710 (such as 1, 0, 2, and 3) pointing to the block 1770 of the physical page. log unit into a chain shape or set block and page directly identified in the mapping table of the original cell, and each is of the original unit can be added with K associated to record updated. log unit of additional chain or into groups that can be shared by the N original unit. In addition, there will be different to the chain or a set of additional log unit. Therefore, the original block 1771 and 1772 both can be connected into a chain shape or into log block groups 1783 and 1785 are associated, where original block with physical block numbers 0 and 1771 are respectively 1772 and 1. Furthermore, for example, the original block 1769 and 1770 can be associated with a log block 1780. Here, the original block with physical block numbers 101 and 1769 are respectively 1770 and 102. N: M + K 1700 using data block mapping structure between relevance of the log block and the log block and the data block of the combined delay. maximum number of combination or log block of delay, where N is the number of user data blocks, M is the write number of the cache or log block, K is the log block group (LBG), LBG can be dynamically controlled. data block group (DBG) associated with each LBG. In this example, each DBG comprises two data blocks, but in an alternative embodiment, the DBG may include any number of data blocks. storing the block generally is an erase size units of memory "page" is generally read/write size of the unit. ordinary skill in the related art should be understood that alternative embodiments can use alternative memory cells, block and/or page size, these sizes are not limited to those described in the exemplary embodiment. unit of the other unit or sector (sector) of any size can be used to instead described herein, block and/or page. For example, in hybrid mapping embodiments, preferably, using a write cache or log unit of page
mapping, and using the block mapping for data unit, but alternative embodiments can use two or more mapping units of selectable size. Although the exemplary embodiments are described with reference to the

accompanying drawings herein, but it should be understood that the invention is not limited to those precise embodiments, and without departing from the scope or spirit of the disclosure condition of ordinary skill in the related art can realize the embodiment of various other changes and modification. All these change and modification is included in the scope of the present disclosure in claim.

Claims

1. A method of memory mapping comprises: receiving write data for a logical unit number from a host; determining which of a plurality of data block groups comprises the logical unit number; and storing the write data in any unfilled log unit of a log block group corresponding to the determined data block group.
2. The method according to claim 1, wherein said size of a unit is a smallest writable number of bits, the size of a block is a smallest erasable number of bits and a plural multiple of writable unit size, and the size of a group is a variable multiple of erasable block size.
3. The method according to claim 1, wherein a unit is at least one of a byte, a page or a sector.
4. The method according to claim 1, wherein an association between the logical unit number and a physical unit number is saved in a page mapping table.
5. The method according to claim 4, wherein page mapping table further comprises data group numbers each associated with a plurality of logical unit numbers.
6. The method according to claim 1, wherein step of determining data block groups comprises modulus arithmetic based on dividing the logical unit number by the number of the plurality of data block groups.
7. The method according to claim 1, further comprising initializing the log block group by assigning a free block as a log block in the log block group.
8. The method according to claim 7, further comprising expanding the log block group by assigning a free block as the next log block in the log block group.
9. The method according to claim 7, further comprising assigning a plurality of free blocks as a corresponding plurality of log blocks in the log block group.
10. The method according to claim 1, further comprising assigning a free block as a log block.
11. The method according to claim 10, wherein step of assigning log comprises enrolling the log block in a group mapping table.
12. The method according to claim 11, wherein said association of the group mapping table is each group to a plurality of physical block numbers for log blocks.
13. The method according to claim 10, wherein said log block is used for all data blocks within the data block group.
14. The method according to claim 1, further comprising: receiving a write request for a second logical unit number from the host; and storing the second logical unit number in association with a second physical unit number in the log block of the log block group.
15. The method according to claim 14, wherein step of storing the logical unit number comprises updating a page mapping table in accordance with new mapping information for the logical unit number.
16. The method according to claim 14, further comprising: checking whether the log block is full, and if so, assigning an additional log block to the log block group for storing further logical unit numbers.
17. The method according to claim 16, wherein step of assigning additional log block comprises updating a group mapping table in accordance with new mapping information for the physical block number of the additional log block.
18. The method according to claim 16, further comprising: receiving a write request for a third logical unit number from the host; and storing the third logical unit number in association with a third physical unit number in an existing log block of the log block group.
19. The method according to claim 1, further comprising: receiving a write request for another logical unit number from the host where the other logical unit number is not included in the first data block group; and storing the other logical unit number in association with another physical unit number in a log block of another log block group and updating a page mapping table.
20. The method according to claim 19, further comprising: assigning a free block as a log block to create the other log block group corresponding to another data block group that includes the other logical unit number.

21. The method according to claim 1, wherein one log block is assigned to all data blocks within a data block group.
22. The method according to claim 1, wherein a log block uses a plurality of physical unit numbers.
23. The method according to claim 1, wherein said memory comprises at least one of flash memory, DRAM, PRAM, MRAM, FRAM or EEPROM.
24. The method according to claim 1, wherein at least one mapping table is organized to map a first plurality of whole user data blocks to a second plurality of log blocks or write buffers with a third plurality of delayed merge blocks.
25. The method according to claim 24, wherein said third plurality is dynamically controlled.
26. The method according to claim 1, wherein any number of log blocks can be associated with any number of logical blocks.
27. The method according to claim 1, wherein said number of log blocks per log block group is greater than one.
28. The method according to claim 1, wherein a log block stores write data units for a plurality of data blocks.
29. A memory mapping system comprising: an input device for receiving write data for a logical unit number from a host; a mapping device for determining which of a plurality of data block groups comprises the logical unit number; and a memory device for storing the write data in any unfilled log unit of a log block group corresponding to the determined data block group.
30. The memory mapping system comprising: at least one mapping table of logical unit and physical unit; a plurality of data unit groups in signal communication with the at least one mapping table; and a plurality of log unit groups, each associated with a corresponding one of the plurality of data unit groups; wherein updated data for any data unit within one of the plurality of data unit groups is stored in any log unit within the corresponding one of the plurality of log unit groups.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 12/06 (2006.01)

[12] 发明专利申请公布说明书

[21] 申请号 200810009911.5

[43] 公开日 2008 年 8 月 13 日

[11] 公开号 CN 101241474A

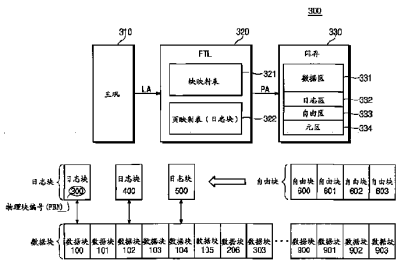
[22] 申请日 2008.2.13
[21] 申请号 200810009911.5
[30] 优先权
[32] 2007.2.6 [33] KR [31] 10-2007-0012198
[32] 2007. 8. 3 [33] US [31] 11/833,750
[71] 申请人 三星电子株式会社
地址 韩国京畿道水原市灵通区梅滩洞 416
[72] 发明人 千源汶 李洋燮
[74] 专利代理机构 北京铭硕知识产权代理有限公司
代理人 韩明星 韩素云
权利要求书 3 页 说明书 11 页 附图 17 页

[54] 发明名称

存储器映射系统及方法

[57] 摘要

本发明公开了一种存储器映射系统和方法，该系统包括：逻辑单元与物理单元的映射表；数据单元组，与映射表进行信号通信；日志单元组，每个日志单元组与对应的一个数据单元组相关联，其中，用于一个数据单元组内的任何数据单元的更新数据被存储在对应的一个日志单元组中内的任何日志单元中。该方法包括：从主机接收用于逻辑单元编号的写数据；确定多个数据块组中的哪一个包括所述逻辑单元编号；将写数据存储在与确定的数据块组对应的日志块组中的任意的未填充的日志单元中。



1、一种存储器映射方法，所述方法包括:

从主机接收用于逻辑单元编号的写数据;

确定多个数据块组中的哪一个包括所述逻辑单元编号;

将写数据存储在与确定的数据块组对应的日志块组中的任意的未填充的日志单元中。

2、如权利要求 1 所述的方法，其中，单元的大小是最小的可写位数，块的大小是最小的可擦除位数且是可写单元的大小的多倍，组大小是可擦除块的大小的可变的倍数。

3、如权利要求 1 所述的方法，其中，单元是字节、页或扇区中的至少一种。

4、如权利要求 1 所述的方法，其中，逻辑单元编号与物理单元编号之间的关联性保存在页映射表中。

5、如权利要求 4 所述的方法，页映射表还包括分别与多个逻辑单元编号相关联的数据块组编号。

6、如权利要求 1 所述的方法，其中，确定数据块组的步骤包括基于逻辑单元编号除以多个数据块组的数量的模算术。

7、如权利要求 1 所述的方法，还包括通过将自由块指定为日志块组中的日志块来将日志块组初始化。

8、如权利要求 7 所述的方法，还包括通过将自由块指定为日志块组中的下一个日志块来扩展日志块组。

9、如权利要求 7 所述的方法，还包括将多个自由块指定为日志块组中的对应的多个日志块。

10、如权利要求 1 所述的方法，还包括将自由块指定为日志块。

11、如权利要求 10 所述的方法，指定日志块的步骤包括将所述日志块登记在组映射表中。

12、如权利要求 11 所述的方法，其中，组映射表的关联性是指每一组与用于日志块的多个物理块编号之间的关联性。

13、如权利要求 10 所述的方法，其中，日志块用于数据块组内的所有数据块。

Samsung Ex. 1002, Page 71 of 622

14、如权利要求 1 所述的方法，还包括:

从主机接收第二逻辑单元编号的写请求;

将与第二物理单元编号相关联的第二逻辑单元编号存储在日志块组的日志块中。

15、如权利要求 14 所述的方法，存储第二逻辑单元编号的步骤包括: 根据用于逻辑单元编号的新映射信息来更新页映射表。

16、如权利要求 14 所述的方法，还包括检查日志块是否被填满，如果被填满，则将另外的日志块指定到日志块组，以用于存储更多的日志单元编号。

17、如权利要求 16 所述的方法，指定另外的日志块的步骤包括根据用于另外的日志块的物理块编号的新映射信息来更新组映射表。

18、如权利要求 16 的方法，还包括:

从主机接收第三逻辑单元编号的写请求;

将与第三物理单元编号相关联的第三逻辑单元编号存储在日志块组现有的日志块中。

19、如权利要求 1 所述的方法，还包括:

从主机接收另一逻辑单元编号的写请求，其中，所述另一逻辑单元编号不包括在所述数据块组中;

将与另一物理单元编号相关联的所述另一逻辑单元编号存储在另一日志块组的日志块中，并更新页映射表。

20、如权利要求 19 所述的方法，还包括:

将自由块指定为日志块，以创建与包括另一逻辑单元编号的另一数据块组对应的另一日志块组。

21、如权利要求 1 所述的方法，其中，一个日志块被指定到数据块组内的所有数据块。

22、如权利要求 1 所述的方法，其中，日志块利用了多个物理单元编号。

23、如权利要求 1 所述的方法，其中，存储器包括闪存、DRAM、PRAM、MRAM、FRAM 和 EEPROM 中的至少一种。

24、如权利要求 1 所述的方法，其中，至少一个映射表的构造方式为: 将第一多个所有的用户数据块映射到具有第三多个延迟合并块的写缓存或第二多个日志块。

25、如权利要求 24 所述的方法，其中，第三多个延迟合并块被动态地控

Samsung Ex. 1002, Page 72 of 622

制。

26、如权利要求 1 所述的方法，其中，任意数量的日志块能与任意数量的逻辑块相关联。

27、如权利要求 1 所述的方法，其中，每个日志块组的日志块的数量大于 1。

28、如权利要求 1 所述的方法，其中，日志块存储用于多个数据块的写数据单元。

29、一种存储器映射系统，包括:

输入装置，用于从主机接收用于逻辑单元编号的写数据;

映射装置，用于确定多个数据块组中的哪一个包括所述逻辑单元编号;

存储装置，用于将写数据存储在与确定的数据块组对应的日志块组中的任意的未填充的日志单元中。

30、一种存储器映射系统，包括:

至少一个逻辑单元与物理单元的映射表;

多个数据单元组，与至少一个映射表进行信号通信;

多个日志单元组，每个日志单元组与多个数据单元组中对应的一个数据单元组相关联，

其中，用于多个数据单元组中的一个数据单元组内的任意数据单元的更新数据被存储在多个日志单元组中的对应的一个日志单元组内的任意日志单元中。

存储器映射系统及方法

本申请要求于 2007 年 2 月 6 日在韩国知识产权局提交的第 P2007-0012198 号韩国专利申请的外国优先权，其公开通过引用完全包含于此。

技术领域

总体来说，本公开涉及闪存系统。更具体地讲，本公开涉及具有映射表（mapping table）的闪存系统和映射方法。

背景技术

新兴的便携式电子设备（比如计算机、数码相机、数码音乐播放器、蜂窝式电话和个人数字助理等）已经增加了对闪存的使用，具体地说，已经增加了对闪存卡的使用。闪存卡可以是 SSD、SD 卡、MMC、记忆棒（Memory Stick）或嵌入式卡（比如 moviNAND、GBNAND 和 iNAND 等）。

主机通常利用闪存转换层（flash translation layer，FTL）与闪存进行通信。FTL 可包括存储在控制器或闪存中的固件（firmware）。FTL 通常用于有效地管理闪存卡。

地址映射操作是 FTL 的一项功能，该操作从主机接收逻辑地址（LA），随后将接收到的 LA 转换成物理地址（PA）。PA 是实际用于将数据存储在闪存内并从闪存检索数据的地址。

可利用地址映射表来帮助地址的转换。例如，地址映射表可被存储在闪存内，并被加载到控制器内的缓存（buffer）中。在地址映射表内，逻辑地址和对应的物理地址相关联。地址映射表的大小取决于被限定的映射单元。地址映射表的大小越大，所需的缓存的大小越大。通常的映射单元包括块或页，块通常是擦除单元的大小，而页通常是读/写单元的大小。

现有许多对应于各种映射单元可用的映射方法。在页映射方法中，通常由页单元确定映射表的大小。即，数据地址的逻辑页被转换成对应的物理页。这里，一个存储块可包括数十页或数百页。不利的是，例如，页映射方法所

Samsung Ex. 1002, Page 74 of 622

需的映射表的大小比块映射方法所需的映射表的大小大得多。

在块映射方法中，通常由块单元确定映射表的大小。这里，必须顺序地将页写入块内。不利的是，当使用这种方法时，为了创建自由块（free block），需要大量的合并操作（merge operation）。

发明内容

通过一种用于存储器映射的系统和方法来解决这些和其它问题。提供了示例实施例。

存储器映射的示例性方法包括：从主机接收用于逻辑单元编号的写数据；确定多个数据块组中的哪一个包括所述逻辑单元编号；将写数据存储在与确定的数据块组对应的日志块组中的任意的未填充的日志单元中。

另一示例性方法还包括从主机接收第二逻辑单元编号的写请求；将与第二物理单元编号对应的第二逻辑单元编号存储在日志块组的日志块中。

示例性存储器映射系统包括：输入装置，用于从主机接收用于逻辑单元编号的写数据；映射装置，用于确定多个数据块组中的哪一个包括所述逻辑单元编号；存储装置，用于将写数据存储在与确定的数据块组对应的日志块组中的任意的未填充的日志单元中。

示例性存储器映射系统包括：至少一个逻辑单元与物理单元的映射表；多个数据单元组，与至少一个映射表进行信号通信；多个日志单元组，每个日志单元组与多个数据单元组中对应的一个数据单元组相关联，其中，用于多个数据单元组中的一个数据单元组内的任意数据单元的更新数据被存储在多个日志单元组中的对应的一个日志单元组内的任意日志单元中。

将根据下面对示例性实施例的描述来进一步理解本公开，其中，将要结合附图来阅读对示例性实施例的描述。

附图说明

本公开提供了根据下面的示例性附图的一种用于存储器映射的系统和方法，其中：

图 1 示出了根据本公开示例性实施例的可用的存储器映射系统的示意性框图；

图 2 示出了根据本公开示例性实施例的可用的闪存卡系统的示意性框

Samsung Ex. 1002, Page 75 of 622

图；

图 3 示出了根据本公开示例性实施例的混合映射系统的示意性框图；

图 4 示出了根据本公开示例性实施例的合并顺序的示意性框图；

图 5 示出了根据本公开示例性实施例的具有组映射表的混合映射系统的示意性框图；

图 6 示出了根据本公开示例性实施例的组映射闪存转换层的示意性框图；

图 7 示出了根据本公开示例性实施例的逻辑闪存结构的示意性框图；

图 8 示出了根据本公开示例性实施例的页映射结构的示意性框图；

图 9 示出了根据本公开示例性实施例的块映射结构的示意性框图；

图 10 示出了根据本公开示例性实施例的混合映射结构的示意性框图；

图 11 示出了根据本公开示例性实施例的合并方法的示意性框图；

图 12 示出了根据本公开示例性实施例的复制-合并方法的示意性框图；

图 13 示出了根据本公开示例性实施例的交换-合并方法的示意性框图；

图 14 示出了根据本公开示例性实施例的 1:1 映射结构的示意性框图；

图 15 示出了根据本公开示例性实施例的 1:2 映射结构的示意性框图；

图 16 示出了根据本公开示例性实施例的 1:N 映射结构的示意性框图；

图 17 示出了根据本公开示例性实施例的 N:M+K 映射结构的示意性框图。

具体实施方式

如图 1 所示，存储器映射系统总地用参考标号 100 来表示。存储器映射系统 100 包括处理器 116、与处理器进行信号通信的闪存 110、与处理器进行信号通信的只读存储器（ROM）112 及与处理器进行信号通信的随机存取存储器（RAM）114。例如，ROM 112 可包括处理器 116 可执行的程序步骤，用于提供读命令和写命令，以从闪存 110 或 RAM 114 读取数据或将数据写入闪存 110 或 RAM 114。根据本公开的存储器映射实施例，在闪存 110 内执行响应于读命令和写命令的读操作和写操作。此外，ROM 112 和 RAM 114 可存储相关的数据结构和/或处理器 116 可执行的应用程序步骤。

转到图 2，闪存卡系统总地用参考标号 200 来表示。闪存卡系统 200 可以是便携式电子设备，比如计算机、数码相机、数码音乐播放器、蜂窝式电话和个人数字助理（PDA）等。闪存卡系统 200 包括与闪存卡 220 进行信号

Samsung Ex. 1002, Page 76 of 622

通信的主机 210。闪存卡 220 可以是固态盘（SSD）、SD 卡、MMC、记忆棒、嵌入式卡（比如 moviNAND、GBNAND 和 iNAND 等）。

闪存卡 220 包括与闪存 250 进行信号通信的控制器 230。主机 210 利用闪存转换层（FTL）与闪存 250 进行通信，FTL 可包括用于有效地管理闪存卡 220 的固件和/或逻辑。FTL 可被存储在控制器 230 或闪存 250 中，或可在控制器 230 或闪存 250 中实现。

控制器 230 包括与控制器总线 232 进行信号通信的主机接口 231、与控制器总线 232 进行信号通信的闪存接口 233、与控制器总线 232 进行信号通信的缓冲存储器 235、与控制器总线 232 进行信号通信的 CPU 237 及与控制器总线 232 进行信号通信的 ROM 239。

在操作中，地址映射操作是 FTL 的一项功能，地址映射操作从主机 210 接收逻辑地址（LA），然后将接收到的 LA 转换成物理地址（PA）。PA 是实际用于将数据存储在闪存 250 内的地址。可使用地址映射表来帮助 LA 到 PA 的地址转换。在地址映射表内，逻辑地址和对应的物理地址相关联。该地址映射表可被存储在闪存 250 内，并被加载到控制器 230 内的缓冲存储器 235。地址映射表的大小取决于被限定的映射单元。

现在转到图 3，混合映射系统总地用参考标号 300 来表示。混合映射系统 300 包括主机 310，主机 310 与 FTL 320 进行信号通信，FTL 320 接着与闪存 330 进行信号通信。逻辑地址（LA）在主机 310 和 FTL 320 之间传递。物理地址（PA）在 FTL 320 和闪存 330 之间传递。FTL 320 包括块映射表 321 和页映射表 322。例如，可以对日志块（log block）应用页映射表，而可以对数据块应用块映射表。

闪存 330 包括数据区 331，这里，数据区 331 包括具有物理块编号（PBN）100、101、102、103、104、105、206、303...、900、901、902 和 903 的数据块。闪存 330 还包括日志区 332，这里，日志区 332 包括具有 PBN 300、400 和 500 的日志块。闪存 330 还包括自由区 333，这里，自由区 333 包括自由块 600、601、602 和 603。此外，闪存 330 包括元区（meta region）334。

在操作中，例如，混合映射方法可用于在闪存卡系统中映射地址。在混合映射方法中，无论对页单元还是对块单元，映射表可以将逻辑地址与物理地址相关联。当混合映射方法用于日志块的页映射和数据块的块映射时，映射表的大小和合并操作的数量均会减小。

Samsung Ex. 1002, Page 77 of 622

例如，当执行写操作时，将要被存储在指定数据块中的页数据首先存储在指定的日志块中。如果没有自由块来用作日志块，则 FTL 执行合并操作。在合并操作中，日志块中的页数据和与日志块对应的数据块中的页数据被存储或复制到新的数据块中。在这些操作期间发生变化的映射信息被存储在元块中。一旦内容被指定到其它块，则日志块和旧的数据块可以被安全地擦除。

如图 4 所示，合并顺序总地用参考标号 400 来表示。合并顺序 400 接收数据块 100 和日志块 300，并提供合并后的数据块 101。这里，当与逻辑页编号（LPN）1 对应的数据已经存储在数据块 100 的物理页编号（PPN）2 中且主机按 LPN 2、3 和 0 的次序来请求写操作时，日志块 300 的 PPN 1、2 和 3 按顺序被写入。当 FTL 比如因缺少自由块而执行日志块 300 的合并操作时，日志块 300 的 LPN 2、3 和 0 和对应的数据块 100 的 LPN 1 被按 LPN 0、1、2 和 3 的顺序分别复制到新数据块 101 的 PPN 1、2、3 和 4。当比如对于新的日志块或自由块进行再指定时，日志块 300 和数据块 100 被擦除。

该方法使用了许多页复制操作和块擦除操作。假设一个块由 4 页组成，则每个合并操作需要四个页复制操作和两个块擦除操作。此外，该方法在日志块内具有低的页使用率。日志块 300 在通过合并被转换成自由块之前，只使用 4 页中的 3 页。用于频繁合并操作的许多复制操作和擦除操作以及低的页使用率，导致闪存卡系统的整体性能降低。

转到图 5，具有组映射表的混合映射系统总地用参考标号 500 来表示。混合映射系统 500 包括与 FTL 520 进行信号通信的主机 510，FTL 520 随后与闪存 530 进行信号通信。逻辑地址（LA）在主机 510 和 FTL 520 之间传递。物理地址（PA）在 FTL 520 和闪存 530 之间传递。FTL 520 包括数据块映射表 521、组映射表 522 和页映射表 523。例如，可以对日志块应用页映射表，而可以对数据块应用块映射表。

闪存 530 包括数据区 531，这里，数据区 531 包括具有物理块编号（PBN）100、101、102、103、104、105、206、303...、900、901、902 和 903 的数据块。这些数据块分布在数据块组之中。闪存 530 还包括日志区 532，这里，日志区 532 包括具有 PBN 300、400 和 500 的日志块。这些日志块分布在日志块组中。闪存 530 还包括自由区 533，这里，自由区 533 包括自由块 600、601、602 和 603。此外，闪存 530 包括元区 534。

在操作中，采用组映射方法，日志块组（LBG）被指定到数据块组（DBG）。

Samsung Ex. 1002, Page 78 of 622

LBG 包括多个日志块且 DBG 包括多个数据块。这里，LBG 1 由日志块 300 和 400 组成，并被指定到 DBG 1。DBG 1 由数据块 100、101、102 和 103 组成。因此，编程到数据块 100~103 的数据首先被编程到日志块 300 或 400。LBG 内的每个日志块可以被指定到被指定的 DBG 内的任意的数据块或所有的数据块。每个 DBG 或 LBG 内的块的数量是可变化的。通过组映射方法，可减少合并操作的数量，且日志块内的平均页使用率会提高。

现在转到图 6，组映射闪存转换层（FTL）总地用参考标号 600 来表示。在该实施例中，FTL 600 包括块映射表（BMT）610、组映射表（GMT）620 和页映射表（PMT）630。这里，块映射表 610 将逻辑块编号（LBN） 0~7 分别与物理块编号（PBN） 100~105、206 和 303 相关联。组映射表 620 将 DBG 1 与日志块 PBN 300 和 400 相关联，并将 DBG 2 与日志块 PBN 500 相关联。对于 DBG 1，页映射表 630 将逻辑页编号（LPN）3、4、11、12、13 和 14 分别与物理页编号（PPN）1200、1201、1202、1203、1600 和 1601 相关联。此外，对于 DBG 2，页映射表 630 将 LPN 17~20 分别与 PPN 2000~2003 相关联。

因此，LBG 1 的 PBN 300 具有分别与 LPN 3、4、11 和 12 相关联的 PPN 1200~1203。LBG 1 的 PBN 400 具有分别与 LPN 13 和 14 相关联的 PPN 1600~1601。LBG 1 与 DBG 1 相关联。DBG 1 的 PBN 100 具有与 LPN 0~3 相关联的 PPN 1~4。DBG 1 的 PBN 101 具有与 LPN 4~7 相关联的 PPN 1~4。DBG 1 的 PBN 102 具有与 LPN 8~11 相关联的 PPN 1~4。DBG 1 的 PBN 103 具有与 LPN 12~15 相关联的 PPN 1~4。

此外，LBG 2 的 PBN 500 具有分别与 LPN 17~20 相关联的 PPN 2000~2003。LBG 2 与 DBG 2 相关联。DBG2 的 PBN 104 具有与 LPN 16~19 相关联的 PPN 1~4。DBG 2 的 PBN 105 具有与 LPN 20~23 相关联的 PPN 1~4。DBG 2 的 PBN 206 具有与 LPN 24~27 相关联的 PPN 1~4。DBG 2 的 PBN 303 具有与 LPN 28~31 相关联的 PPN 1~4。

因此，块映射表 610 将逻辑块编号（LBN）转换成物理块编号（PBN）。组映射表 620 具有每个 DBG 与对应于 DBG 的日志块的 PBN 之间的映射信息。页映射表 630 具有每个 DBG、LPN 和对应的 PPN 之间的映射信息。

在该示例性的组映射实施例的操作中，从主机接收 LPN 3 和 4 的写请求。FTL 从自由块中指定日志块 300，并产生与 DBG 1 对应的 LBG 1。FTL 将日

Samsung Ex. 1002, Page 79 of 622

志块 300 登记在组映射表 620 中。日志块 300 可以被指定到 DBG 1 内所有的数据块。这里，LPN 0~15 可被存储在日志块 300 中。LPN 3 和 4 被存储在日志块 300 的 PPN 1200 和 1201 中。映射信息被存储在页映射表 630 中。

接着，从主机接收 LPN 11~14 的写请求。LPN 11 和 12 被存储在先前指定的日志块 300 的 PPN 1202 和 1203 中。根据这个新的映射信息来更新页映射表 630。

另外地将新的日志块 400 指定给 LBG 1，以存储 LPN 13 和 14。FTL 将日志块 400 登记在组映射表 620 中。LPN 13 和 14 被存储在日志块 400 的 PPN 1600 和 1601 中，且 PMT 630 被更新。写请求从主机到达 LPN 17~20。

因为 LPN 17~20 不属于 DBG 1，所以它们不能被存储在日志块 400 中。FTL 从自由块中指定新的日志块 500，并产生与 DBG 2 对应的 LBG 2。FTL 将日志块 500 登记在组映射表 620 中。LPN 16~31 可以被存储在日志块 500 中。LPN 17~20 被存储在日志块 500 的 PPN 2000~2003 中，且 PMT 630 被更新。

通过利用组映射方法，日志块的页使用率提高。这与例如日志块 300 只会使用一页（比如 PPN 1200）的传统方法形成对比。在本实施例中，因为一个日志块被指定到数据块组内的所有数据块，所以日志块 300 使用所有的页（比如 PPN 1200~1203）。此外，对应于日志块的页使用率的提高，合并操作的数量减少。此外，复制操作和擦除操作的数量减少，系统的性能提高。

如图 7 所示，逻辑闪存结构总地用参考标号 700 来表示。闪存 700 包括元数据区 710 和用户数据区 720，其中，元数据区 710 对于用户来说是不可见的，用户数据区 720 对于用户来说是可见的。元数据区 710 包括固件块 712、保留块 714、MAP 块 716 和写缓冲块 718，其中，保留块 714 可用于替换坏的块。MAP 块 716 包括块映射表 717，该块映射表 717 将逻辑块编号 0、1...、31 分别映射到物理块编号 3、15...、0。多个写缓冲块 718 包括写缓冲块 719，这里，写缓冲块 719 存储逻辑页编号 1 和 3 的替代数据（即 1'和 3'）。

用户数据区 720 包括数据块 722~726。数据块 725 包括逻辑页编号 0、1、2 和 3。在操作中，合并操作将数据块 725 与写缓冲块 719 组合，以形成新的数据块 726。这里，新的数据块 726 包括逻辑页编号 0、1'、2 和 3'。

转到图 8，页映射结构总地用参考标号 800 来表示。页映射结构 800 包括页映射表 810 和闪存 870。页映射表 810 包括逻辑页编号 830 和对应的物

Samsung Ex. 1002, Page 80 of 622

理页编号 850。逻辑页编号 832 用于查询对应的物理页编号。

在操作中，逻辑页编号（LPN）1 初始地对应于物理页编号（PPN）2。新的 PPN 5 被写入闪存 870 中，且页映射表 810 被更新以将 LPN 1 与 LPN 5 相关联。这可以被称作“任意位置（out-of-place）映射”。更新的页可以被写入新块中的不同位置。页映射表更新使用了如上的相对大的映射表大小。例如，128KB 的映射表用于 128MB 的 NAND 闪存。

现在转到图 9，块映射结构总地用参考标号 900 来表示。块映射结构 900 包括块映射表 920 和闪存 970。块映射表 920 包括逻辑块编号 940 和对应的物理块编号 960。逻辑块编号 942 用于查询对应的物理块编号。物理页编号 962 是加到物理块编号的偏移量。

在操作中，在闪存 970 中，物理页编号（PPN）2 被更新为 PPN 2'。这可被称作“顺序位置（in-place） 映射”。这里，更新的页被写入新块的相同位置，且块映射表被更新。当任意位置页更新导致块复制操作时在写操作期间存在上述复制。

如图 10 所示，混合映射结构总地用参考标号 1000 来表示。混合映射结构 1000 包括页映射表 1010、块映射表 1020、日志块 1070 和数据块 1080。

页映射表 1010 包括逻辑页编号 1030 和对应的物理页编号 1050。块映射表 1020 包括逻辑块编号 1040 和对应的物理块编号 1060。逻辑块编号 1032 用于查询对应的物理块编号。

在操作中，逻辑块编号 1032 用于在块映射表 1020 中查询对应的物理块编号，或者，逻辑页编号 1052 用于在页映射表 1010 中查询对应的物理页编号。

例如，页映射可用于写缓存或日志块，而块映射可用于数据块。当逻辑页编号 1 初始地与日志块 1070 中的物理页编号 2 对应时，页映射表 1010 可被更新，以使逻辑页编号 1 变化成与物理页编号 4 对应。当日志块 1070 用于写入数据块 1080 时，块映射表 1020 可被更新。

转到图 11，合并方法总地用参考标号 1100 来表示。这里，日志入口（entry）信息 1110 用于更新日志块 1120。随后，日志块 1120 与数据块 1130 合并。合并后的数据被写入自由块 1140，由此，自由块 1140 变成新的数据块。因此，合并方法 1100 指定自由块，并将有效页复制到被指定的自由块，通过将被指定的自由块设置为数据块来更新映射页，并将旧的日志块和旧的数据块设置

Samsung Ex. 1002, Page 81 of 622

为可擦除状态。

现在转到图 12，复制-合并方法总地用参考标号 1200 来表示。这里，日志入口信息 1210 用于更新日志块 1220 的页。数据块 1230 的未变化的页被复制到日志块 1220。因此，复制-合并方法 1200 将数据块中的有效页复制到日志块，通过将旧的日志块设置为数据块来更新映射页，并将旧的数据块设置为可擦除状态。

如图 13 所示，交换-合并方法总地用参考标号 1300 来表示。这里，日志入口信息 1310 被用于更新日志块 1320 中的所有页。通过将旧的日志块设置为数据块来更新映射页，并将旧的数据块设置为可擦除状态。

转到图 14，1:1 的映射结构总地用参考标号 1400 来表示。1:1 的映射结构 1400 包括单元映射表 1410，该单元映射表 1410 具有与物理单元编号部分或列 1430 对应的逻辑单元编号部分或列 1420。物理单元编号对应于闪存内的包括传送单元 1450 的物理单元 1440。例如，这些单元可以是块。即，每个映射表入口可以与一个物理单元对应。

现在转到图 15，1:2 映射结构总地用参考标号 1500 来表示。1:2 映射结构 1500 包括页映射表 1510 和块映射表 1520。页映射表 1510 具有与物理页编号部分或列 1550 对应的逻辑页编号部分或列 1530。块映射表 1520 具有与物理块编号部分或列 1560 对应的逻辑块编号部分或列 1540。

物理块编号和物理页编号对应于闪存或物理存储装置内的物理块 1570~1573 和它们的页。这里，来自块映射表 1520 的物理块编号（例如 101）指向物理块 1570，来自页映射表 1510 的物理页编号（比如 1、0、2 和 3）指向块 1570 内的物理页。直接被标识在映射表中的块和页是原始单元，每个原始单元可具有与其相关联的一个附加的日志单元，用于记录更新。因此，原始块 1570 和 1571 可以分别与日志块 1572 和 1573 相关联。即，每个映射表入口可以与一个或两个物理单元对应。

如图 16 所示，1:N 映射结构总地用参考标号 1600 来表示。1:N 结构 1600 包括页映射表 1610 和块映射表 1620。页映射表 1610 具有与物理页编号部分或列 1650 对应的逻辑页编号部分或列 1630。块映射表 1620 具有与物理块编号部分或列 1660 对应的逻辑块编号部分或列 1640。

物理块编号和物理页编号对应于物理存储装置或闪存内的物理块 1670~1675 和它们的页。这里，来自块映射表 1620 的物理块编号（比如 101）

Samsung Ex. 1002, Page 82 of 622

指向物理块 1670，来自页映射表 1610 的物理页编号（比如 1、0、2 和 3）指向物理块 1670 内的物理页。直接被标识在映射表中的块和页是原始单元，每个原始单元可以具有与之相关联的 N 个附加的日志单元，用于记录更新。因此，原始块 1671 可以与日志块 1673、1675...相关联。即，每个映射表入口可以与 1~N 个物理单元对应。1:N 映射结构 1600 利用数据块与日志块的延迟的合并。

转到图 17，N:M+K 映射结构总地用参考标号 1700 来表示。N:M+K 映射结构 1700 包括页映射表 1710 和块映射表 1720。页映射表 1710 具有与物理页编号部分或列 1750 对应的逻辑页编号部分或列 1730。块映射表 1720 具有与物理块编号部分或列 1760 对应的逻辑块编号部分或列 1740。

物理块编号和物理页编号对应于闪存或物理存储装置内的物理块 1769~1775 和它们的页。这里，来自块映射表 1720 的物理块编号（比如 101）指向物理块 1770，来自页映射表 1710 的物理页编号（比如 1、0、2 和 3）指向块 1770 内的物理页。直接在映射表中标识的块和页是原始单元，且每个原始单元可以与 K 个附加的成链状或成组的日志单元相关联，以记录更新。附加的成链状或成组的日志单元可以被 N 个原始单元所共用。此外，会存在 M 个不同的成链状或成组的附加日志单元。因此，原始块 1771 和 1772 均可与成链状的或成组的日志块 1783 和 1785 相关联，这里，原始块 1771 和 1772 分别具有物理块编号 0 和 1。此外，例如，原始块 1769 和 1770 均可与日志块 1780 相关联，这里，原始块 1769 和 1770 分别具有物理块编号 101 和 102。N:M+K 映射结构 1700 利用数据块之间的日志块的关联性以及日志块与数据块的延迟的合并。这里，N 是所有的用户数据块的数量，M 是写缓存或日志块的数量，K 是日志块组（LBG）中的延迟的合并或日志块的最大数量，其中，LBG 可以被动态地控制。数据块组（DBG）与每个 LBG 相关联。在该示例中，每个 DBG 包括两个数据块，但是在可选的实施例中，DBG 可包括任意数量的数据块。

存储“块”通常是擦除单元的大小，存储“页”通常是读/写单元的大小。相关领域的普通技术人员应该理解的是，可选的实施例可以使用可选的存储单元、块和/或页大小，这些大小都不限于示例性实施例中描述的那些。任意大小的其它单元或扇区（sector）可以用来替代这里描述的单元、块和/或页。例如，在混合映射实施例中，可优选地是，利用用于写缓存或日志单元的页

Samsung Ex. 1002, Page 83 of 622

映射，并利用用于数据单元的块映射，但是可选的实施例可利用两个或更多个可选大小的映射单元。

虽然在此已经参照附图描述了示例性实施例，但是应该理解，本公开不限于那些精确的实施例，并且在不脱离本公开的范围或精神的情况下，相关领域的普通技术人员可以实现对实施例的各种其它变化和更改。所有这些变化和更改意在被包括在如权利要求中阐述的本公开的范围内。

Samsung Ex. 1002, Page 84 of 622

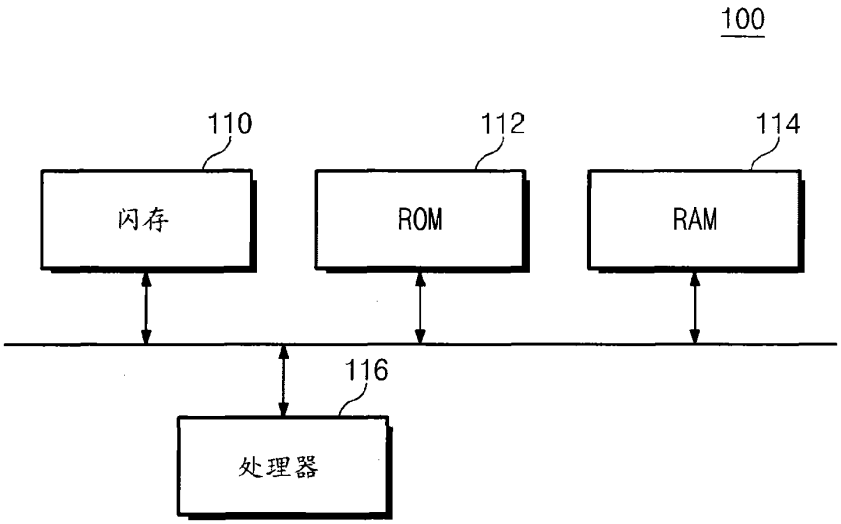


图1

Samsung Ex. 1002, Page 85 of 622

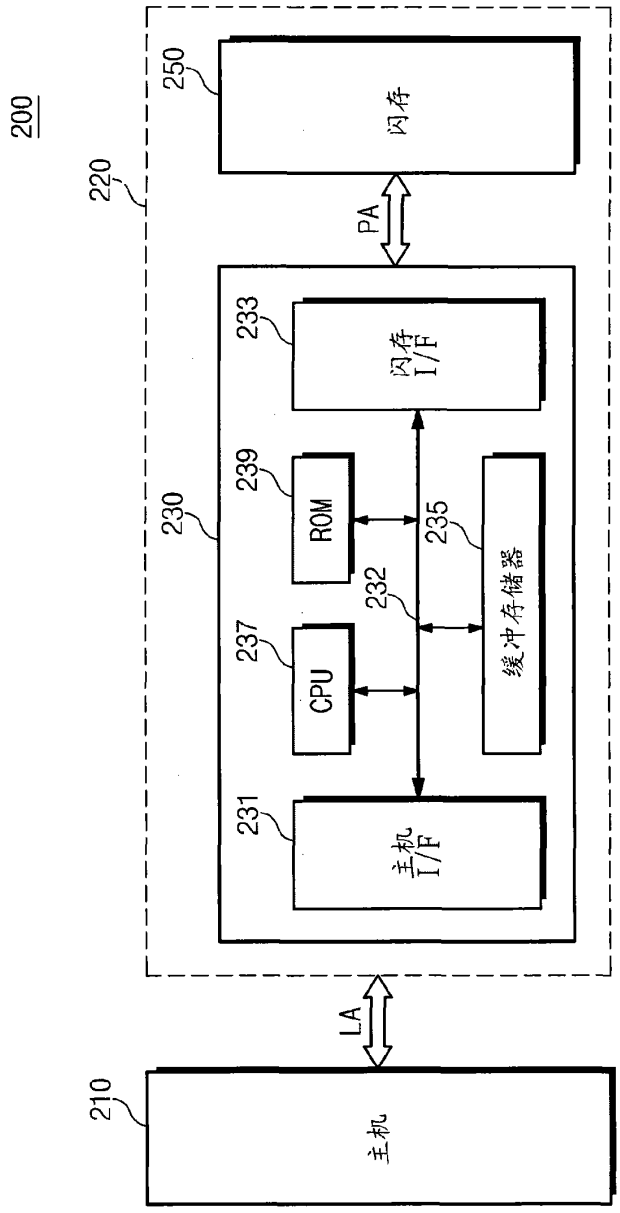


图 2

Samsung Ex. 1002, Page 86 of 622

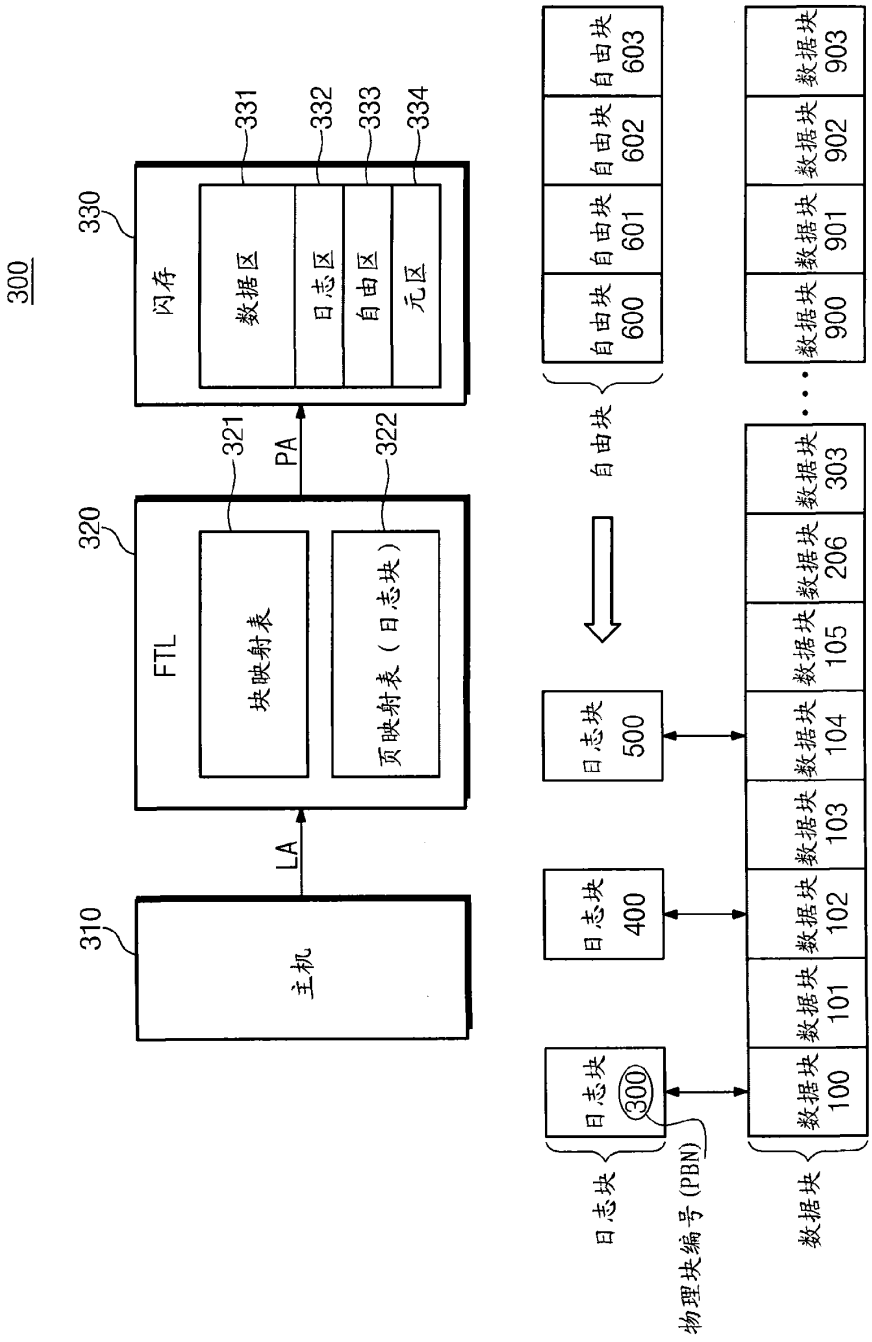


图3

Samsung Ex. 1002, Page 87 of 622

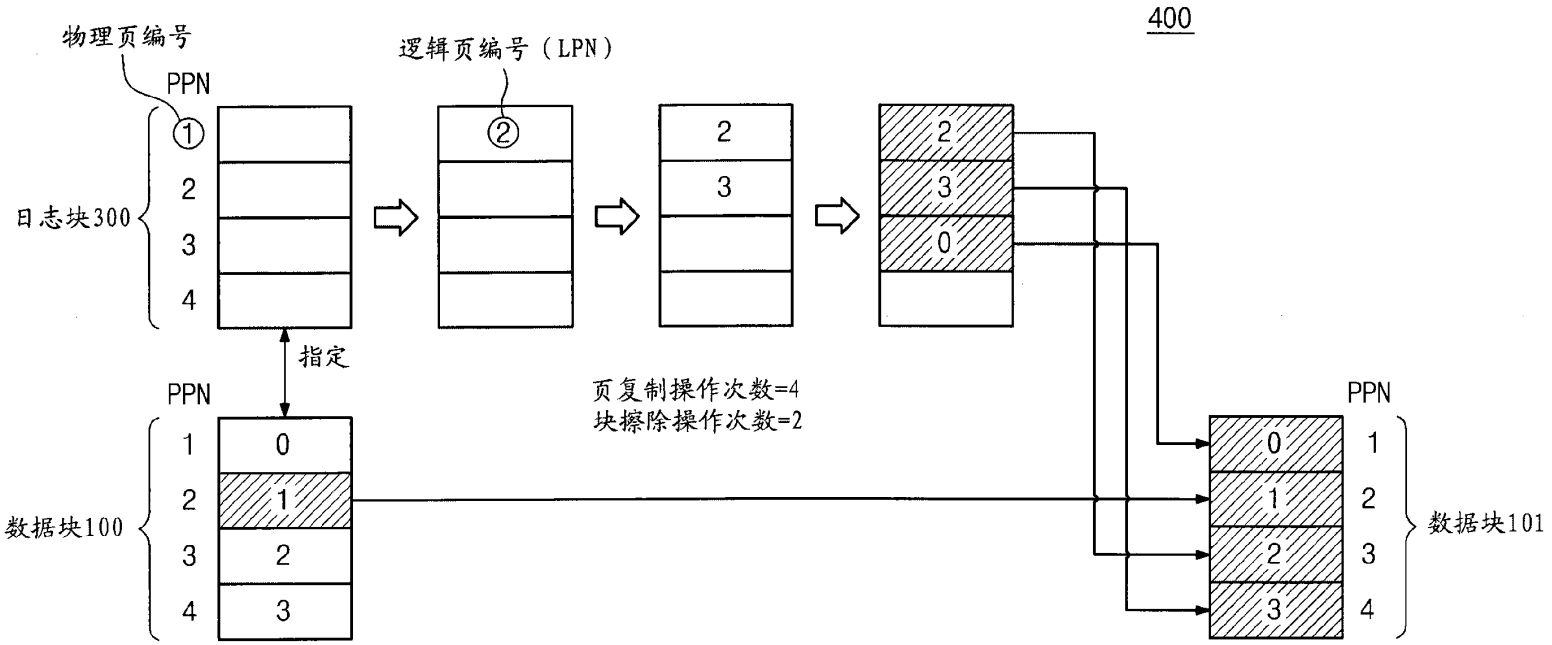


图4

Samsung Ex. 1002, Page 88 of 622

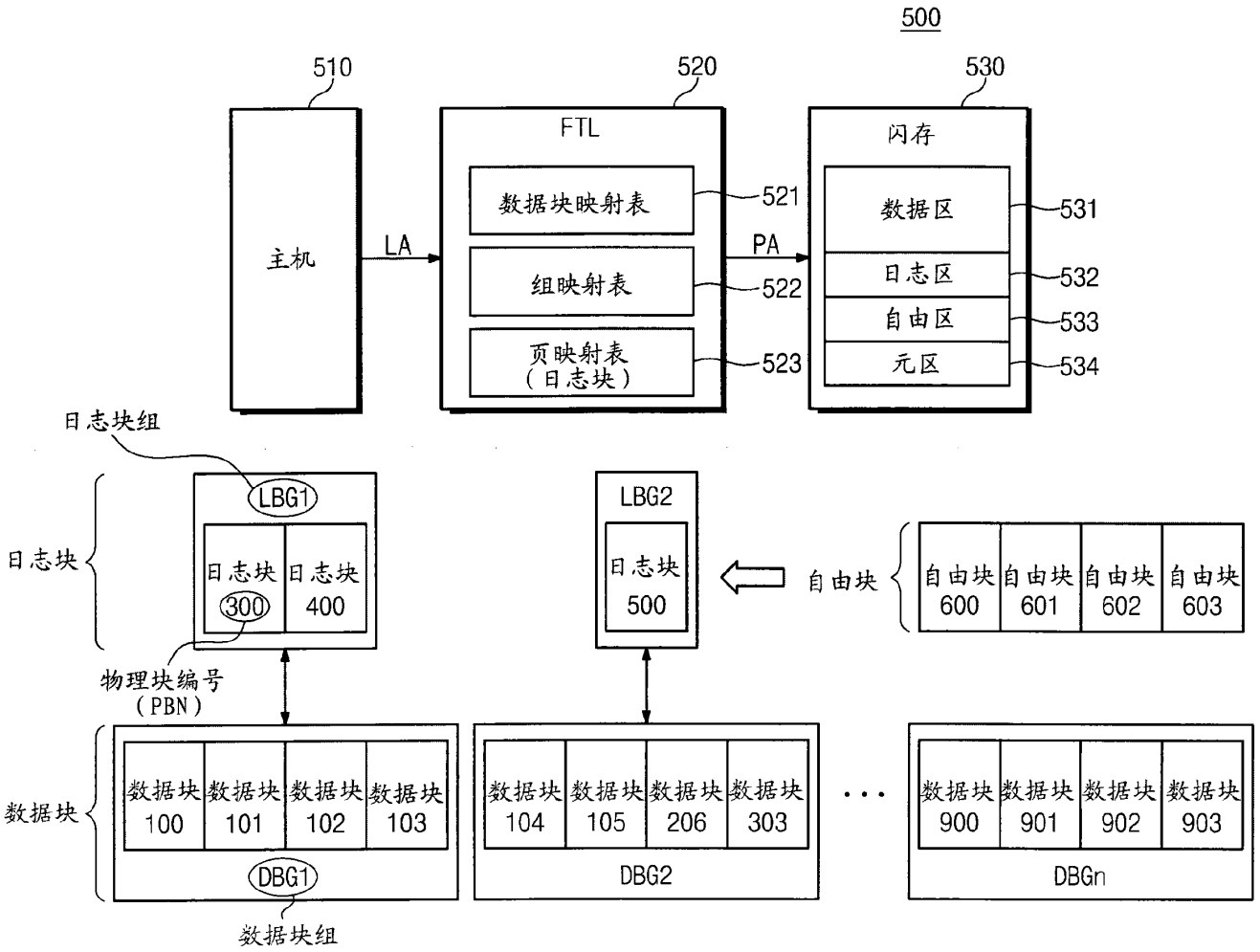


图5

Samsung Ex. 1002, Page 89 of 622

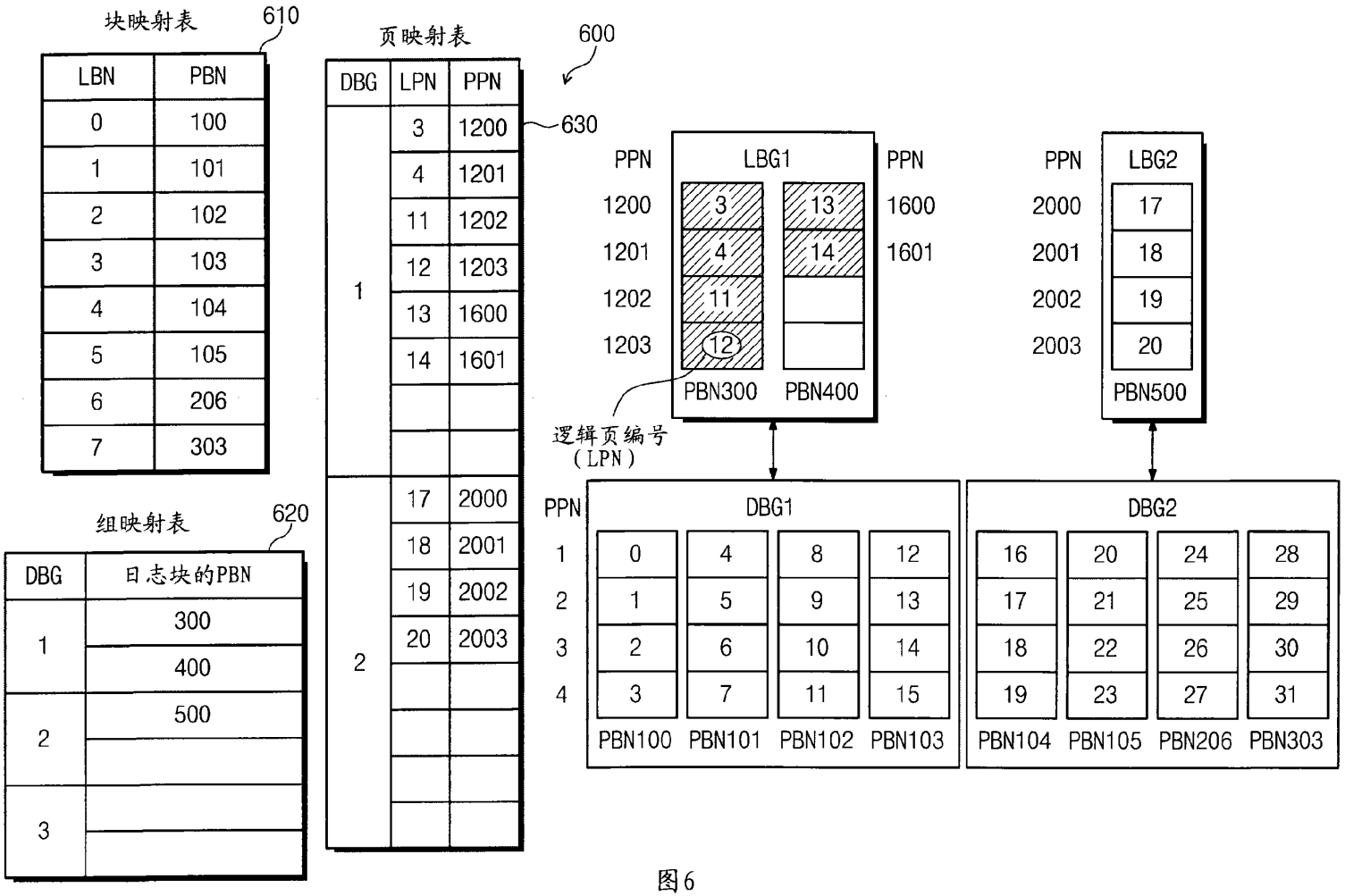


图6

Samsung Ex. 1002, Page 90 of 622

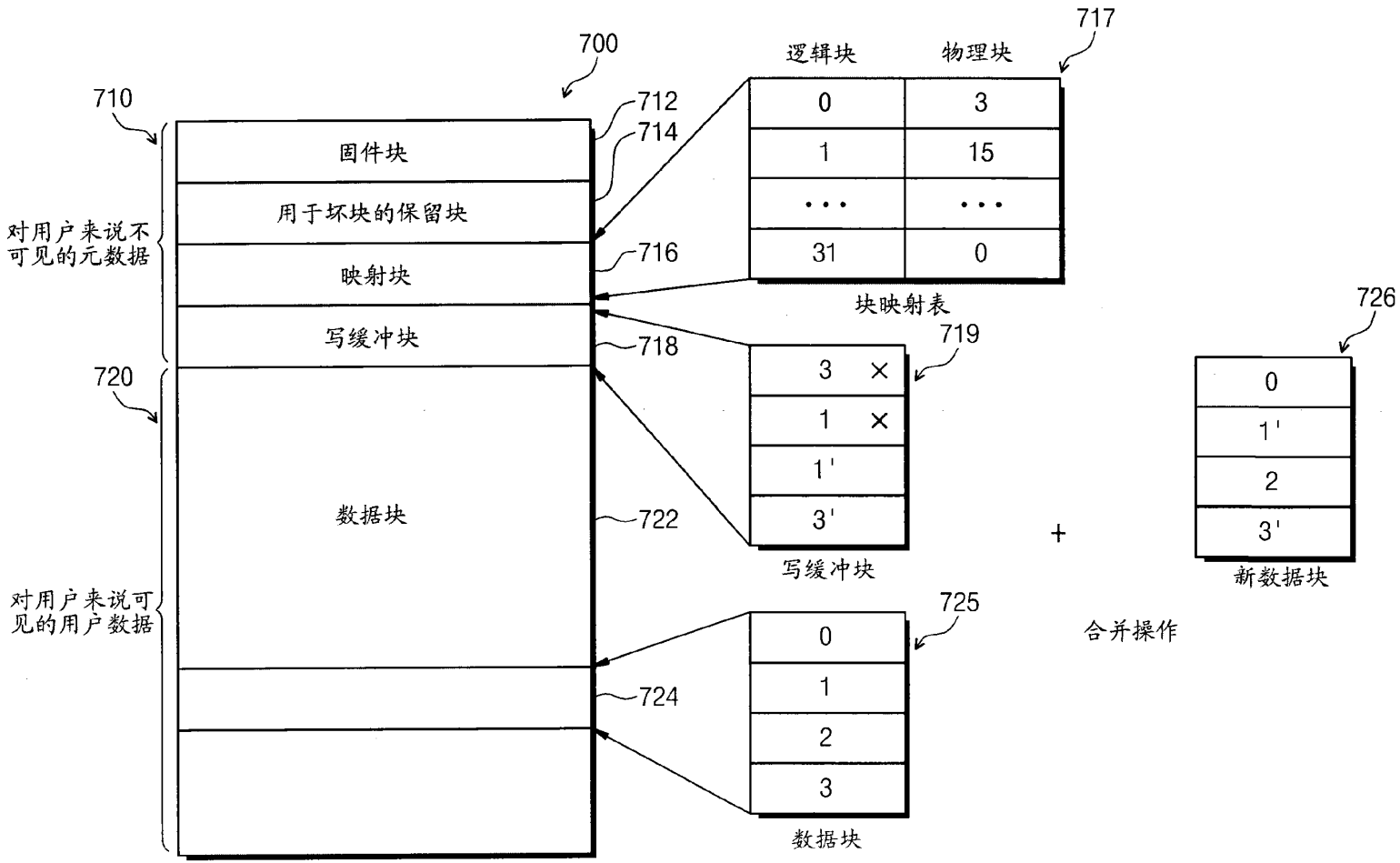


图7

Samsung Ex. 1002, Page 91 of 622

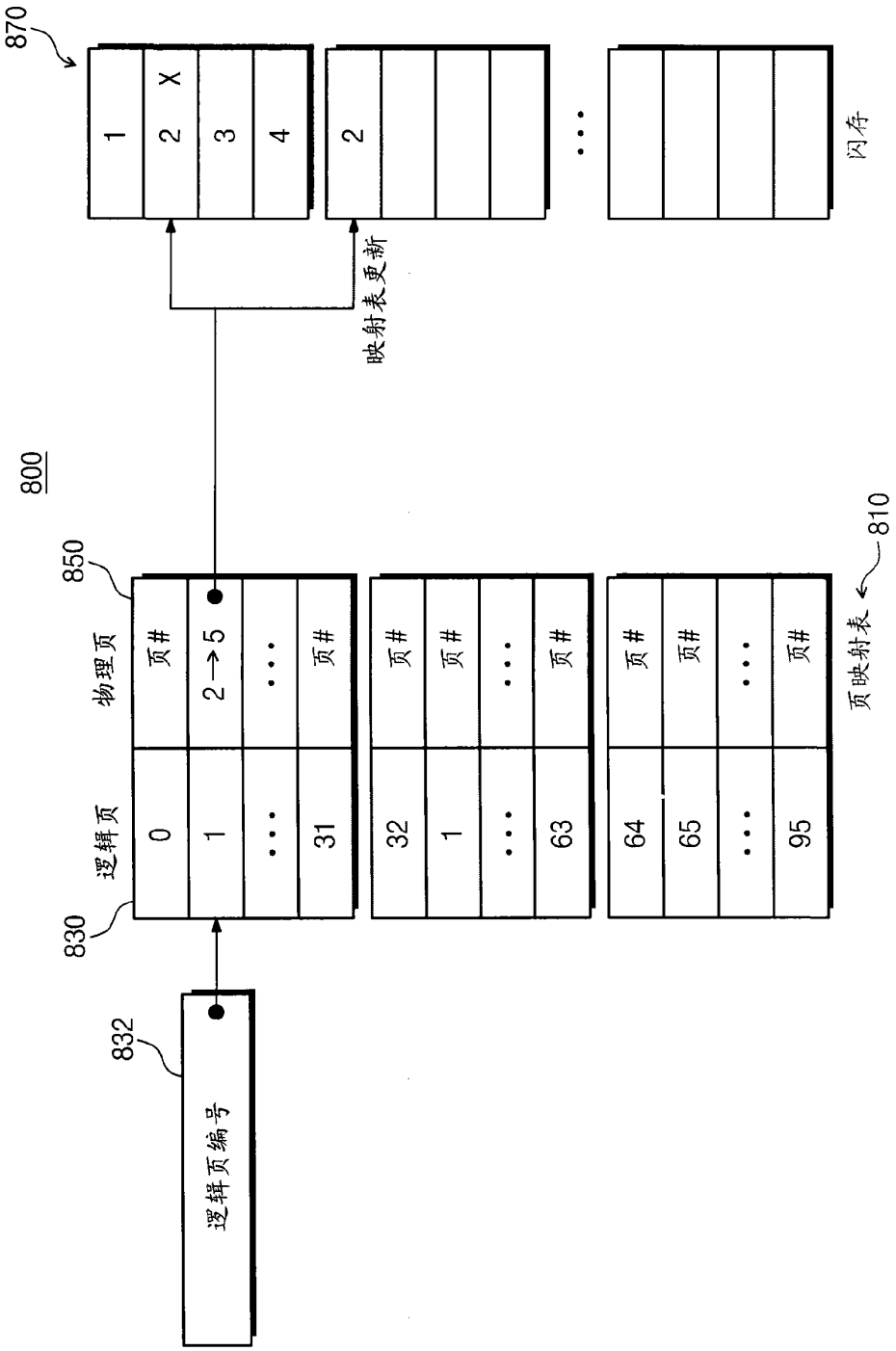


图8

Samsung Ex. 1002, Page 92 of 622

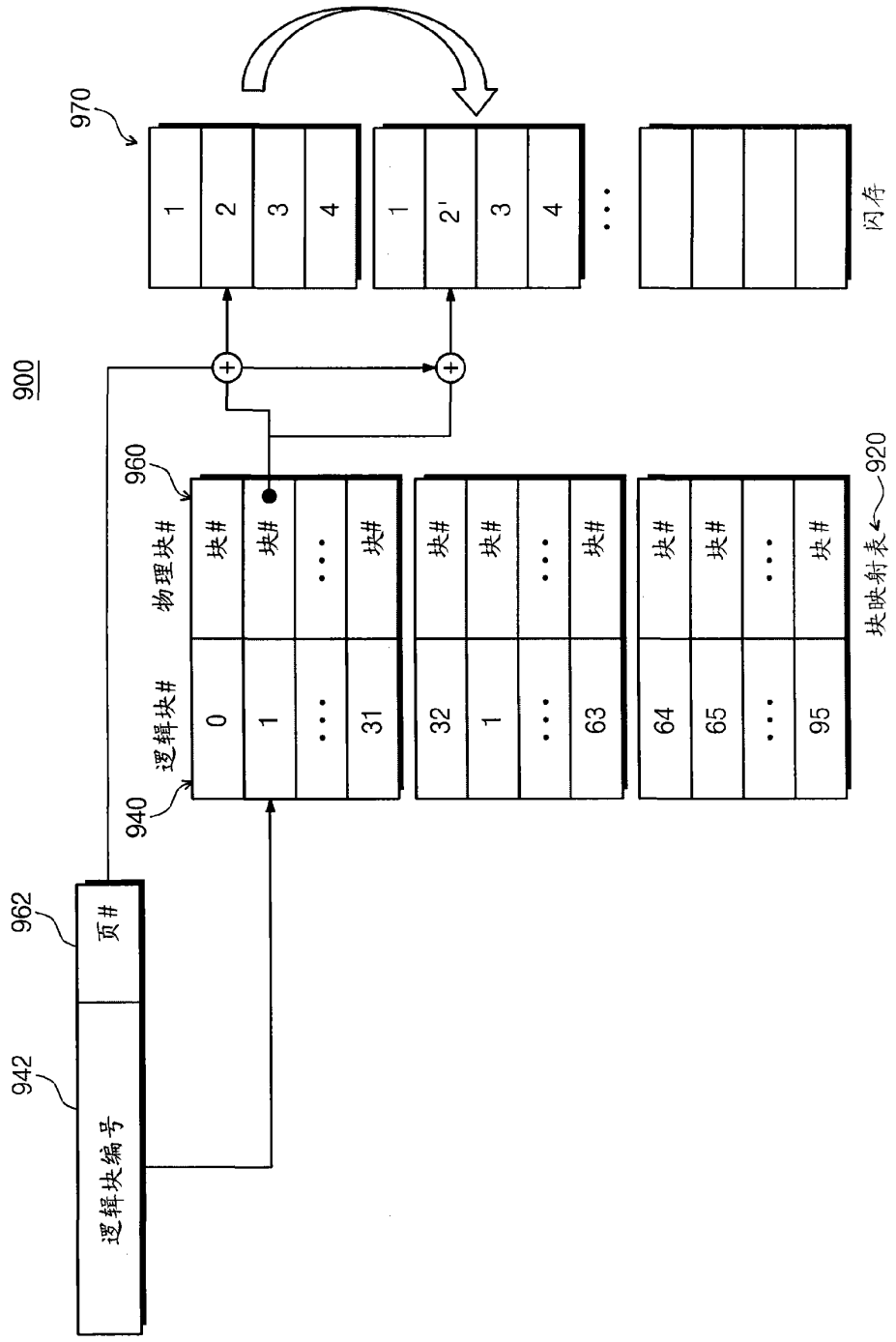


图9

Samsung Ex. 1002, Page 93 of 622

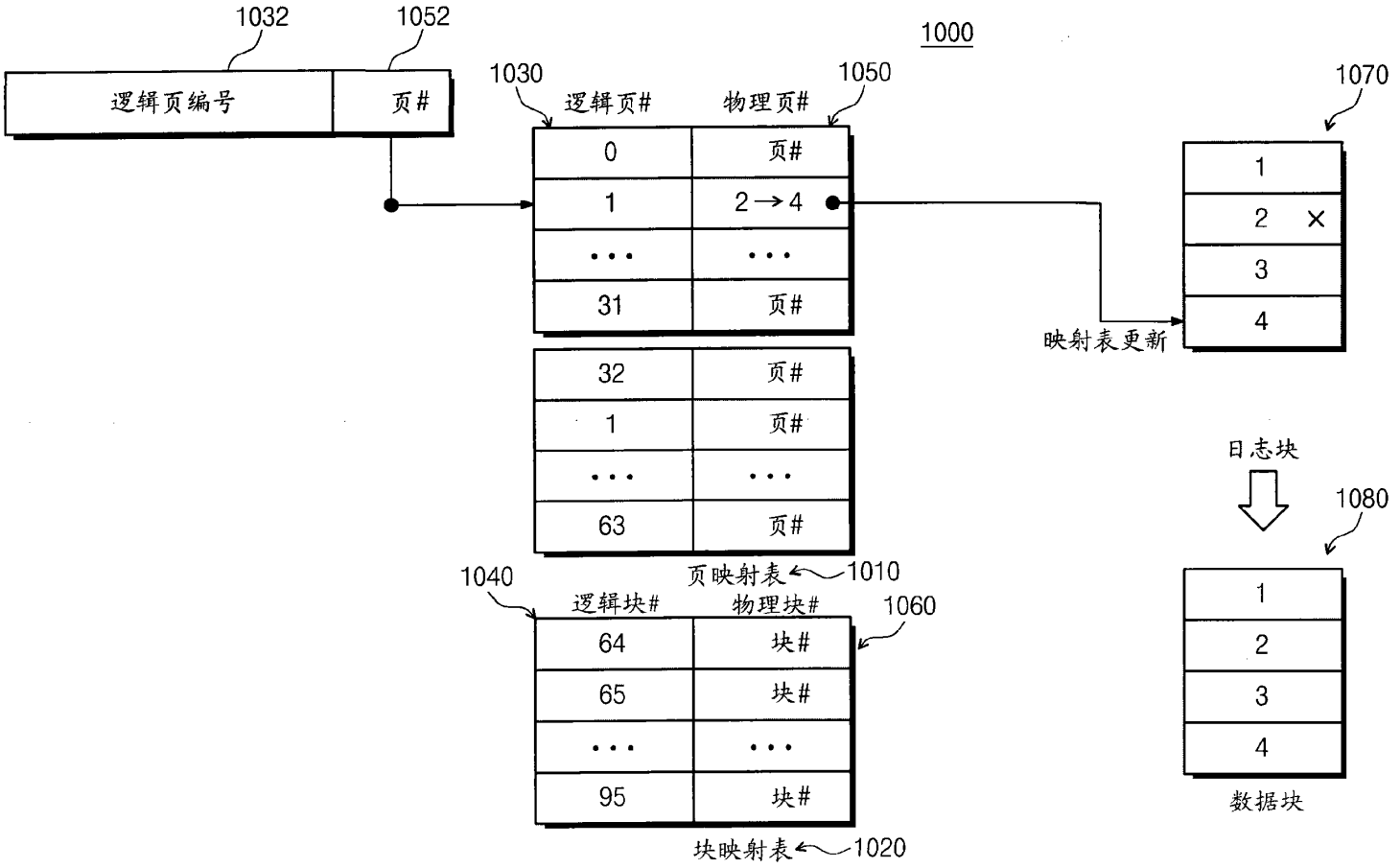


图10

Samsung Ex. 1002, Page 94 of 622

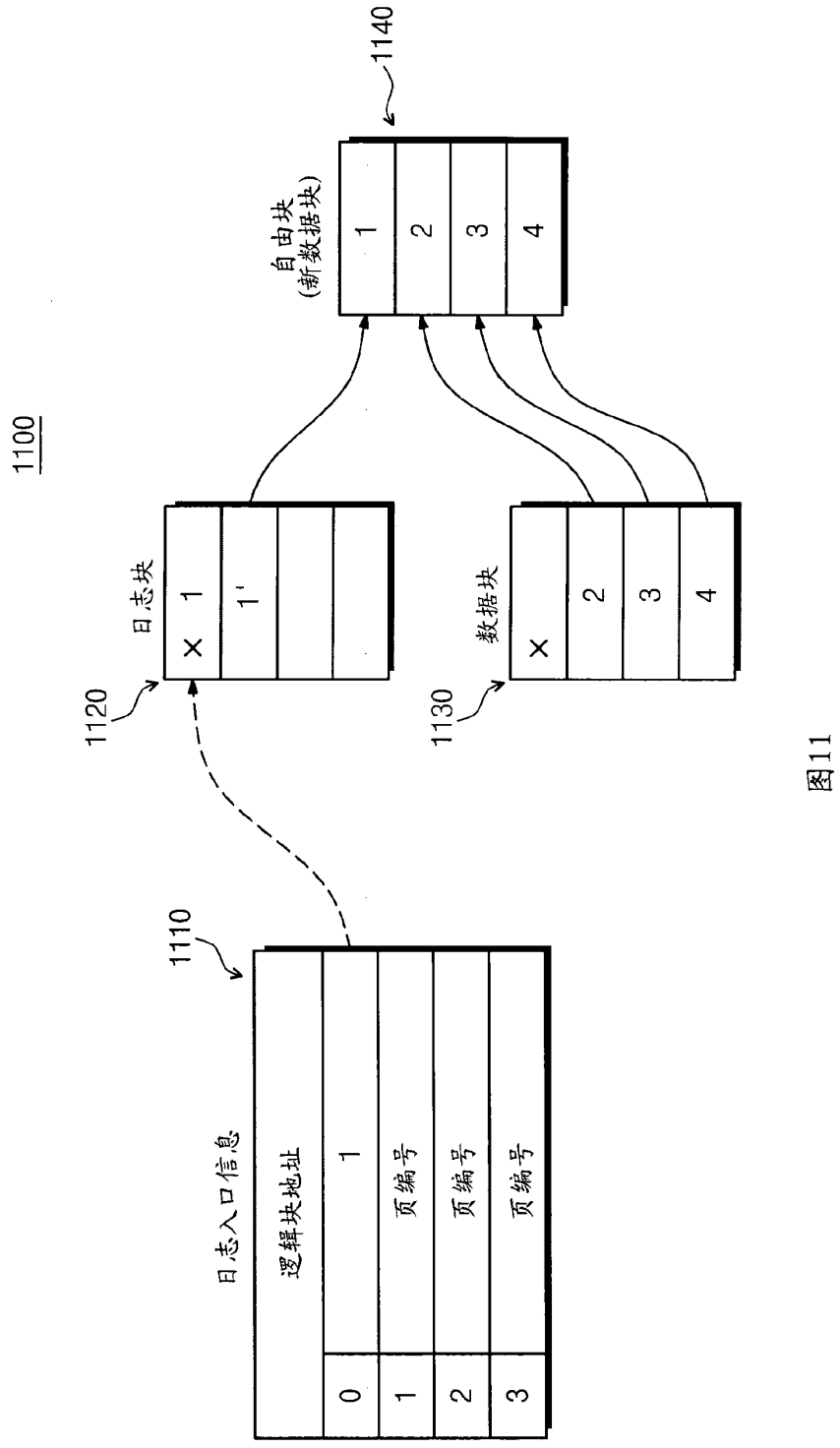


图11

Samsung Ex. 1002, Page 95 of 622

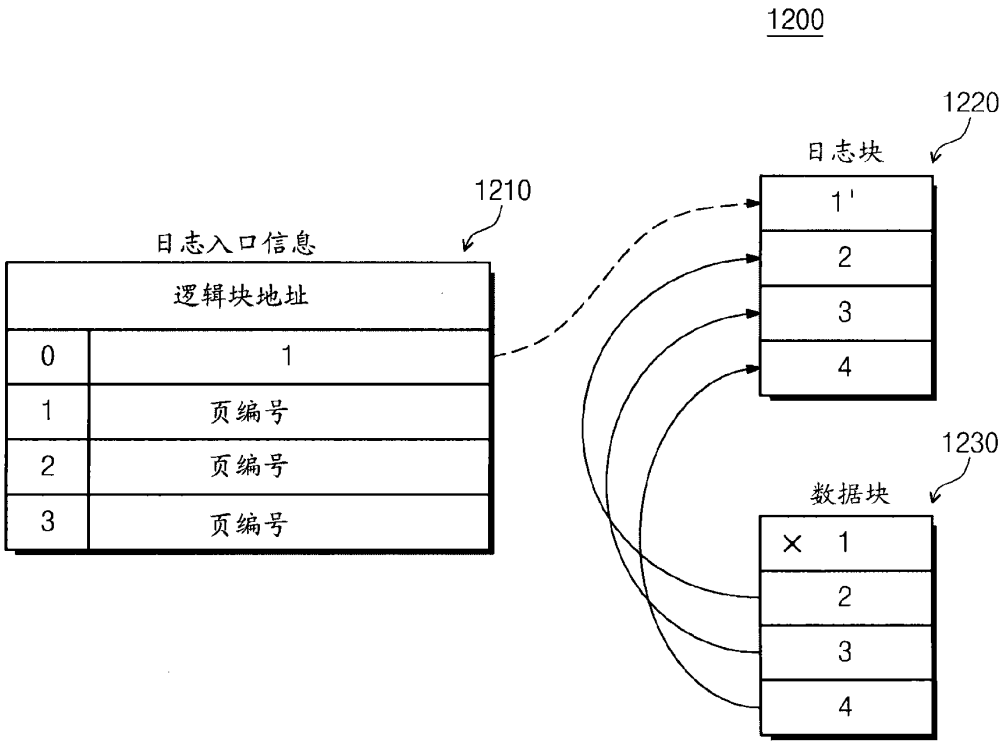


图12

Samsung Ex. 1002, Page 96 of 622

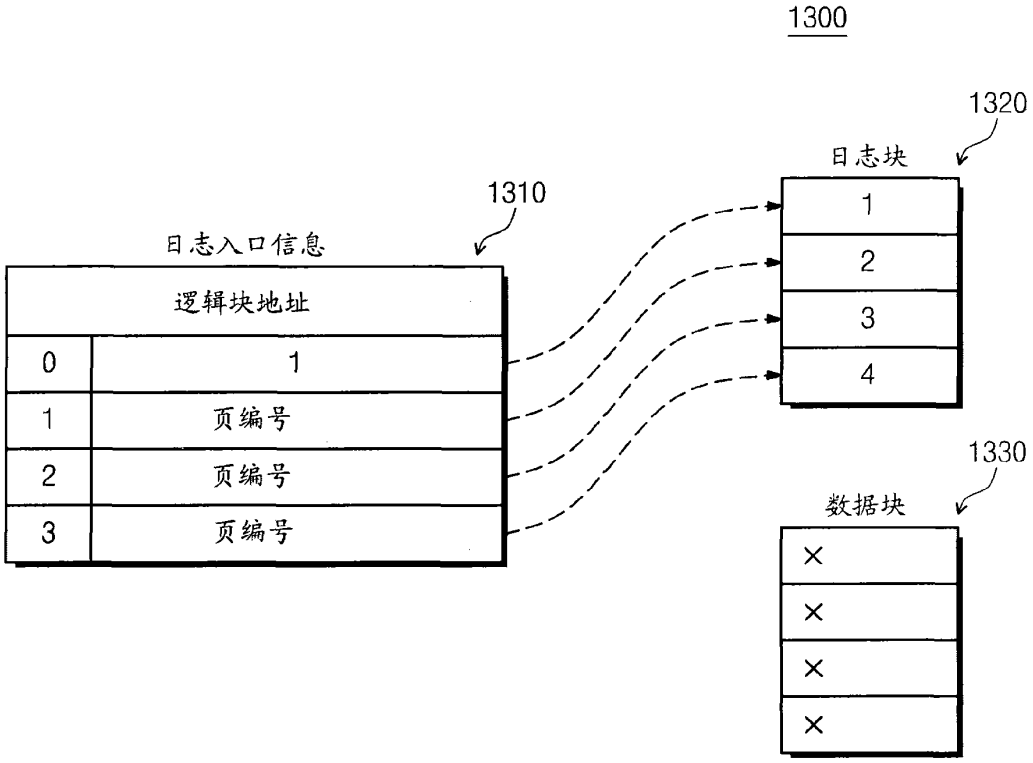


图13

Samsung Ex. 1002, Page 97 of 622

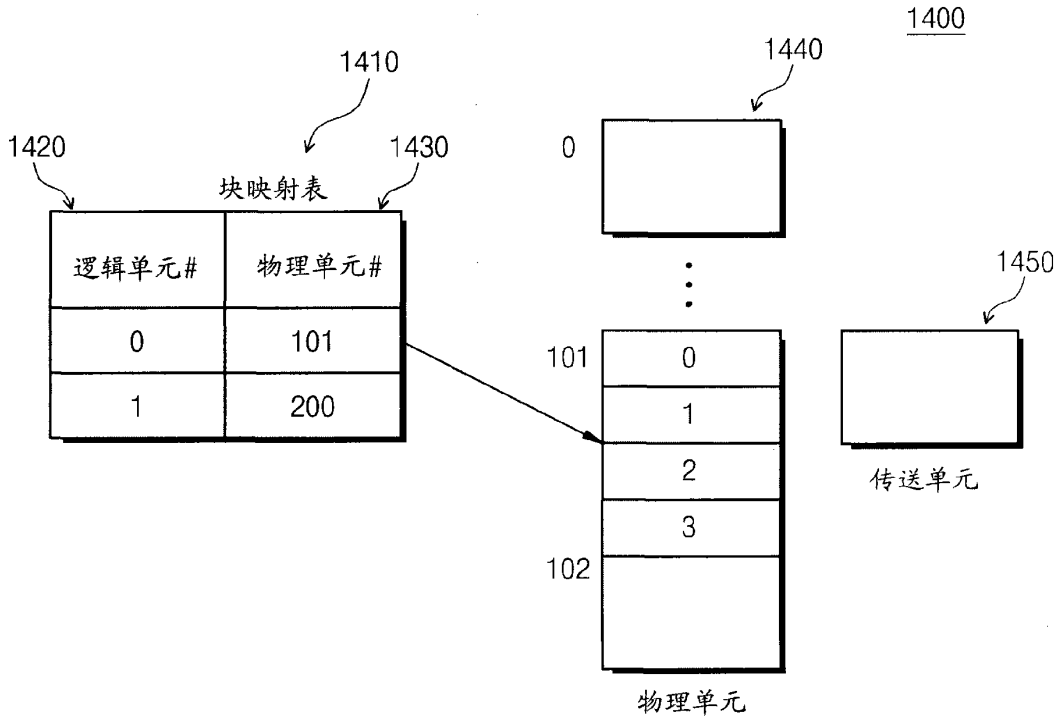


图14

Samsung Ex. 1002, Page 98 of 622

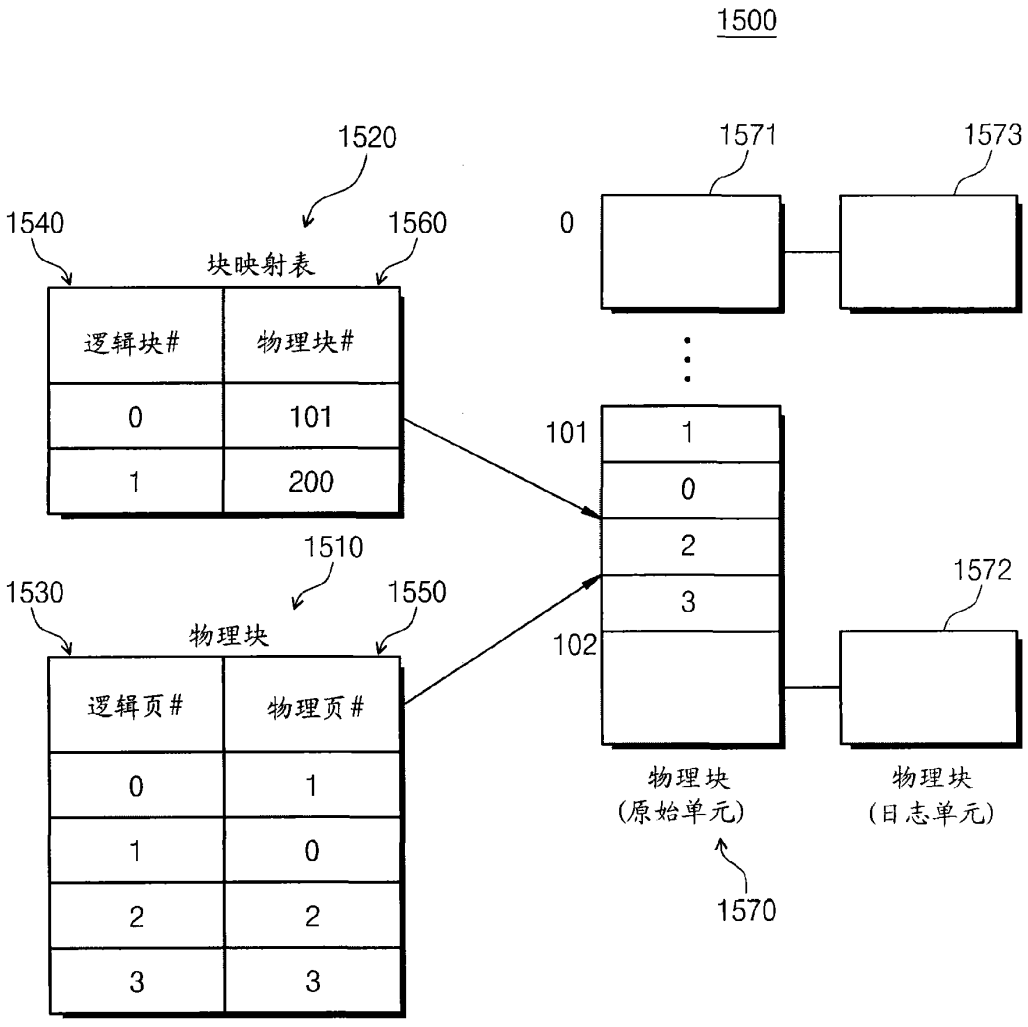


图15

Samsung Ex. 1002, Page 99 of 622

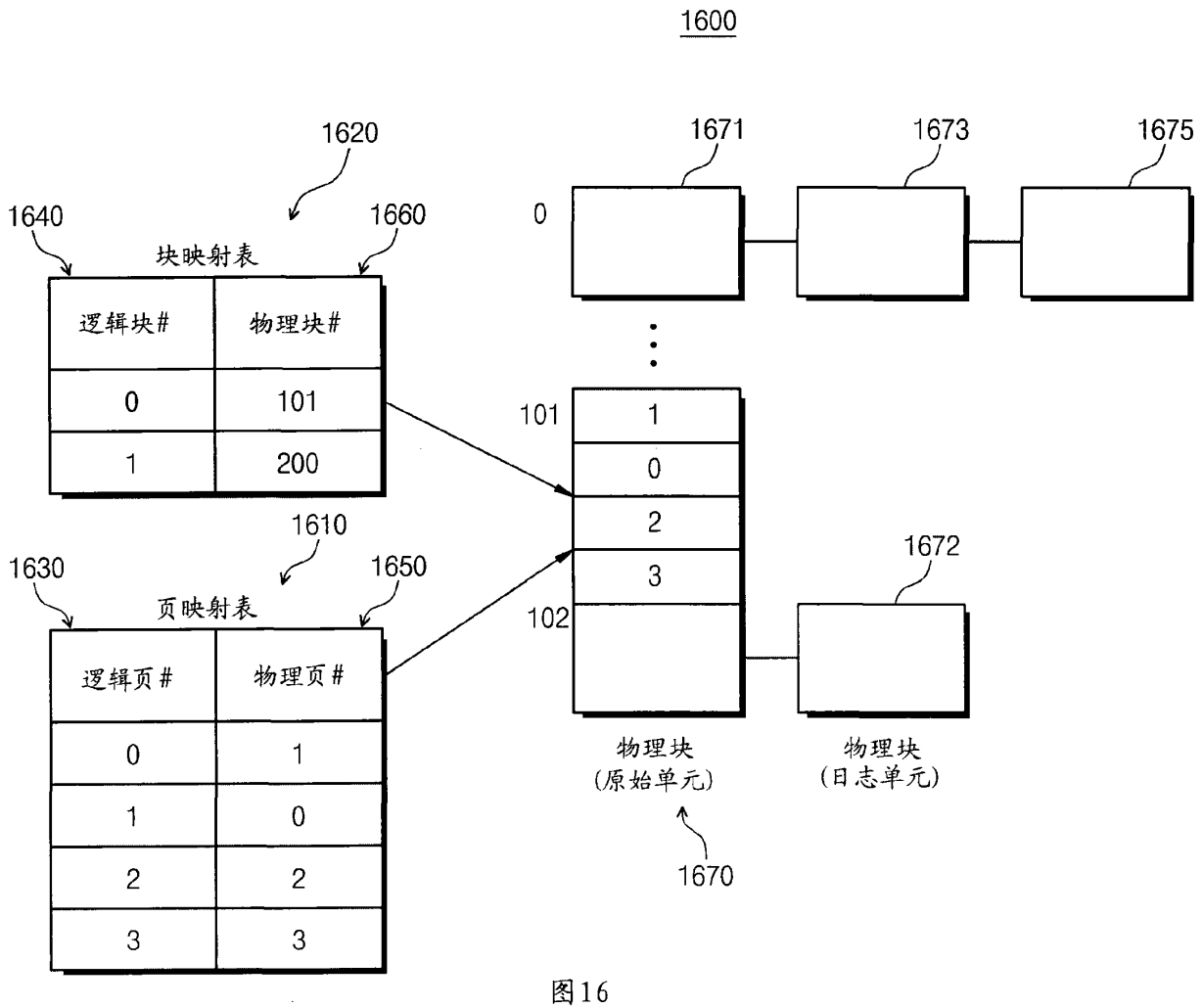


图16

Samsung Ex. 1002, Page 100 of 622

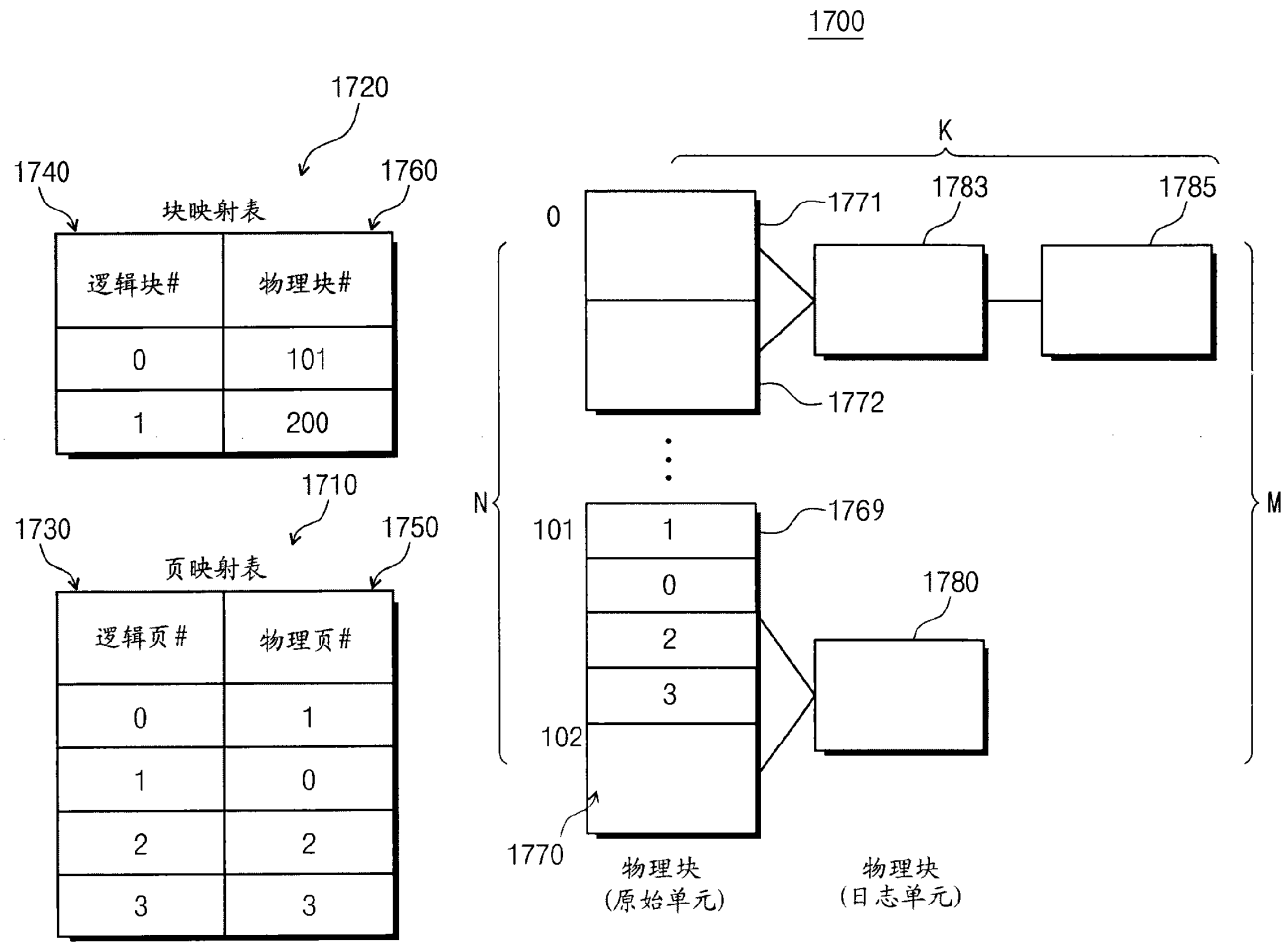

1700
1720
1740
1760
块映射表
逻辑块#
物理块#
0
101
1
200
1710
1730
1750
页映射表
逻辑页#
物理页#
0
1
1
0
2
2
3
3
K
0
1771
1783
1785
1772
N
M
101
1
1769
0
1780
2
3
102
1770
物理块
(原始单元)
物理块
(日志单元)

图17

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

73091 7590 08/23/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	

ART UNIT	PAPER NUMBER
2136	

MAIL DATE	DELIVERY MODE
08/23/2022	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Applicant-Initiated Interview Summary	**Application No.** 16/997,471	**Applicant(s)** Kuzmin et al.		
	Examiner CHRISTOPHER D BIRKHIMER	**Art Unit** 2136	**AIA (First Inventor to File) Status** No	**Page** 1 of 1

All Participants (applicant, applicants representative, PTO personnel)	**Title**	**Type**
CHRISTOPHER D BIRKHIMER	Primary Examiner	Telephonic
Robert Lercari	Claim developer	
Tim Kang (Reg. 46423)	Attorney of Record	
Mike Jadon	CEO	

Date of Interview: 19 August 2022

Issues Discussed:

35 U.S.C. 103

Discussed the inventive concept of the claims compared to the prior art. Suggested amendments that indicate the page utilization information of a subdivision is sent to the host. Discussed differences between where the erase operation is sent from in the instant application and the prior art references. Discussed claim amendments change the scope of the claims requiring further search and consideration of the prior art.

☑ Attachment

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

Applicant is reminded that a complete written statement as to the substance of the interview must be made of record in the application file. It is the applicants responsibility to provide the written statement, unless the interview was initiated by the Examiner and the Examiner has indicated that a written summary will be provided. See MPEP 713.04
Please further see:
MPEP 713.04
Title 37 Code of Federal Regulations (CFR) § 1.133 Interviews, paragraph (b)
37 CFR § 1.2 Business to be transacted in writing

Applicant recordation instructions: The formal written reply to the last Office action must include the substance of the interview. (See MPEP section 713.04). If a reply to the last Office action has already been filed, applicant is given a non-extendable period of the longer of one month or thirty days from this interview date, or the mailing date of this interview summary form, whichever is later, to file a statement of the substance of the interview.

Examiner recordation instructions: Examiners must summarize the substance of any interview of record. A complete and proper recordation of the substance of an interview should include the items listed in MPEP 713.04 for complete and proper recordation including the identification of the general thrust of each argument or issue discussed, a general indication of any other pertinent matters discussed regarding patentability and the general results or outcome of the interview, to include an indication as to whether or not agreement was reached on the issues raised.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application Serial Number:	16/997,471
Confirmation Number:	1070
First Named Inventor:	Andrey V. Kuzmin
Title:	Nonvolatile Memory Controller Host-Issued Address Delimited Erasure and Memory Controller Remapping of Host-Address Space for Bad Blocks
Filing Date:	08/19/2020
Attorney Docket:	2020058 / RMS-02C1C1C14

AGENDA FOR TELEPHONIC EXAMINER INTERVIEW

Dear Examiner Birkhimer:

Further to our earlier conversation, please find below topics that we would like to discuss during our interview scheduled for tomorrow, August 19, 2022, at 2:00 pm EDT.

Respectfully submitted,

Dated: August 18, 2022 By /Timothy B. Kang/

Timothy B. Kang
Registration No. 46,423

(703) 652-3817
MANNAVA & KANG, P.C.

3201 Jermantown Road
Suite 525
Fairfax, VA 22030
(703) 865-5150 (facsimile)

DRAFT

Independent claim 1:

1. (Previously Presented) A memory controller to control flash memory, the flash memory having subdivisions, each of the subdivisions corresponding to a respective group of erase blocks, each of the erase blocks such that it must be erased before storage locations in that erase block can be reprogrammed, the memory controller comprising:

a host interface;

a memory interface;

logic to track information indicating an extent to which each of the subdivisions is full;

logic to receive a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, wherein the logic to track is to, in association with execution of the write request, update the information indicating the extent to which the specific one of the subdivisions is full;

logic to send the host the information indicating the extent to which the specific one of the subdivisions is full and an address corresponding to the specific one of the subdivisions;

logic to receive an erasure request from the host, via the host interface, the erasure request accompanied by the address corresponding to the specific one of the subdivisions, to responsively control erasure of each unerased erase block in the group respective to the specific one of the subdivisions, and to update the information indicating the extent to which the specific one of the subdivisions is full; and

logic to, in connection with the erasure, detect a defect in one of the erase blocks in the group respective to the specific one of the subdivisions, to responsively substitute one or more different erase blocks of the flash memory for the one of the erase blocks for which the defect was detected;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

DRAFT

REMARKS

The pending claims were previously amended to recite that the memory controller is to send the host information indicating the extent to which a specific one of the subdivisions is full along with an address corresponding to the specific one of the subdivisions. The memory controller receives an erasure request from the host that is accompanied by the address corresponding to the specific one of the subdivisions. In this regard, the host communicates to the memory controller which of the subdivisions is to undergo write and erase operations and the memory controller carries out the operations. This is similar to the amendments made in the responses that were filed in a couple of other applications that we discussed. Accordingly, it is respectfully submitted that independent claim 1 is allowable over Selinger for reasons similar to those presented in the other applications.

It is also respectfully submitted that the proposed combination of Selinger and Gorobets fails to render independent claim 1 *prima facie* obvious for at least the following reasons.

1. Selinger fails to disclose "logic to track information indicating an extent to which each of the erase blocks is full."

The Office Action asserts that pars. 62, 76, and 100 of Selinger disclose this feature because they disclose "A status of a program operation being good shows tracking information indicating the associated block(s) where successively written and are thus full of valid data now." *Office Action*, page 5. That assertion is respectfully traversed because a status of a program operation being good or bad does not correspond to an extent to which each of the erase blocks in subdivisions of a flash memory is full. Clarification of this assertion is therefore respectfully requested.

2. Gorobets fails to disclose that the memory controller receives an erasure request from the host, in which the erasure request is accompanied by the address corresponding to the specific one of the subdivisions.

Gorobets pertains to a phased garbage collection, in which a memory controller 110 controls operations including erasing data and attending to garbage collection operations during, for instance, timeout periods allocated to write commands. Par. [0023]. Gorobets does not appear to disclose that the host sends erase commands to the memory controller. Instead, as is generally known, when memory controllers perform garbage collection, hosts do not send erase commands to the memory controllers. Accordingly, it is respectfully submitted that the proposed combination of Selinger and Gorobets fails to disclose "logic to receive an erasure request from the host, via the host interface, the erasure request accompanied by the address corresponding to the specific one of the subdivisions."

For similar reasons, one of ordinary skill in the art would not have found it obvious to combine Selinger and Gorobets as suggested in the Office Action.

DRAFT

Electronic Acknowledgement Receipt

EFS ID:	46461126
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Timothy B. Kang/Ingrid Hebron
Filer Authorized By:	Timothy B. Kang
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	22-AUG-2022
Filing Date:	19-AUG-2020
Time Stamp:	18:23:25
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1		1135-0006-Resp.pdf	217438 ce5090acce2be241c5c21b8c657c0ebbe74f36cc	yes	18

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Transmittal Letter	1	1
	Amendment/Request for Reconsideration-After Non-Final Rejection	2	18
Warnings:			
Information:			
	Total Files Size (in bytes):	217438	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.
National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.
New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Doc Code: TRAN.LET
Document Description: Transmittal Letter

PTO/SB/21 (07-09)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission	18

Application Number	16/997,471
Filing Date	August 19, 2020
First Named Inventor	Andrey V. KUZMIN, et al.
Art Unit	2136
Examiner Name	Christopher D. Birkhimer
Attorney Docket Number	2020058-RMS-02C1C1C14

ENCLOSURES *(Check all that apply)*

- ☐ Fee Transmittal Form
  - ☐ Fee Attached
- ☑ Amendment/Reply
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Reply to Missing Parts/ Incomplete Application
  - ☐ Reply to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) _____
  - ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☐ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☐ Other Enclosure(s) (please Identify below):

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	MANNAVA & KANG, P.C.		
Signature	/Timothy B. Kang/		
Printed name	Timothy B. Kang		
Date	August 22, 2022	Reg. No.	46,423

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature			
Typed or printed name		Date	

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and1.14. This collection is estimated to 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s):	Andrey V. KUZMIN, et al.	**Confirmation No.:**	1070
Serial No.:	16/997,471	**Examiner:**	Christopher D. Birkhimer
Filed:	August 19, 2020	**Group Art Unit:**	2136

Title: NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

AMENDMENT UNDER 37 CFR § 1.111

Sir:

In response to the Office Action dated June 28, 2022, kindly amend the application identified above as follows. In the following, underlines indicate insertions and strikethroughs and double brackets indicate deletions.

IN THE CLAIMS

Please find below a listing of all of the pending claims. The status of each claim is set forth in parentheses. This listing will replace all prior versions, and listings, of claims in the present application.

1. (Currently amended) A memory controller to control flash memory, the flash memory having subdivisions, each of the subdivisions corresponding to a respective group of erase blocks, each of the erase blocks such that it must be erased before storage locations in that erase block can be reprogrammed, the memory controller comprising:

a host interface;

a memory interface;

logic to track information indicating extent of page utilization of ~~an extent to which~~ each of the subdivisions ~~is full~~;

logic to receive a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, wherein the logic to track is to, in association with execution of the write request, update the information indicating extent of page utilization of ~~to which~~ the specific one of the subdivisions ~~is full~~;

logic to send the host the information indicating [[the]] extent of page utilization ~~to which~~ of the specific one of the subdivisions ~~is full~~ and an address corresponding to the specific one of the subdivisions;

logic to receive an erasure request from the host, via the host interface, the erasure request accompanied by the address corresponding to the specific one of the subdivisions, to responsively control erasure of each unerased erase block in the group respective to the specific one of the subdivisions, and to update the information indicating [[the]] extent of page utilization of ~~to which~~ the specific one of the subdivisions ~~is full~~; and

logic to, in connection with the erasure, detect a defect in one of the erase blocks in the group respective to the specific one of the subdivisions, to responsively substitute one or more different erase blocks of the flash memory for the one of the erase blocks for which the defect was detected;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

2. (Previously presented) The memory controller of claim 1, wherein the host interface is also to receive a data copy request from the host, wherein the memory controller is to responsively retrieve data from a source location in the flash memory, the source location associated with the specific one of the subdivisions, and write the retrieved data to a destination location in the flash memory, without sending the retrieved data to the host, and wherein the logic to track is to responsively update the tracked information for one of the subdivisions which corresponds to the destination location, in association with the write of the retrieved data into the destination location.

3. (Currently amended) The memory controller of claim 2, wherein:

[[the]] a plurality of data copy requests are each accompanied by a source address and a destination address; and

each source address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the flash memory, the mapping such that

different values of the field compel the memory controller to access respective storage structures at the specific tier,

a value of the field for each source address is effective to cause the memory controller to select one of the respective storage structures corresponding to the source location, and

a value of the field for each destination address is effective to cause the memory controller to select one of the respective storage structures corresponding to the destination location.

4. (Original) The memory controller of claim 3, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

5-10. (Canceled).

11. (Currently amended) A method of operation in a memory controller to control flash memory, where the flash memory has subdivisions, each of the subdivisions corresponding to a respective group of erase blocks, each of the erase blocks such that it must be erased before storage locations in that erase block can be reprogrammed, the memory controller having a host interface, a memory interface and at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that, when executed, are to control the operation of hardware circuitry, the method comprising:

tracking information indicating ~~the extents~~ extent of page utilization of ~~an extent to which~~ each of the subdivisions ~~is full~~;

receiving a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, and, in association with execution of the write request, updating the information indicating [[the]] extent of page utilization of ~~to which~~ the specific one of the subdivisions ~~is full~~;

sending the host the information indicating [[the]] extent of page utilization of ~~to which~~ the specific one of the subdivisions ~~is full~~ and an address corresponding to the specific one of the subdivisions;

receiving an erasure requests from the host, via the host interface, the erasure request accompanied by the address corresponding to the specific one of the subdivisions, responsively controlling erasure of each unerased erase block in the group respective to the specific one of the subdivisions. and updating the information

indicating [[the]] extent of page utilization of ~~to which~~ the specific one of the subdivisions ~~is full~~; and

in connection with the erasure, detecting a defect in one of the erase blocks in the group respective to the specific one of the subdivisions, responsively substituting one or more different erase blocks of the flash memory for the one of the erase blocks for which the defect was detected.

12. (Previously presented) The method of claim 11, further comprising also receiving via the host interface a data copy request from the host and responsively retrieving data from a source location in the flash memory, the source location associated with the specific one of the subdivisions, and writing the retrieved data which to a destination location in the flash memory, without sending the data which has been read to the host, and responsively updating the tracked information for the one of the subdivisions which corresponds to the destination location, in association with the write of the retrieved data into the destination location.

13. (Currently amended) The method of claim 12, wherein:

[[the]] a plurality of data copy requests are each accompanied by a source address and a destination address;

each source address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the flash memory, the mapping such that different values of the field compel the memory controller to access respective storage structures at the specific tier

a value of the field for each source address is effective to cause the memory controller to select one of the respective storage structures corresponding to the source location, and

a value of the field for each destination address is effective to cause the memory controller to select one of the respective storage structures corresponding to the destination location.

14. (Original) The method of claim 13, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

15-20. (Canceled)

21. (Previously Presented) The memory controller of claim 2, wherein:

the memory controller further comprises logic to track metadata for each of the subdivisions, the metadata indicating a time since data was programmed, to compare the tracked metadata to a threshold, and, responsive to a determination that the tracked metadata for the specific one of the subdivisions satisfies the threshold, to transmit to the host the address corresponding to the specific one of the subdivisions; and

the data copy request is accompanied by the address corresponding to the specific one of the subdivisions.

22. (Previously Presented) The memory controller of claim 21, wherein the address accompanying the data copy request is a source address for the retrieved data and wherein the data copy request is also accompanied by a destination address, wherein the specific one of the subdivisions is a first one of the subdivisions, and wherein the destination address is free to designate a second one of the subdivisions.

23. (Previously Presented) The memory controller of claim 3, wherein the specific tier is a tier associated with the subdivisions, and wherein the respective storage structures comprise respective ones of the subdivisions.

24. (Previously Presented) The memory controller of claim 1, wherein the memory controller further comprises logic to receive a query from the host, and wherein the logic to send the information to the host is to do so as a response to the query.

25. (Previously Presented) The memory controller of claim 1, wherein:

the memory controller further comprises logic to assign a write destination in the flash memory, in which data accompanying the write request is to be stored, the write destination corresponding to the specific one of the subdivisions, and to transmit to the host a read address that will be used to retrieve the data specifically from the write destination; and

the memory controller further comprises logic to receive, from the host, and via the host interface, a read request for the data, accompanied by the read address, and to

identify both of the specific one of the subdivisions and the write destination from the read address.

26. (Previously Presented) The memory controller of claim 1, wherein the memory controller further comprises logic to store erase count information for each of the subdivisions, and to send to the host, via the host interface, the erase count information for the specific one of the subdivisions, together with the address corresponding to the specific one of the subdivisions, and wherein the logic to store the erase count information is to update the erase count information for the specific one of the subdivisions in association with the erasure.

27. (Previously Presented) The method for claim 12, wherein:

the method further comprises tracking metadata for each of the subdivisions, the metadata indicating a time since data was programmed, comparing the tracked metadata to a threshold, and, responsive to a determination that the tracked metadata for the specific one of the subdivision satisfies the threshold, transmitting to the host the address corresponding to the specific one of the subdivisions; and

the data copy request is accompanied by the address corresponding to the specific one of the subdivisions.

28. (Canceled).

29. (Canceled).

30. (Previously Presented) The method of claim 11, further comprising receiving a query from the host, and sending the information to the host as a response to the query.

31. (Previously Presented) The method of claim 11, wherein:

the method further comprises assigning a write destination in the flash memory, in which data accompanying the write request is to be stored, the write destination corresponding to the specific one of the subdivisions, and transmitting to the host a read address that will be used to retrieve the data specifically from the write destination; and

the method further comprises receiving, from the host, and via the host interface, a read request for the data, accompanied by the read address, and identifying both of the specific one of the subdivisions and the write destination from the read address.

32. (Currently amended) The method of claim 11, wherein the method further comprises:

storing ~~erasing~~ erase count information for each of the subdivisions;

sending to the host, via the host interface, the erase count information for the specific one of the subdivisions, together with the address corresponding to the specific one of the subdivisions; and

updating the erase count information for the specific one of the subdivisions in association with the erasure.

33. (New) The memory controller of claim 1, wherein the extent of the page utilization of the specific one of the subdivisions represents the amount of valid data written to the specific one of the subdivisions since the specific one of the subdivisions was last erased.

34. (New) The memory controller of claim 1, wherein the extent of the page utilization of the specific one of the subdivisions represents a remaining space of the specific one of the subdivisions that is available for writing without first erasing the specific one of the subdivisions.

REMARKS

Favorable reconsideration of this application is respectfully requested in view of the claim amendments and following remarks.

Examiner Interview Conducted

Examiner Birkhimer is thanked for the courtesies extended to the undersigned, Robert Lercari, and Mike Jadon during the interview conducted on August 19, 2022. During the interview, the features of independent claim 1 and how those features differ from the disclosures contained in the cited Selinger and Gorobets documents were discussed. For instance, the undersigned asserted that the cited documents fail to disclose that information indicating an extent to which each of the subdivisions is full is tracked. Examiner Birkhimer stated that this feature is unclear and suggested that we amend this language. It was proposed that this language be changed to "logic to track information indicating the extent of page utilization of each of the subdivisions." Examiner Birkhimer agreed that the cited documents do not appear to disclose this feature. However, the Examiner said that an updated search would be needed before a determination as to whether the claims are allowable may be made.

Independent claims 1 and 11 are amended to incorporate the above-cited feature. The Examiner is therefore respectfully requested to reconsider the claims based on these amendments.

Status of Claims

Claims 1, 3, 11, 13, and 32 are amended and new claims 33 and 34 are added. Support for the amendments to independent claims 1 and 11 and new claims 33 and 34 may at least be found in paragraph [0040] of the originally filed specification.

Claims 5-10 and 15-20 were previously canceled without prejudice or disclaimer of the subject matter contained therein. Claims 28 and 29 are canceled without prejudice or disclaimer of the subject matter contained therein.

Claims 1-4, 11-14, 21-27, and 30-34 are pending in the application of which claims 1 and 11 are independent.

No new matter has been introduced by way of the amendments above. Entry thereof is therefore respectfully requested.

Summary of the Office Action

Claims 3, 4, 13, 14, 23, and 29 are objected to for minor informalities.

Claims 3, 4, 13, 14, 23, and 29 are rejected under 35 U.S.C. § 112, second paragraph, as allegedly being indefinite.

Claims 1-4, 11-14, 23-25, and 29-31 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. Patent Application Publication No. 2011/0161784 to Selinger, et al. (hereinafter "Selinger") in view of U.S. Patent Application Publication No. 2008/0082596 to Gorobets (hereinafter "Gorobets").

Claims 21, 22, 27, and 28 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Selinger in view of Gorobets as applied to claims 2 and 12 respectively above,

and further in view of U.S. Patent Application Publication No. 2011/0231730 to Allen (hereinafter "Allen").

Claims 26 and 32 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Selinger in view of Gorobets as applied to claims 1 and 11 respectively above, and further in view of U.S. Patent Application Publication No. 2011/0197014 to Yeh (hereinafter "Yeh").

The rejections above are respectfully traversed for at least the reasons set forth below.

Information Disclosure Statement

The indication that the documents cited in the Information Disclosure Statement submitted on June 14, 2022 have been considered is hereby acknowledged with appreciation.

Claim Objections

Claims 3, 4, 13, 14, 23, and 29 are objected to because of the following informalities:

Claims 3 and 13 are amended as suggested in the Office Action. The Examiner is therefore respectfully requested to withdraw the objection to the claims.

Claim Rejection Under 35 U.S.C. § 112, 2nd Paragraph

Claims 3, 4, 13, 14, 23, and 29 are rejected under 35 U.S.C. § 112, second paragraph, as allegedly being indefinite.

Claims 3 and 13 are amended in minor respects to better comply with the provisions of 35 U.S.C. § 112, second paragraph. The Examiner is therefore respectfully requested to withdraw the rejection of these claims.

Claim Rejections Under 35 U.S.C. § 103(a)

Claims 1-4, 11-14, 23-25, and 29-31

Claims 1-4, 11-14, 23-25, and 29-31 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Selinger in view of Gorobets.

Independent claim 1 is amended to recite:

> logic to track information indicating the extent of page utilization of each of the subdivisions;
> logic to receive a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, wherein the logic to track is to, in association with execution of the write request, update the information indicating extent of page utilization of the specific one of the subdivisions;
> logic to send the host the information indicating extent of page utilization of the specific one of the subdivisions and an address corresponding to the specific one of the subdivisions;
> logic to receive an erasure request from the host, via the host interface, the erasure request accompanied by the address corresponding to the specific one of the subdivisions, to responsively control erasure of each unerased erase block in the group respective to the specific one of the subdivisions, and to update the information indicating extent of page utilization of the specific one of the subdivisions

Independent claim 11 is amended in similar respects.

As noted above, independent claims 1 and 11 are amended to include the features agreed upon during the Examiner interview as being deficient in Selinger and Gorobets. Accordingly, even assuming for the sake of argument that one of ordinary skill in the art were somehow motivated to combine Selinger and Gorobets as suggested in the Office Action, the proposed combination fails to disclose each and every element recited in independent claim 1. The proposed combination therefore fails to render this claim *prima facie* obvious.

The Examiner is therefore respectfully requested to withdraw the rejection of independent claims 1 and 11 and the claims that depend therefrom.

Claims 21, 22, 26-28, and 32

Claims 21, 22, 26-28, and 32 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Selinger in view of Gorobets as applied to claims 2 and 12 respectively above, and further in view of Allen. Claims 26 and 32 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Selinger in view of Gorobets as applied to claims 1 and 11 respectively above, and further in view of Yeh.

Claims 21, 22, 26-28, and 32 depend upon one of independent claims 1 and 11 and are thus allowable over the proposed combination of Selinger and Gorobets at least by virtue of these dependencies. The Office Action has not asserted that Allen or Yeh make up for the deficiencies the proposed combination of Selinger and Gorobets nor do they appear to do so.

Accordingly, even assuming for the sake of argument that one of ordinary skill in the art were somehow motivated to combine Selinger and Gorobets with either of Allen and Yeh as suggested in the Office Action, the proposed combinations fail to disclose each and every element recited in independent claims 1 and 11. The proposed combinations therefore fail to render these claims *prima facie* obvious.

The Examiner is therefore respectfully requested to withdraw the rejection of claims 21, 22, 26-28, and 32.

Conclusion

In light of the foregoing, withdrawal of the rejections of record and allowance of this application are earnestly solicited. Should the Examiner believe that a telephone conference with the undersigned would assist in resolving any issues pertaining to the allowability of the above-identified application, please contact the undersigned at the telephone number listed below. Please grant any required extensions of time and charge any fees due in connection with this request to Deposit Account No. 50-3290.

Respectfully submitted,

Dated: August 22, 2022 By /Timothy B. Kang/

Timothy B. Kang
Registration No. 46,423
(703) 652-3817

MANNAVA & KANG, P.C.
3201 Jermantown Road
Suite 525
Fairfax, VA 22030
(703) 865-5150 (facsimile)

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 16/997,471	Filing Date 08/19/2020	☐ To be Mailed

ENTITY: ☐ LARGE ☑ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $230 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	08/22/2022	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $50 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $240 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	LIE
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/CAROL A BARNES/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

73091 7590 06/28/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	

ART UNIT	PAPER NUMBER
2136	

MAIL DATE	DELIVERY MODE
06/28/2022	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 16/997,471	**Applicant(s)** Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	**Art Unit** 2136	**AIA (FITF) Status** No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☑ Responsive to communication(s) filed on 06/14/2022.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☐ This action is **FINAL.** 2b) ☑ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 1-4,11-14 and 21-32 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 1-4,11-14 and 21-32 is/are rejected.

8) ☐ Claim(s) _____ is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☐ The specification is objected to by the Examiner.

11)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)

2) ☑ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date 06/14/2022.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

DETAILED ACTION

The current Office Action is in response to the papers submitted 06/14/2022.

Claims 1 – 4, 11 – 14, and 21 - 32 are pending.

Notice of Pre-AIA or AIA Status

1. The present application is being examined under the pre-AIA first to invent provisions.

Claim Objections

2. Claims **3 – 4, 13 – 14, 23, and 29** are objected to because of the following informalities:

Claim 3 is worded oddly to the examiner at line 19. It appears "structure" should be "structures".

Claim 13 is objected for containing the same language as claim 3 and is thus objected to for similar reasons.

All remaining claims are objected to for being dependent on an objected base claim.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

3. The following is a quotation of 35 U.S.C. 112(b):
(b) CONCLUSION.—The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the inventor or a joint inventor regards as the invention.

The following is a quotation of 35 U.S.C. 112 (pre-AIA), second paragraph:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims **3 – 4, 13 – 14, 23, and 29** are rejected under 35 U.S.C. 112(b) or 35 U.S.C. 112 (pre-AIA), second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the inventor or a joint inventor (or for applications subject to pre-AIA 35 U.S.C. 112, the applicant), regards as the invention.

Claim 3 recites the limitation "the data copy requests" in line 5. There is no previous mention of data copy requests in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim. Claim 2 does disclose a data copy request in lines 1 – 2. It will be assumed for examination that the data copy requests refer to the single copy request in claim 2.

Claim 13 is rejected for containing similar language as rejected claim 3 and is thus rejected for similar reasons.

All remaining claims are rejected for being dependent on a rejected base claim.

Claim Rejections - 35 USC § 103

5. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of pre-AIA 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries for establishing a background for determining obviousness under pre-AIA 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.

2. Ascertaining the differences between the prior art and the claims at issue.

3. Resolving the level of ordinary skill in the pertinent art.

4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under pre-AIA 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of pre-AIA 35 U.S.C. 103(c) and potential pre-AIA 35 U.S.C. 102(e), (f) or (g) prior art under pre-AIA 35 U.S.C. 103(a).

6. Claims **1 – 4, 11 – 14, 23 – 25, and 29 - 31** is/are rejected under pre-AIA 35 U.S.C. 103(a) as being unpatentable over Selinger et al. (Pub. No.: US 2011/0161784) referred to as Selinger in view of Gorobets (Pub. No.: US 2008/0082596) referred to as Gorobets.

Regarding **claim 1**, Selinger teaches a memory controller **[700, Fig 7]** to control flash memory **[730, Fig 7]**, the flash memory **[730, Fig 7]** having a group of erase blocks **[1200 and 1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The memory is group of erase blocks]**, each of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** such that it must be erased before storage locations **[Paragraph 0062; The pages in a block]** in the erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** can be reprogrammed **[Paragraph 0062]**, the memory controller **[700, Fig 7]** comprising:

a host interface **[725, Fig 7]**;

a memory interface **[735, Fig 7]**;

logic to track information **[908, Fig 9]** indicating an extent to which each of the erase blocks is full **[Paragraphs 0062, 0076, and 0100; A status of a program operation being good shows tracking information indicating the associated block(s) where successively written and are thus full of valid data now]**;

logic **[700, Fig 7]** to receive a write request **[510, Fig 5]** from a host **[720, Fig 7]** via the host interface **[725, Fig 7]**, the write request **[510, Fig 5]** accompanied by an address that designates a specific one of the erase blocks **[Paragraphs 0025, 0062, and 0065; The command write command includes a specific address where to write data]**, wherein the logic to track is to, in association with execution of the write request, update the information indicating the extent to which the specific one of the erase blocks is full **[908, Fig 9; Paragraphs 0062, 0076, and 0100;**

Updating status information for bit 3 to indicate the program was successful indicates the block(s) are programmed and full of valid data];

logic **[700, Fig 7]** to send the host **[720, Fig 7]** the information **[908, Fig 9]** indicating the extent to which the specific one of the erase blocks is full **[Paragraphs 0062, 0076, and 0100; Updating status information for bit 3 to indicate the program was successful indicates the block(s) programmed are full of valid data and that status information is sent to the host]** and an address of a block **[Paragraphs 0065, 0093, and 0159 – 0161; The host knows the block addresses since it performs the logical to physical mapping. The controller performs block management by moving data from one block to another block. This shows the host would need to receive new block addresses, as a result of a block management operation performed by the controller, to update the mapping table in the host to correctly address memory after the moving of data]**;

logic **[700, Fig 7]** to receive an erasure request from the host **[720, Fig 7]**, via the host interface **[725, Fig 7]** and to update the information **[908, Fig 9]** indicating the extent to which the specific one of the erase blocks is full **[Paragraphs 0025, 0062, 0073 – 0074, 0076, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information**

sent to the host from the controller. That status of erased correctly indicate the block(s) is full of erase data];

logic **[700, Fig 7]** to, in connection with the erasure **[Paragraphs 0025, 0073 – 0074, 0100 and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]**, detect a defect **[908 and 914, Fig 9; Both status bits when set indicate a defect in an erase block]** in one of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** in the group **[1200 and 1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The memory is group of erase blocks]**, to responsively substitute one or more different erase blocks of the flash memory **[730, Fig 7]** for the one of the erase blocks for which the defect was detected **[Paragraphs 0093 and 0159 – 161; controller substitutes a good spare erase block for a bad block when a bad block is detected]**; and

wherein each said logic **[700, Fig 7]** comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry **[725, 735, 740, 750, 760, and 770, Fig 7; Paragraph 0025]**.

However, Selinger may not specifically disclose the limitation of flash memory having subdivisions where each of the subdivisions corresponds to a respective group of erase blocks and the erasure request accompanied by the address corresponding to

the specific one of the subdivisions, to responsively control erasure of each unerased block in the group respective to the specific one of the subdivisions.

Gorobets discloses flash memory **[118, Fig 1; Paragraph 0024]** having subdivisions where each of the subdivisions corresponds to a respective group of erase blocks **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks]** the erasure request accompanied by the address corresponding to the specific one of the subdivisions, to responsively control erasure of each unerased block in the group respective to the specific one of the subdivisions **[Figs 9A – 9B; Paragraphs 0056 – 0061; The memory is addressed in metablocks and when a metablock is erased all blocks that make up the metablock are erased that where not previously erased putting the entire metablock in the erased state]**.

Therefore, it would have been obvious to a person of ordinary skill in the art before the effective filing date of the claimed invention to incorporate Gorobets in Selinger, because it improves the garbage collection process performed on flash memory by preventing timeout errors by using phased garbage collection **[Paragraphs 0004 – 0006]**.

Regarding **claim 2**, Selinger teaches the host interface **[725, Fig 7]** is also to receive a data copy request **[1410 and 1440, Fig 14]** from the host **[720, Fig 7]**, wherein the memory controller **[700, Fig 7]** is to responsively retrieve data from a source location **[1430, Fig 14]** in the flash memory **[730, Fig 7]**, the source location associated with the specific one of the erase blocks **[1410, Fig 14; The source address is an erase block of memory where data is read from for the copy**

operation], and write the retrieved data to a destination location **[1440, 1460, and 1470, Fig 14]** in the flash memory **[730, Fig 7]** without sending the retrieved data to the host **[720, Fig 7; 1460 and 1470, Fig 14; Paragraph 0152; The data that is copied is not sent to the host]**, and wherein the logic to track is to responsively update the tracked information **[908, Fig 9]** for one of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** which corresponds to the destination location, in associated with the write of the retrieved data into the destination location **[Paragraphs 0062, 0076, and 0100; A status of a program operation being good shows tracking information indicating the associated block(s) where successively written and are thus full of valid data now]**.

Gorobets discloses flash memory **[118, Fig 1; Paragraph 0024]** having subdivisions where each of the subdivisions corresponds to a respective group of erase blocks **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks]**.

Regarding **claim 3**, Selinger teaches the data copy requests **[1410 and 1440, Fig 14]** are each accompanied by a source address **[1410, Fig 14]** and a destination address **[1440, Fig 14]**; and

each source address **[1410, Fig 14]** includes a field that has been mapped in advance to a specific tier of a structural hierarchy **[Paragraphs 0065, 0108, and 0110; Each address provided from the host is mapped in advance in the mapping table to a physical structural tier address**

such as erase block addresses] associated with storage in the flash memory **[730, Fig 7]**, the mapping such that

different values of the field compel the memory controller **[700, Fig 7]** to access respective storage structures at the specific tier **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; Different erase block addresses will cause different erase blocks to be accessed on the erase block tier of the memory]**; and

a value of the field for each source address **[1410, Fig 14]** is effective to cause the memory controller **[700, Fig 7]** to select one of the respective storage structure corresponding to the source location **[1420, Fig 14]**, and

a value of the field for each destination address **[1440, Fig 14]** is effective to cause the memory controller to select one of the respective storage structures corresponding to the destination location **[1460 or 1470, Fig 14]**.

Regarding **claim 4**, Selinger teaches the specific tier is a tier associated with the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks which make up a tier of erase blocks]**, and wherein the respective storage structures comprise respective erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]**.

With regard to **claim 23**, Selinger teaches the specific tier is a tier associated with the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The erase**

blocks make up a block tier of storage], and wherein the respective storage structures comprise respective ones of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; Each erase block is a storage structure that is accessed for reads and writes]**.

Gorobets discloses the specific tier is associated with subdivisions and wherein the storage structures comprise respective ones of the subdivisions **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks. Each block and metablock is a storage structure]**.

With regard to **claim 24**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic **[700, Fig 7]** to receive a query from the host **[510, Fig 5; 610; Fig 6; Paragraphs 0074 and 0076; Each read or write results in status information sent to the host making each read and write command a query also for the status information**, and wherein the logic **[700, Fig 7]** to send the information to the host is to do so as a response to the query **[Paragraphs 0074 and 0076; Status information is sent to the host due to a read and write query operation to indicate to the host the status of the blocks due to the read and write requests]**.

With regard to **claim 25**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic **[700, Fig 7]** to assign a write destination in the flash memory **[730, Fig 7]**, in which data accompanying the write request **[510, Fig 5]** is to be stored, the write destination corresponding to the specific one of the erase blocks **[Paragraphs 0093 and 0159 – 0161; The controller handles block management my relocating data from a bad**

block to a new spare block. This requires the host to assign a write destination to the new spare block], and to transmit to the host **[720, Fig 7]** a read address that will be used to retrieve the data specifically from the write destination **[Paragraphs 0065, 0093, and 0159 – 0161; The host knows the block addresses since it performs the logical to physical mapping. The controller performs block management by moving data from one block to another block. This shows the host would need to receive new block addresses, as a result of a block management operation performed by the controller, to update the mapping table in the host to correctly address memory after the moving of data. The new address is a read address of where the data is stored for later access]**; and

the memory controller **[700, Fig 7]** further comprises logic **[700, Fig 7]** to receive, from the host, and via the host interface **[725, Fig 7]**, a read request for the data **[610, Fig 6]**, accompanied by the read address, and to identify both of the specific one of the erase blocks and the write destination from the read address **[Fig 6; Paragraphs 0065, 0093, and 0159 – 0161; A read request for data relocated previously to a spare block would identify the spare block that is used to store the data and also the write address of the spare block used previously to access the spare block to write the relocated data]**.

Gorobets discloses flash memory **[118, Fig 1; Paragraph 0024]** having subdivisions where each of the subdivisions corresponds to a respective group of erase

blocks **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks]**.

Claims 11 – 14 and 29 - 31 are the method claims relating to the memory controller claims 1 – 4 and 23 - 25 and are thus rejected using the same prior art and reasoning.

7. Claims **21 – 22 and 27 - 28** are rejected under pre-AIA 35 U.S.C. 103(a) as being unpatentable over Selinger et al. (Pub. No.: US 2011/0161784) referred to as Selinger in view of Gorobets (Pub. No.: US 2008/0082596) referred to as Gorobets as applied to **claims 2 and 12** respectively above, and further in view of Allen (Pub. No.: US 2011/0231730) referred to as Boyle.

With regard to **claim 21**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to track metadata for each of the erase blocks **[900, Fig 9; 1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, the metadata indicating status **[900, Fig 9; The status information indicates the status of an erase block due to a read or write operation]** of an erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** and to transmit to the host **[720, Fig 7]** the address corresponding to the specific one of the erase blocks **[Paragraphs 0065, 0093, and 0159 – 0161; The host knows the block addresses since it performs the logical to physical mapping. The controller performs block management by**

moving data from one block to another block. This shows the host would need to receive new block addresses, as a result of a block management operation performed by the controller, to update the mapping table in the host to correctly address memory after the moving of data]; and

the data copy request **[1410 and 1440, Fig 14]** is accompanied by the address corresponding to the specific one of the erase block **[1410 and 1440, Fig 14; Paragraphs 0065, 0074, 0093, and 0159 – 0161; A relocation of a block that was already relocated due to an error and managed by the controller would include the address the controller provided to the host due to the previous error management done by the controller]**.

Gorobets discloses flash memory **[118, Fig 1; Paragraph 0024]** having subdivisions where each of the subdivisions corresponds to a respective group of erase blocks **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks]**.

However, Selinger in view of Gorobets may not specifically disclose the limitations of the metadata indicating a time since data was programmed, to compare the tracked metadata to a threshold, and, responsive to a determination that the tracked metadata for the specific location satisfies the threshold, to transmit the address corresponding to the specific location.

Allen discloses the metadata indicating a time since data was programmed, to compare the tracked metadata to a threshold, and, responsive to a determination that

the tracked metadata for the specific location satisfies the threshold, to transmit the address corresponding to the specific location **[Paragraphs 0020 – 0021; The age of data is an indication of how long data was last programmed. Marking a block as bad indicates the age has reached or exceeded some threshold level to mark the block as bad]**.

Therefore, it would have been obvious to a person of ordinary skill in the art before the effective filing date of the claimed invention to incorporate Allen in Selinger in view of Gorobets, because it provides another metric to analyze to manage data in a system to prevent data lose.

With regard to **claim 22**, Selinger teaches the address accompanying the data copy request **[1410 and 1440, Fig 14]** is a source address **[1410, Fig 14]** for the retrieved data **[1420 and 1430, Fig 14]** and wherein the data copy request **[1410 and 1440, Fig 14]** is also accompanied by a destination address **[1440, Fig 14]**, wherein the specific one of the erase blocks is a first one of the erase blocks **[1410, Fig 14; The source address is an address of a first erase block where data is read from]**, and wherein the destination address is free to designate a second one of the erase blocks **[1440, Fig 14; The destination address is an address of an erase block]**.

Claim 27 and 28 are the corresponding method claims of controller claims 21 and 22 and are thus rejected using the same prior art and similar reasoning.

8. Claims **26 and 32** are rejected under pre-AIA 35 U.S.C. 103(a) as being unpatentable over Selinger et al. (Pub. No.: US 2011/0161784) referred to as Selinger in view of Gorobets (Pub. No.: US 2008/0082596) referred to as Gorobets as applied to

claims 1 and 11 respectively above, and further in view of Yeh (Pub. No.: US 2011/0197014) referred to as Yeh.

With regard to **claim 26**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic **[700, Fig 7]** to store additional status information **[900, Fig 9; Each row represents a different status indicator for an erase block as a result of a read or write operation]** for each of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, and to send to the host **[720, Fig 7]**, via the host interface **[725, Fig 7]**, the additional status information **[900, Fig 9; Each row represents a different status indicator for an erase block as a result of a read or write operation]** for the specific one of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The status information represents the status of a given erase block]**, together with the address corresponding to the specific one of the erase blocks **[Paragraphs 0065, 0093, and 0159 – 0161; The host knows the block addresses since it performs the logical to physical mapping. The controller performs block management by moving data from one block to another block. This shows the host would need to receive new block addresses, as a result of a block management operation performed by the controller, to update the mapping table in the host to correctly address memory after the moving of data]**, and wherein the logic **[700, Fig 7]** to store the additional status information **[900, Fig 9; Each row represents a different status indicator for an erase block as a result of a read or write operation]** is to update the additional status information **[900, Fig 9; Each row represents a different status indicator for an erase block as a result of a read or write operation]** for the specific one of the erase blocks in

association with the erasure **[Paragraphs 0025, 0062, 0073 – 0074, 0076, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller. That status of erased correctly indicate the block(s) is full of erase data]**.

Gorobets discloses flash memory **[118, Fig 1; Paragraph 0024]** having subdivisions where each of the subdivisions corresponds to a respective group of erase blocks **[Fig 2; Paragraph 0027; The metablock 210 – 213 and 220 – 223 are each a subdivision of the overall flash memory and each are comprised of a group of erase blocks]** the erasure request accompanied by the address corresponding to the specific one of the subdivisions, to responsively control erasure of each unerased block in the group respective to the specific one of the subdivisions **[Figs 9A – 9B; Paragraphs 0056 – 0061; The memory is addressed in metablocks and when a metablock is erased all blocks that make up the metablock are erased that where not previously erased putting the entire metablock in the erased state]**.

However, Selinger in view of Gorobets may not specifically disclose the limitations of storing erase count information for each erase block and updating the erase count information for the specific one of the erase blocks in association with the erasure.

Yeh discloses storing erase count information for each erase block and updating the erase count information for the specific one of the erase blocks **[Paragraph 0052; The memory in managed in erase blocks and pages]** in association with the erasure

[S1605, Fig 16; Paragraphs 0014, 0071 – 0073, 0076 – 0077, 0079 – 0080, 0082, 0085, 0088, 0092, 0095 – 0097, 0099].

Therefore, it would have been obvious to a person of ordinary skill in the art before the effective filing date of the claimed invention to incorporate Yeh in Selinger in view of Gorobets, because the use of erase counts is used to extend the life of memory **[Paragraphs 0009 and 0011; Abstract]**.

Claim 32 is the corresponding method claims of controller claims 21 and 22 and are thus rejected using the same prior art and similar reasoning.

Response to Arguments

9. Applicant's arguments, see pages 10 - 11, filed 06/14/2022, with respect to the rejection(s) of claim(s) 1 - 20 under 35 USC 102 in view of Selinger alone have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Selinger, Gorobets, Allen, and Yeh as indicated in the rejections above.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER D BIRKHIMER whose telephone number is (571)270-1178. The examiner can normally be reached 8-5 Hoteling.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an

interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on 571-272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Christopher D Birkhimer/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 1 of 2

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20130097236-A1	04-2013	KHORASHADI; Behrooz	G06F16/951	709/204
*	B	US-20120303875-A1	11-2012	Benhase; Michael T.	G06F11/1076	711/E12.019
*	C	US-20120155492-A1	06-2012	Abel; Francois	H04J3/1682	370/474
*	D	US-5963977-A	10-1999	Gold; Clifford M.	G06F3/0656	711/208
*	E	US-20130166824-A1	06-2013	SHIM; GYU-DONG	G06F12/0246	711/E12.008
*	F	US-6892287-B1	05-2005	Millard; Paul C.	H04L12/5601	711/170
*	G	US-20110264843-A1	10-2011	Haines; Jonathan W.	G06F12/0246	711/170
*	H	US-20110041039-A1	02-2011	Harari; Eliyahou	G06F11/1068	713/193
*	I	US-20110238892-A1	09-2011	Tsai; Song-Feng	G11C16/349	711/E12.008
*	J	US-20060004957-A1	01-2006	Hand; Leroy C. III	G06F12/0866	711/E12.019
*	K	US-8341339-B1	12-2012	Boyle; William B.	G06F3/0685	711/112
*	L	US-20110197014-A1	08-2011	Yeh; Chih-Kang	G06F12/0246	711/E12.078
*	M	US-20080082596-A1	04-2008	Gorobets; Sergey Anatolievich	G06F12/0253	1/1

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N	JP-2001051889-A	02-2001	JP	TOMORI N	G11C16/102
	O	KR-2000025612-A	05-2000	KR	BAE S H	H04L12/64
	P	RD-527033-A	03-2008	RD		
	Q	JP-2011123863-A	06-2011	JP	HSU S	G06F3/061
	R	JP-2011203916-A	10-2011	JP	SUKEGAWA H	G06F12/0246
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 2 of 2

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20110231730-A1	09-2011	Allen; William J.	G06F12/0246	711/E12.008
	B					
	C					
	D					
	E					
	F					
	G					
	H					
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

CPC - Searched*		
Symbol	**Date**	**Examiner**
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	09/30/2021	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	02/10/2022	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	06/23/2022	CDB

CPC Combination Sets - Searched*		
Symbol	**Date**	**Examiner**

US Classification - Searched*			
Class	**Subclass**	**Date**	**Examiner**

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	**Date**	**Examiner**
EAST search performed	09/30/2021	CDB
Assignee/Inventor search performed in EAST	09/30/2021	CDB
Prior Art/Assignee/Inventor/Google Internet database search	02/10/2022	CDB
Prior Art/Assignee/Inventor search performed	06/23/2022	CDB

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	09/30/2021	02/10/2022	06/23/2022						
	1	✓	✓	✓						
	2	✓	✓	✓						
	3	✓	✓	✓						
	4	✓	✓	✓						
	5	✓	✓	-						
	6	✓	✓	-						
	7	✓	✓	-						
	8	✓	✓	-						
	9	✓	✓	-						
	10	✓	✓	-						
	11	✓	✓	✓						
	12	✓	✓	✓						
	13	✓	✓	✓						
	14	✓	✓	✓						
	15	✓	✓	-						
	16	✓	✓	-						
	17	✓	✓	-						
	18	✓	✓	-						
	19	✓	✓	-						
	20	✓	✓	-						
	21			✓						
	22			✓						
	23			✓						
	24			✓						
	25			✓						
	27			✓						
	28			✓						
	29			✓						
	30			✓						
	31			✓						
	32			✓						

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/
/Christopher Birkhimer/ 1/6/2023

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	16997471
Filing Date	2021-12-27
First Named Inventor	Andrey V. Kuzmin
Art Unit	2136
Examiner Name	Birkhimer, Christopher
Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	8626989	B2	2014-01-07	Van Aken	
	2	9383926	B2	2016-07-05	Law	
	3	9134918	B2	2015-09-15	Yurzola	
	4	8898410	B1	2014-11-25	Ehrenberg	
	5	11347639	B1	2022-05-31	Kuzmin	
	6	11347638	B1	2022-05-31	Kuzmin	
	7	11354234	B1	2022-06-07	Kuzmin	
	8	11354235	B1	2022-06-07	Kuzmin	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

EFS Web 2.1.17

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/
Tabsoft, pu 1d1e2A0082182022

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENT APPLICATION PUBLICATIONS						Remove
Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20130297880	A1	2013-11-07	Flynn	
	2	20130275682	A1	2013-10-17	Ramanujan	
	3	20120072645	A1	2012-03-22	Olbrich	
	4	20080034153	A1	2008-02-07	Lee	
	5	20170192901	A1	2017-07-06	Coffin	
	6	20150339187	A1	2015-11-26	Sharon	
	7	20170199703	A1	2017-07-13	Ravimohan	
	8	20170109078	A1	2017-04-20	Shaharabany	
	9	20050160227	A1	2005-07-21	Todd	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/
Tabesh, Farida 08/15/2022

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	10	20070136555	A1	2007-06-14	Sinclair	
	11	20140122781	A1	2014-05-01	Smith	
	12	20160070496	A1	2016-03-10	Cohen	
	13	20100287327	A1	2010-11-11	Li	
	14	20090216992	A1	2009-08-27	Greiner	
	15	20090144497	A1	2009-06-04	Withers	
	16	20110231623	A1	2011-09-22	Goss	
	17	20090172499	A1	2009-07-02	Olbrich	
	18	20170075620	A1	2017-03-16	Yamamoto	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/
Examiner Initials* Date Considered 06/21/2022

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button [Add]

NON-PATENT LITERATURE DOCUMENTS [Remove]

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	Y.. Hu et al., "Achieving page-mapping FTL performance at block-mapping FTL cost by hiding address translation," 2010 IEEE 26th Symposium On Mass Storage Systems And Technologies (MSST), 2010, pp1-12.	

If you wish to add additional non-patent literature document citation information please click the Add button [Add]

EXAMINER SIGNATURE

Examiner Signature	/Christopher D Birkhimer/	Date Considered	06/21/2022

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/
Tracking ID: 6008218

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Marc P. Schuyler/	Date (YYYY-MM-DD)	2022-06-14
Name/Print	Marc P. Schuyler	Registration Number	35675

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L91	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"2	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM

		0150324264"|"20150347041"|"20150347296"|"20160018998"|"20160026564"|"20160342509"|"20160357462"|"20160364179"|"20170031699"|"5568423"|"5652857"|"5860082"|"6134631"|"7096378"|"7339823"|"7404031"|"7406563"|"7581078"|"7702846"|"7710777"|"7752381"|"7801561"|"7818489"|"7836244"|"7861122"|"7941692"|"7970983"|"7991944"|"8001318"|"8024545"|"8055833"|"8065471"|"8065473"|"8068365"|"8072463"|"8074022"|"8082389"|"8086790"|"8347042"|"8572331"|"8645634"|"9229854"|"9335939"|"9400749"|"9519578"|"9542118"|"9588904"|"9652376"|"9696917"|"9710377"|"9727454"|"9785572"|"9846541").PN.				
L92	54	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L93	131	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L94	20736	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L95	1	(L22 L23) and (constituent near2 storage near2 location$1) and cop$4 and (erase near2 request)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L96	20736	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L97	63	L26 and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L98	54	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L99	131	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM

		Richard)).inv.					
L100	1	(L28 L29) and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L101	1414	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L102	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L103	17	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L104	68	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L105	33	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L106	326	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L107	108	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L108	54	((controller$1) near10 (transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM

		moved or moving))					
L109	24	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L110	2348	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L111	1358	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L112	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L113	17	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L114	66	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L115	32	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L116	325	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L117	108	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L118	54	((controller$1) near10	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21

		(transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or moved or moving))					09:55 AM
L119	24	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L120	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM

		"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"20150324264"|"20150347041"|"20150347296"|"20160018998"|"20160026564"|"20160342509"|"20160357462"|"20160364179"|"20170031699"|"5568423"|"5652857"|"5860082"|"6134631"|"7096378"|"7339823"|"7404031"|"7406563"|"7581078"|"7702846"|"7710777"|"7752381"|"7801561"|"7818489"|"7836244"|"7861122"|"7941692"|"7970983"|"7991944"|"8001318"|"8024545"|"8055833"|"8065471"|"8065473"|"8068365"|"8072463"|"8074022"|"8082389"|"8086790"|"8347042"|"8572331"|"8645634"|"9229854"|"9335939"|"9400749"|"9519578"|"9542118"|"9588904"|"9652376"|"9696917"|"9710377"|"9727454"|"9785572"|"9846541").PN.				
L121	0	"store the storage location release information"	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L122	0	"store the storage location release information"	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L123	54	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L124	131	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:55 AM
L125	5	(L54 L53) and ((error$1 defect$1)).clm.	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 09:55 AM
L126	26	("11347638" OR "11347639" OR "11354234" OR	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/21 09:56 AM

		"11354235" OR "20050160227" OR "20070136555" OR "20080034153" OR "20090144497" OR "20090172499" OR "20090216992" OR "20100287327" OR "20110231623" OR "20120072645" OR "20130275682" OR "20130297880" OR "20140122781" OR "20150339187" OR "20160070496" OR "20170075620" OR "20170109078" OR "20170192901" OR "20170199703" OR "8626989" OR "8898410" OR "9134918" OR "9383926").pn.					
L127	6	L94 AND (updat$4 near10 full near5 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:07 AM
L128	68	((track$4 updat$4) near10 full near5 status near10 block)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:09 AM
L129	7	((track$4 updat$4) near10 full near5 status	(US-PGPUB; USPAT; USOCR; FIT (AU, AP,	OR	ON	ON	2022/06/21 10:12 AM

		near10 (block page) near10 host)	AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L130	180	((track$4 updat$4) near10 full near10 (block page) near10 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:20 AM
L131	9	"20130024423"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:24 AM
L132	10	"20080082596"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO,	OR	ON	ON	2022/06/21 10:33 AM

			RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L133	70	((track$4 updat$4) near10 full near10 (page) near10 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:40 AM
L134	20	((track$4 updat$4) near10 full near10 (page) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:42 AM
L135	0	((track$4 updat$4) near10 full near10 (die) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:43 AM
L136	0	((track$4 updat$4) near10 full near10 (plane) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU,	OR	ON	ON	2022/06/21 10:43 AM

			CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L137	26	((track$4 updat$4) near10 full near10 (cell) near10 status)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:43 AM
L138	38	(controller near10 write near5 address near5 back near5 host)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 10:59 AM
L142	6371	((vm (virtual ADJ machine)) near5 live near5 migrat$4)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA,	OR	ON	ON	2022/06/21 01:52 PM

			VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L143	5251	(((virtual ADJ machine)) near5 live near5 migrat$4)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/21 01:53 PM
L145	57	2011/0145473	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 11:56 AM
L146	5	"20110145473"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 11:57 AM
L147	73	L94 AND ((hot cold) near5 data near5 host) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB,	OR	ON	ON	2022/06/22 12:04 PM

			HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L148	333	((hot cold) near5 data near5 time) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:09 PM
L149	85	((hot cold) near5 data near5 time) SAME wear SAME host	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:09 PM
L150	10	"20140006688"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT;	OR	ON	ON	2022/06/22 12:10 PM

			IBM_TDB)				
L151	333	((hot cold) near5 data near5 time) SAME wear	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:11 PM
L152	192	((hot cold) near5 data near5 time) near10 cop$3	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:14 PM
L153	5	((hot cold) near5 data near5 time) near10 cop$3 near5 host	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:14 PM
L154	192	((hot cold) near5 data near5 time) near10 cop$3	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV,	OR	ON	ON	2022/06/22 12:15 PM

			MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L155	3920	copy near5 old near5 data	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:16 PM
L156	409	L94 AND ((copy migrate) near5 old near5 data)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:17 PM
L157	3	"20090276588"	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 12:20 PM
L158	7	"20060004957"	(US-PGPUB; USPAT;	OR	ON	ON	2022/06/22

			USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				12:20 PM
L159	0	L94 AND ((copy migrate) near5 timestamp near5 threshold)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:12 PM
L160	0	L94 AND ((copy migrate) near10 timestamp near5 threshold)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:13 PM
L161	5	L94 AND ((copy migrate) SAME (timestamp near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO,	OR	ON	ON	2022/06/22 02:13 PM

			NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L162	1885	L94 AND ((copy migrate) SAME (timenear5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:14 PM
L163	146	L94 AND ((copy migrate) SAME (time near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:14 PM
L164	6	L94 AND ((copy migrate) SAME (age near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:17 PM
L165	1803	((copy migrate) SAME (age near5 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY,	OR	ON	ON	2022/06/22 02:27 PM

			CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L166	13376	((copy migrate) SAME (old near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:28 PM
L167	969	L94 AND ((copy migrate) SAME (old near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:28 PM
L168	3	L94 AND ((copy migrate) SAME (old near5 data near10 threshold))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK,	OR	ON	ON	2022/06/22 02:29 PM

			TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L169	81	L94 AND ((copy migrate) SAME (age near5 data))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 02:31 PM
L170	0	L94 AND (send near5 host near5 erase near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 03:10 PM
L171	5	(send near5 host near5 erase near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 03:10 PM
L172	0	(send near5 host near5 P/E near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA,	OR	ON	ON	2022/06/22 03:12 PM

			EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L173	0	(send near5 host near5 P/E)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 03:12 PM
L174	9	(send near5 host near5 (program erase) near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 03:12 PM
L175	0	(send near5 host near5 (P/E))	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO;	OR	ON	ON	2022/06/22 03:14 PM

			JPO; DERWENT; IBM_TDB)				
L176	534	(host near5 (program erase) near5 count)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/06/22 03:14 PM

PE2E SEARCH - Search History (Interference)

There are no Interference searches to show.

METHOD AND APPARATUS FOR CONTROLLING MULTICHANNEL VOICE DATA

DOCUMENT ID
KR-20000025612-A

DATE PUBLISHED
2000-05-06

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
BAE SEONG HAN	N/A	N/A	N/A	KR

DATE FILED
1998-10-13

CPC CURRENT

TYPE	CPC	DATE
CPCI	H04L65/65	2022-05-01
CPCI	H04L65/65	2022-05-01
CPCI	H04L49/90	2013-01-01
CPCI	H04L47/2416	2013-01-01
CPCI	H04L65/1026	2013-01-01
CPCI	H04L65/1036	2013-01-01
CPCI	H04L12/64	2013-01-01
CPCI	H04M7/125	2013-01-01
CPCI	H04L12/66	2013-01-01
CPCI	H04L49/90	2013-01-01
CPCI	H04L47/2416	2013-01-01
CPCI	H04M7/125	2013-01-01
CPCI	H04L12/64	2013-01-01
CPCI	H04L65/1036	2013-01-01
CPCI	H04L65/1026	2013-01-01
CPCA	H04L65/1101	2022-05-01
CPCA	H04L2012/6408	2013-01-01
CPCA	H04L2012/6408	2013-01-01

Abstract

PURPOSE: A method and apparatus for controlling a multichannel voice data is to improve a processing speed by transmitting a voice data between an RTP(real time protocol) block and a multichannel packet controller. CONSTITUTION: A method for controlling a multichannel voice data comprises the steps of: fixing and allocating in advance a memory region by a predetermined unit in initializing a system, and fixing and allocating in advance an RTP header to the region; storing a voice packet data from an internet network and a public telephone network for a multichannel into the previously allocated memory region; and judging

whether or not one interrupt is periodically generated; transmitting the stored voice packet date in sequence whenever the periodic one interrupt is generated. An UDP(User Data gram Protocol) block transmits and receives the voice packet data to/from the internet network. A multichannel packet controller transmits and receives the voice packet data to/from the public telephone network and performs a packet separation.

Description

INVENTION-TITLE
BACKGROUND-ART
The present invention relates to a gateway system, and more particularly, to a method and apparatus for controlling multi-channel voice data in a gateway system between a public telephone network and an internet protocol packet network.
Recently, as a voice call terminal based on Internet Protocol (IP), such as an Internet phone, is commercialized, a gateway (Gate-Way) system that enables voice calls with an existing public telephone network has been developed and commercially available.
In general, the gateway system is divided into a call control conversion part and a voice data processing conversion part between both networks.
In particular, voice data processing conversion is a key technology for system performance and efficiency because it must be processed in high speed in real time.
In general, the voice processing portion of the gateway system is implemented using a standardized protocol having a structure as shown in FIG.
In the gateway system, the voice signal of the public telephone network is transmitted to the LAN network or the Internet network. When the analog voice signal received from the public telephone network is rearranged to a predetermined format in the public telephone network interface 101, the audio codec 102 When the digital voice data is converted into digital voice data and the multi-channel packet controller 103 periodically packetizes the digital voice data by a predetermined amount unit, a real-time protocol (RTP / RTCP) block 104 And UDP / IP (User Datagram Protocol / Internet Protocol) block 105 to add a UDP header and an RTP header for predetermined signal processing, and then rearrange the LAN interface 106 to a predetermined format for LAN or Internet. To the network.
On the contrary, the process of transmitting voice data in a LAN or Internet network to a public telephone network is performed when the user datagram protocol / Internet protocol (UDP / IP) block 105 and real time when the digital voice data is re-arranged in a predetermined format in the LAN interface 106. Real-Time Protocol (RTP) block 104 sequentially transmits the voice packet data to the multi-channel packet controller 103 by interpreting the UDP header and the RTP header and controlling the corresponding data. When 103 separates the voice packet data into individual voice data, the audio codec 102 converts the voice packet data into an analog voice signal and rearranges the voice packet data into a predetermined format in the public telephone network interface 101 to transmit the voice packet data to the public telephone network.
In the above processing / conversion process of voice data, multiple channels must be simultaneously processed in real time.
Therefore, since standardized protocols are used to interface between protocol layers, standardized protocol processing must first be satisfied when exchanging data between protocol layers, but the performance of voice calls varies depending on how voice data is controlled.
In general, the voice data format between protocol interfaces in a gateway system is illustrated in the example of FIG. 2.
This shows a process in which a voice signal on one line of a public telephone network corresponding to one audio codec is relayed to one UDP port.
That is, when the number of digital voice data converted by the audio codec 102 reaches a predetermined number, the multi-channel packet controller 103 generates RTP data by bundleing them into one packet and generates the RTP data in the real-time protocol (RTP) block 104. UDP data is generated by adding an RTP header to the RTP data, and the UDP block 105 adds a UDP header to the UDP data and transmits the UDP data to the Internet.
On the contrary, the voice data in the internet network is processed in the reverse order from the above, and

the voice data are separated into respective voice data in the multi-channel packet controller 103 and then converted into analog voice signals in the audio codec 102 and transmitted to the public telephone network. . In this case, one public telephone network channel is mapped to one port of the UDP block as a result of call control performed at the beginning of one voice call, and multi-channel voice processing / transmission in the gateway system is performed on several channels and UDP blocks of the public telephone network. It is to perform voice data relay between multiple ports.
The real-time protocol (RTP) is a standard protocol for transmitting real-time data such as voice or video in the Internet network, and the RTP header contains control information for controlling RTP data including voice data.
On the other hand, since the voice data processing in the multi-channel packet controller 103 and the real-time protocol (RTP) 104 has to be performed at high speed, the implementation structure and scheduling algorithm of both modules dominate overall system performance.
TECHNICAL-FIELD
However, conventionally, an RTP header is added to voice data when transmitting voice data to the Internet network, and then a UDP header is added to the data to which the RTP header is added, and conversely, after removing the UDP header when transmitting voice data to a public telephone network, the RTP header is removed. There was a problem that the processing speed is lowered by sequentially performing the decoding operation to the original voice data.
Accordingly, in order to solve the conventional problem, the present invention divides a buffer area into voice packet units and then allocates an RTP header to each area in a fixed manner so that the buffer area is controlled in a ring manner when transmitting voice data. It is an object of the present invention to provide a method and apparatus for controlling multi-channel voice data, which is designed to improve processing speed by performing voice data transmission between a block and a multi-channel packet controller.
DISCLOSURE
TECH-SOLUTION

In order to achieve the above object, the present invention provides an apparatus and method for implementing a multi-channel packet controller driver and a real-time protocol (RTP) block in one application program synchronized by one timer interrupt generated periodically. It is done.

In addition, in order to achieve the above object, the present invention relates to voice packets transmitted from a multi-channel packet controller to a real-time protocol (RTP) block in advance when the multi-channel packet controller controls the voice packets for each channel in advance. For real-time protocol blocks to control buffers allocated memory areas and for real-time protocol blocks to control multi-channel packet controllers from real-time protocol (RTP) blocks, the real-time protocol block is fixed in advance at system initialization. And the memory area is configured to control the allocated buffers.

When the memory area is fixed in advance, a 12-byte fixed RTP header as well as a voice packet is allocated to each unit buffer in advance.

Hereinafter, the present invention will be described in detail with reference to the drawings.

3, 5 and 6, the system for the implementation of the present invention, UDP block 205 for performing voice packet data transmission and reception with the Internet network, and voice signal transmission and reception with the public telephone network Multi-channel packet controller 204 for performing packetization or packet separation, and memory blocks 210a and 210b for storing voice packet data in the UDP block 205 and multi-channel packet controller 204. And an application task block 203 for controlling voice data transmission between the UDP block 205 and the multi-channel packet controller 204 at each occurrence of an interrupt, a timer 201 for periodically generating an interrupt, and this timer At 201, an interrupt service block 202 is provided to the application program block 203 when an interrupt occurs periodically.

The application program block 203 includes a circuit for driving a real-time protocol (RTP) and a multi-channel packet controller.

The memory blocks 210a and 210b are configured by fixedly pre-allocating a memory region corresponding to a voice packet of each channel at system initialization and by pre-allocating an RTP header for each memory region.

The reception buffer descriptor 220 and the transmission buffer descriptor having a flag indicating whether to store the voice packet corresponding to the memory areas of the memory blocks 210a and 210b and a fixed address indicating the start position of the voice packet data. It is further provided with 230.

The memory areas pre-allocated to the memory blocks 210a and 210b are controlled in a ring manner.

Referring to the operation and effect of the embodiment of the present invention configured as described above are as follows.

First, the task scheduling process will be described as follows.

The application task block 203 processes data transmitted and received from the UDP block 205 and the multichannel packet controller 204 in one task in consideration of periodically repeated packet data.

In this case, whenever an interrupt event from the multi-channel packet controller 204 occurs, a polling method is used instead of a data processing method.

That is, in case of the interrupt event driving method from the traffic source, an overhead of interrupt service switching time or task switching time is applied to each channel and each packet.

For example, if the packetization period is '30msec' and the number of channels is 'N', 'N' interrupt service switching and task switching should occur every 30ms on the public telephone network.

In addition, the processing similar to the above is applied to the events in the opposite direction, so that switching overhead due to numerous events is applied in a short time of '30 msec ', thereby lowering the process efficiency.

Therefore, in the present invention, when the task scheduling method as shown in FIG. 3 is applied, an interrupt is generated only once every '30 msec 'by the periodic interrupt timer 201, and the real time protocol (RTP) / multichannel packet is interrupted by the interrupt service routine 202. It is only necessary to pass the event flag once to the application task block 203 which can execute controller drive simultaneously.

This operation is performed in the same process as the signal flow of FIG.

The paused application task block 203 is redone whenever a timer interrupt occurs in the timer 201.

The memory blocks 210a and 210b are pre-allocated memory areas for storing voice packets from the multi-channel packet controller 204 and voice packets from the UDP block 205 during system initialization. The RTP header is also preallocated.

Accordingly, the multi-channel packet controller 204 continuously stores the digitally converted voice data transmitted from the public telephone network in the corresponding area of the memory block 210a, and the UDP block 205 stores the digital voice data transmitted from the Internet network. Stored successively in the area of block 210b.

At this time, when the data is completely filled in the corresponding memory area of the memory block 210a, the multi-channel packet controller 204 updates the status bit of the corresponding area of the reception buffer descriptor 220 to '1' and the corresponding memory area of the memory block 210b. When the data is

full in the memory area, the UDP block 205 updates the status bit of the corresponding area of the transmission buffer descriptor 230 to '1'.

Thereafter, when a periodic interrupt occurs in the timer 201, the interrupt service block 202 provides an application flag to the application task block 203 informing the occurrence of the periodic interrupt.

At this time, the application task block 203 checks the status bits of the reception buffer descriptor (RxBD) 220, which are updated by the multi-channel packet controller 204, for the voice call channel that is currently active.

Accordingly, if the status bit of the corresponding area of the reception buffer descriptor 220 corresponding to the currently active channel is '1' and there is data received from the public telephone network, the application task block 203 may determine the protocol data in the RTP header. After the update is performed, the data is transmitted to the UDP block 205 together with the RTP data filled by the multi-channel packet controller 204, and the status bit is updated to '0'.

The application program block 203 checks the port of the UDP block 205 corresponding to the channel to determine whether there is data received from the Internet network.

At this time, if there is a voice packet in the UDP block 205, the application task block 203 processes the RTP header and then copies the voice packet to the corresponding area of the memory block 210b and transmit buffer descriptor (TxBD) 230. Updates the status bit of the corresponding area of) to '1'.

Accordingly, the multi-channel packet controller 204 reads and transmits the RTP data stored in the corresponding area of the memory block 210b to the audio codec and updates the status bit of the corresponding area of the transmission buffer descriptor 230 to '0'.

The above operation is repeatedly performed for each of the 'N' channels, and when all of these operations are completed, the application program block 203 immediately enters the pause state.

That is, the buffer control process according to the present invention will be described with reference to FIGS. 5 and 6 as follows.

The memory blocks 210a and 210b are included in one memory circuit and have a fixed allocation area for storing the RTP header and the voice packet during system initialization in order to have a buffering structure for each voice channel.

FIG. 5 illustrates a case of storing and transmitting a voice packet from a public telephone network. The multi-channel packet controller 204 packetizes voice data from each audio codec and transmits the data to a corresponding buffer area of the memory block 210a. Copy the RTP data (voice packet) and update the status bits of the corresponding area of the receive buffer descriptor 220.

Accordingly, the application task block 203 may include the RTP header and the RTP data when the status bit of the region of the reception buffer descriptor 220 corresponding to the currently active channel is '1' when a periodic interrupt occurs. The data is sent to the UDP block 205 and its RTP header is updated.

6 illustrates a case of storing and transmitting a voice packet from the Internet network. The application task block 203 checks the UDP block 205 when a periodic interrupt occurs, and if there is received data, the memory block (After processing the RTP header of the area corresponding to the currently active channel in the area of 210b) and copying the received voice packet, the status bit of the corresponding area of the transmission buffer descriptor 230 is updated.

Accordingly, the multi-channel packet controller 204 checks whether the status bit of the corresponding

region of the transmission buffer descriptor 230 is '1' and if it is '1', the audio codec of the corresponding region of the memory block 210b. And update the status bit of the corresponding area of the transmission buffer descriptor 230 to '0'.

The operation is performed simultaneously at every periodic interrupt occurrence.
ADVANTAGEOUS-EFFECTS

As described in detail above, the present invention improves the operation speed by minimizing the load on the microprocessor by an application task for relaying voice packets received from a public telephone network and voice packets received from an Internet network. There is an effect that can accommodate a large number of channels.

In addition to the public telephone network, the present invention can be applied to a gateway system between a network for transmitting voice using a TDM scheme such as an ISDN or a frame relay and a packet network such as the Internet network, thereby obtaining the same effect.

Claims

A method for transmitting and receiving adaptive data between an internet network and a public telephone network, comprising: a first step of pre-assigning a memory area in predetermined units at the time of system initialization and pre-assigning an RTP header to each area thereof; and an internet network for a plurality of channels. And a second step of continuously storing voice data from a public telephone network in the corresponding memory area previously allocated, a third step of determining whether one interrupt occurs periodically, and one interrupt periodically And performing a fourth step of sequentially transmitting voice packet data in the internet network and a public telephone network for the plurality of channels stored in the memory.
A method for storing and transmitting voice data between a public telephone network and an Internet network, the method comprising: a first step of pre-allocating a memory area in predetermined units at system initialization and pre-allocating an RTP header to each of the areas; A second step of determining a periodic interrupt occurrence in the state, a third step of sequentially searching for a plurality of channels to determine an active channel, and determining whether data of a corresponding channel exists when determining the active channel. In the fourth step, if the data of the corresponding channel is present, voice data in the public telephone network is transmitted to the Internet network together with the RTP header, and at the same time, voice data in the Internet network is processed through the RTP header to transmit only the voice data to the public telephone network. The fifth step, and when the voice data transmission for the plurality of channels ends, Again stop the multi-channel, characterized in that to perform step 6. Method for controlling the voice data temporarily.
A gateway system for transmitting and receiving voice data between an internet network and a public telephone network, comprising: a UDP block for transmitting and receiving voice data with an internet network, a multi-channel packet controller for transmitting and receiving voice signals with a public telephone network, and a plurality of channels. A memory block for storing voice data from the UDP block and the multi-channel packet controller in a corresponding area, and controlling the memory block whenever a periodic interrupt occurs, thereby controlling the voice between the UDP block and the multi-channel packet controller for multiple channels. And an application program block for executing data transfer, and a timer for periodically generating an interrupt and providing the real-time protocol to the real-time protocol.
4. The apparatus of claim 3, wherein the memory block is configured by pre-allocating a region corresponding to the voice packet of each channel at system initialization.
5. The apparatus of claim 4, wherein the memory block is configured by fixedly pre-allocating a Real-Time Protocol (RTP) header for each region that is fixedly pre-allocated at system initialization.
4. The memory block of claim 3, wherein the memory block further comprises a transmission / reception buffer descriptor for storing an address indicating a storage bit and a status bit indicating whether to store a voice packet and a storage bit corresponding to each pre-allocated area. Multi-channel voice data control

device characterized in that the configuration.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl.[6]
H04L 12/66

(11) 공개번호 특2000-0025612
(43) 공개일자 2000년05월06일

(21) 출원번호 10-1998-0042765
(22) 출원일자 1998년10월13일

(71) 출원인 엘지전자 주식회사 구자홍
서울특별시 영등포구 여의도동 20번지
(72) 발명자 배성한
서울특별시 서초구 반포동 577-21
(74) 대리인 박장원

심사청구 : 있음

(54) 멀티 채널 음성 데이터 제어 방법 및 장치

요약

본 발명은 멀티채널 음성 데이터 제어 방법 및 장치에 관한 것으로 특히, 버퍼의 영역을 음성 패킷 단위로 분할한 후 각 영역에 RTP 헤더를 고정적으로 할당하여 음성 데이터 전송시 버퍼 영역을 링 방식으로 제어함에 의해 실시간 프로토콜(RTP) 블럭과 멀티 채널 패킷 제어기간의 음성 데이터 전송을 수행함으로써 처리 속도를 향상시키도록 함에 목적이 있다. 이러한 목적의 본 발명은 시스템 초기화시 소정량 단위로 메모리 영역을 미리 할당하고 그 각각의 영역에 RTP 헤더를 미리 할당하는 단계와, 태스크 일시 정지 상태에서 주기적인 인터럽트 발생을 판단하는 단계와, N개의 채널을 순차적으로 검색하여 액티브되는 채널을 판단하는 단계와, 상기에서 액티브되는 채널을 판단하면 해당 채널의 데이터가 존재하는지 판단하는 단계와, 상기에서 해당 채널의 데이터가 존재하면 멀티채널 패킷 제이기로부터의 데이터는 RTP 헤더를 갱신하여 UDP 블럭으로 전송하고 동시에 UDP 블럭으로부터의 데이터는 RTP 헤더를 처리하여 멀티채널 패킷 제어기로 전송하는 단계와, 상기에서 N개의 채널에 대한 데이터 전송이 종료되면 태스크를 일시 정지시키는 단계를 수행한다.

대표도

도4

명세서

도면의 간단한 설명

도 1은 일반적인 게이트웨이 시스템의 프로토콜 구조를 보인 예시도.

도 2는 도1 에서의 음성 데이터의 포맷을 보인 예시도.

도 3은 본 발명의 실시를 위한 시스템의 태스크 스케쥴링 구조를 보인 예시도.

도 4는 본 발명의 실시를 위한 신호 흐름도.

도 5 및 도 6은 본 발명에서의 버퍼 할당 구조 및 처리 과정을 보인 예시도.

* 도면의 주요부분에 대한 부호 설명 *

201 : 타이머 202 : 인터럽트 서비스 블럭
203 : 응용 프로그램 태스크 블럭 204 : 멀티 채널 패킷 제어기
205 : UDP 블럭 210a,210b : 메모리 블럭
220 : 수신 버퍼 디스크립터 230 : 송신 버퍼 디스크립터

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 게이트웨이(GateWay) 시스템에 관한 것으로 특히, 공중 전화망과 인터넷 프로토콜 패킷망간의 게이트웨이 시스템에 있어서 멀티 채널 음성 데이터 제어 방법 및 장치에 관한 것이다.

Samsung Ex. 1002, Page 187 of 622

최근 인터넷 폰과 같이 인터넷 프로토콜(IP)을 기반으로 하는 음성 통화 단말기가 상용화되면서 기존의 공중 전화망과 음성 통화를 가능하게 해 주는 게이트웨이(Gate-Way) 시스템이 개발되어 시판되고 있다.

통상적으로 게이트웨이 시스템은 양쪽 망간의 호 제어(Call Control) 변환 부분과 음성 데이터 처리 변환 부분으로 구분된다.

특히, 음성 데이터 처리 변환은 실시간으로 고속 처리되어야 하기 때문에 시스템의 성능과 효율성을 위한 핵심 기술이다.

일반적으로 게이트웨이 시스템에서의 음성 처리 부분은 도1 과 같은 구조의 표준화된 프로토콜을 사용하여 구현한다.

이러한 게이트웨이 시스템에서 공중 전화망의 음성 신호가 랜망 또는 인터넷망으로 전송되는 과정은 공중 전화망 인터페이스(101)에서 공중 전화망에서 수신된 아날로그 음성 신호를 소정 포맷으로 재배열하면 오디오 코덱(CODEC)(102)에서 디지털 음성 데이터로 변환하고 멀티 채널 패킷 제어기(103)에서 상기 디지털 음성 데이터를 주기적으로 소정량 단위씩 패킷화하면 실시간 프로토콜(RTP/RTCP ; Real-Time Protocol/Real-Time Transmission Control Protocol) 블럭(104) 및 UDP/IP(User Datagram Protocol/Internet Protocol) 블럭(105)에 의해 UDP 헤더, RTP 헤더를 부가하여 소정 신호 처리된 후 랜 인터페이스(106)에서 소정 포맷으로 재배열되어 랜(LAN) 또는 인터넷망으로 전송한다.

반대로, 랜망 또는 인터넷망에서의 음성 데이터가 공중 전화망으로 전송되는 과정은 디지털 음성 데이터가 랜 인터페이스(106)에서 소정 포맷으로 재열되면 UDP/IP(User Datagram Protocol/Internet Protocol) 블럭(105) 및 실시간 프로토콜(RTP : Real-Time Protocol) 블럭(104)에서 순차적으로 UDP 헤더, RTP 헤더를 해석하여 해당 데이터를 제어함에 의해 음성 패킷 데이터를 멀티 채널 패킷 제어기(103)으로 전송하고 그 멀티 채널 패킷 제어기(103)가 음성 패킷 데이터를 각각의 음성 데이터로 분리하면 오디오 코덱(102)에서 아날로그 음성 신호로 변환함에 따라 공중 전화망 인터페이스(101)에서 소정 포맷으로 재배열되어 공중 전화망으로 전송된다.

상기에서 음성 데이터의 처리/변환 과정은 실시간으로 동시에 여러 채널을 처리해야 한다.

따라서, 표준화된 프로토콜을 사용하여 각 프로토콜 계층간에 인터페이스하기 때문에 각 프로토콜 계층간에 데이터를 주고 받을 때 표준화된 프로토콜 처리가 우선적으로 만족되어야 하지만 음성 데이터를 어떻게 제어하느냐에 따라 음성 통화의 성능이 달라진다.

일반적으로 게이트웨이 시스템에서 프로토콜 인터페이스간의 음성 데이터 포맷은 도2 의 예시도와 같다.

이는 하나의 오디오 코덱에 대응되는 공중 전화망의 한 회선에서의 음성 신호가 하나의 UDP 포트까지 중계되는 과정을 도시한 것이다.

즉, 오디오 고덱(102)에서 변환되는 디지털 음성 데이터가 소정 개수가 되면 멀티채널 패킷 제어기(103)에서 하나의 패킷으로 묶어 RTP 데이터를 생성하고 실시간 프로토콜(RTP) 블럭(104)에서 상기에서 생성된 RTP 데이터에 RTP 헤더를 부가하여 UDP 데이터를 생성하며 UDP 블럭(105)에서 UDP 데이터에 UDP 헤더를 부가하여 인터넷망으로 전송하게 된다.

반대로, 인터넷망에서의 음성 데이터는 상기와 역순으로 순차 처리됨에 의해 멀티 채널 패킷 제어기(103)에서 각각의 음성 데이터로 분리된 후 오디오 코덱(102)에서 아날로그 음성 신호로 변환되어 공중 전화망으로 전송된다.

이 경우 하나의 음성 통화 초기에 이루어지는 호 제어의 결과에 의해 한 개의 공중전화망 채널은 UDP 블럭의 한 개의 포트와 매핑되는데, 게이트웨이 시스템에서의 다중 채널 음성 처리/전송은 공중전화망의 여러 채널과 UDP 블럭의 여러 포트간의 음성 데이터 중계를 수행하는 것이다.

상기에서 실시간 프로토콜(RTP)은 인터넷망에서 음성 또는 비디오와 같은 실시간성 데이터를 전송하기 위한 표준 프로토콜로서, RTP 헤더에는 음성 데이터가 실려 있는 RTP 데이터를 제어하기 위한 제어 정보들을 담고 있다.

한편, 상기에서 멀티채널 패킷 제어기(103)와 실시간 프로토콜(RTP)(104)에서의 음성 데이터 처리 과정이 고속으로 이루어져야 하기 때문에 양 모듈의 구현 구조 및 스케줄링 알고리즘이 전체 시스템 성능을 좌우한다.

발명이 이루고자하는 기술적 과제

그러나, 종래에는 인터넷망으로 음성 데이터 전송시 음성 데이터에 RTP 헤더를 부가한 후 그 RTP 헤더가 부가된 데이터에 UDP 헤더를 부가하고 반대로 공중 전화망으로 음성 데이터 전송시 UDP 헤더를 제거한 후 RTP 헤더를 제거하여 원래의 음성 데이터로 복호하는 동작을 순차적으로 수행함으로 처리 속도가 저하되는 문제점이 있었다.

따라서, 본 발명은 종래의 문제점을 개선하기 위하여 버퍼의 영역을 음성 패킷 단위로 분할한 후 각 영역에 RTP 헤더를 고정적으로 할당하여 음성 데이터 전송시 비피 영역을 링 방식으로 제이함에 의해 실시간 프로토콜(RTP) 블럭과 멀티 채널 패킷 제어기간의 음성 데이터 전송을 수행함으로써 처리 속도를 향상시키도록 창안한 멀티 채널 음성 데이터 제어 방법 및 장치을 제공함에 목적이 있다.

발명의 구성 및 작용

본 발명은 상기의 목적을 달성하기 위하여 멀티채널 패킷 제어기 드라이버와 실시간 프로토콜(RTP) 블럭은 주기적으로 발생되는 하나의 타이미 인터럽트에 의해 동기화되는 하나의 응용 프로그램 태스크내에

Samsung Ex. 1002, Page 188 of 622

구현하는 장치 및 방법을 제공함을 특징으로 한다.

또한, 본 발명은 상기의 목적을 달성하기 위하여 멀티채널 패킷 제어기로부터 실시간 프로토콜(RTP) 블럭으로 전달되는 음성 패킷들에 대해 멀티채널 패킷 제어기가 각 채널에 대한 음성 패킷을 제어하기 위해 시스템 초기화시 미리 고정적으로 메모리 영역이 할당된 버퍼들을 제어하며 실시간 프로토콜(RTP) 블럭으로부터 멀티 채널 패킷 제어기로 전달되는 음성 패킷들에 대해 실시간 프로토콜 블럭이 각 채널에 대한 음성 패킷을 제어하기 위해 시스템 초기화시 미리 고정적으로 메모리 영역이 할당된 버퍼들을 제어하도록 구성함을 특징으로 한다.

상기에서 미리 고정적으로 메모리 영역 할당시 각 단위 버퍼에 대해 음성 패킷뿐만이 아니라 12바이트의 고정적인 RTP 헤디도 미리 할당하도록 구성함을 특징으로 한다.

이하, 본 발명을 도면에 의거 상세히 설명하면 다음과 같다.

본 발명의 실시를 위한 시스템은 도3, 도5 및 도6 의 예시도에 도시한 바와 같이, 인터넷망과의 음성 패킷 데이터 송수신을 수행하는 UDP 블럭(205)과, 공중 전화망과의 음성 신호 송수신을 수행하면서 패킷화 또는 패킷 분리를 수행하는 멀티 채널 패킷 제어기(204)와, 상기 UDP 블럭(205) 및 멀티 채널 패킷 제어기(204)에서의 음성 패킷 데이터를 저장하는 메모리 블럭(210a)(210b)과, 인터럽트 발생때마다 상기 UDP 블럭(205)과 멀티 채널 패킷 제어기(204)간의 음성 데이터 전송을 제어하는 응용 프로그램 태스크 블럭(203)과, 주기적으로 인터럽트를 발생시키는 타이머(201)와, 이 타이머(201)에서 주기적으로 인터럽트 발생시 상기 응용 프로그램 태스크 블럭(203)으로 제공하는 인터럽트 서비스 블럭(202)으로 구성한다.

상기 응용 프로그램 태스크 블럭(203)은 실시간 프로토콜(RTP)과 멀티 채널 패킷 제어기 구동을 위한 회로를 포함하여 구성한다.

상기 메모리 블럭(210a)(210b)은 시스템 초기화시 각 채널의 음성 패킷에 대응하는 메모리 영역이 고정적으로 미리 할당되며 각 메모리 영역마다 RTP 헤더를 고정적으로 미리 할당하여 구성한다.

상기에서 메모리 블럭(210a)(210b)의 메모리 영역에 대응하여 음성 패킷의 지장 여부를 표시하는 플래그 및 음성 패킷 데이터의 저장 시작 위치를 나타내는 고정 어드레스를 구비한 수신 버퍼 디스크립터(220)와 전송 버퍼 디스크립터(230)를 더 구비하여 구성한다.

상기에서 메모리 블럭(210a)(210b)에 미리 할당된 메모리 영역은 링 방식으로 제어하는 것을 특징으로 한다.

이와같이 구성한 본 발명의 실시예에 대한 동작 및 작용 효과를 설명하면 다음과 같다.

먼저, 태스크 스케쥴링(Task Scheduling) 과정을 설명하면 다음과 같다.

응용 프로그램 태스크 블럭(203)은 주기적으로 반복되는 패킷 데이터의 특성을 고려하여 UDP 블럭(205)와 멀티채널 패킷 제어기(204)로부터 송수신되는 데이터들을 일괄적으로 하나의 태스크에서 처리한다.

이 경우 멀티채널 패킷 제어기(204)로부터의 인터럽트 이벤트가 발생할 때마다 데이터를 처리하는 방식을 사용하지 않고 주기적으로 폴링하는 방식을 사용한다.

즉, 트래픽 소스로부터의 인터럽트 이벤트 구동 방식의 경우 매 채널, 매 패킷 각각에 대해 인터럽트 서비스 스위칭 시간이나 태스크 스위칭 시간의 오버헤드가 적용된다.

예를 들어, 패킷화 주기가 '30msec'라 하고 하고 채널 개수가 'N'개라고 하면 공중 전화망에서는 수신 방향에 대해서 평균적으로 '30msec'마다 'N'번의 인터럽트 서비스 스위칭과 태스크 스위칭이 일어나야 한다.

또한, 반대 방향의 이벤트들에 대해서도 상기와 유사한 처리를 함으로 '30msec'라는 짧은 시간내에 수많은 이벤트들로 인한 스위칭 오버헤드가 적용되어 프로세스 효율이 훨씬 저하된다.

따라서, 본 발명은 도3 과 같은 태스크 스케쥴링 방식을 적용하면 주기적인 인터럽트 타이머(201)에 의해 '30msec'마다 한 번만 인터럽트가 발생하고 인터럽트 서비스 루틴(202)에서는 실시간 프로토콜(RTP)/멀티채널 패킷 제어기 구동을 동시에 실행할 수 있는 응용 프로그램 태스크 블럭(203)으로 이벤트 플래그를 한 번 전달하기만 하면 된다.

이러한 동작은 도4 의 신호 흐름과 동일한 과정으로 이루어진다.

일시 정지되어 있는 응용프로그램 태스크 블럭(203)은 타이머(201)에서 한 번의 타이머 인터럽트가 발생될 때마다 재실행된다.

그리고, 메모리 블럭(210a)(210b)은 시스템 초기화시 멀티 채널 패킷 제이기(204)로부터의 음성 패킷과 UDP 블럭(205)로부터의 음성 패킷을 저장하기 위한 메모리 영역이 미리 할당되고 그 각 메모리 영역에는 RTP 헤더도 미리 할당된다.

따라서, 멀티 채널 패킷 제어기(204)는 공중 전화망으로부터 전송되어 디지털 변환된 음성 데이터를 메모리 블럭(210a)의 해당 영역에 연속적으로 저장하며 UDP 블럭(205)은 인터넷 망으로부터 전송된 디지털 음성 데이터를 메모리 블럭(210b)의 해당 영역에 연속적으로 저장한다.

이때, 메모리 블럭(210a)의 해당 메모리 영역에 데이터가 모두 차면 멀티 채널 패킷 제어기(204)이 수신 버퍼 디스크립터(220)의 해당 영역의 상태 비트를 '1'로 갱신하며 메모리 블럭(210b)의 해당 메모리 영역에 데이터가 모두 차면 UDP 블럭(205)이 전송 버퍼 디스크립터(230)의 해당 영역의 상태 비트를 '1'로 갱신한다.

Samsung Ex. 1002, Page 189 of 622

이 후, 타이머(201)에서 주기적인 인터럽트가 발생하면 인터럽트 서비스 블럭(202)은 주기적인 인터럽트의 발생을 알리는 이벤트 플래그를 응용 프로그램 태스크 블럭(203)에 제공한다.

이때, 응용 프로그램 태스크 블럭(203)은 현재 액티브되어 있는 음성 통화 채널에 대해 각각 멀티 채널 패킷 제이기(204)가 갱신하는 수신 버피 디스크립티(RxBD)(220)의 상태 비트를 점검한다.

이에 따라, 현재 액티브된 채널에 대응하는 수신 버퍼 디스크립터(220)의 해당 영역의 상태 비트가 '1'로서 공중 전화망으로부터 수신된 데이터가 존재하면 응용 프로그램 태스크 블럭(203)은 RTP 헤더에 프로토콜 데이터의 갱신을 수행한 후 멀티 채널 패킷 제어기(204)에 의해 채워진 RTP 데이터와 함께 UDP 블럭(205)으로 전송한 후 상기 상태 비트를 '0'을 갱신한다.

그리고, 응용 프로그램 태스크 블럭(203)은 해당 채널에 대응하는 UDP 블럭(205)의 포트를 점검하여 인터넷망으로부터 수신된 데이터가 존재하는지 판단한다.

이때, UDP 블럭(205)에서의 음성 패킷이 존재하면 응용 프로그램 태스크 블럭(203)은 RTP 헤더를 처리한 후 음성 패킷을 메모리 블럭(210b)의 해당 영역에 복사하고 송신 버퍼 디스크립터(TxBD)(230)의 해당 영역의 상태 비트를 '1'로 갱신한다.

이에 따라, 멀디 채널 패킷 제어기(204)는 메모리 블럭(210b)의 해당 영역에 저장된 RTP 데이터를 읽어 오디오 코덱으로 전송하고 송신 버퍼 디스크립터(230)의 해당 영역의 상태 비트를 '0'으로 갱신한다.

상기와 같은 동작을 'N'개의 채널 각각에 대하여 반복적으로 수행하며 이러한 동작이 모두 종료되면 응용 프로그램 태스크 블럭(203)은 바로 일시 정지 상태로 돌입한다.

즉, 본 발명에서의 버퍼 제어 과정을 도5 및 도6 을 참조하여 설명하면 다음과 같다.

메모리 블럭(210a)(210b)는 하나의 메모리 회로에 포함되며 각 음성 채널당 버퍼링 구조를 가지기 위해 시스템 초기화시 RTP 헤더와 음성 패킷을 저장하는 영역이 연속적으로 고정 할당한다.

도5 는 공중 전화망으로부터의 음성 패킷을 저장하여 전송하는 경우를 도시한 것으로, 멀티채널 패킷 제어기(204)는 각 오디오 코덱으로부터 음성 데이터를 패킷화하여 데이터를 메모리 블럭(210a)의 해당 버퍼 영역에 RTP 데이터(음성 패킷)을 복사하고 수신 버퍼 디스크립터(220)의 해당 영역의 상태 비트를 업데이더한다.

이에 따라, 응용 프로그램 태스크 블럭(203)은 주기적인 인터럽트 발생시 수신 버퍼 디스크립터(220)의 영역중 현재 액티브된 채널에 대응하는 영역의 상태 비트가 '1'인 경우 RTP 헤더와 RTP 데이터를 포함하는 UDP 데이터를 UDP 블럭(205)로 전송하고 그 RTP 헤더를 갱신한다.

또한, 도6 은 인터넷망으로부터의 음성 패킷을 저장하여 전송하는 경우를 도시한 것으로, 응용 프로그램 대스크 블럭(203)은 주기적인 인더럽트 발생시 UDP 블럭(205)를 점검하여 수신 데이더가 있는 경우 메모리 블럭(210b)의 영역중 현재 액티브된 채널에 대응하는 영역의 RTP 헤더 처리하고 수신된 음성 패킷을 복사한 후 송신 버퍼 디스크립터(230)의 해당 영역의 상태 비트를 갱신한다.

이에 따라, 멀티 채널 패킷 제어기(204)는 송신 버퍼 디스크립터(230)의 해당 영역의 상태 비트가 '1'인지를 확인하여 '1'인 경우 메모리 블럭(210b)의 해당 영역의 음성 패킷을 오디오 코덱으로 전송하고 상기 송신 버퍼 디스크립터(230)의 해당 영역의 상태 비트를 '0'으로 갱신한다.

상기 동작은 주기적인 인터럽트 발생시마다 동시에 수행한다.

발명의 효과

상기에서 상세히 설명한 바와 같이 본 발명은 공중전화망으로부터 수신되는 음성 패킷과 인터넷망으로부터 수신되는 음성 패킷을 중계하는 응용 프로그램 태스크에 의해 마이크로 프로세서에 가해지는 부하(load)를 최소화하여 동작 속도를 개선함으로써 보다 많은 채널을 수용할 수 있는 효과가 있다.

또한, 본 발명은 공중전화망 이외에도 종합정보통신망(ISDN)등이나 프레임 릴레이등 TDM 방식으로 음성을 전송하는 망과 인터넷망과 같은 패킷망간의 게이트웨이 시스템에도 적용하여 동일한 효과를 얻을 수 있는 효과가 있다.

(57) 청구의 범위

청구항 1

인터넷망과 공중 전화망간의 음성 데이터 송수신 방법에 있어서, 시스템 초기화시 소정 단위로 메모리 영역을 미리 고정 할당하고 그 각각의 영역에 RTP 헤더를 미리 고정 할당하는 제1 단계와, 복수개 채널에 대한 인터넷망과 공중 전화망으로부터의 음성 데이터를 상기에서 미리 할당된 해당 메모리 영역에 각기 연속으로 저장하는 제2 단계와, 주기적으로 하나의 인더럽트가 발생하는지 판단하는 제3 단계와, 상기에서 주기적인 하나의 인터럽트가 발생할 때마다 상기에서 저장된 복수개의 채널에 대한 인터넷망과 공중 전화망에서의 음성 패킷 데이터를 순차적으로 상호 전송하는 제4 단계를 수행함을 특징으로 하는 멀티 채널 음성 데이터 제어 방법.

청구항 2

공중 전화망과 인터넷망간의 음성 데이터를 저장하여 상호 전송하는 방법에 있어서, 시스템 초기화시 소정량 단위로 메모리 영역을 미리 할당하고 그 각각의 영역에 RTP 헤더를 미리 할당하는 제1 단계와, 태스크 일시 정지 상태에서 주기적인 인터럽트 발생을 판단하는 제2 단계와, 복수의 채널을 순차적으로 검색하여 액티브되는 채널을 판단하는 제3 단계와, 상기에서 액티브되는 채널을 판단하면 해당 채널의 데

Samsung Ex. 1002, Page 190 of 622

이터가 존재하는지 판단하는 제4 단계와, 상기에서 해당 채널의 데이터가 존재하면 공중 전화망에서의 음성 데이터는 RTP 헤더와 함께 인터넷망으로 전송하고 동시에 인터넷망에서의 음성 데이터는 RTP 헤더를 처리하여 음성 데이터만을 공중 전화망으로 전송하는 제5 단계와, 상기에서 복수의 채널에 대한 음성 데이터 전송이 종료되면 대스크를 다시 일시 정지시키는 제6 단계를 수행하는 것을 특징으로 하는 멀티 채널 음성 데이터 제어 방법.

청구항 3

인터넷망과 공중 전화망간의 음성 데이터를 송수신하는 게이트웨이 시스템에 있어서, 인터넷망과의 음성 데이터 송수신을 수행하는 UDP 블럭과, 공중 전화망과의 음성 신호 송수신을 수행하는 멀티 채널 패킷 제어기와, 복수 채널에 대한 상기 UDP 블럭 및 멀티 채널 패킷 제어기에서의 음성 데이터를 해당 영역에 저장하는 메모리 블럭과, 주기적인 인터럽트 발생때마다 상기 메모리 블럭을 제어하여 복수 채널에 대한 상기 UDP 블럭과 멀티 채널 패킷 제어기간의 음성 데이터 전송을 실행시키는 응용 프로그램 태스크 블럭과, 주기적으로 인터럽트를 발생시켜 상기 실시간 프로토콜에 제공하는 타이머로 구성함을 특징으로 하는 멀티 채널 음성 데이터 제어 장치.

청구항 4

제3항에 있어서, 메모리 블럭은 시스템 초기화시 각 채널의 음성 패킷에 대응하는 영역을 고정적으로 미리 할당하여 구성함을 특징으로 하는 멀티 채널 음성 데이터 제어 장치.

청구항 5

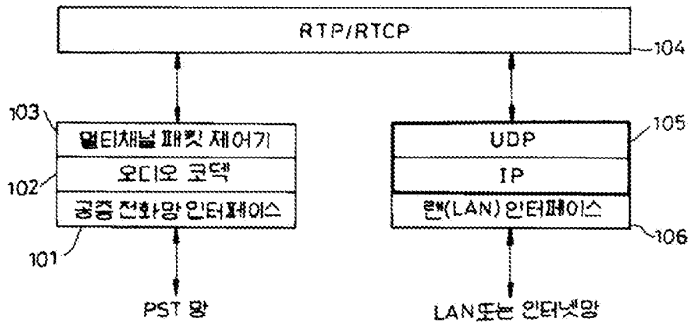
제4항에 있어서, 메모리 블럭은 시스템 초기화시 고정적으로 미리 할당되는 각 영역마다 RTP(Real-Time Protocol) 헤더를 고정적으로 미리 할당하여 구성함을 특징으로 하는 멀티 채널 음성 데이터 제어 장치.

청구항 6

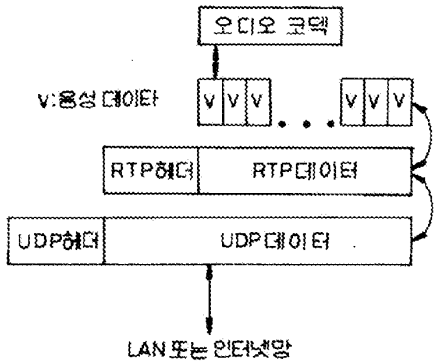
제3항에 있어서, 메모리 블럭은 미리 할당된 각 영역에 대응하여 음성 패킷의 저장 여부를 표시하는 상태 비트 및 음성 패킷 데이터의 지장 시작 위치를 나타내는 이드레스를 지장하는 송,수신 비피 디스크립터를 더 포함하여 구성함을 특징으로 하는 멀티 채널 음성 데이터 제어 장치.

도면

도면1

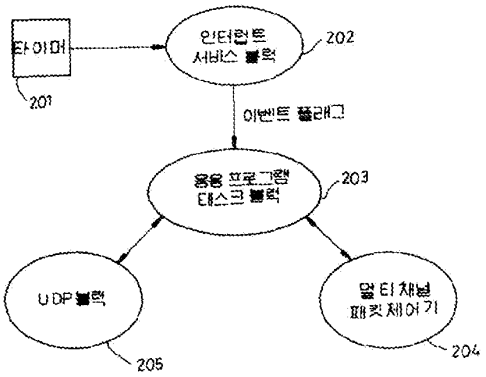


도면2

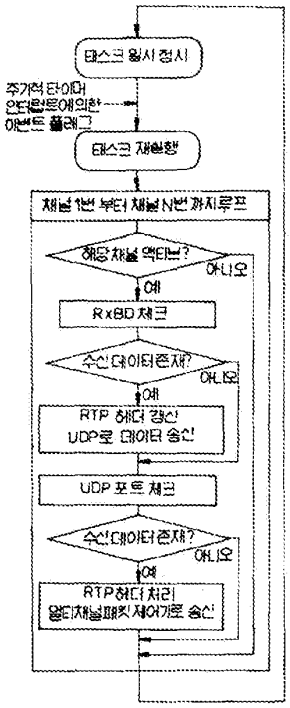


Samsung Ex. 1002, Page 191 of 622

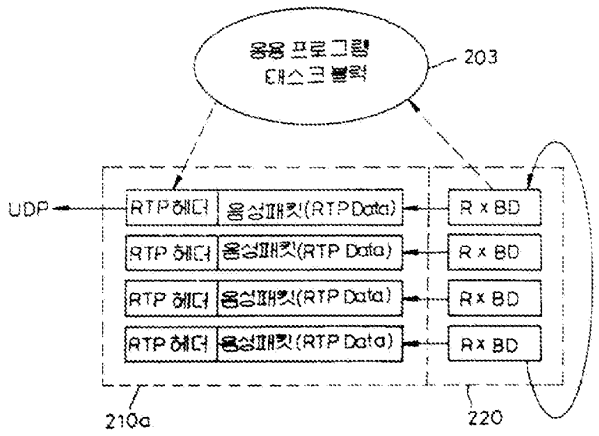
도면3


타이머
201
인터럽트 서비스 블럭
202
이벤트 플래그
응용 프로그램 태스크 블럭
203
UDP 블럭
205
멀티 채널 패킷 제어기
204

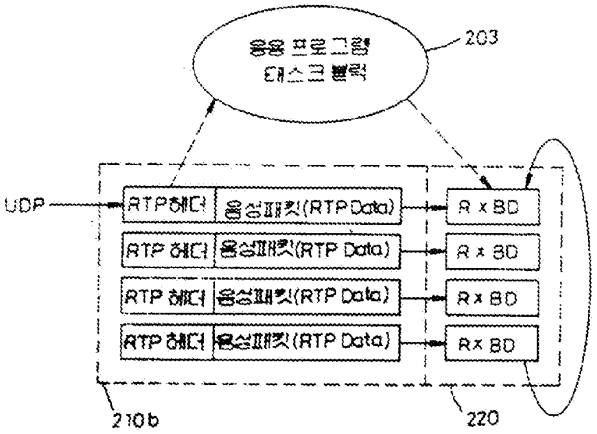
도면4


태스크 일시 정지
주기적 타이머 인터럽트에의한 이벤트 플래그
태스크 재실행
채널 1번 부터 채널 N번 까지 루프
해당 채널 액티브?
아니오
예
RxBD 체크
수신 데이터 존재?
아니오
예
RTP 헤더 갱신 UDP로 데이터 송신
UDP 포트 체크
수신 데이터 존재?
아니오
예
RTP헤더 처리 멀티채널패킷 제어기로 송신

도면5



도면6



Samsung Ex. 1002, Page 193 of 622

FILE SYSTEM USING NONVOLATILE SEMICONDUCTOR STORAGE DEVICE

DOCUMENT ID	DATE PUBLISHED
JP-2001051889-A	2001-02-23

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
TOMORI NOBUAKI	N/A	N/A	N/A	N/A

DATE FILED
1999-09-08

CPC CURRENT

TYPE	CPC	DATE
CPCI	G11C16/102	2013-01-01
CPCI	G11C16/105	2013-01-01

Abstract

PROBLEM TO BE SOLVED: To read data normally out of a nonvolatile semiconductor storage device even if an unexpected break of power supply is made in operation. SOLUTION: This file system stores some of pieces of state information on erasure blocks for each of the erasure blocks and stores some of pieces of state information on sectors in a file system memory part 3 and a flash memory part 5, sector by sector. The state information on each of the erasure blocks and the state information on each of the sectors in the file system memory part 3 are generated by reading data out of the flash memory part 5 when the file system 1 is started, and a file system control part 2 reads the data out of the flash memory part 5 according to the state information on the erasure blocks and the state information on the sectors in the file system memory part 3 and also secures the data in the flash memory part 5 even in case of an unexpected break of power supply in operation.

Description

DETAILED DESCRIPTION OF THE INVENTION
[0001]
[0001] 1. Field of the Invention [0002] The present invention relates to a file system using a nonvolatile semiconductor memory device such as a flash memory.
[0002]
2. Description of the Related Art Recently, a nonvolatile semiconductor memory device (for example, a flash memory) capable of electrically writing and erasing data has been replaced with a hard disk drive capable of writing and erasing data magnetically. The technology to use is drawing attention.
A flash memory has a data reading speed of about 10,000 times and a data writing speed of about 1,000 times as fast as a hard disk drive, and can be easily reduced in size and weight, and has excellent shock resistance. It has features such as high portability and is suitable for use in storing and reading low-volume data at high speed.

However, in a flash memory, in order to newly write data in a storage area in which data has already been written, an operation of temporarily erasing the written data is necessary. In addition, the unit for erasing data is as large as several kilobytes, a relatively long data erasing time of several tens of milliseconds to several seconds is required, and the number of guaranteed data writing and erasing is limited.
For this reason, when a flash memory is used as a substitute for a hard disk drive, the data handling unit is reduced to several hundreds of bytes, comparable to that of a hard disk drive, in consideration of the above drawbacks. Mechanism for managing
That is, it is necessary to provide a file system corresponding to the flash memory. Such a file system is disclosed in, for example, JP-A-6-95955.
Further, US Pat. No. 5,544,119 discloses a processing method at the time of power restoration after interruption of power supply.
U.S. Pat. No. 5,544,356 discloses a method for reconstructing a file system.
[0007]
However, in the technique described in the above publication, a flash memory and a file system for performing various controls are used in an environment where power supply using a battery or the like as a power supply is unstable. If the power supply is interrupted even momentarily due to battery exhaustion while the flash memory or file system is operating, that is, if unintended interruption of the power supply occurs, the data is stored in the flash memory. There is a possibility that the read data cannot be read normally.
Specifically, first, the current state of a storage area (hereinafter, referred to as a sector) in which data is stored is represented by sector state information. Since only one of the three states of “in use” and “invalid” is indicated, when the power supply is cut off when updating data in the sector,
Even if the data is not updated normally, the data may be regarded as normal based on the sector status information. The process of updating the data of this sector will be described below with reference to the flowchart of FIG.
According to the flowchart of FIG. 10, first, the sector status information of one sector storing the existing data to be updated is changed from "in use" to "invalid" (step 101), and then the other is changed. The sector status information of one sector is changed from "unused" to "used" (step 10).
2) After that, update data is written to the sector whose sector status information is "in use" (step 10).
3). Steps 101 to 103 are repeated until there is no more data to be written.
Here, immediately after step 102 or during the execution of step 103, when the power supply is interrupted and the data writing to the flash memory is interrupted, the writing of the update data to the sector is not completed normally. Nevertheless, the sector status information of this sector is "in use". Therefore, when the power supply is restored, the file system mistakenly recognizes the data stored in this sector as valid.
Second, when power is cut off when erasing all data in one block and reconstructing a file system, the same data is stored in two blocks or one block is lost. In some cases, a state in which all the data in the data was not completely erased occurred. This file system rebuilding process is shown in FIG.
This will be described below according to the flowchart of FIG.
According to the flowchart of FIG. 11, first, all the sectors are erased, and spare blocks whose sector status information of all the sectors is “unused” are reserved in advance.
The data to be stored in the block from which all data is to be erased, that is, the data of the sector whose sector status information is "invalid" and all other data except the sector status information are copied to the spare block (step 111). The block number included in the position information other than the sector whose sector state information is “invalid” in the block from which all data is erased is updated to the block number of the spare block (step 112). Finally, the block from which the data is copied is erased and secured as a spare block for the next file system rebuilding (step 113).
Here, if the power supply is interrupted before the execution of step 113 or during the execution of step 113, when the power supply is restored, the same data is stored in two blocks, or one block is stored. In this case, a state occurs in which all data in the data cannot be completely erased, and a new spare block cannot be secured.
In view of the above, the present invention has been made in view of the above-mentioned conventional problems, and is intended to provide a nonvolatile semiconductor device capable of normally reading data stored in a flash memory even when power supply is interrupted. It is an object to provide a file system using

a storage device.
[0015]
In order to solve the above-mentioned problems, a file system according to the present invention is a file system using a nonvolatile semiconductor memory device having a plurality of erase blocks each including a plurality of sectors. Storage means for storing, for each erase block, one of a plurality of pieces of state information about the erase block, and for each of the sectors, storing any one of a plurality of pieces of state information about the sector; And a guarantee unit for guaranteeing data already stored in the non-volatile semiconductor memory device based on the status information of each erase block and the status information of each sector in the storage unit at the time of access.
According to the present invention, a plurality of state information about a block and a plurality of state information about a sector are stored in the storage means. Therefore, even if unintended interruption of power supply occurs while the file system is operating,
When the power supply is restored, at the time of the initialization process at the time of startup, based on a plurality of state information about the block and a plurality of state information about the sector, the state of the block and the sector is accurately grasped and the flash memory is finely read. A file system that can be managed and has been executed immediately before the power supply was cut off without destroying the existing data stored in the flash memory and causing inconsistency in the existing data. Can be automatically recovered.
In one embodiment, the storage means is provided on a RAM having no restriction on updating data, and the assurance means is configured to access each of the erase blocks on the RAM when accessing the nonvolatile semiconductor memory device. And the status information of each sector.
When the RAM is used in this manner, there is no restriction on updating data, and the processing speed is increased.
In one embodiment, at least five pieces of status information on the erased block include “unused”, “data present”, “block full”, “data transfer”, and “original block erase”. A status for setting the status information of the spare block reserved for reconstruction from "unused" to "data transfer in progress"; and Means for transferring and storing the data of the spare block to the spare block, and after the transfer and storage of the data are completed, updating the block state information of the spare block from "data transfer" to "erasing the original block". The data in the erase block of the data transfer source is erased, the erase block of the data transfer source is secured as a new spare block, and the status information of the new spare block is referred to as “data”. And means for updating "or from" block full "to" unused ", and means for updating the status information of the data transfer destination of the spare block from the" source block erasing "to" Data Available ".
As described above, when five states of "unused", "data present", "block full", "data transfer", and "original block erase" are used as the state information of the erased block, Flash memory can be managed precisely by grasping the state accurately, without destroying existing data stored in flash memory, and without causing inconsistency in existing data,
It is possible to automatically recover the file system to the previous stage that was being executed immediately before the power supply was cut off.
In one embodiment, at least one spare block reserved in advance for reconstruction is provided for each erase block included in one nonvolatile semiconductor memory device and having a different block size.
In this case, there is an advantage that it is possible to adopt a flash memory that is suitable for the purpose of the system or is less expensive from among the flash memories having the equal block or boot block configuration.
In one embodiment, at least one spare block reserved in advance for reconstruction is provided for each erase block included in a plurality of nonvolatile semiconductor memory devices having different block sizes.
In this case, there is an advantage that the number of spare blocks required by the file system can be reduced. Therefore, there is an advantage that the file system can be configured at a low cost.
In one embodiment, the spare block reserved in advance for reconstruction has a size equal to or larger than the size of the erase block.
In this case, the boot block (for example, 8K
As a spare block for (a size of a byte), a block having a size larger than 8 Kbytes (for example, a size of 64 Kbytes) is allocated to the spare block, and the size of the spare block can be kept constant, thereby

simplifying management.
A file system according to the present invention is a file system using a nonvolatile semiconductor memory device having a plurality of sectors, wherein for each of the sectors, storage means for storing any one of a plurality of state information on the sector; And a guarantee unit for guaranteeing data already stored in the nonvolatile semiconductor memory device based on status information of each sector in the storage unit when accessing the nonvolatile semiconductor memory device. As multiple pieces of status information about
At least five states of "unused", "data writing", "data writing completed", "data valid", and "data invalid" are set, and the assurance means includes state information of a sector to which data is written. Means for updating from "unused" to "data writing" and means for updating the status information of the sector from "data writing" to "data writing completed" after data writing to the sector is completed. When new data is written to the sector, the status information of the sector is updated from "data write completed" to "data valid", and when updated data of existing data is written to the sector, After updating the status information of the other sector to which the existing data is written from “data valid” to “data invalid”, the state of the sector to which the updated data is written is updated. To update the information from the "data write complete" to "data valid".
According to the present invention, five pieces of status information on a sector are stored in the storage means. For this reason, even if unintended interruption of power supply occurs while the file system is operating, when power supply is restored, initialization of the sector at the time of startup is performed based on a plurality of state information about the sector. The flash memory can be managed precisely by accurately grasping the power supply, without destroying the existing data stored in the flash memory and causing inconsistency in the existing data, immediately before the power supply is cut off. The file system can be automatically restored to the previous stage that was being executed.
In another embodiment, a plurality of state information on the erase block includes “unused”, “data present”, “data transfer”, “erase original block”, “former original block”. At least five states may be set.
In this case, means for updating the state information of the spare erase block previously reserved for reconstruction from "unused" to "data transfer in progress", and updating the data in the erase block of the data transfer source to the spare erase block. Means for transferring and storing the data to the erased block, and after completion of the data transfer and storage, updates the block status information of the spare erased block from "data transfer" to "erasing the original block", and Means for erasing the data of the erase block, securing the erase block of the data transfer source as a new preliminary erase block, and updating the state information of the new preliminary erase block from "data present" to "unused"; The status information of the preliminary erase block at the data transfer destination is updated from “erasing the original block” to “formatting the original block”, and the new preliminary erase block of the data transfer source is updated. Kuhe writing and erasable number of new spare erased block and (formats) means,
After formatting is completed, a configuration may be considered that includes means for updating the status information of the spare erase block at the data transfer destination from “former block format” to “data present”.
As described above, when five states of "unused", "data present", "data transfer", "erasing of original block", and "formatting of original block" are used as the state information of the erased block. Flash memory can be managed precisely by accurately grasping the state of the erase block, without destroying existing data stored in the flash memory, and without causing inconsistency in existing data, It is possible to automatically recover the file system to the previous stage that was being executed immediately before the power supply was cut off. The method of not setting the state of “block full” as the state information of the erase block is particularly effective when the erase block is divided into sectors of unequal size and used. This is because, when writing data of unequal size to consecutive sectors in the same erase block, if the size of data to be written next is unknown, the state information of the erase block becomes “block full”. It is difficult to determine whether or not.
In the initialization of the file system when the power supply is restored, it is searched whether or not an erase block of “original block is being erased” exists as the state information of the erase block, and if the erase block exists, Having the same logical block number as the logical block number of the erase block; Means for acquiring an erase block whose erase block status information is "data present" or "block full" from the erase block status information in the storage means; erasing the acquired erase block data; Then,

the means for updating the state information of the erase block from "data present" or "block full" to "unused" and the state information of the erase block whose erase block state information is "erasing the original block" to the "original block". Means for updating from “block erasing” to “data present”.
At least five states of "unused", "data present", "data transfer", "erase of original block", and "formation of original block" are set as a plurality of state information about the erase block. The assurance means includes means for updating the state information of the spare erase block previously reserved for rebuilding the file system from "unused" to "data transfer in progress", and a data transfer source erase block. Means for transferring the data in the block to the spare erase block and storing the data, and after the transfer and storage of the data are completed, the status information of the spare erase block is "transferring data".
To the original block being erased, erase the data in the erased block of the data transfer source, and after erasure is complete, secure this erased block of the data transfer source as a new spare erase block. Means for updating the status information from "data present" to "unused", and updating the status information of the spare erase block at the data transfer destination from "erasing the original block" to "formatting the original block" and transferring the data. Means for formatting the original new pre-erased block, and after formatting is completed, the status information of the pre-erased block at the data transfer destination is changed to "formation of original block".
Means for updating the data to “data available”.
In the erase block, the number of times that the block can be erased is stored in each erase block. At the time of formatting, the number of times a new spare erase block is subtracted by one from the number of times that the spare erase block can be erased is stored in the spare block. Means for storing in the erase block, and means for selecting, as a data transfer source, the erase block having the largest number of block erase times among all the erase blocks except the spare erase block when reconstructing the file system. May be.
When the number of erasable blocks becomes equal to or less than 0, use of the erase block may be prohibited.
In the erase block, the number of times that the block can be erased is stored in each erase block. At the time of the formatting, the number of times that the new erase block is increased by one to the number of erases of the preliminary erase block is changed to the number of times of the preliminary erase. Means for storing in the block, and means for selecting the erase block having the largest number of block erase times as the data transfer source among all the erase blocks except the spare erase block when reconstructing the file system. Good.
When the number of erasable blocks becomes equal to or more than the specification value (maximum value) of the number of erasable times, use of the erase block may be prohibited.
At least one spare erase block reserved in advance for reconstructing the file system may be provided for each erase block included in one nonvolatile semiconductor memory device and having a different block size. .
At least one spare erase block reserved in advance for reconstructing the file system may be provided for each erase block included in a plurality of nonvolatile semiconductor memory devices having different block sizes. .
The spare erase block reserved in advance for reconstructing the file system may have a size larger than the erase block size.
In a file system using a nonvolatile semiconductor memory device having a plurality of sectors, storage means for storing any one of a plurality of pieces of state information on the sector for each of the sectors; Means for assuring data already stored in the nonvolatile semiconductor memory device based on the state information of each sector in the storage means at the time of access, and providing a plurality of pieces of state information for the sector. "Unused", "Writing data", "Data writing completed", "Data valid",
At least five states of "data invalid" are set, and the assurance means updates the state information of the sector to which data is written from "unused" to "data is being written"; After the writing is completed, means for updating the status information of the sector from "data writing in progress" to "data writing completed", and when data is newly written in the sector, the status information of the sector is changed to "data writing".
When updating from “Complete” to “Data valid” and writing the updated data of the existing data to the sector, the status information of the other sector where the existing data is written is changed from “Data valid” to “Data invalid”. Means for updating the status information of the sector in which the updated data is

written from "data write completed" to "data valid". There.
In the initialization of the file system when the power supply is restored, it is searched whether or not an erase block of “original block is being erased” exists as the state information of the erase block. Means for acquiring the data transfer source block to the erase block from the state information of each erase block in the storage means; erasing the acquired data of the data transfer source block; From data available to unused
Means for updating the status information of the erased block whose erase block status information is "erasing the original block" from "erasing the original block" to "formatting the original block",
Means for formatting the erased block from which data has been erased and securing it as a new spare erased block, and after formatting is completed, the status information of the erased block whose format information is "former block format" is changed to "former block format". Means for updating the data to “data available”.
In the initialization of the file system when the power supply is restored, it is searched whether or not an erase block of “former block format” exists as the state information of the erase block. Means for acquiring the data transfer source block to the erase block from the state information of each erase block in the storage means, means for formatting the acquired data transfer source block and securing it as a new spare erase block, Means for updating the status information of the erased block whose erase block status information is "former block format" from "former block format" to "data present".
[0044]
Embodiments of the present invention will be described below with reference to the accompanying drawings.
FIG. 1 shows an embodiment of the file system of the present invention. In FIG. 1, 1 is a file system, 2 is a file system control unit that controls the processing of the file system 1, 3 is a file system memory unit that the file system 1 uses to store control information,
Reference numeral 4 denotes a flash memory control unit that controls data processing to the flash memory unit 5, reference numeral 5 denotes a flash memory unit, and reference numeral 6 denotes an application or an operating system that requests the file system 1 to process data.
FIG. 2 is a configuration diagram of data stored in the flash memory unit 5 in FIG. Each erase block 51 of the flash memory unit 5 is divided into a plurality of physical sectors 52, which are data handling units, and has a unique physical sector number within each erase block 51. Assuming that each erase block 51 has a size of 64 Kbytes and each physical sector 52 has a size of 512 bytes, each erase block 51 has 128 physical sectors 52.
Here, an arbitrary one of the plurality of erase blocks 51 in the flash memory unit 5 is erased.
1 is a data-unused spare erase block used for rebuilding the file system 1, and the first sector 52 of each erase block 51 is a block control sector 52 for containing control information of each erase block.
a is reserved in advance.
In the erase block 51, the block control sector 52a includes the erase block 5 from the beginning.
2 bytes as status information of the first sector, 2 bytes as status information of the physical sector 52 of the sector # 1, and sectors # 2 to # 1.
2 bytes as status information of each of the 27 physical sectors 52 are sequentially arranged, so that 2 bytes × 128 =
It has 256 bytes of information.
The state information of the erasure block 51 is composed of logical block numbers 0 to 255 to which the file system control unit 2 allocates, and data indicating any of five types of block states. The five types of block states are (1)
11111111b representing "unused" (b represents a binary number; the same applies hereinafter); (2) 11 representing "during data transfer"
111110b, (3) 1 representing “under erasing of original block”
1111100b, (4) 1111 representing "data available"
1000b, (5) 11110 representing “block full”
000b.
Like the status information of the erasure block 51, the status information of each sector 52 of the erasure block 51 is a logical sector number unique to the system of 0 to 4095 to which the file system control unit 2 allocates, and five types. Data indicating any one of the sector states. The five types of sector states are (I)

1111b representing "unused", and (II).
1110b representing "writing data", (III)
1100b representing "data writing completed", (IV)
1000b representing “data valid” and (V) 0000b representing “data invalid”.
FIG. 3 is a configuration diagram of data relating to each erase block 51 stored in the file system memory unit 3 in FIG. 1 (hereinafter, referred to as a block information table 10).
In FIG. 3, a physical block number, a logical block number, and a block state are defined as one information unit. For example, an erase block in which a physical block number = 0, a logical block number = 0, and a block state = data exists. 51, physical block number = 1, logical block number =
1, indicating that there is an erase block 51 or the like in which the block state = block full.
FIG. 4 is a diagram showing the structure of data relating to each sector 52 stored in the file system memory unit 3 in FIG. 1 (hereinafter referred to as a sector information table 11).
In FIG. 4, a physical block number, a physical sector number, a logical sector number, and a sector state are defined as one information unit. For example, a physical block number = 0, a physical sector number = 0, a logical sector number = 100, a sector State = data valid sector 52, physical block number = 0, physical sector number = 1, logical sector number = 10,
Sector status = indicates that there is a sector 52 in which data is invalid.
FIG. 5 is a configuration diagram of data (hereinafter, referred to as a sector) in which the number of sectors 52 is arranged according to each sector state for each erase block 51 stored in the file system memory unit 3 in FIG. Information table 12).
In FIG. 5, the physical block number, the number of unused sectors, the number of valid data sectors, and the number of invalid data sectors are defined as one information unit. For example, the physical block number = 0, the number of unused sectors = 100, the valid data number Erase block 51 with sector number = 20, data invalid sector number = 7, physical block number = 1, unused sector number =
0, number of valid data sectors = 50, number of invalid data sectors =
This indicates that the erase block 51 or the like at 77 exists.
The block information table 10 of FIG. 3, the sector information table 11 of FIG. 4, and the sector information table 12 of FIG. 5 indicate the block control of each erase block 51 from the flash memory unit 5 when the file system 1 starts. Each is created by reading the sector 52a.
The block information table 10, the sector information table 11,
Even when the sector information table 12 is not created, the information in these tables is stored in the block control sector 52a of each erase block 51, so that the file system 1 can be used. However,
Since the operation speed of the file system memory unit 3 is faster than that of the flash memory unit 5, it is better to create and store the block information table 10, the sector information table 11, and the sector information table 12 in the file system memory unit 3. The internal state of the flash memory unit 5 and the storage location of data can be grasped at high speed. Further, since data exchange with the flash memory unit 5 is reduced, the file system memory unit 3
Is more effective in increasing the operation speed of the file system 1.
FIG. 6 is a flowchart relating to a process of reading data stored in the flash memory unit 5 in the present embodiment.
First, when a request to read data from the flash memory unit 5 occurs in the application or the operating system 6, a logical sector number is given from the application or operating system 6 to the file system control unit 2 together with the read request. The file system control unit 2 accesses the sector information table 11 in the file system memory unit 3 to search for a given logical sector number, and the sector state corresponding to this logical sector number becomes “data valid”. Is confirmed, a physical sector number corresponding to the logical sector number is obtained, and the physical sector number is given to the flash memory control unit 4 (step 121). The flash memory control unit 4 accesses the flash memory unit 5 based on the physical sector number, reads a sector at a physical storage location corresponding to the physical sector number, and stores the read data in the file system control. (Step 122). If there is remaining data to be read, the process returns to step 121 (step 1).
(23, Yes), if there is any remaining data to be read, the process is terminated (Step 123, No).
FIGS. 7A to 7C are flowcharts relating to a process of writing data to the flash memory unit 5 in the present embodiment.

First, when a request for writing data to the flash memory unit 5 occurs in the application or the operating system 6, a logical sector number and data to be written are given from the application or the operating system 6 to the file system control unit 2 together with the write request. Can be The file system control unit 2 accesses the sector information table 12 in the file system memory unit 2 and acquires the physical block number (indicating the writable erase block 51) having the largest number of unused sectors (step 131). Is confirmed (step 132), and if this physical block number could not be obtained (step 132, N
o), the file system 1 to be described later is reconstructed to secure the writable erase block 51 (step 133).
If the physical block number with the largest number of unused sectors can be obtained (step 132, Yes),
The file system control unit 2 accesses the sector information table 11 and, among the physical sector numbers corresponding to the physical block numbers acquired in step 131 or 133, the physical sector number whose sector status is “unused” And further accesses the sector information table 12 to set the number of unused sectors corresponding to the physical block number to −1,
The number of valid data sectors is set to +1 (step 134).
Thereafter, the file system control unit 2 accesses the block information table 10 and checks whether the block state corresponding to the physical block number acquired in step 131 or 133 is “unused” (In step 135), if the block state is “unused” (step 135, Yes), the flash memory unit 5 is accessed through the flash memory control unit 4 and the block control sector 52a of the erase block 51 corresponding to the physical block number is accessed. Is updated to the physical block number, and the block state of the block control sector 52a is updated from “unused” to “data present”.
Similarly, the block information table 10 is accessed and step 1
The logical block number corresponding to the physical block number acquired in 31 or 133 is updated to the physical block number, the block status is updated from “unused” to “data present” (step 136), and the process proceeds to step 137. If the block state is not "unused" (step 135, N
o), jump from step 136 to step 137;
Next, the file system control unit 2 accesses the sector information table 12 and executes step 131 or 133. It is checked whether the number of unused sectors corresponding to the physical block number acquired in (1) is 0 (step 137), and if the number of unused sectors is 0 (step 137, Yes), the flash memory unit is sent through the flash memory control unit 4. 5
And updates the block state of the block control sector 52a of the erase block 51 corresponding to the physical block number from “data present” to “block full”. Similarly, the block information table 10 is accessed, and the block state corresponding to the physical block number acquired in step 131 or 133 is also updated from "data present" to "block full" (step 138).
Next, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4, and deletes the physical block number obtained in step 131 or 133 and the erasure corresponding to the physical sector number obtained in step 134. The logical sector number given to the file system control unit 2 is written to the logical sector number of the block control sector 52a of the block 51, and the sector status of this sector 52 is updated from "unused" to "data writing". Similarly, the sector information table 11 is accessed, and the physical block number acquired in step 131 or 133 and the logical sector number corresponding to the physical sector number acquired in step 134 are assigned to the logical sector number given to the file system control unit 2. The sector status is updated from "unused" to "data writing" (step 139).
Thereafter, the file system control unit 2 sends the flash memory unit 5 through the flash memory control unit 4.
And writes the data given to the file system control unit 2 into the physical block number obtained in step 131 or 133 and the sector 52 obtained in step 134 and corresponding to the physical sector number (step 140).
After completion of writing to the sector 52 in the flash memory unit 5, the file system control unit 2 deletes the physical block number acquired in step 131 or 133 and the erase block 51 corresponding to the physical sector number acquired in step 134. The sector state of the block control sector 52a is updated from "data writing" to "data writing completed". Similarly, the sector information table 11 is accessed, and the physical block number acquired in step 131 or 133 and the sector state corresponding to the physical sector number acquired in step 134 are updated from "data writing" to "data writing completed". (Step 141).
Then, the file system control unit 2 checks whether the data written in the sector 52 in the flash memory unit 5 is update data of the data already written in the flash memory unit 5 (step 14).
2). Here, if the existing data is to be updated (step 1

42, Yes), the file system control unit 2 accesses the sector information table 11, has the same logical sector number as the logical sector number of the sector 52 for which the writing was performed in step 140, and executed the writing in step 140. A physical sector number and a physical block number of another sector 52 different from the physical sector number of the sector 52,
That is, the physical sector number and physical block number of the sector 52 in which the existing data is stored are obtained (step 14).
3). Then, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4, and changes the physical block number and the sector state of the block control sector 52a of the erase block 51 corresponding to the physical sector number to “data valid”. "To" data invalid ". Similarly, the sector information table 11 is accessed to update the physical block number and the sector state corresponding to the physical sector number from “data valid” to “data invalid”. Further, the sector information table 12 is accessed, the number of valid data sectors corresponding to this physical block number is set to -1, and the number of invalid data sectors is incremented to +1 (step 144).
Thereafter, the file system control unit 2 sends the flash memory unit 5 through the flash memory control unit 4.
And updates the sector state of the block control sector 52a of the erase block 51 corresponding to the physical block number acquired in step 131 or 133 and the physical sector number acquired in step 134 from "data writing completed" to "data valid". I do. Similarly, the sector information table 11 is accessed, and the physical block number and the sector state corresponding to the physical sector number are updated from "data write completed" to "data valid" (step 145).
Subsequently, the file system control unit 2
Confirms whether write data still remains (step 146), and if write data still remains (step 146, Yes), returns to step 131, and if there is no write data (step 146).
No), end.
FIGS. 8A and 8B are flowcharts relating to the rebuilding process of the file system 1 in the present embodiment.
First, as described with reference to the flowcharts of FIGS. 7A to 7C, when the writable erase block 51 cannot be obtained (Step 132, No), the file system is used to secure the writable erase block 51. Is reconstructed (step 133). The processing of step 133 corresponds to the processing shown in the flowchart of FIG.
8A and 8B, the file system control unit 2 accesses the sector information table 12, acquires the physical block number having the largest number of data invalid sectors, and accesses and acquires the block information table 10. The logical block number corresponding to the physical block number is obtained (step 151). Then, the file system control unit 2
The flash memory unit 5 is accessed through the flash memory control unit 4, and the logical block number of the block control sector 52a of the spare erase block 51 previously reserved in an unused state for the rebuilding process of the file system is obtained in step 151. The block state of the block control sector 52a is updated from "unused" to "data transfer in progress". Similarly, the block information table 10 is accessed, the logical block number of the preliminary erase block 51 is updated to the logical block number acquired in step 151, and the block state is updated from "unused" to "data transfer in progress" (step 152).
Next, the file system control unit 2 accesses the sector information table 11 in order to efficiently copy the data from the erasure block 51 of the physical block number obtained in step 151 to the preliminary erasure block 51, and The physical sector number corresponding to the physical block number obtained in 151 is searched for and obtained from the physical sector numbers whose sector status is "data valid" (step 153). If there is a sector 52 whose sector state is “data valid” (step 154, Yes), the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and proceeds to step 1.
The sector state “data valid” of the block control sector 52 a of the erase block 51 corresponding to the physical block number acquired at 51 and the physical sector number acquired at step 153 is changed to the block control sector corresponding to the same physical sector number of the preliminary erase block 51. Copy to sector state 52a. Similarly, the file system control unit 2 accesses the sector information table 11 and changes the physical block number acquired in step 151 and the sector state “data valid” of the physical sector number acquired in step 153 to the same physical Copies to the sector state of the sector number. At the same time, the file system control unit 2 accesses the flash memory unit 5 through the flash memory

control unit 4, and
The data is copied from the sector 52 of the erase block 51 corresponding to the physical block number acquired at 51 and the physical sector number acquired at step 153 to the sector 52 of the same physical sector number of the spare erase block 51 (step 155).
The file system control unit 2 executes step 1
It is confirmed whether or not there is an unprocessed sector 52 in the erase block 51 of the physical block number obtained in 51 (step 156). If there is an unprocessed sector 52, the process returns to step 153. If there are no unprocessed sectors 52,
Since the data copy is completed, the file system control unit 2 accesses the sector information table 12 and The number of valid data sectors corresponding to the physical block number acquired in step 1 is copied to the number of valid data sectors in the spare erase block 51, the number of invalid data sectors is set to 0, and the number of valid data sectors, the number of invalid data sectors, and the unused sectors are set. The number of unused sectors is calculated and set so that the sum of the numbers becomes the total number of sectors (127) (step 157).
Next, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and changes the block state of the block control sector 52a of the preliminary erase block 51 from “data transfer” to “original block erase”. Medium ”(step 158),
The data of the erase block 51 of the physical block number acquired in step 151 is erased (step 159),
Block control sector 5 of preliminary erase block 51
The block status of 2a is updated from "erasing the original block" to "data present". Similarly, the file system control unit 2
Accesses the block information table 10 and updates the block state of the preliminary erase block 51 from "data transfer" to "original block erase" (step 158), and step 1
After 59, the block state of the preliminary erase block 51 is updated from "original block erase" to "data present". As a result, the block having the physical block number acquired in step 151, that is, the block from which data has been erased, is secured as a new spare erase block 51 (step 160).
Here, the block state of the block control sector 52a of the erase block 51 from which data has been erased in step 159 is automatically changed from "data present" or "block full" to "unused" by data erase.
Becomes Also, a block information table 10 and a sector information table 11
In the sector information table 12, the erase block 51 from which data has been erased in step 159 becomes a spare erase block, and the spare erase block is not referred to when writing data to the flash memory unit 5. It is not necessary to update the item corresponding to the physical block number.
In the present embodiment, the process of rebuilding the file system 1 has been described in the case of erasing one erase block. However, a series of processes from step 151 may be executed a plurality of times. is there.
FIGS. 9A to 9D are flowcharts relating to the initialization of the file system 1 in the present embodiment. This initialization is performed when the file system 1 is started up.
In other words, it is executed once when the power supply is resumed after the power supply is once stopped, and guarantees reading of data in the flash memory unit 5.
First, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4, reads the block state of the block control sector 52a of each erase block 51, and, based on the read block information, The block information in the block information table 10 is updated (step 171), and it is confirmed whether the remaining erased block 51 exists (step 17).
2) The process of step 171 is repeated until the remaining erase block 51 is exhausted.
Next, the file system control unit 2 accesses the block information table 10 updated in step 171 to confirm whether or not the block state “data transfer is in progress” (step 173). If "transferring" exists, the physical block number of this block state is acquired, and the block state corresponding to this physical block number is updated from "data transfer" to "unused" (step 174). At the same time, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and erases the data of the block having this physical block number.
Here, step 151 in FIGS. 8A and 8B
When the power supply is interrupted in the middle of steps 157 to 157 and the data transfer to the preliminary erase block 51 is interrupted, the block state of the preliminary erase block 51 is held as "data

transfer in progress". When power supply is resumed after this,
In the processing of steps 173 and 174, the data of the preliminary erase block 51 in the block state “data transfer” in which the data transfer is interrupted is erased, and this preliminary erase block 5
The block status of No. 1 becomes “unused”. This prevents incomplete data from being read from the preliminary erase block 51.
Next, the file system control unit 2 accesses the block information table 10 updated in step 171 and checks whether there is a block whose block state is “erasing the original block” (step 175). ,
If there is a block whose block state is “erasing the original block”, a physical block number having the same logical block number as the logical block number of this block and the block state of “data present” is acquired. Then, the file system control unit 2 accesses the block information table 10 and updates the block state corresponding to the physical block number to “unused”. At the same time, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and erases the data of the block having this physical block number.
Further, the file system controller 2 deletes the erasure block 51 of the physical block number whose block status acquired in step 175 is “the original block is being erased”.
Update the block status information of the block control sector 52a from “being erased from the original block” to “data present”,
The block state corresponding to the same physical block number in the block information table 10 is updated to "data present" (step 176).
Thereafter, the file system control section 2 reads the sector state of the block control sector 52a of the erase block 51, and updates the sector information table 11 in the file system memory section 3 based on the sector information (step 177).), It is confirmed whether or not the remaining sector information exists (Step 178). If there is the remaining sector information, the process returns to Step 177 to return to the remaining sector 5
The process is repeated until there are no more erased blocks 51, and if there are remaining erased blocks 51, the process returns to step 177 and the process is repeated until there are no more erased blocks 51 remaining.
Here, step 158 in FIGS. 8A and 8B
When the power supply is interrupted in the middle of steps 160 to 160, the block state of the preliminary erase block 51 is held as "the original block is being erased".
Data transfer to has been completed. After that, when the power supply is restarted, in the processing of steps 175 to 178,
The data of the preliminary erase block 51 is retained, and the data of the erase block 51 of the data transfer source is erased. This prevents incomplete data from being read from the erase block 51 of the data transfer source.
Next, the file system control unit 2 accesses the sector information table 11 updated in step 177 and checks whether there is a sector 52 whose sector status is “data writing” (step 179). If there is a sector 52 whose sector status is "data writing", the physical block number and physical sector number corresponding to the sector status "data writing" obtained in step 179 are obtained. Then, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and deletes the erase block 51 corresponding to the physical block number and the physical sector number.
Of the block control sector 52a from "data writing" to "data invalid".
Similarly, the file system control unit 2 sets the sector information table 1
1 is accessed and the sector status corresponding to these physical block numbers and physical sector numbers is changed to "data writing".
Is updated to "data invalid" (step 180).
Next, the file system control unit 2 accesses the sector information table 11 and checks whether or not there is a sector 52 whose sector status is "data write completed" (step 181). If there is a sector 52 of “data write completed”, this sector 5
It is confirmed whether there is a sector 52 having the same logical sector number as the logical sector number of No. 2 and the sector status is "data valid" (step 182). Obtain the physical block number and physical sector number. Then, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and changes the physical block number and the sector state of the block control

sector 52a of the erase block 51 corresponding to the physical sector number to “data valid”. "To" data invalid ". Similarly, the file system control unit 2 accesses the sector information table 11 and changes the sector state corresponding to these physical block numbers and physical sector numbers from “data valid” to “data invalid”.
Is updated to (step 183).
Next, the file system control unit 2 accesses the sector information table 11 and obtains the physical block number and physical sector number of the sector 52 whose sector status obtained in step 181 is “data write completed”. Then, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4, and the block control sector 5 of the erase block 51 corresponding to these physical blocks and physical sector numbers.
The sector status of 2a is updated from "data write completed" to "data valid". Similarly, the file system control unit 2 accesses the sector information table 11 and updates the sector status corresponding to these physical block numbers and physical sector numbers from “data write completed” to “data valid” (step 184). .
Here, step 139 in FIG. 7A to FIG.
If the power supply is interrupted in the middle of steps 141 to 141 and the data writing is interrupted, the sector state is held as "data writing", and if the data writing is completed, the sector state becomes "data writing completed". Is held as After that, when the power supply is restarted, in the processing of steps 179 to 184, the data of the sector 52 in which the data writing is interrupted is erased, and the sector status of this sector 52 is changed to "data invalid". ". This prevents incomplete data from being read from the sector 52 in which data writing has been interrupted. Further, the data of the sector 52 in the sector state “data write completed” is held and this sector state is updated to “data valid”, so that data can be correctly read from this sector 52.
Next, the file system control unit 2 accesses the sector information table 11 and, for each physical block number, the number of sector states “unused” corresponding to the same physical block number and the number of sector states “data valid”. And the number of sector states “data invalid” are counted, and based on these count values, the physical block number, unused sector number, data valid sector number and data invalid of the sector information table 12 in the file system memory unit 3 are calculated. Update the number of sectors (step 18
5) It is confirmed whether or not the remaining physical block number exists (step 186). If the remaining physical block number exists, the process returns to step 185 and repeats until there is no remaining physical block number.
Thereafter, based on the block information table 10 updated in steps 171, 174, and 175, the file system control unit 2 assigns one block corresponding to the physical block number whose block status is "unused". The erase block 51 is secured as the preliminary erase block 51 (step 187).
As described above, in this embodiment, when writing data or reconstructing a file system, the state information of each erase block 51 and each sector 52 is finely rewritten. When the power supply is resumed thereafter, each erase block 5
1 and the erase block 5 based on the status information of each sector 52.
It is possible to erase the uncertain data of 1 or the sector 52 and save only the correct data.
In the above embodiment, (1) 11111111 representing “unused” is set as the current block state.
b (b represents a binary number; the same applies hereinafter); (2) 11111110b representing "data transfer in progress"; (3) 11111100b representing "original block erasing"; (4) 11111000b representing "data present"; 5) Five types of block states of 1111110000b representing "block full" have been exemplified. However, the combination of 1 and 0 may be complicated or the number of bits may be increased, whereby the difference between the states is more reliably determined. It becomes possible.
Further, in order to manage the block state more finely, the number of states of the block may be set to six or more, which makes it possible to manage the state more finely.
In the above embodiment, the current sector status is (I) 1111b representing "unused",
(II) 1110b representing "writing data", (I
II) 1100b representing "data writing completed", (I
V) Five types of sector states of 1000b representing "data valid" and (V) 0000b representing "data invalid" have been exemplified. However, the combination of 1 and 0 may be complicated or the number of bits may be increased. This makes it possible to more reliably determine the difference between the states.

Further, in order to manage the sector state more finely, the number of states of the sector 52 may be set to six or more, which makes it possible to manage the state more finely.
Although the above embodiment has been described on the assumption that the flash memory has a uniform block size,
Also in the flash memory having the boot block configuration, any one erase block (8 Kbytes) for the boot block (for example, 8 Kbytes) and any one block for the other blocks (for example, 64 Kbytes) May be reserved in advance as a spare block for reconstructing the file system 1.
This makes it possible to adopt a flash memory suitable for the purpose of the system or a cheaper flash memory from among flash memories having a uniform block or boot block configuration. Further, as a spare block for a boot block (for example, 8 Kbytes), a block having a size larger than 8 Kbytes (for example, 64 Kbytes) can be allocated to the spare block. In this case, the size of the spare block can be made constant. Therefore, management becomes simple.
When the file system 1 of the present embodiment includes a plurality of flash memories, one arbitrary erase block is reserved for each flash memory as a spare block for reconstructing the file system in advance. Instead, one spare block is reserved for a plurality of flash memories, or at least one spare block is reserved for each erase block included in the plurality of flash memories and having different block sizes. In this case, the number of spare blocks to be secured can be reduced.
In Step 176 shown in FIG. 9B,
The file system control unit 2 determines the physical block number having the same logical block number as the logical block number of the erasure block 51 in the block state “erasing the original block” acquired in step 175 and the block state being “data present”. To get. Then, the file system control unit 2 accesses the block information table 10 and updates the block state corresponding to the physical block number to “unused”. At the same time, the file system control unit 2 accesses the flash memory unit 5 through the flash memory control unit 4 and erases the data of the erase block 51 of this physical block number. Also, the file system control unit 2
In step 175, the block state of the block control section 52a of the erase block 51 having the physical block number of the block state "erasing the original block" acquired from step 175 is updated from "erasing the original block" to "data present", and the block information table 10 is updated. Is updated to "data available" corresponding to the same physical block number.
Here, a plurality of state information about the erasure block 51 include five states of “unused”, “data present”, “block full”, “data transfer”, and “original block erase”. In addition, the following processing flow can be used as an example when “original block erase completed” is set.
FIG. 13 is a flowchart showing a part of the flowchart relating to the initialization of the file system 1 shown in FIGS. 9A to 9D, which corresponds to the portion between steps 174 and 177 in the present embodiment.
First, the file system control unit 2 accesses the block information table 10 updated in step 171 to confirm whether or not the block state “original block erase completed” exists (step 191). If "erase complete" exists, the process proceeds to step 1.
Steps 19 skipping 92 and step 193
Jump to 4.
Next, the file system control unit 2 accesses the block information table 10 updated in step 171 to confirm whether or not the block state “erasing the original block” exists (step 192). If "block erasing" is present, the erase block 51
The same logical block number as the logical block number of
The file system control unit 2 acquires the physical block number of which the block status is “data present”, accesses the flash memory unit 5 through the flash memory control unit 4, and erases the data of the erase block 51 of this physical block number. I do.
After that, the file system control unit 2 deletes the erasure block 5 of the physical block number whose block status acquired in step 192 is “erasing the original block”.
The block state of one block control sector 52a is updated from "original block erase" to "original block erase completed".
Further, the file system control unit 2 accesses the block information table 10 and updates the block state corresponding to the physical block number of the erased block 51 from which data has been erased to “unused” (step 193).

Next, the block state of the block control sector 52a of the erasure block 51 of the physical block number updated to the block state “completed original block erase” is changed from “completed original block erase” to “data present”, and the block information is updated. The block state corresponding to this physical block number in Table 10 is updated from "erase of original block" to "data present" (step 194).
As described above, by adding the block state “original block erasure completed”, it is possible to skip the block erasure processing which has a relatively long processing time, ie,
If the block status immediately before the power supply was shut down was changed from “Original block erased” to “Original block erase completed”, the initialization processing time when the power supply was restored can be reduced. It is.
In another embodiment, as shown in FIG.
As a plurality of state information about the erase block 51,
Five types of “unused”, “with data”, “during data transfer”, “under erasing original block”, and “under formatting original block” may be set.
In this case, means for updating the state information of the spare erase block 51 previously reserved for reconstruction from “unused” to “data transfer in progress”, and updating the data in the erase block 51 of the data transfer source. Means for transferring and storing the data to the preliminary erasure block 51, and after completion of the data transfer and storage, updating the block state information of the preliminary erasure block 51 from "data transfer" to "original block erasing";
Erase the data in the erase block 51 of the data transfer source,
Means for securing the erase block 51 of the data transfer source as a new preliminary erase block 51 and updating the status information of the new preliminary erase block 51 from "data present" to "unused"; The status information of the preliminary erase block 51 is updated from “being erased from the original block” to “being formatted in the original block”, and the number of erasable times of the new preliminary erase block 51 is written into the new preliminary erase block 51 of the data transfer source (Formatting) means, and after formatting is completed, the preliminary erase block 5 of the data transfer destination.
A configuration that combines means for updating the status information of No. 1 from “in the original block format” to “data available” is conceivable.
As described above, the status information of the erase block 51 includes “unused”, “data present”, “data transfer”,
"Erasing original block", "Formatting original block"
When the five states are used, the state of the erase block can be accurately grasped and the flash memory can be finely managed, without destroying the existing data stored in the flash memory, and without changing the existing data. It is possible to automatically recover the file system to the previous stage that was being executed immediately before the power supply was cut off without causing inconsistency in the file system.
In the other embodiment, when the erase block 51 in the block state “erasing the original block” is detected in the initialization of the file system 1 at the time of power recovery, the data in the erase block 51 of the data transfer source is erased. . Alternatively, in order to further improve the accuracy of the existing data, when the erase block 51 not only “erasing the original block” but also “formatting the original block” is detected in the initialization, the erase block 51 of the data transfer source may be detected. There is a method of erasing data.
Unlike the above-described embodiment in which the state "block full" is set as the state information of the erase block 51, the method in which the block state "block full" is not set is such that the erase block is divided into sectors of unequal size. It is particularly effective when used. This is because, when writing data of unequal size to consecutive sectors in the same erase block, if the size of data to be written next is unknown, the state information of the erase block becomes “block full”. It is difficult to determine whether or not.
In order to average the number of times of block erasure of each erase block 51, there is a method of storing the number of erasable times of each erase block in each erase block 51.
As another embodiment shown in FIGS. 8A and 8B, when the file system 1 is reconstructed, the file system control unit 2 removes each of the spare erase blocks 51 as the erase blocks from which data is transferred. The number of erasable times of the erasing block 51 is compared. If the difference between the maximum value and the minimum value of the erasing block exceeds a certain value, the erasing block 51 having the

maximum number of erasing is selected, and the difference exceeds the certain value. If not, the sector information table 12 in the erase block 51 is accessed, and the erase block 51 having the largest number of invalid data sectors is selected.
In this case, it is possible to suppress the variation of the erasable number of each erasing block 51 within a certain number and increase the number of rewriting as a flash memory.
The number of erasable times of the erasing block 51 is increased by one from 0 and incremented to the maximum value of the specification value of the number of erasable times (in this case, the number of erasures is indicated instead of the number of erasable times). Conversely, a method is conceivable in which the number of erasable times is reduced by one from the maximum value of the specification value and decremented to 0. Since it can be determined that the life of the block 51 has expired, the algorithm for determining the number of erasable times can be simplified.
In the formatting means of the erasing block 51 in another embodiment, it is also possible to store the date and time of formatting, and in this case, it is possible to perform a refresh operation of the data in the erasing block 51 after a certain time has elapsed. Become.
[0120]
According to the file system of the present invention, there is provided a file system using a nonvolatile semiconductor memory device having a plurality of erase blocks each including a plurality of sectors. Storage means for storing, for each of the erase blocks, one of a plurality of pieces of state information about the erase block, and for each of the sectors, storing any one of a plurality of pieces of state information about the sector; And a guarantee unit for guaranteeing data already stored in the nonvolatile semiconductor memory device based on the status information of each erase block and the status information of each sector in the storage unit when accessing the storage device. I do.
According to the present invention, a plurality of pieces of state information about blocks and a plurality of pieces of state information about sectors are stored in the storage means. Therefore, even if unintended interruption of power supply occurs while the file system is operating,
When the power supply is restored, at the time of the initialization process at the time of startup, based on a plurality of state information about the block and a plurality of state information about the sector, the state of the block and the sector is accurately grasped and the flash memory is finely read. A file system that can be managed and has been executed immediately before the power supply was cut off without destroying the existing data stored in the flash memory and causing inconsistency in the existing data. Can be automatically recovered.
In one embodiment, the storage means is provided on a RAM having no restriction on data updating, and the assurance means is configured to access each of the erase blocks on the RAM when accessing the nonvolatile semiconductor memory device. And the status information of each sector.
When the RAM is used in this way, there is no restriction on updating data, and the processing speed is increased.
In one embodiment, at least five status information items of “unused”, “data present”, “block full”, “data transferring”, and “original block erasing” are provided as the plurality of status information about the erased block. A status for setting the status information of the spare block reserved for reconstruction from "unused" to "data transfer in progress"; and Means for transferring and storing the data of the spare block to the spare block, and after the transfer and storage of the data are completed, updating the block state information of the spare block from "data transfer" to "erasing the original block". The data in the erase block of the data transfer source is erased, the erase block of the data transfer source is secured as a new spare block, and the status information of the new spare block is referred to as “data”. And means for updating "or from" block full "to" unused ", and means for updating the status information of the data transfer destination of the spare block from the" source block erasing "to" Data Available ".
As described above, when the five statuses “unused”, “data present”, “block full”, “data transfer”, and “original block erased” are used as the status information of the erased block, Flash memory can be managed precisely by grasping the state accurately, without destroying existing data stored in flash memory, and without causing inconsistency in existing data,
It is possible to automatically recover the file system to the previous stage that was being executed immediately before the power supply was cut off.
In one embodiment, at least one spare block reserved in advance for reconstruction is provided for each

erase block included in one nonvolatile semiconductor memory device and having a different block size.
In this case, there is an advantage that it is possible to adopt a flash memory suitable for the purpose of the system or a cheaper one of the flash memories having the equal block or boot block configuration.
In one embodiment, at least one or more spare blocks reserved in advance for reconstruction are provided for each erase block having a different block size included in a plurality of nonvolatile semiconductor memory devices.
In this case, there is an advantage that the number of spare blocks required by the file system can be reduced. Therefore, there is an advantage that the file system can be configured at a low cost.
In one embodiment, the spare block reserved in advance for reconstruction has a size equal to or larger than the size of the erase block.
In this case, the boot block (for example, 8K
As a spare block for (a size of a byte), a block having a size larger than 8 Kbytes (for example, a size of 64 Kbytes) is allocated to the spare block, and the size of the spare block can be kept constant, thereby simplifying management.
A file system according to the present invention is a file system using a nonvolatile semiconductor memory device having a plurality of sectors, wherein for each of the sectors, storage means for storing any one of a plurality of pieces of state information on the sector; And a guarantee unit for guaranteeing data already stored in the nonvolatile semiconductor memory device based on status information of each sector in the storage unit when accessing the nonvolatile semiconductor memory device. As multiple pieces of status information about
At least five states of "unused", "data writing", "data writing completed", "data valid", and "data invalid" are set, and the assurance means includes state information of a sector to which data is written. Means for updating from "unused" to "data writing" and means for updating the status information of the sector from "data writing" to "data writing completed" after data writing to the sector is completed. When new data is written to the sector, the status information of the sector is updated from "data write completed" to "data valid", and when updated data of existing data is written to the sector, After updating the status information of the other sector to which the existing data is written from “data valid” to “data invalid”, the state of the sector to which the updated data is written is updated. To update the information from the "data write complete" to "data valid".
According to the present invention, five pieces of status information on a sector are stored in the storage means. For this reason, even if unintended interruption of power supply occurs while the file system is operating, when power supply is restored, initialization of the sector at the time of startup is performed based on a plurality of state information about the sector. The flash memory can be managed precisely by accurately grasping the power supply, without destroying the existing data stored in the flash memory and causing inconsistency in the existing data, immediately before the power supply is cut off. The file system can be automatically restored to the previous stage that was being executed.
In one embodiment, as the plurality of state information on the erase block, “unused”, “data present”, “block full”, “data transfer”, “erase original block”, “former block erase”, Means for updating the status information of the spare erase block previously reserved for rebuilding from "unused" to "data transfer in progress". Means for transferring the data in the erase block of the data transfer source to the spare erase block and storing the data, and after completion of the transfer and storage of the data, the block status information of the preliminary erase block is changed from "data transfer" to "source". Update to `` erasing block ", erase the data of the erase block of the data transfer source,
Means for updating the block status information of the preliminary erase block from "erasing the original block" to "completed erase of the original block"; securing the erase block of the data transfer source as a new preliminary erase block; Means for updating the status information of the spare erase block at the data transfer destination from "data available" or "block full" to "unused", and changing the status information of the spare erase block of the data transfer destination from "completed original block erase" to "data available" Means.
As described above, the status information of the erased block includes the following six information: “unused”, “data present”, “block full”, “data transfer”, “erase of original block”, and “completed erase of original block”. When the state is used, the state of the erase block can be more accurately grasped and the flash memory can be managed more minutely than when the above five states are used, and the existing

data stored in the flash memory is destroyed. This makes it possible to automatically restore the file system to the previous stage that was being executed immediately before the power supply was cut off, without causing inconsistency in existing data.
In one embodiment, at least a plurality of pieces of state information on the erase block include “unused”, “data present”, “data transfer”, “erase original block”, and “former original block”. Five states are set, and the assurance means updates the state information of the spare erase block, which is secured in advance for reconstruction, from "unused" to "data transfer in progress", and erases the data transfer source. Means for transferring the data in the block to the spare erase block for storage, and after the transfer and storage of the data are completed, the block status information of the spare erase block is changed from "data transfer" to "original block erase". Update, erase the data of the erase block of the data transfer source, secure this erase block of the data transfer source as a new spare erase block, and Means for updating the status information of the block from "data present" to "unused", and updating the status information of the spare erase block of the data transfer destination from "erasing the original block" to "formatting the original block", Means for writing (formatting) the number of erasable times of the new preliminary erase block into the new preliminary erase block at the transfer source, and after formatting is completed, the state information of the preliminary erase method block at the data transfer destination is changed to “formation of original block. ”To“ data available ”.
As described above, when five states of “unused”, “data present”, “data transfer”, “erase of original block”, and “formation of original block” are used as the state information of the erase block. Flash memory can be managed precisely by accurately grasping the state of the erase block, without destroying existing data stored in the flash memory, and without causing inconsistency in existing data, It is possible to automatically recover the file system to the previous stage that was being executed immediately before the power supply was cut off.
In this case, unlike the embodiment in which the state “block full” is set as the state information of the erase block, the method of not setting the block state “block full” is that the erase block is divided into sectors of unequal size. It is particularly effective when used. This is because, when writing data of unequal size to consecutive sectors in the same erase block, if the size of data to be written next is unknown, the state information of the erase block becomes “block full”. It is difficult to determine whether or not.
In one embodiment, the number of erasable times of each erase block is stored in each erase block in order to average the number of times of block erase of each erase block. By using the erasable number at the time of rebuilding the file system, it is possible to suppress the variation of the erasable number of each erase block within a certain number and to improve the total rewrite number as the flash memory. .
The erasable number of erase blocks is 0.
Is incremented by one, and incremented to the maximum value of the erasable number specification value (in this case, the number of erasures is shown instead of the erasable number). Conversely, the maximum value of the erasable number specification value Is decremented to 0 and decremented to 0. In the latter method, it is possible to determine that the life of the erase block has expired when the number of erasable times reaches 0. Algorithm can be simplified.
In the erase block formatting means in one embodiment, the formatted date and time can be stored. In this case, the data in the erase block can be refreshed after a certain time has elapsed.
[Brief description of the drawings]
FIG. 1 is a block diagram showing an embodiment of a file system according to the present invention.
FIG. 2 is a configuration diagram of data stored in a flash memory unit of the file system of FIG. 1;
FIG. 3 is a configuration diagram of data on each erase block stored in a file system memory unit in the file system of FIG. 1;
FIG. 4 is a configuration diagram of data relating to each sector stored in a file system memory unit in the file system of FIG. 1;
5 is a configuration diagram of data in which the number of each sector is arranged according to each sector state for each erase block stored in a file system memory unit in the file system of FIG. 1;
FIG. 6 is a flowchart relating to data read processing in the file system of FIG. 1;
FIG. 7A is a flowchart relating to a process of writing data in the file system of FIG. 1;
FIG. 7B is a flowchart following FIG. 7A relating to a process of writing data in the file system of FIG. 1;
FIG. 7C is a flowchart following FIG. 7B relating to a process of writing data in the file system of FIG. 1;

FIG. 8A is a flowchart relating to a rebuilding process of the file system of FIG. 1;
FIG. 8B is a flowchart following FIG. 8A relating to the rebuilding process of the file system of FIG. 1;
FIG. 9A is a flowchart relating to initialization of the file system of FIG. 1;
FIG. 9B is a flowchart following FIG. 9A regarding initialization of the file system of FIG. 1;
FIG. 9C is a flowchart following FIG. 9B relating to the initialization of the file system of FIG. 1;
FIG. 9D is a flowchart following FIG. 9C relating to the initialization of the file system of FIG. 1;
FIG. 10 is a flowchart related to a process of updating data stored in a flash memory according to a conventional technique.
And FIG. 11 is a flowchart relating to a file system rebuilding process according to the related art.
FIG. 12 is a diagram showing another embodiment representing a block state of block state information in a configuration diagram of data stored in a flash memory unit in the file system of FIG. 1;
13 is a diagram showing a flowchart of a part related to the first embodiment in which “original block erase completed” is set as status information in FIGS. 9A to 9D relating to the initialization of the file system in FIG. 1;
[Explanation of symbols]
DESCRIPTION OF SYMBOLS 1 File system 2 File system control part 3 File system memory part 4 Flash memory control part 5 Flash memory part 6 Application, OS (operating system) 51 Erase block, preliminary erase block 52 sector 52a Block control sector

Claims

In a file system using a nonvolatile semiconductor memory device having a plurality of erase blocks each including a plurality of sectors, any one of a plurality of state information on the erase blocks is stored for each of the erase blocks. A storage unit for storing, for each of the sectors, one of a plurality of pieces of state information on the sector; and a state information of each of the erase blocks in the storage unit and each of the plurality of state information when the nonvolatile semiconductor memory device is accessed. A file system using a nonvolatile semiconductor memory device, comprising: a guarantee unit for guaranteeing data already stored in the nonvolatile semiconductor memory device based on sector state information.
2. The storage unit is provided on a RAM having no restriction on data update, and the assurance unit is configured to access status information of each of the erase blocks on the RAM when accessing the nonvolatile semiconductor storage device. 2. A file system using the non-volatile semiconductor storage device according to claim 1, wherein status information of each sector is referred to.
3. A plurality of state information of “unused”, “data present”, “block full”, “data transfer”, and “original block erase” are set as a plurality of state information about the erase block. The assurance means includes means for updating the state information of the spare block reserved in advance for reconstruction from "unused" to "data transfer in progress", and updating the data in the erase block of the data transfer source. Means for transferring the data to the spare block and storing the data; and after completion of the data transfer and storage, updating the block status information of the spare block from "data transfer" to "erasing the original block", Erase the data of the erase block, secure the erase block of the data transfer source as a new spare block, and change the status information of the new spare block to “data present” or 2. The non-volatile memory according to claim 1, further comprising: means for updating from "block full" to "unused"; and means for updating the status information of the spare block at the data transfer destination from "erasing the original block" to "data present". File system using a volatile semiconductor storage device.
4. The nonvolatile memory device according to claim 3, wherein at least one spare block reserved in advance for reconstructing the file system is provided for each erase block included in one nonvolatile semiconductor memory device and having a different block size. A file system using the non-volatile semiconductor storage device according to 1.
5. A spare block reserved in advance for rebuilding a file system has at least one spare block for each erase block included in a plurality of nonvolatile semiconductor memory devices having different block sizes. A file system using the non-volatile semiconductor storage device according to 1.
6. The file system using the nonvolatile semiconductor memory device according to claim 4, wherein the spare block reserved in advance for reconstructing the file system has a size equal to or larger than the size

of the erase block.
7. A file system using a nonvolatile semiconductor memory device having a plurality of sectors, a storage means for storing any one of a plurality of state information for each sector, and the nonvolatile semiconductor memory. Means for assuring data already stored in the non-volatile semiconductor storage device based on the status information of each sector in the storage means at the time of access to the device; As information, at least five states of "unused", "data writing", "data writing completed", "data valid", and "data invalid" are set. Means for updating the status information of the sector from "unused" to "data writing", and the status of the sector after the data writing to the sector is completed. Means for updating the status information from "data writing" to "data writing completed"; and, when data is newly written to the sector, updating the status information of the sector from "data writing completed" to "data valid". However, when the updated data of the existing data is written in the sector, the status information of the other sector in which the existing data is written is updated from "data valid" to "data invalid", and then the updated data is updated. A file system using a nonvolatile semiconductor memory device that updates the status information of the written sector from “data write completed” to “data valid”.
8. When the file system is initialized when the power supply is restored, it is searched for the existence of an erase block of “original block is being erased” as the state information of the erase block, and the erase block is present. Indicates an erase block having the same logical block number as that of the erase block, and the status information of the erase block being "data present" or "block full". Means for acquiring from the information, means for erasing the data of the acquired erasure block, and after completion of the erasure, means for updating the status information of the erasure block from "data present" or "block full" to "unused"; The status information of the erased block whose status information is "erasing the original block" is changed from "erasing the original block" to "data present".
4. A file system using the nonvolatile semiconductor memory device according to claim 3, further comprising:
9. At least five states of “unused”, “data present”, “during data transfer”, “erasing of original block”, and “formatting of original block” as a plurality of state information about the erase block. Wherein the assurance means updates the state information of the spare erase block previously reserved for rebuilding the file system from "unused" to "data transfer in progress"; Means for transferring the data in the erased block to the spare erased block and storing the data, and after the transfer and storage of the data are completed, the status information of the spare erased block is "transferred"
To the original block being erased, erase the data in the erased block of the data transfer source, and after erasure is complete, secure this erased block of the data transfer source as a new spare erase block. Means for updating the status information from "data present" to "unused", and updating the status information of the spare erase block at the data transfer destination from "erasing the original block" to "formatting the original block" and transferring the data. Means for formatting the original new pre-erased block, and after formatting is completed, the status information of the pre-erased block at the data transfer destination is changed to "formation of original block".
2. A file system using the nonvolatile semiconductor memory device according to claim 1, further comprising:
means for updating the data to "data available".
10. The erase block stores the number of erasable blocks in each erase block. At the time of formatting, a new spare erase block has a number obtained by subtracting one from the number of erases of the spare erase block. Means for storing in the spare erase block, means for selecting the erase block having the largest number of block erase times as a data transfer source among all erase blocks except the spare erase block when reconstructing the file system. A file system using the nonvolatile semiconductor memory device according to claim 9.
11. The file system using the nonvolatile semiconductor memory device according to claim 10, wherein the use of the erase block is prohibited when the number of times of erasable block becomes a value equal to or less than 0.
12. The erase block stores the number of erasable blocks in each erase block. At the time of formatting, a new spare erase block includes the number of times the erase count of the spare erase block is increased by one. Means for storing the data in the spare erase block and means for selecting the erase block having the largest number of block erase times as the data transfer source among all the erase blocks except the spare erase block when reconstructing the file system. A file system using the nonvolatile semiconductor memory device according to claim 9.

13. The nonvolatile semiconductor memory device according to claim 12, wherein when the block erasable number reaches a value equal to or more than a specification value (maximum value) of the erasable number, use of the erase block is prohibited. The file system used.
14. A nonvolatile semiconductor memory device according to claim 14, wherein at least one spare erase block reserved in advance for reconstructing the file system is provided for each erase block included in one nonvolatile semiconductor memory device having a different block size. 10. A file system using the nonvolatile semiconductor memory device according to 9.
15. The nonvolatile memory device according to claim 15, wherein at least one spare erase block reserved in advance for reconstructing the file system is provided for each erase block included in the plurality of nonvolatile semiconductor memory devices having a different block size. 10. A file system using the nonvolatile semiconductor memory device according to 9.
16. The file system using the nonvolatile semiconductor memory device according to claim 14, wherein the spare erase block reserved in advance for reconstructing the file system has a size equal to or larger than an erase block size.
17. A file system using a nonvolatile semiconductor memory device having a plurality of sectors, a storage means for storing any one of a plurality of pieces of status information on the sector for each sector, and the nonvolatile semiconductor memory. Means for assuring data already stored in the nonvolatile semiconductor memory device based on the status information of each sector in the storage means at the time of access to the device; At least five states of "unused", "data writing", "data writing completed", "data valid", and "data invalid" are set as the state information, and the assurance means writes data. Sector status information changed from "unused" to "writing data"
Means for updating the status information of the sector to "data writing in progress" after the data writing to the sector is completed.
Means for updating from "data writing completed" to "data writing completed", and when new data is written to the sector, the status information of the sector is updated from "data writing completed" to "data valid", and existing data in the sector is updated. When the updated data is written, the status information of the other sector in which the existing data is written is updated from "data valid" to "data invalid", and then the status of the sector in which the updated data is written is updated. 10. The file system using the nonvolatile semiconductor memory device according to claim 9, further comprising: means for updating information from "data write completion" to "data valid".
18. In the initialization of the file system when the power supply is restored, it is searched whether or not an erase block of “original block is being erased” exists as the state information of the erase block, and if the erase block exists, Means for acquiring a data transfer source block to the erase block from the state information of each erase block in the storage means; erasing the acquired data of the data transfer source block;
Means for updating the status information of the erase block from "data present" to "unused"; and changing the status information of the erase block whose erase block status information is "erasing the original block" from "erasing the original block" to "the original block". Means to update the block to `` being formatted '' and format the erased block from which data has been erased, to secure it as a new spare erased block; 10. The file system using the nonvolatile semiconductor memory device according to claim 9, further comprising: means for updating information from "former block format" to "data present".
19. In the initialization of the file system when power supply is restored, it is searched for the existence of an erase block of "former block format" as the state information of the erase block, and when the erase block exists, Means for acquiring a data transfer source block to the erase block from status information of each erase block in the storage means;
Means for formatting the acquired data transfer source block to secure it as a new spare erase block, and after formatting is completed, change the status information of the erase block whose erase block status information is "former block format" to "former block format". 10. A file system using the nonvolatile semiconductor memory device according to claim 9, further comprising: means for updating from "data" to "data available".

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（Ａ）**　(11)特許出願公開番号
特開2001－51889
（P2001－51889A）
(43)公開日　平成13年２月23日(2001.2.23)

(51)Int.Cl.⁷		識別記号	FI			ﾃｰﾏｺｰﾄﾞ(参考)
Ｇ０６Ｆ	12/00	５４２	Ｇ０６Ｆ	12/00	５４２Ｍ	５Ｂ０１８
	12/16	３４０		12/16	３４０Ｐ	５Ｂ０８２

審査請求　未請求　請求項の数19　ＯＬ　（全 23 頁）

(21)出願番号　特願平11－254973

(22)出願日　平成11年９月８日(1999.9.8)

(31)優先権主張番号　特願平11－153090
(32)優先日　平成11年５月31日(1999.5.31)
(33)優先権主張国　日本（ＪＰ）

(71)出願人　000005049
シャープ株式会社
大阪府大阪市阿倍野区長池町22番22号
(72)発明者　戸森　宣昭
大阪府大阪市阿倍野区長池町22番22号　シャープ株式会社内
(74)代理人　100078282
弁理士　山本　秀策

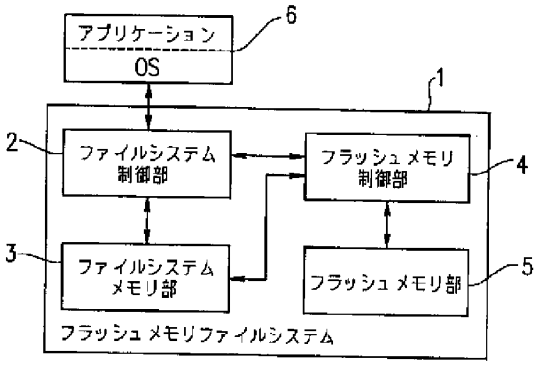
Ｆターム(参考)　5B018 GA04 LA06 MA22 NA06 QA05
QA15
5B082 JA07

(54)【発明の名称】　不揮発性半導体記憶装置を用いたファイルシステム

(57)【要約】

【課題】　動作中に意図しない電力供給の遮断が発生しても、不揮発性半導体記憶装置内に格納されているデータを正常に読み出すことを可能にする。

【解決手段】　各消去ブロック毎に、消去ブロックについての複数の状態情報のいずれかを格納すると共に、該各セクタ毎に、セクタについての複数の状態情報のいずれかをファイルシステムメモリ部３及びフラッシュメモリ部５に格納している。ファイルシステムメモリ部３内の各消去ブロックの状態情報及び各セクタの状態情報は、ファイルシステム１の起動時に、フラッシュメモリ部５内の該データを読み出すことによって作成され、ファイルシステム制御部２は、ファイルシステムメモリ部３内の各消去ブロックの状態情報及び各セクタの状態情報に基いて、フラッシュメモリ部５からデータを読み出すと共に、動作中に意図しない電力供給の遮断時においてもフラッシュメモリ部５内のデータを保証する。



Samsung Ex. 1002, Page 215 of 622

【特許請求の範囲】

【請求項１】　複数のセクタをそれぞれ含む複数の消去ブロックを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、

該各消去ブロック毎に、消去ブロックについての複数の状態情報のいずれかを格納すると共に、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、

該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各消去ブロックの状態情報及び該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備する不揮発性半導体記憶装置を用いたファイルシステム。

【請求項２】　前記記憶手段は、データの更新に制約の無いＲＡＭ上に設けられ、

前記保証手段は、前記不揮発性半導体記憶装置へのアクセス時に、該ＲＡＭ上の前記各消去ブロックの状態情報及び前記各セクタの状態情報を参照する請求項１に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項３】　前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」という少なくとも５つの状態を設定しており、

前記保証手段は、

再構築用に予め確保されている予備ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、

データ転送元の消去ブロック内のデータを該予備ブロックへ転送して記憶させる手段と、

該データの転送並びに記憶の完了後、該予備ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、このデータ転送元の消去ブロックを新たな予備ブロックとして確保し、この新たな予備ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へ更新する手段と、

データ転送先の該予備ブロックの状態情報を「元ブロック消去中」から「データ有」へ更新する手段とを含む請求項１に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項４】　ファイルシステムの再構築用に予め確保されている予備ブロックは、１個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する請求項３に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項５】　ファイルシステムの再構築用に予め確保されている予備ブロックは、複数個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する請求項３に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項６】　ファイルシステムの再構築用に予め確保されている予備ブロックは、消去ブロックのサイズ以上のサイズを有する請求項４又は５に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項７】　複数のセクタを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、

該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、

該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備し、

前記セクタについての複数の状態情報として、「未使用」、「データ書き込み中」、「データ書き込み完了」、「データ有効」、「データ無効」という少なくとも５つの状態を設定しており、

前記保証手段は、

データが書き込まれるセクタの状態情報を「未使用」から「データ書き込み中」へ更新する手段と、

該セクタへのデータの書き込みが完了した後、該セクタの状態情報を「データ書き込み中」から「データ書き込み完了」へ更新する手段と、

該セクタにデータを新規に書き込んだ場合は、該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新し、該セクタに既存のデータの更新データを書き込んだ場合は、該既存のデータが書き込まれている他のセクタの状態情報を「データ有効」から「データ無効」に更新後、該更新データが書き込まれている該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新する不揮発性半導体記憶装置を用いたファイルシステム。

【請求項８】　電力供給が回復した時のファイルソステムの初期化において、前記消去ブロックの状態情報として「元ブロック消去中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックの論理ブロック番号と同一の論理ブロック番号を有し、消去ブロックの状態情報が「データ有」あるいは「ブロックフル」である消去ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得した消去ブロックのデータを消去し、消去完了後、この消去ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へ更新する手段と、消去ブロックの状態情報が「元ブロック消去中」の消去ブロックの状態情報を「元ブロック消去中」から「データ有」へ更新する手段とを含む請求項３に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項９】　前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という少なくとも５つの状態を設定しており、前記保証手段は、ファイルシステムの再構築用に予め確保

Samsung Ex. 1002, Page 216 of 622

されている予備消去ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備消去ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロックの状態情報を「データ転送中」から「元ブロック消去中」へ更新し、データ転送元の消去ブロックのデータを消去し、消去完了後、このデータ転送元の消去ブロックを新たな予備消去ブロックとして確保し、この新たな予備消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、データ転送先の該予備消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データ転送元の新たな予備消去ブロックをフォーマットする手段と、フォーマット完了後、データ転送先の該予備消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含む請求項１に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１０】　前記消去ブロックは、ブロック消去可能回数を各消去ブロック内に記憶しており、前記フォーマット時において、新たな予備消去ブロックに該予備消去ブロックの消去可能回数から１減じた回数を該予備消去ブロック内に記憶する手段と、ファイルシステムの再構築時には、該予備消去ブロックを除いた全ての消去ブロックの中で、ブロック消去回数が最大の消去ブロックをデータ転送元として選択する手段とを含む請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１１】　前記ブロック消去可能回数が０以下の値になった時点で、該消去ブロックの使用を禁止する請求項１０に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１２】　前記消去ブロックは、ブロック消去可能回数を各消去ブロック内に記憶しており、前記フォーマット時において、新たな予備消去ブロックに該予備消去ブロックの消去回数に１増加した回数を該予備消去ブロック内に記憶する手段と、ファイルシステムの再構築時には、該予備消去ブロックを除いた全ての消去ブロックの中で、ブロック消去回数が最大の消去ブロックをデータ転送元として選択する手段とを含む請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１３】　前記ブロック消去可能回数が消去可能回数の仕様値（最大値）以上の値になった時点で、該消去ブロックの使用を禁止する請求項１２に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１４】　ファイルシステムの再構築用に予め確保されている予備消去ブロックは、１個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１５】　ファイルシステムの再構築用に予め確保されている予備消去ブロックは、複数個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１６】　ファイルシステムの再構築用に予め確保されている予備消去ブロックは、消去ブロックサイズ以上のサイズを有する請求項１４または１５に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１７】　複数のセクタを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置へのアクセス時に、該記憶手段内の該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備し、前記セクタについての複数の状態情報として、「未使用」、「データ書き込み中」、「データ書き込み完了」、「データ有効」、「データ無効」という少なくとも５つの状態を設定しており、前記保証手段は、データが書き込まれるセクタの状態情報を「未使用」から「データ書き込み中」へ更新する手段と、該セクタへのデータの書き込みが完了した後、該セクタの状態情報を「データ書き込み中」から「データ書き込み完了」へ更新する手段と、該セクタにデータを新規に書き込んだ場合は、該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新し、該セクタに既存のデータの更新データを書き込んだ場合は、該既存のデータが書き込まれている他のセクタの状態情報を「データ有効」から「データ無効」に更新後、該更新データが書き込まれている該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新する手段とを含む請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１８】　電力供給が回復した時のファイルシステムの初期化において、前記消去ブロックの状態情報として「元ブロック消去中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックへのデータ転送元ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得したデータ転送元ブロックのデータを消去し、消去完了後、この消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、消去ブロックの状態情報が「元ブロック消去中」の消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データを消去した消去ブロックをフォーマットして新たな予備消去ブロックとして確保する手段と、フォーマット完了後、消去ブロックの状態情報が「元ブロックフォーマット中」の消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段

Samsung Ex. 1002, Page 217 of 622

とを含む請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【請求項１９】　電力供給が回復した時のファイルシステムの初期化において、前記消去ブロックの状態情報として「元ブロックフォーマット中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックへのデータ転送元ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得したデータ転送元ブロックをフォーマットして新たな予備消去ブロックとして確保する手段と、フォーマット完了後、消去ブロックの状態情報が「元ブロックフォーマット中」の消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含む請求項９に記載の不揮発性半導体記憶装置を用いたファイルシステム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、フラッシュメモリ等の不揮発性半導体記憶装置を用いたファイルシステムに関するものである。

【０００２】

【従来の技術】最近、データを電気的に書き込み及び消去することが可能な不揮発性半導体記憶装置（例えばフラッシュメモリ）を、データを磁気的に書き込み及び消去することが可能なハードディスクドライブの代替として用いるための技術が注目されている。

【０００３】フラッシュメモリは、ハードディスクドライブと比較して、データの読み出し速度が約１万倍程度かつデータの書き込み速度が約１千倍程度速く、また小型軽量化を図り易く耐衝撃性に優れ、可搬性が高い等の特長を備え、更に低容量のデータを高速に格納しかつ読み出すという用途には好適である。

【０００４】しかしながら、フラッシュメモリにおいては、既にデータが書き込まれている記憶領域に新しくデータを書き込むには、書き込まれているデータを一旦消去するという動作が必要となる。しかも、データを消去する単位が数キロバイトと大きく、数十ミリ秒から数秒と比較的長いデータ消去時間が必要であり、データの書き込み及び消去の保証回数が有限であるという欠点がある。

【０００５】このため、フラッシュメモリをハードディスクドライブの代替として用いる場合は、上記欠点を踏まえ、データの取り扱い単位をハードディスクドライブ並みの数百バイト程度に小さくし、なるべくデータ消去を行なわず、効率良くデータを管理するための仕組み、つまりフラッシュメモリに対応したファイルシステムを提供する必要がある。この様なファイルシステムは、例えば特開平６－９５９５５号公報に開示されている。

【０００６】また、米国特許５，５４４，１１９号には、電力供給遮断後の電力復旧時の処理方法が示され、米国特許５，５４４，３５６号には、ファイルシステムの再構築方法が示されている。

【０００７】

【発明が解決しようとする課題】しかしながら、上記公報に記載の技術においては、電源として電池等を用いた電力供給が不安定な環境下で、フラッシュメモリや各種の制御を行うファイルシステムが使用される状況を想定しておらず、フラッシュメモリあるいはファイルシステムの動作中に、電池の消耗等により電力供給が一瞬でも遮断されたとき、つまり意図しない電力供給の遮断が発生したときには、フラッシュメモリに格納されているデータを正常に読み出せなくなるという可能性があった。

【０００８】具体的には、第１に、データが格納された記憶領域（以下セクタと呼称する）の現在の状態をセクタ状態情報によって表しているが、このセクタ状態情報が「未使用」、「使用中」、「無効」という３種類の状態のいずれかを表しているに過ぎないため、セクタのデータを更新するに際し、電力供給の遮断が発生すると、データが正常に更新されなくても、セクタ状態情報に基いてデータを正常とみなしてしまうことがあった。このセクタのデータの更新処理を図１０のフローチャートに従って次に述べる。

【０００９】図１０のフローチャートによれば、まず更新される既存のデータを格納している１つのセクタのセクタ状態情報を「使用中」から「無効」へ変更し（ステップ１０１）、次にもう１つのセクタのセクタ状態情報を「未使用」から「使用中」へ変更し（ステップ１０２）、この後にセクタ状態情報が「使用中」となったセクタに対して更新データを書き込む（ステップ１０３）。ステップ１０１～１０３を書き込むデータがなくなるまで繰り返す。

【００１０】ここで、ステップ１０２の直後あるいはステップ１０３の実行中において、電力供給の遮断が発生し、フラッシュメモリに対するデータの書き込みが中断すると、更新データのセクタへの書き込みが正常に終了していないにもかかわらず、このセクタのセクタ状態情報が「使用中」となる。このため、電力供給が回復したときに、ファイルシステムは、このセクタに格納されているデータを有効であると誤認してしまう。

【００１１】第２に、１つのブロック内の全てのデータを消去して、ファイルシステムを再構築するに際し、電源の遮断が発生すると、２つのブロックにおいて同じデータを格納した状態、あるいは１つのブロック内の全てのデータを完全に消去していない状態が発生することがあった。このファイルシステムの再構築の処理を図１１のフローチャートに従って次に述べる。

【００１２】図１１のフローチャートによれば、まず全てのセクタが消去され、全てのセクタのセクタ状態情報が「未使用」である予備ブロックを予め確保しておき、全てのデータが消去されるブロックにおける保存すべき

Samsung Ex. 1002, Page 218 of 622

データ、つまりセクタ状態情報が「無効」であるセクタのデータ及びセクタ状態情報を除く他の全てのデータを予備ブロックにコピーし（ステップ１１１）、更に全てのデータが消去されるブロックにおけるセクタ状態情報が「無効」であるセクタ以外の位置情報に含まれるブロック番号を予備ブロックのブロック番号に更新する（ステップ１１２）。最後に、データのコピー元であるブロックを消去し、次回のファイルシステム再構築時の予備ブロックとして確保する（ステップ１１３）。

【００１３】ここで、ステップ１１３の実行前あるいはステップ１１３の実行中において、電力供給の遮断が発生すると、電力供給が回復したときに、２つのブロックにおいて同じデータを格納した状態、あるいは１つのブロック内の全てのデータを完全に消去できていない状態が発生し、新たな予備ブロックを確保できない。

【００１４】そこで、本発明は、上記従来の問題に鑑みなされたものであり、電力供給の遮断が発生しても、フラッシュメモリに格納されているデータを正常に読み出すことが可能な不揮発性半導体記憶装置を用いたファイルシステムを提供することを目的とする。

【００１５】

【課題を解決するための手段】上記課題を解決するために、本発明のファイルシステムは、複数のセクタをそれぞれ含む複数の消去ブロックを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各消去ブロック毎に、消去ブロックについての複数の状態情報のいずれかを格納すると共に、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各消去ブロックの状態情報及び該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備する。

【００１６】本発明によれば、ブロックについての複数の状態情報及びセクタについての複数の状態情報を記憶手段に格納している。このため、ファイルシステムが稼働中において、意図しない電力供給の遮断が生じても、電力供給が回復したときには、起動時の初期化処理に際し、ブロックについての複数の状態情報及びセクタについての複数の状態情報に基いて、ブロック及びセクタの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【００１７】一実施形態では、前記記憶手段は、データの更新に制約の無いＲＡＭ上に設けられ、前記保証手段は、前記不揮発性半導体記憶装置へのアクセス時に、該ＲＡＭ上の前記各消去ブロックの状態情報及び前記各セクタの状態情報を参照する。

【００１８】この様にＲＡＭを利用すると、データの更新に制約がなくなり処理速度が早くなる。

【００１９】一実施形態では、前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」という少なくとも５つの状態を設定しており、前記保証手段は、再構築用に予め確保されている予備ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、このデータ転送元の消去ブロックを新たな予備ブロックとして確保し、この新たな予備ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へ更新する手段と、データ転送先の該予備ブロックの状態情報を「元ブロック消去中」から「データ有」へ更新する手段とを含む。

【００２０】この様に消去ブロックの状態情報として「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」という５つの状態を利用した場合、ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【００２１】一実施形態では、再構築用に予め確保されている予備ブロックは、１個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する。

【００２２】この場合、均等ブロックあるいはブートブロック構成のフラッシュメモリのうちからシステムの目的に合わせた、あるいはより安価なフラッシュメモリを採用することが可能となる利点がある。

【００２３】一実施形態では、再構築用に予め確保されている予備ブロックは、複数個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する。

【００２４】この場合、ファイルシステムが必要とする予備ブロックの数を少なくできる利点がある。このため、ファイルシステムが安価に構成できる利点もある。

【００２５】一実施形態では、再構築用に予め確保されている予備ブロックは、消去ブロックのサイズ以上のサイズを有する。

【００２６】この場合は、ブートブロック（例えば８Ｋバイトのサイズ）用の予備ブロックとして、８Ｋバイトより大きいサイズのブロック（例えば６４Ｋバイトのサイズ）を予備ブロックに割り当て、予備ブロックのサイ

Samsung Ex. 1002, Page 219 of 622

ズを一定にできるため、管理が簡単になる。

【００２７】本発明のファイルシステムは、複数のセクタを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備し、前記セクタについての複数の状態情報として、「未使用」、「データ書き込み中」、「データ書き込み完了」、「データ有効」、「データ無効」という少なくとも５つの状態を設定しており、前記保証手段は、データが書き込まれるセクタの状態情報を「未使用」から「データ書き込み中」へ更新する手段と、該セクタへのデータの書き込みが完了した後、該セクタの状態情報を「データ書き込み中」から「データ書き込み完了」へ更新する手段と、該セクタにデータを新規に書き込んだ場合は、該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新し、該セクタに既存のデータの更新データを書き込んだ場合は、該既存のデータが書き込まれている他のセクタの状態情報を「データ有効」から「データ無効」に更新後、該更新データが書き込まれている該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新する。

【００２８】本発明によれば、セクタについての５つの状態情報を記憶手段に格納している。このため、ファイルシステムが稼働中において、意図しない電力供給の遮断が生じても、電力供給が回復したときには、起動時の初期化処理に際し、セクタについての複数の状態情報に基いて、セクタの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【００２９】他の実施形態では、前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という少なくとも５つの状態を設定しても良い。

【００３０】この場合、再構築用に予め確保されている予備消去ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備消去ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、このデータ転送元の消去ブロックを新たな予備消去ブロックとして確保し、この新たな予備消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、データ転送先の該予備消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データ転送元の新たな予備消去ブロックへ新たな予備消去ブロックの消去可能回数などを書き込む（フォーマットする）手段と、フォーマット完了後、データ転送先の該予備消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含む構成が考えられる。

【００３１】この様に、消去ブロックの状態情報として「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という５つの状態を利用した場合、消去ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。また、消去ブロックの状態情報として「ブロックフル」という状態を設定しない方法は、消去ブロックが不均等なサイズのセクタに分割されて使用される場合に特に有効である。なぜならば、不均等なサイズのデータを同一の消去ブロック内の連続したセクタに書き込む場合において、次に書き込むべきデータのサイズが不明である場合に、消去ブロックの状態情報が「ブロックフル」となるかどうかの判断が難しいためである。

【００３２】電力供給が回復した時のファイルソステムの初期化において、前記消去ブロックの状態情報として「元ブロック消去中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックの論理ブロック番号と同一の論理ブロック番号を有し、消去ブロックの状態情報が「データ有」あるいは「ブロックフル」である消去ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得した消去ブロックのデータを消去し、消去完了後、この消去ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へ更新する手段と、消去ブロックの状態情報が「元ブロック消去中」の消去ブロックの状態情報を「元ブロック消去中」から「データ有」へ更新する手段とを含んでもよい。

【００３３】前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という少なくとも５つの状態を設定しており、前記保証手段は、ファイルシステムの再構築用に予め確保されている予備消去ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備消去ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロックの状態情報を「データ転送中」から「元ブロック消去中」へ更新し、データ転送元の消

Samsung Ex. 1002, Page 220 of 622

去ブロックのデータを消去し、消去完了後、このデータ転送元の消去ブロックを新たな予備消去ブロックとして確保し、この新たな予備消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、データ転送先の該予備消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データ転送元の新たな予備消去ブロックをフォーマットする手段と、フォーマット完了後、データ転送先の該予備消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含んでもよい。

【００３４】前記消去ブロックは、ブロック消去可能回数を各消去ブロック内に記憶しており、前記フォーマット時において、新たな予備消去ブロックに該予備消去ブロックの消去可能回数から１減じた回数を該予備消去ブロック内に記憶する手段と、ファイルシステムの再構築時には、該予備消去ブロックを除いた全ての消去ブロックの中で、ブロック消去回数が最大の消去ブロックをデータ転送元として選択する手段とを含んでもよい。

【００３５】前記ブロック消去可能回数が０以下の値になった時点で、該消去ブロックの使用を禁止してもよい。

【００３６】前記消去ブロックは、ブロック消去可能回数を各消去ブロック内に記憶しており、前記フォーマット時において、新たな予備消去ブロックに該予備消去ブロックの消去回数に１増加した回数を該予備消去ブロック内に記憶する手段と、ファイルシステムの再構築時には、該予備消去ブロックを除いた全ての消去ブロックの中で、ブロック消去回数が最大の消去ブロックをデータ転送元として選択する手段とを含んでもよい。

【００３７】前記ブロック消去可能回数が消去可能回数の仕様値（最大値）以上の値になった時点で、該消去ブロックの使用を禁止してもよい。

【００３８】ファイルシステムの再構築用に予め確保されている予備消去ブロックは、１個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有してもよい。

【００３９】ファイルシステムの再構築用に予め確保されている予備消去ブロックは、複数個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有してもよい。

【００４０】ファイルシステムの再構築用に予め確保されている予備消去ブロックは、消去ブロックサイズ以上のサイズを有してもよい。

【００４１】複数のセクタを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置へのアクセス時に、該記憶手段内の該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備し、前記セクタについての複数の状態情報として、「未使用」、「データ書き込み中」、「データ書き込み完了」、「データ有効」、「データ無効」という少なくとも５つの状態を設定しており、前記保証手段は、データが書き込まれるセクタの状態情報を「未使用」から「データ書き込み中」へ更新する手段と、該セクタへのデータの書き込みが完了した後、該セクタの状態情報を「データ書き込み中」から「データ書き込み完了」へ更新する手段と、該セクタにデータを新規に書き込んだ場合は、該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新し、該セクタに既存のデータの更新データを書き込んだ場合は、該既存のデータが書き込まれている他のセクタの状態情報を「データ有効」から「データ無効」に更新後、該更新データが書き込まれている該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新する手段とを含んでもよい。

【００４２】電力供給が回復した時のファイルシステムの初期化において、前記消去ブロックの状態情報として「元ブロック消去中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックへのデータ転送元ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得したデータ転送元ブロックのデータを消去し、消去完了後、この消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、消去ブロックの状態情報が「元ブロック消去中」の消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データを消去した消去ブロックをフォーマットして新たな予備消去ブロックとして確保する手段と、フォーマット完了後、消去ブロックの状態情報が「元ブロックフォーマット中」の消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含んでもよい。

【００４３】電力供給が回復した時のファイルシステムの初期化において、前記消去ブロックの状態情報として「元ブロックフォーマット中」の消去ブロックが存在するか検索し、該消去ブロックが存在する場合は、該消去ブロックへのデータ転送元ブロックを、前記記憶手段内の各消去ブロックの状態情報から取得する手段と、取得したデータ転送元ブロックをフォーマットして新たな予備消去ブロックとして確保する手段と、フォーマット完了後、消去ブロックの状態情報が「元ブロックフォーマット中」の消去ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含んでもよい。

【００４４】

【発明の実施の形態】以下、本発明の実施形態を添付図面を参照して説明する。

【００４５】図１は、本発明のファイルシステムの一実施形態を示している。図１において、１はファイルシス

Samsung Ex. 1002, Page 221 of 622

テム、２はファイルシステム１の処理を制御するファイルシステム制御部、３はファイルシステム１が制御情報を格納するために使用するファイルシステムメモリ部、４はフラッシュメモリ部５へのデータ処理を制御するフラッシュメモリ制御部、５はフラッシュメモリ部、６はファイルシステム１へデータの処理を依頼するアプリケーションもしくはオペレーティングシステムである。

【００４６】図２は、図１におけるフラッシュメモリ部５に格納されているデータの構成図である。フラッシュメモリ部５の各消去ブロック５１は、データの取扱い単位である複数の物理セクタ５２に分割されて配置されており、各消去ブロック５１内で一意の物理セクタ番号を有する。各消去ブロック51のサイズが６４Ｋバイトで、各物理セクタ５２のサイズが５１２バイトであるとした場合、各消去ブロック５１には１２８個の物理セクタ５２が存在することになる。

【００４７】ここで、フラッシュメモリ部５内の複数の消去ブロック５１のうちの任意の１個の消去ブロック５１をファイルシステム１の再構築用に用いられるデータ未書き込みの予備消去ブロックとして、また各消去ブロック５１の先頭のセクタ５２をそれぞれの消去ブロックの制御情報を格納するブロックコントロールセクタ５２ａとして、予め確保しておく。

【００４８】消去ブロック５１において、ブロックコントロールセクタ５２ａには、先頭から該消去ブロック５１の状態情報としての２バイト、セクタ＃１の物理セクタ５２の状態情報としての２バイト、セクタ＃２～＃１２７の各物理セクタ５２の状態情報としてのそれぞれの２バイトが順次配列され、よって２バイト×１２８個＝２５６バイトの情報を有している。

【００４９】消去ブロック５１の状態情報は、ファイルシステム制御部２が割り当てを行う０～２５５の論理ブロック番号、及び５種類のブロック状態のいずれかを表すデータからなる。５種類のブロック状態とは、（１）「未使用」を表す１１１１１１１１ｂ（ｂは２進数を表わす。以下同様）、（２）「データ転送中」を表す１１１１１１１０ｂ、（３）「元ブロック消去中」を表す１１１１１１００ｂ、（４）「データ有」を表す１１１１１０００ｂ、（５）「ブロックフル」を表す１１１１００００ｂである。

【００５０】消去ブロック５１の各セクタ５２の状態情報は、消去ブロック５１の状態情報と同様に、ファイルシステム制御部２が割り当てを行う０～４０９５のシステム内で一意の論理セクタ番号、及び５種類のセクタ状態のいずれかを表すデータからなる。５種類のセクタ状態とは、（Ｉ）「未使用」を表す１１１１ｂ、（ＩＩ）「データ書き込み中」を表す１１１０ｂ、（ＩＩＩ）「データ書き込み完了」を表す１１００ｂ、（ＩＶ）「データ有効」を表す１０００ｂ、（Ｖ）「データ無効」を表す００００ｂである。

【００５１】図３は、図１におけるファイルシステムメモリ部３に格納されている各消去ブロック５１に関するデータの構成図（以下、ブロック情報表１０と呼称）である。

【００５２】図３において、物理ブロック番号、論理ブロック番号、及びブロック状態を１つの情報単位としており、例えば物理ブロック番号＝０、論理ブロック番号＝０、ブロック状態＝データ有となっている消去ブロック５１、物理ブロック番号＝１、論理ブロック番号＝１、ブロック状態＝ブロックフルとなっている消去ブロック５１等が存在することを示している。

【００５３】図４は、図１におけるファイルシステムメモリ部３に格納されている各セクタ５２に関するデータの構成図（以下、セクタ情報表１１と呼称）である。

【００５４】図４において、物理ブロック番号、物理セクタ番号、論理セクタ番号、及びセクタ状態を１つの情報単位としており、例えば物理ブロック番号＝０、物理セクタ番号＝０、論理セクタ番号＝１００、セクタ状態＝データ有効となっているセクタ５２、物理ブロック番号＝０、物理セクタ番号＝１、論理セクタ番号＝１０、セクタ状態＝データ無効となっているセクタ５２等が存在することを示している。

【００５５】図５は、図１におけるファイルシステムメモリ部３に格納されている各消去ブロック５１毎に、各セクタ５２の数をそれぞれのセクタ状態に応じて整理したデータの構成図（以下、セクタ情報表１２と呼称）である。

【００５６】図５において、物理ブロック番号、未使用セクタ数、データ有効セクタ数、及びデータ無効セクタ数を１つの情報単位としており、例えば物理ブロック番号＝０、未使用セクタ数＝１００、データ有効セクタ数＝２０、データ無効セクタ数＝７となっている消去ブロック５１、物理ブロック番号＝１、未使用セクタ数＝０、データ有効セクタ数＝５０、データ無効セクタ数＝７７となっている消去ブロック５１等が存在することを示している。

【００５７】図３のブロック情報表１０、図４のセクタ情報表１１、及び図５のセクタ情報表１２は、ファイルシステム１が起動する段階において、フラッシュメモリ部５から各消去ブロック５１のブロックコントロールセクタ５２ａを読み出すことによってそれぞれ作成される。

【００５８】ブロック情報表１０、セクタ情報表１１、及びセクタ情報表１２を作成しない場合においても、これらの表の情報が各消去ブロック５１のブロックコントロールセクタ５２ａに格納されているため、ファイルシステム１を使用することは可能である。しかしながら、ファイルシステムメモリ部３の動作速度がフラッシュメモリ部５よりも高速なため、ファイルシステムメモリ部３にブロック情報表１０、セクタ情報表１１、及びセク

Samsung Ex. 1002, Page 222 of 622

タ情報表１２を作成して置いておく方が、フラッシュメモリ部５の内部状態やデータの格納場所を高速に把握することができる。また、フラッシュメモリ部５とのデータのやり取りが減るので、ファイルシステムメモリ部３を利用した方が、ファイルシステム１の動作速度を高速にする上で非常に有効である。

【００５９】図６は、本実施形態におけるフラッシュメモリ部５に格納されたデータの読み出し処理に関するフローチャートである。

【００６０】まず、アプリケーションもしくはオペレーティングシステム６においてフラッシュメモリ部５からのデータの読み出し要求が発生した場合、この読み出し要求と共に論理セクタ番号がアプリケーションもしくはオペレーティングシステム６からファイルシステム制御部２に与えられる。ファイルシステム制御部２は、ファイルシステムメモリ部３内のセクタ情報表１１をアクセスして、与えられた論理セクタ番号を検索し、この論理セクタ番号に対応するセクタ状態が「データ有効」となっていることを確認し、この論理セクタ番号に対応する物理セクタ番号を取得し、この物理セクタ番号をフラッシュメモリ制御部４に与える（ステップ１２１）。フラッシュメモリ制御部４は、この物理セクタ番号に基いてフラッシュメモリ部５をアクセスし、この物理セクタ番号に該当する物理的な格納場所のセクタに対して読み出しを行い、読み出したデータをファイルシステム制御部２に与える（ステップ１２２）。読み出すべき残りのデータがある場合は、ステップ１２１へ戻り（ステップ１２３，Ｙｅｓ）、読み出すべき残りのデータがある場合は、終了となる（ステップ１２３，Ｎｏ）。

【００６１】図７Ａ～図７Ｃは、本実施形態におけるフラッシュメモリ部５へのデータの書き込み処理に関するフローチャートである。

【００６２】まず、アプリケーションもしくはオペレーティングシステム６においてフラッシュメモリ部５へのデータの書き込み要求が発生した場合、この書き込み要求と共に論理セクタ番号及び書き込むデータがアプリケーションもしくはオペレーティングシステム６からファイルシステム制御部２に与えられる。ファイルシステム制御部２は、ファイルシステムメモリ部２内のセクタ情報表１２をアクセスして、未使用セクタ数が最も多い物理ブロック番号（書き込み可能な消去ブロック５１を示す）を取得し（ステップ１３１）、取得できたかどうかの確認を行い（ステップ１３２）、この物理ブロック番号が取得出来なかった場合は（ステップ１３２，Ｎｏ）、書き込み可能な消去ブロック５１を確保するために後述するファイルシステム１の再構築を行う（ステップ１３３）。

【００６３】また、未使用セクタ数が最も多い物理ブロック番号を取得できれば（ステップ１３２，Ｙｅｓ）、ファイルシステム制御部２は、セクタ情報表１１をアクセスして、ステップ１３１又は１３３において取得した物理ブロック番号に対応する各物理セクタ番号のうちからセクタ状態が「未使用」となっている物理セクタ番号を取得し、更にセクタ情報表１２をアクセスして、該物理ブロック番号に対応する未使用セクタ数を－１とし、データ有効セクタ数を＋１とする（ステップ１３４）。

【００６４】この後、ファイルシステム制御部２は、ブロック情報表１０をアクセスして、ステップ１３１又は１３３において取得した物理ブロック番号に対応するブロック状態が「未使用」であるかの確認を行い（ステップ１３５）、ブロック状態が「未使用」であれば（ステップ１３５，Ｙｅｓ）、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、該物理ブロック番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａの論理ブロック番号を該物理ブロック番号に更新し、ブロックコントロールセクタ５２ａのブロック状態を「未使用」から「データ有」へ更新する。同様に、ブロック情報表１０をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号に対応する論理ブロック番号を該物理ブロック番号に更新し、ブロック状態を「未使用」から「データ有」へ更新し（ステップ１３６）、ステップ１３７に移る。ブロック状態が「未使用」でなければ（ステップ１３５，Ｎｏ）、ステップ１３６をジャンプしてステップ１３７に移る。

【００６５】次に、ファイルシステム制御部２は、セクタ情報表１２をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号に対応する未使用セクタ数が０かの確認を行い（ステップ１３７）、未使用セクタ数が０であれば（ステップ１３７，Ｙｅｓ）、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、該物理ブロック番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を「データ有」から「ブロックフル」へ更新する。同様に、ブロック情報表１０をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号に対応するブロック状態も「データ有」から「ブロックフル」へ更新する（ステップ１３８）。

【００６６】次に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得した物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａの論理セクタ番号にファイルシステム制御部２に与えられた論理セクタ番号を書き込み、このセクタ５２のセクタ状態を「未使用」から「データ書き込み中」へ更新する。同様に、セクタ情報表１１をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得した物理セクタ番号に対応する論理セクタ番号

Samsung Ex. 1002, Page 223 of 622

をファイルシステム制御部２に与えられた論理セクタ番号に更新し、セクタ状態を「未使用」から「データ書き込み中」へ更新する（ステップ１３９）。

【００６７】この後、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得し物理セクタ番号に対応するセクタ５２に、ファイルシステム制御部２に与えられたデータを書き込む(ステップ１４０）。

【００６８】ファイルシステム制御部２は、フラッシュメモリ部５内のセクタ５２への書き込み終了後、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得した物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を「データ書き込み中」から「データ書き込み完了」へ更新する。同様に、セクタ情報表１１をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得した物理セクタ番号に対応するセクタ状態を「データ書き込み中」から「データ書き込み完了」へ更新する（ステップ１４１）。

【００６９】そして、ファイルシステム制御部２は、フラッシュメモリ部５内のセクタ５２に書き込んだデータが既にフラッシュメモリ部５に書き込まれていたデータの更新データであるかの確認を行う（ステップ１４２）。ここで、既存データの更新であれば（ステップ１４２，Ｙｅｓ）、ファイルシステム制御部２は、セクタ情報表１１をアクセスして、ステップ１４０において書き込みを実行したセクタ５２の論理セクタ番号と同じ論理セクタ番号を有しかつステップ１４０において書き込みを実行したセクタ５２の物理セクタ番号とは異なる他のセクタ５２の物理セクタ番号及び物理ブロック番号、つまり既存のデータが格納されたセクタ５２の物理セクタ番号及び物理ブロック番号を取得する（ステップ１４３）。そして、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、これらの物理ブロック番号並びに物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を「データ有効」から「データ無効」へ更新する。同様に、セクタ情報表１１をアクセスし、これらの物理ブロック番号並びに物理セクタ番号に対応するセクタ状態を「データ有効」から「データ無効」へ更新する。更に、セクタ情報表１２をアクセスし、この物理ブロック番号に対応するデータ有効セクタ数を－１とし、データ無効セクタ数を＋１する（ステップ１４４）。

【００７０】この後、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ステップ１３１又は１３３において取得した物理ブロック番号並びにステップ１３４において取得した物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を「データ書き込み完了」から「データ有効」へ更新する。同様に、セクタ情報表１１をアクセスし、これらの物理ブロック番号並びに物理セクタ番号に対応するセクタ状態も「データ書き込み完了」から「データ有効」へ更新する（ステップ１４５）。

【００７１】引き続いて、ファイルシステム制御部２は、書き込みデータがまだ残っているかの確認を行い（ステップ１４６）、書き込みデータがまだ残っている場合は（ステップ１４６，Ｙｅｓ）、ステップ１３１へ戻り、書き込みデータがなくなれば（ステップ１４６，Ｎｏ）、終了する。

【００７２】図８Ａ～図８Ｂは、本実施形態におけるファイルシステム１の再構築処理に関するフローチャートである。

【００７３】まず、図７Ａ～図７Ｃのフローチャートで説明した様に書き込み可能な消去ブロック５１が取得出来なかった場合は（ステップ１３２，Ｎｏ）、書き込み可能な消去ブロック５１を確保するためにファイルシステムの再構築を行う（ステップ１３３）。このステップ１３３の処理が図８のフローチャートに示す処理に該当する。

【００７４】図８Ａ～図８Ｂにおいて、ファイルシステム制御部２は、セクタ情報表１２をアクセスし、データ無効セクタ数が最も多い物理ブロック番号を取得し、更にブロック情報表１０をアクセスし、取得した物理ブロック番号に対応する論理ブロック番号を取得する（ステップ１５１）。そして、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ファイルシステムの再構築処理用に予め未使用状態で確保しておいた予備消去ブロック５１のブロックコントロールセクタ５２ａの論理ブロック番号をステップ１５１において取得した論理ブロック番号へ更新し、このブロックコントロールセクタ５２ａのブロック状態を「未使用」から「データ転送中」へ更新する。同様に、ブロック情報表１０をアクセスし、予備消去ブロック５１の論理ブロック番号をステップ１５１において取得した論理ブロック番号へ更新し、ブロック状態を「未使用」から「データ転送中」へ更新する（ステップ１５２）。

【００７５】次に、ファイルシステム制御部２は、データをステップ１５１において取得した物理ブロック番号の消去ブロック５１から予備消去ブロック５１へ効率良くコピーするにために、セクタ情報表１１をアクセスし、ステップ１５１において取得した物理ブロック番号に対応する各物理セクタ番号のうちからセクタ状態が「データ有効」である物理セクタ番号を検索して取得する（ステップ１５３）。セクタ状態が「データ有効」の

Samsung Ex. 1002, Page 224 of 622

セクタ５２があれば（ステップ１５４，Ｙｅｓ）、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ステップ１５１において取得した物理ブロック番号並びにステップ１５３において取得した物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態「データ有効」を予備消去ブロック５１の同じ物理セクタ番号に対応するブロックコントロールセクタ５２ａのセクタ状態にコピーする。同様に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、ステップ１５１において取得した物理ブロック番号並びにステップ１５３において取得した物理セクタ番号のセクタ状態「データ有効」を予備消去ブロック５１の同じ物理セクタ番号のセクタ状態にコピーする。同時に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、ステップ１５１において取得した物理ブロック番号並びにステップ１５３において取得した物理セクタ番号に対応する消去ブロック５１のセクタ５２から予備消去ブロック５１の同じ物理セクタ番号のセクタ５２へとデータをコピーする（ステップ１５５）。

【００７６】ファイルシステム制御部２は、ステップ１５１において取得した物理ブロック番号の消去ブロック５１に未処理のセクタ５２があるかの確認を行う（ステップ１５６）。未処理のセクタ５２がある場合は、ステップ１５３へ戻る。未処理のセクタ５２がない場合は、データのコピーが完了したため、ファイルシステム制御部２は、セクタ情報表１２をアクセスし、ステップ１５１において取得した物理ブロック番号に対応するデータ有効セクタ数を予備消去ブロック５１のデータ有効セクタ数へコピーし、またデータ無効セクタ数を０とし、データ有効セクタ数とデータ無効セクタ数と未使用セクタ数の和が総セクタ数（１２７）となる様に、未使用セクタ数を計算して設定する（ステップ１５７）。

【００７７】次に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、予備消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を「データ転送中」から「元ブロック消去中」へ更新し（ステップ１５８）、ステップ１５１において取得した物理ブロック番号の消去ブロック５１のデータを消去し（ステップ１５９）、予備消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を「元ブロック消去中」から「データ有」へ更新する。同様に、ファイルシステム制御部２は、ブロック情報表１０をアクセスし、予備消去ブロック５１のブロック状態を「データ転送中」から「元ブロック消去中」へ更新し（ステップ１５８）、ステップ１５９を経た後に予備消去ブロック５１のブロック状態を「元ブロック消去中」から「データ有」へ更新する。これによって、ステップ１５１において取得した物理ブロック番号のブロック、つまりデータを消去したブロックが新しい予備消去ブロック５１として確保される（ステップ１６０）。

【００７８】ここで、ステップ１５９においてデータを消去した消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態は、データ消去により自動的に「データ有」あるいは「ブロックフル」から「未使用」となる。また、ブロック情報表１０、セクタ情報表１１及びセクタ情報表１２では、ステップ１５９においてデータを消去した消去ブロック５１が予備消去ブロックとなり、フラッシュメモリ部５へのデータの書き込みにおいて予備消去ブロックが参照されることはないため、この予備消去ブロックの物理ブロック番号に対応する項目の更新は不要である。

【００７９】なお、本実施例においては、ファイルシステム１の再構築の処理は、消去ブロック１つを消去する場合を示したが、ステップ１５１からの一連の処理を複数回実行することも可能である。

【００８０】図９Ａ～図９Ｄは、本実施形態におけるファイルシステム１の初期化に関するフローチャートである。この初期化は、ファイルシステム１の立ち上げ時、つまり電力供給が一旦遮断された後の電力供給再開時に一度実行され、フラッシュメモリ部５内のデータの読み出しを保証するものである。

【００８１】まず、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、各消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を読み出し、この読み出したブロック情報に基いて、ブロック情報表１０のブロック情報を更新し（ステップ１７１）、残りの消去ブロック５１が存在するかの確認を行い（ステップ１７２）、残りの消去ブロック５１が無くなるまで、ステップ１７１の処理を繰り返す。

【００８２】次に、ファイルシステム制御部２は、ステップ１７１において更新したブロック情報表１０をアクセスし、ブロック状態「データ転送中」が存在するかの確認を行い（ステップ１７３）、ブロック状態「データ転送中」が存在する場合は、このブロック状態の物理ブロック番号を取得し、この物理ブロック番号に対応するブロック状態を「データ転送中」から「未使用」へ更新する（ステップ１７４）。同時に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、この物理ブロック番号のブロックのデータを消去する。

【００８３】ここで、図８Ａ～図８Ｂのステップ１５１～１５７の途中で電力供給が遮断され、予備消去ブロック５１へのデータ転送が中断した場合は、この予備消去ブロック５１のブロック状態が「データ転送中」として保持される。この後に電力供給が再開されたときには、ステップ１７３、１７４の処理において、データ転送が

Samsung Ex. 1002, Page 225 of 622

中断したブロック状態「データ転送中」の予備消去ブロック５１のデータが消去され、この予備消去ブロック５１のブロック状態が「未使用」となる。これによって不完全なデータが予備消去ブロック５１から読み出されることが防止される。

【００８４】次に、ファイルシステム制御部２は、ステップ１７１において更新したブロック情報表１０をアクセスし、ブロック状態が「元ブロック消去中」であるブロックが存在するかの確認を行い（ステップ１７５）、ブロック状態が「元ブロック消去中」であるブロックが存在する場合は、このブロックの論理ブロック番号と同じ論理ブロック番号であってかつブロック状態が「データ有」である物理ブロック番号を取得する。そして、ファイルシステム制御部２は、ブロック情報表１０をアクセスし、この物理ブロック番号に対応するブロック状態を「未使用」へ更新する。同時に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、この物理ブロック番号のブロックのデータを消去する。

【００８５】また、ファイルシステム制御部２は、ステップ１７５において取得したブロック状態が「元ブロック消去中」である物理ブロック番号の消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態情報を「元ブロック消去中」から「データ有」へ更新し、ブロック情報表１０における同じ物理ブロック番号に対応するブロック状態を「データ有」へ更新する（ステップ１７６）。

【００８６】この後、ファイルシステム制御部２は、消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を読み出し、このセクタ情報に基いてファイルシステムメモリ部３内のセクタ情報表１１を更新し（ステップ１７７）、残りのセクタ情報が存在するかの確認を行い（ステップ１７８）、残りのセクタ情報が存在する場合は、ステップ１７７へ戻り、残りのセクタ５２がなくなるまで繰り返し、同様に残りの消去ブロック５１が存在する場合は、ステップ１７７へ戻り、残りの消去ブロック５１がなくなるまで繰り返す。

【００８７】ここで、図８Ａ～図８Ｂのステップ１５８～１６０の途中で電力供給が遮断された場合は、この予備消去ブロック５１のブロック状態が「元ブロック消去中」として保持されており、この予備消去ブロック５１へのデータ転送が終了している。この後に電力供給が再開されると、ステップ１７５～１７８の処理において、予備消去ブロック５１のデータを保持し、データ転送元の消去ブロック５１のデータを消去している。これによってデータ転送元の消去ブロック５１から不完全なデータが読み出されることが防止される。

【００８８】次に、ファイルシステム制御部２は、ステップ１７７において更新したセクタ情報表１１をアクセスし、セクタ状態が「データ書き込み中」であるセクタ５２が存在するかの確認を行い（ステップ１７９）、セクタ状態が「データ書き込み中」であるセクタ５２が存在する場合は、ステップ１７９において取得したセクタ状態「データ書き込み中」に対応する物理ブロック番号及び物理セクタ番号を取得する。そして、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、これらの物理ブロック番号及び物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を「データ書き込み中」から「データ無効」へ更新する。同様に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、これらの物理ブロック番号及び物理セクタ番号に対応するセクタ状態を「データ書き込み中」から「データ無効」へ更新する(ステップ１８０）。

【００８９】次に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、セクタ状態が「データ書き込み完了」であるセクタ５２が存在するかの確認を行い（ステップ１８１）、セクタ状態が「データ書き込み完了」であるセクタ５２が存在する場合は、このセクタ５２の論理セクタ番号と同じ論理セクタ番号を有しかつセクタ状態が「データ有効」のセクタ５２が存在するかの確認を行い（ステップ１８２）、このセクタ５２が存在する場合は、このセクタ５２の物理ブロック番号及び物理セクタ番号を取得する。そして、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、これらの物理ブロック番号及び物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａのセクタ状態を「データ有効」から「データ無効」へ更新する。同様に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、これらの物理ブロック番号及び物理セクタ番号に対応するセクタ状態を「データ有効」から「データ無効」へ更新する（ステップ１８３）。

【００９０】次に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、ステップ１８１において取得したセクタ状態が「データ書き込み完了」であるセクタ５２の物理ブロック番号及び物理セクタ番号を取得する。そして、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、これらの物理ブロック及び物理セクタ番号に対応する消去ブロック５１のブロックコントロールセクタ５２ａののセクタ状態を「データ書き込み完了」から「データ有効」へ更新する。同様に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、これらの物理ブロック番号及び物理セクタ番号に対応するセクタ状態を「データ書き込み完了」から「データ有効」へ更新する（ステップ１８４）。

【００９１】ここで、図７Ａ～図７ｃのステップ１３９～１４１の途中で電力供給が遮断され、データの書き込みが中断した場合は、セクタ状態が「データ書き込み

Samsung Ex. 1002, Page 226 of 622

中」として保持され、またデータの書き込みが完了した場合は、セクタ状態が「データ書き込み完了」として保持される。この後に電力供給が再開されると、ステップ１７９～１８４の処理において、データの書き込みが中断したセクタ状態「データ書き込み中」のセクタ５２のデータが消去され、このセクタ５２のセクタ状態が「データ無効」となる。これによってデータの書き込みが中断したセクタ５２から不完全なデータが読み出されることが防止される。また、セクタ状態「データ書き込み完了」のセクタ５２のデータが保持され、このセクタ状態が「データ有効」に更新されるので、このセクタ５２からデータを正しく読み出すことができる。

【００９２】次に、ファイルシステム制御部２は、セクタ情報表１１をアクセスし、物理ブロック番号毎に、同一物理ブロック番号に対応するセクタ状態「未使用」の数、セクタ状態「データ有効」の数、セクタ状態「データ無効」の数をそれぞれ計数し、これらの計数値に基いてファイルシステムメモリ部３内のセクタ情報表１２の物理ブロック番号、未使用セクタ数、データ有効セクタ数及びデータ無効セクタ数を更新し（ステップ１８５）、残りの物理ブロック番号が存在するかの確認を行い（ステップ１８６）、残りの物理ブロック番号が存在する場合は、ステップ１８５へ戻り、残りの物理ブロック番号がなくなるま繰り返す。

【００９３】この後、ファイルシステム制御部２は、ステップ１７１、ステップ１７４、ステップ１７５において更新されたブロック情報表１０に基いて、ブロック状態が「未使用」である物理ブロック番号に対応する１つの消去ブロック５１を予備消去ブロック５１として確保する（ステップ１８７）。

【００９４】この様に本実施形態においては、データの書き込みやファイルシステムの再構築に際し、各消去ブロック５１や各セクタ５２の状態情報をきめ細かく書き換えているため、電力供給が不用意に遮断されも、この後に電力供給が再開されたときには、各消去ブロック５１や各セクタ５２の状態情報に基いて、消去ブロック５１やセクタ５２の不確定なデータを消去して、正しいデータのみを保存することができる。

【００９５】上記実施形態においては、現在のブロック状態として、（１）「未使用」を表す１１１１１１１１ｂ（ｂは２進数を表わす。以下同様）、（２）「データ転送中」を表す１１１１１１１０ｂ、（３）「元ブロック消去中」を表す１１１１１１００ｂ、（４）「データ有」を表す１１１１１０００ｂ、（５）「ブロックフル」を表す１１１１０００００ｂという５種類のブロック状態を例示したが、１と０の組み合わせを複雑にしたり、ビット数を増やしても良く、これによって各状態の違いをより確実に判定することが可能になる。

【００９６】また、よりきめ細かくブロック状態を管理するため、ブロックの状態数を６つ以上に設定しても良く、これによってよりきめ細かく状態を管理することが可能になる。

【００９７】また、上記実施形態においては、現在のセクタ状態として、（Ｉ）「未使用」を表す１１１１ｂ、（ＩＩ）「データ書き込み中」を表す１１１０ｂ、（ＩＩＩ）「データ書き込み完了」を表す１１００ｂ、（ＩＶ）「データ有効」を表す１０００ｂ、（Ｖ）「データ無効」を表す００００ｂという５種類のセクタ状態を例示したが、１と０の組み合わせを複雑にしたり、ビット数を増やしても良く、これによって各状態の違いをより確実に判定することが可能になる。

【００９８】また、よりきめ細かくセクタ状態を管理するため、セクタ５２の状態数を６つ以上に設定しても良く、これによってよりきめ細かく状態を管理することが可能になる。

【００９９】上記実施形態においては、均等ブロックサイズであるフラッシュメモリを前提として説明したが、ブートブロック構成のフラッシュメモリにおいても、ブートブロック（例えば８Ｋバイトのサイズ）用の任意の１個の消去ブロック（８Ｋバイト）と、それ以外のブロック（例えば６４Ｋバイトのサイズ）用の任意の１個の消去ブロック（６４Ｋバイト）をファイルシステム１の再構築用の予備ブロックとして、予め確保しても良く、これによって均等ブロックあるいはブートブロック構成のフラッシュメモリのうちからシステムの目的に合うフラッシュメモリあるいはより安価なフラッシュメモリを採用することが可能となる。また、ブートブロック（例えば８Ｋバイト）用の予備ブロックとして、８Ｋバイトより大きいサイズのブロック（例えば６４Ｋバイト）を予備ブロックに割り当てることも可能であり、この場合は、予備ブロックのサイズを一定にできるため、管理が簡単になる。

【０１００】本実施形態のファイルシステム１が複数個のフラッシュメモリを備える場合は、１個のフラッシュメモリ毎に任意の１個の消去ブロックをファイルシステムの再構築用の予備ブロックとして予め確保しておくのではなく、複数個のフラッシュメモリに対し１個の予備ブロックを確保したり、複数個のフラッシュメモリに含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個の予備ブロックを確保しても良く、この場合は、確保する予備ブロックの数を少なくすることができる。

【０１０１】図９Ｂに示すステップ１７６においては、ファイルシステム制御部２は、ステップ１７５において取得したブロック状態「元ブロック消去中」の消去ブロック５１の論理ブロック番号と同じ論理ブロック番号であって、且つブロック状態が「データ有」である物理ブロック番号を取得する。そして、ファイルシステム制御部２は、ブロック情報表１０をアクセスし、この物理ブロック番号に対応するブロック状態を「未使用」へ更新

Samsung Ex. 1002, Page 227 of 622

する。同時に、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、この物理ブロック番号の消去ブロック５１のデータを消去する。また、ファイルシステム制御部２は、ステップ１７５において取得したブロック状態「元ブロック消去中」の物理ブロック番号の消去ブロック５１のブロックコントロールセクク５２ａのブロック状態を「元ブロック消去中」から「データ有」へ更新し、ブロック情報表１０における同じ物理ブロック番号に対応するブロック状態を「データ有」へ更新する。

【０１０２】ここで、消去ブロック５１についての複数の状態惰報としては、『未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」の５つに加え、「元ブロック消去完了」を設定した場合の例として以下のような処理の流れにすることが可能である。

【０１０３】図１３は、図９Ａ～図９Ｄのファイルシステム１の初期化に関するフローチャートの内、本実施形態におけるステップ１７４からステップ１７７の間に相当する部分を抜き出したフローチャートを示す図である。

【０１０４】まず、ファイルシステム制御部２は、ステップ１７１において更新したブロック情報表１０をアクセスし、ブロック状態「元ブロック消去完了」が存在するかを確認し（ステップ１９１）、ブロック状態「元ブロック消去完了」が存在する場合は、処理はステツプ１９２およびステップ１９３をスキップしてステップ１９４へジャンプする。

【０１０５】次に、ファイルシステム制御部２は、ステップ１７１において更新したブロック情報表１０をアクセスし、ブロック状態「元ブロック消去中」が存在するかを確認し（ステップ１９２）、ブロック状態「元ブロック消去中」が存在する場合は、この消去ブロック５１の論理ブロック番号と同じ論理ブロック番号であって、且つブロック状態が「データ有」である物理ブロック番号を取得し、ファイルシステム制御部２は、フラッシュメモリ制御部４を通じてフラッシュメモリ部５をアクセスし、この物理ブロック番号の消去ブロック５１のデータを消去する。

【０１０６】この後、ファイルシステム制御部２は、ステツプ１９２において取得したブロック状態が「元ブロック消去中」である物理ブロック番号の消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を「元ブロック消去中」から「元ブロック消去完了」へ更新する。

【０１０７】また、ファイルシステム制御部２は、ブロック情報表１０をアクセスし、データ消去をした消去ブロック５１の物理ブロック番号に対応するブロック状態を「未使用」へと更新する（ステップ１９３）。

【０１０８】次に、ブロック状態「元ブロック消去完了」へ更新した物理ブロック番号の消去ブロック５１のブロックコントロールセクタ５２ａのブロック状態を「元ブロック消去完了」から「データ有」へ更新し、ブロック情報表１０におけるこの物理ブロック番号に対応するブロック状態を「元ブロック消去中」から「データ有」へ更新する（ステップ１９４）。

【０１０９】この様に、ブロック状態「元ブロック消去完了」を追加することにより、比較的処理時間の長いブロック消去処理をスキップすることが可能、すなわち、電力供給の遮断直前のブロック状態が「元ブロック消去中」から「元ブロック消去完了」へ更新された直後であった場合は、電力供給が回復した時の初期化処理時間を短縮することが可能である。

【０１１０】他の実施形態では、図１２に示すように、消去ブロック５１についての複数の状態情報として、「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」の５つを設定しても良い。

【０１１１】この場合、再構築用に予め確保されている予備消去ブロック５１の状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック５１内のデータを該予備消去ブロック５１へ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロック５１のブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロック５１のデータを消去し、このデータ転送元の消去ブロック５１を新たな予備消去ブロック５１として確保し、この新たな予備消去ブロック５１の状態情報を「データ有」から「未使用」へ更新する手段と、データ転送先の該予備消去ブロック５１の状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データ転送元の新たな予備消去ブロック５１へ新たな予備消去ブロック５１の消去可能回数などを書き込む（フォーマットする）手段と、フォーマット完了後、データ転送先の該予備消去ブロック５１の状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを合む構成が考えられる。

【０１１２】この様に、消去ブロック５１の状態情報として「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という５つの状態を利用した場合、消去ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１１３】他の実施形態における電力回復時のファイルシステム１の初期化においてブロック状態「元ブロック消去中」の消去ブロック５１が検出された場合は、デ

Samsung Ex. 1002, Page 228 of 622

ータ転送元の消去ブロック５１のデータを消去する。または、更に既存データの正確性を期すために、該初期化において「元ブロック消去中」だけでなく「元ブロックフォーマット中」の消去ブロック５１が検出された場合もデータ転送元の消去ブロック５１のデータを消去する方法が考えられる。

【０１１４】また、消去ブロック５１の状態情報として「ブロックフル」という状態を設定する前記実施形態と異なり、ブロック状態「ブロックフル」を設定しない方法は、消去ブロックが不均等なサイズのセクタに分割されて使用される場合に特に有効である。なぜならば、不均等なサイズのデータを同一の消去ブロック内の連続したセクタに書き込む場合において、次に書き込むべきデータのサイズが不明である場合に、消去ブロックの状態情報が「ブロックフル」となるかどうかの判断が難しいためである。

【０１１５】また、各消去ブロック５１のブロック消去回数を平均化するために、各消去ブロック５１内に該消去ブロックの消去可能回数を記憶する方法がある。

【０１１６】図８Ａおよび図８Ｂに示す他の実施形態としては、ファイルシステム１の再構築時において、ファイルシステム制御部２は、データの転送元となる消去ブロックとして予備消去ブロック５１を除いた各消去ブロック５１の消去可能回数を比較し、該回数の最大値と最小値の差が一定値を越えた場合には、該回数が最大の消去ブロック５１を選択し、差が一定値を越えていない場合は、消去ブロック５１内のセクタ情報表１２をアクセスし、データ無効セクタ数が最大の消去ブロック５１を選択する。

【０１１７】この場合、各消去ブロック５１の消去可能回数のバラツキを一定の回数内に抑え、フラッシュメモリとしての書き替え回数を増やすことが可能になる。

【０１１８】また、消去ブロック５１の消去可能回数は、０から1つずつ増加させ、消去可能回数の仕様値の最大値までインクリメントする方法（この場合は消去可能回数でなく消去回数を表す）と、これとは逆に消去可能回数の仕様値の最大値から１つずつ減少させ、０までデクリメントする方法が考えられるが、後者の方法においては、消去可能回数が０になった時点で該消去ブロック５１の寿命が来たと判定できるため、消去可能回数の判定アルゴリズムを簡単にできる。

【０１１９】他の実施形態における消去ブロック５１のフォーマット手段では、フォーマットした日時を記憶することも可能であり、この場合、一定時間経過後に消去ブロック５１内のデータのリフレッシュ動作を行うことが可能になる。

【０１２０】

【発明の効果】本発明のファイルシステムは、複数のセクタをそれぞれ含む複数の消去ブロックを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各消去ブロック毎に、消去ブロックについての複数の状態情報のいずれかを格納すると共に、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各消去ブロックの状態情報及び該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備する。

【０１２１】本発明によれば、ブロックについての複数の状態情報及びセクタについての複数の状態情報を記憶手段に格納している。このため、ファイルシステムが稼働中において、意図しない電力供給の遮断が生じても、電力供給が回復したときには、起動時の初期化処理に際し、ブロックについての複数の状態情報及びセクタについての複数の状態情報に基いて、ブロック及びセクタの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１２２】一実施形態では、前記記憶手段は、データの更新に制約の無いＲＡＭ上に設けられ、前記保証手段は、前記不揮発性半導体記憶装置へのアクセス時に、該ＲＡＭ上の前記各消去ブロックの状態情報及び前記各セクタの状態情報を参照する。

【０１２３】この様にＲＡＭを利用すると、データの更新に制約がなくなり処理速度が早くなる。

【０１２４】一実施形態では、前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」という少なくとも５つの状態を設定しており、前記保証手段は、再構築用に予め確保されている予備ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、このデータ転送元の消去ブロックを新たな予備ブロックとして確保し、この新たな予備ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へ更新する手段と、データ転送先の該予備ブロックの状態情報を「元ブロック消去中」から「データ有」へ更新する手段とを含む。

【０１２５】この様に消去ブロックの状態情報として「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」という５つの状態を利用した場合、ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシ

Samsung Ex. 1002, Page 229 of 622

ュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１２６】一実施形態では、再構築用に予め確保されている予備ブロックは、１個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する。

【０１２７】この場合、均等ブロックあるいはブートブロック構成のフラッシュメモリのうちからシステムの目的に合わせた、あるいはより安価なフラッシュメモリを採用することが可能となる利点がある。

【０１２８】一実施形態では、再構築用に予め確保されている予備ブロックは、複数個の不揮発性半導体記憶装置に含まれるブロックサイズが異なる各消去ブロック毎に、少なくとも１個以上有する。

【０１２９】この場合、ファイルシステムが必要とする予備ブロックの数を少なくできる利点がある。このため、ファイルシステムが安価に構成できる利点もある。

【０１３０】一実施形態では、再構築用に予め確保されている予備ブロックは、消去ブロックのサイズ以上のサイズを有する。

【０１３１】この場合は、ブートブロック（例えば８Ｋバイトのサイズ）用の予備ブロックとして、８Ｋバイトより大きいサイズのブロック（例えば６４Ｋバイトのサイズ）を予備ブロックに割り当て、予備ブロックのサイズを一定にできるため、管理が簡単になる。

【０１３２】本発明のファイルシステムは、複数のセクタを有する不揮発性半導体記憶装置を用いたファイルシステムにおいて、該各セクタ毎に、セクタについての複数の状態情報のいずれかを格納する記憶手段と、該不揮発性半導体記憶装置のアクセス時に、該記憶手段内の該各セクタの状態情報に基いて、該不揮発性半導体記憶装置に既に格納されているデータを保証する保証手段とを具備し、前記セクタについての複数の状態情報として、「未使用」、「データ書き込み中」、「データ書き込み完了」、「データ有効」、「データ無効」という少なくとも５つの状態を設定しており、前記保証手段は、データが書き込まれるセクタの状態情報を「未使用」から「データ書き込み中」へ更新する手段と、該セクタへのデータの書き込みが完了した後、該セクタの状態情報を「データ書き込み中」から「データ書き込み完了」へ更新する手段と、該セクタにデータを新規に書き込んだ場合は、該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新し、該セクタに既存のデータの更新データを書き込んだ場合は、該既存のデータが書き込まれている他のセクタの状態情報を「データ有効」から「データ無効」に更新後、該更新データが書き込まれている該セクタの状態情報を「データ書き込み完了」から「データ有効」へ更新する。

【０１３３】本発明によれば、セクタについての５つの状態情報を記憶手段に格納している。このため、ファイルシステムが稼働中において、意図しない電力供給の遮断が生じても、電力供給が回復したときには、起動時の初期化処理に際し、セクタについての複数の状態情報に基いて、セクタの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１３４】一実施形態では、前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」、「元ブロック消去完了」という少なくとも６つの状態を設定しており、前記保証手段は、再構築用に予め確保されている予備消去ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備消去ブロックへ転送して記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、該予備消去ブロックのブロック状態情報を「元ブロック消去中」から「元ブロック消去完了」へ更新する手段と、データ転送元の消去ブロックを新たな予備消去ブロックとして確保し、この新たな予備消去ブロックの状態情報を「データ有」あるいは「ブロックフル」から「未使用」へと更新する手段と、データ転送先の該予備消去ブロックの状態情報を「元ブロック消去完了」から「データ有」へ変更する手段とを含む。

【０１３５】この様に、消去ブロックの状態情報として「未使用」、「データ有」、「ブロックフル」、「データ転送中」、「元ブロック消去中」、「元ブロック消去完了」という６つの状態を利用した場合、前記５つの状態を利用した場合よりも更に消去ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また、既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１３６】一実施形態では、前記消去ブロックについての複数の状態情報として、「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という少なくとも５つの状態を設定しており、前記保証手段は、再構築に予め確保されている予備消去ブロックの状態情報を「未使用」から「データ転送中」へ更新する手段と、データ転送元の消去ブロック内のデータを該予備消去ブロックへ転送して

Samsung Ex. 1002, Page 230 of 622

記憶させる手段と、該データの転送並びに記憶の完了後、該予備消去ブロックのブロック状態情報を「データ転送中」から「元ブロック消去中」へ更新し、該データ転送元の消去ブロックのデータを消去し、このデータ転送元の消去ブロックを新たな予備消去ブロックとして確保し、この新たな予備消去ブロックの状態情報を「データ有」から「未使用」へ更新する手段と、データ転送先の該予備消去ブロックの状態情報を「元ブロック消去中」から「元ブロックフォーマット中」へ更新し、データ転送元の新たな予備消去ブロックへ新たな予備消去ブロックの消去可能回数などを書き込む（フォーマットする）手段と、フォーマット完了後、データ転送先の該予備消去法ブロックの状態情報を「元ブロックフォーマット中」から「データ有」へ更新する手段とを含む。

【０１３７】この様に、消去ブロックの状態情報として「未使用」、「データ有」、「データ転送中」、「元ブロック消去中」、「元ブロックフォーマット中」という５つの状態を利用した場合、消去ブロックの状態を正確に把握してきめ細かにフラッシュメモリを管理することができ、フラッシュメモリに格納されている既存のデータを破壊することなく、また既存のデータに矛盾を生じさせることなく、電力供給の遮断直前に実行中であった前段階へファイルシステムを自動的に回復させることが可能となる。

【０１３８】この場合、消去ブロックの状態情報として「ブロックフル」という状態を設定する前記実施形態と異なり、ブロック状態「ブロックフル」を設定しない方法は、消去ブロックが不均等なサイズのセクタに分割されて使用される場合に特に有効である。なぜならば、不均等なサイズのデータを同一の消去ブロック内の連続したセクタに書き込む場合において、次に書き込むべきデータのサイズが不明である場合に、消去ブロックの状態情報が「ブロックフル」となるかどうかの判断が難しいためである。

【０１３９】一実施形態では、各消去ブロックのブロック消去回数を平均化するために、各消去ブロック内に該消去ブロックの消去可能回数を記憶する。この消去可能回数をファイルシステムの再構築時において利用することにより、各消去ブロックの消去可能回数のバラツキを一定の回数内に抑え、フラッシュメモリとしての総書き替え回数を向上させることが可能になる。

【０１４０】また、消去ブロックの消去可能回数は、０から１つずつ増加させ、消去可能回数の仕様値の最大値までインクリメントする方法（この場合は消去可能回数でなく消去回数を表す）と、これとは逆に消去可能回数の仕様値の最大値から１つずつ減少させ、０までデクリメントする方法が考えられるが、後者の方法においては、消去可能回数が０になった時点で該消去ブロックの寿命が来たと判定できるため、消去可能回数の判定アルゴリズムを簡単にできる。

【０１４１】一実施形態における消去ブロックのフォーマット手段では、フォーマットした日時を記憶することも可能であり、この場合、一定時間経過後に消去ブロック内のデータのリフレッシュ動作を行うことが可能になる。

【図面の簡単な説明】

【図１】本発明のファイルシステムの一実施形態を示すブロック図である。

【図２】図１のファイルシステムのフラッシュメモリ部に格納されているデータの構成図である。

【図３】図１のファイルシステムにおけるファイルシステムメモリ部に格納されている各消去ブロックに関するデータの構成図である。

【図４】図１のファイルシステムにおけるファイルシステムメモリ部に格納されている各セクタに関するデータの構成図である。

【図５】図１のファイルシステムにおけるファイルシステムメモリ部に格納されている各消去ブロック毎に、各セクタの数をそれぞれのセクタ状態に応じて整理したデータの構成図である。

【図６】図１のファイルシステムにおけるデータの読み出し処理に関するフローチャートである。

【図７Ａ】図１のファイルシステムにおけるデータを書き込む処理に関するフローチャートである。

【図７Ｂ】図１のファイルシステムにおけるデータを書き込む処理に関する図７Ａに続くフローチャートである。

【図７Ｃ】図１のファイルシステムにおけるデータを書き込む処理に関する図７Ｂに続くフローチャートである。

【図８Ａ】図１のファイルシステムの再構築処理に関するフローチャートである。

【図８Ｂ】図１のファイルシステムの再構築処理に関する図８Ａに続くフローチャートである。

【図９Ａ】図１のファイルシステムの初期化に関するフローチャートである。

【図９Ｂ】図１のファイルシステムの初期化に関する図９Ａに続くフローチャートである。

【図９Ｃ】図１のファイルシステムの初期化に関する図９Ｂに続くフローチャートである。

【図９Ｄ】図１のファイルシステムの初期化に関する図９Ｃに続くフローチャートである。

【図１０】従来の技術におけるフラッシュメモリに格納されたデータの更新処理に関するフローチャートである。

【図１１】従来の技術におけるファイルシステムの再構築処理に関するフローチャートである。

【図１２】図１のファイルシステムにおけるフラッシュメモリ部に格納されているデータの構成図の内、ブロック状態情報のブロック状態を表す他の実施形態を示す図

Samsung Ex. 1002, Page 231 of 622

である。

【図１３】図１のファイルシステムの初期化に関する図９Ａ～図９Ｄの内、状態情報として「元ブロック消去完了」を設定した第１の実施形態に関わる部分のフローチャートを示す図である。

【符号の説明】

１　ファイルシステム

２　ファイルシステム制御部

３　ファイルシステムメモリ部

４　フラッシュメモリ制御部

５　フラッシュメモリ部

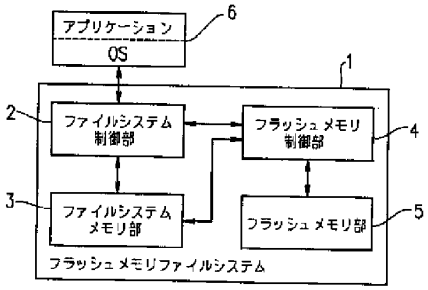
６　アプリケーション、ＯＳ（オペレーティングシステム）

５１　消去ブロック、予備消去ブロック

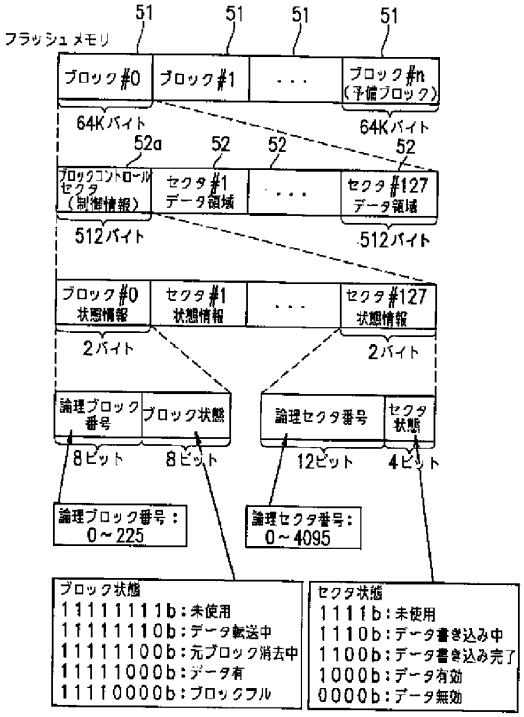
５２　セクタ

５２ａ　ブロックコントロールセクタ

【図１】



【図２】



【図３】

10

物理ブロック番号	論理ブロック番号	ブロック状態
0	0	データ有
1	1	データ・フル
⋮	⋮	⋮

【図４】

11

物理ブロック番号	物理セクタ番号	論理セクタ番号	セクタ状態
0	0	100	データ有効
0	1	10	データ無効
⋮	⋮	⋮	⋮

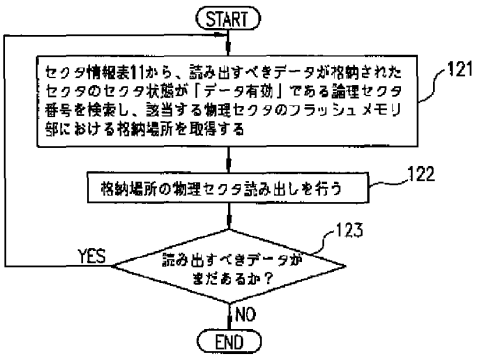
【図５】

12

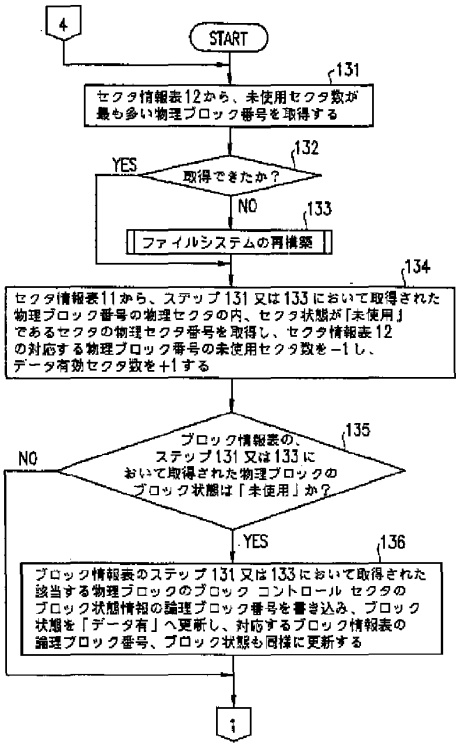
物理ブロック番号	未使用セクタ数	データ有効セクタ数	データ無効セクタ数
0	100	20	7
1	0	50	77
⋮	⋮	⋮	⋮

Samsung Ex. 1002, Page 232 of 622

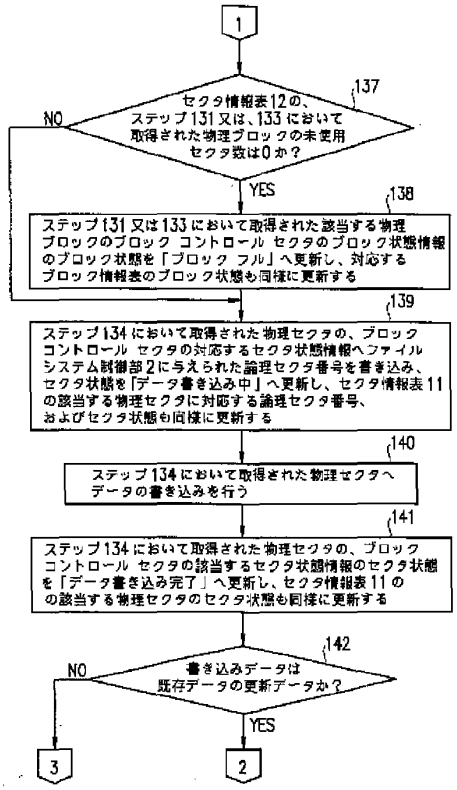
【図６】



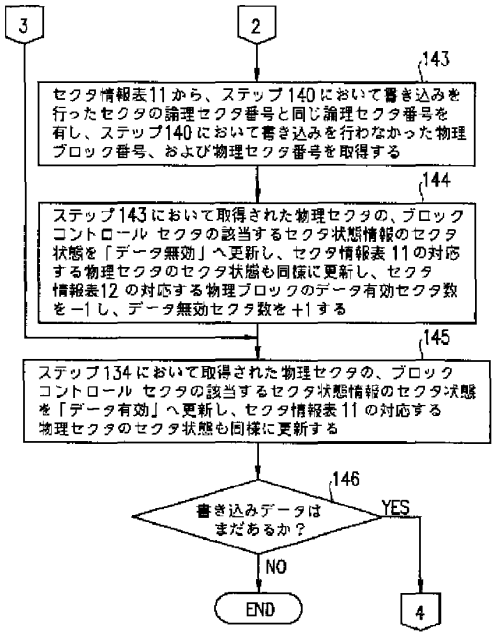
【図７Ａ】



【図７Ｂ】

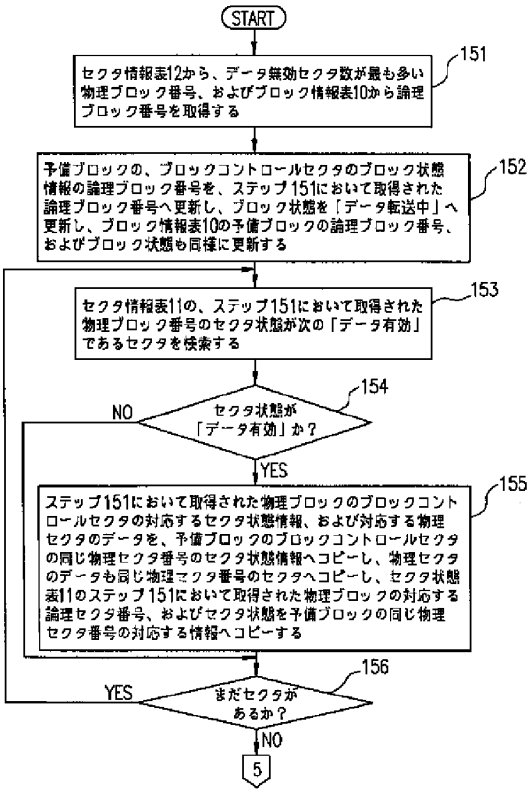


【図７Ｃ】

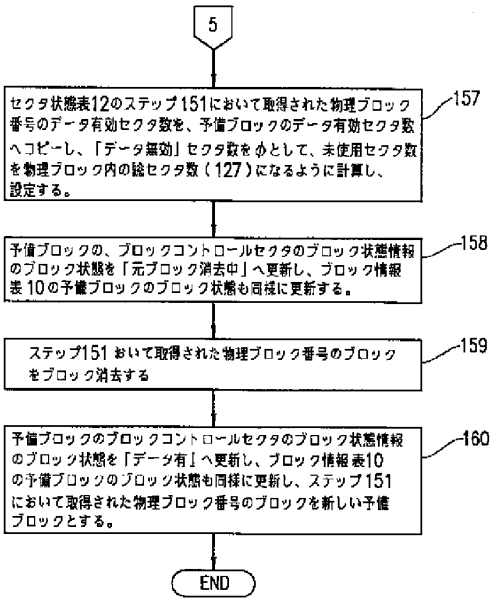


Samsung Ex. 1002, Page 233 of 622

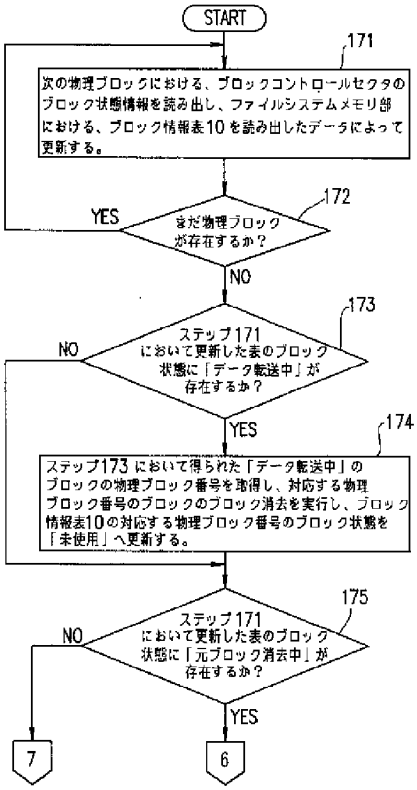
【図８Ａ】



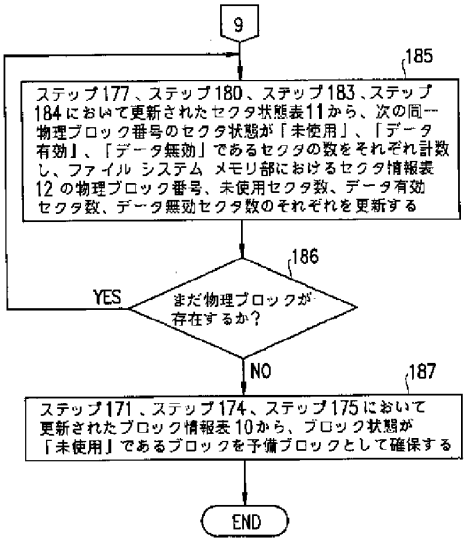
【図８Ｂ】



【図９Ａ】

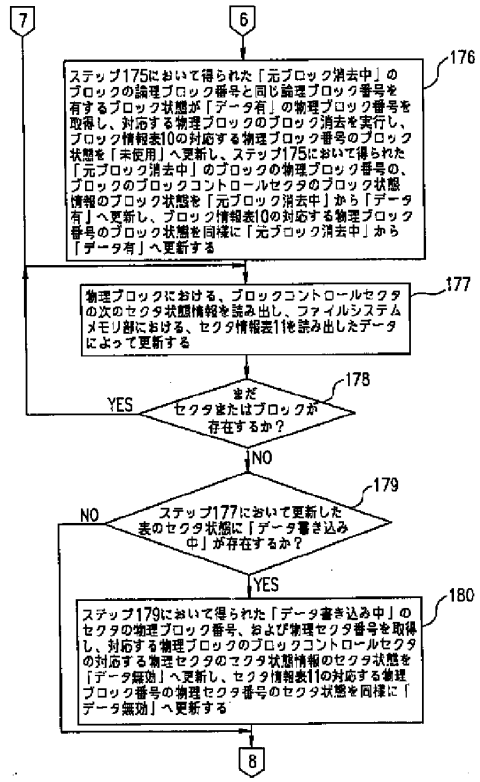


【図９Ｄ】

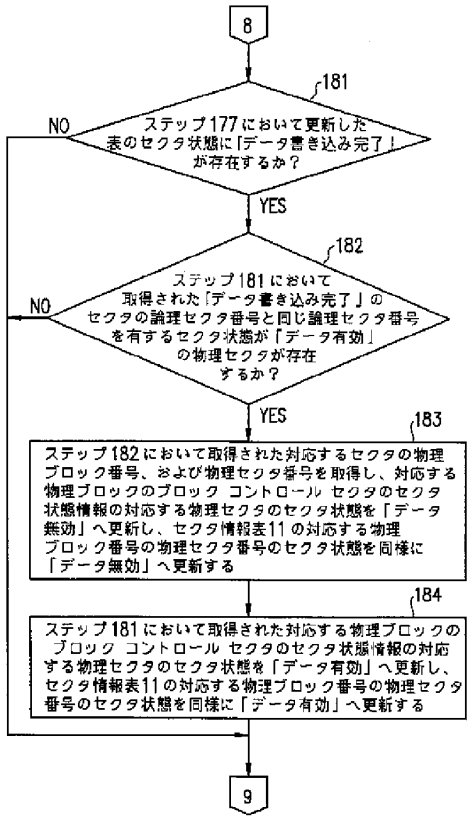


Samsung Ex. 1002, Page 234 of 622

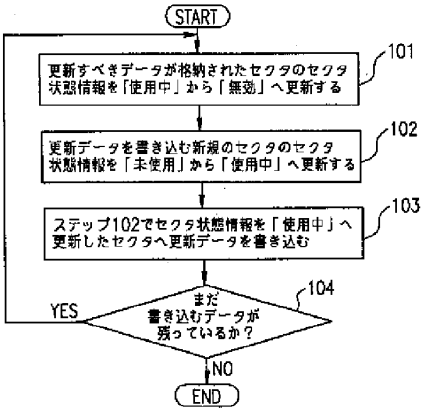
【図９Ｂ】



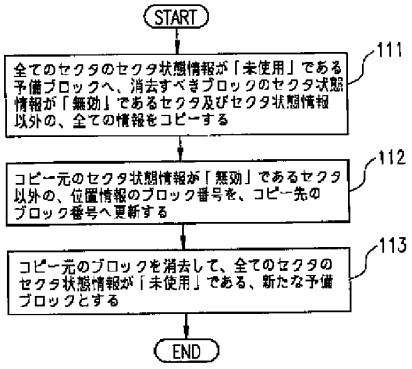
【図９Ｃ】



【図１０】

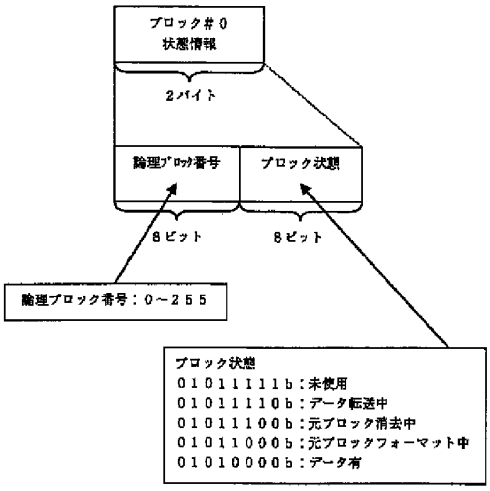


【図１１】



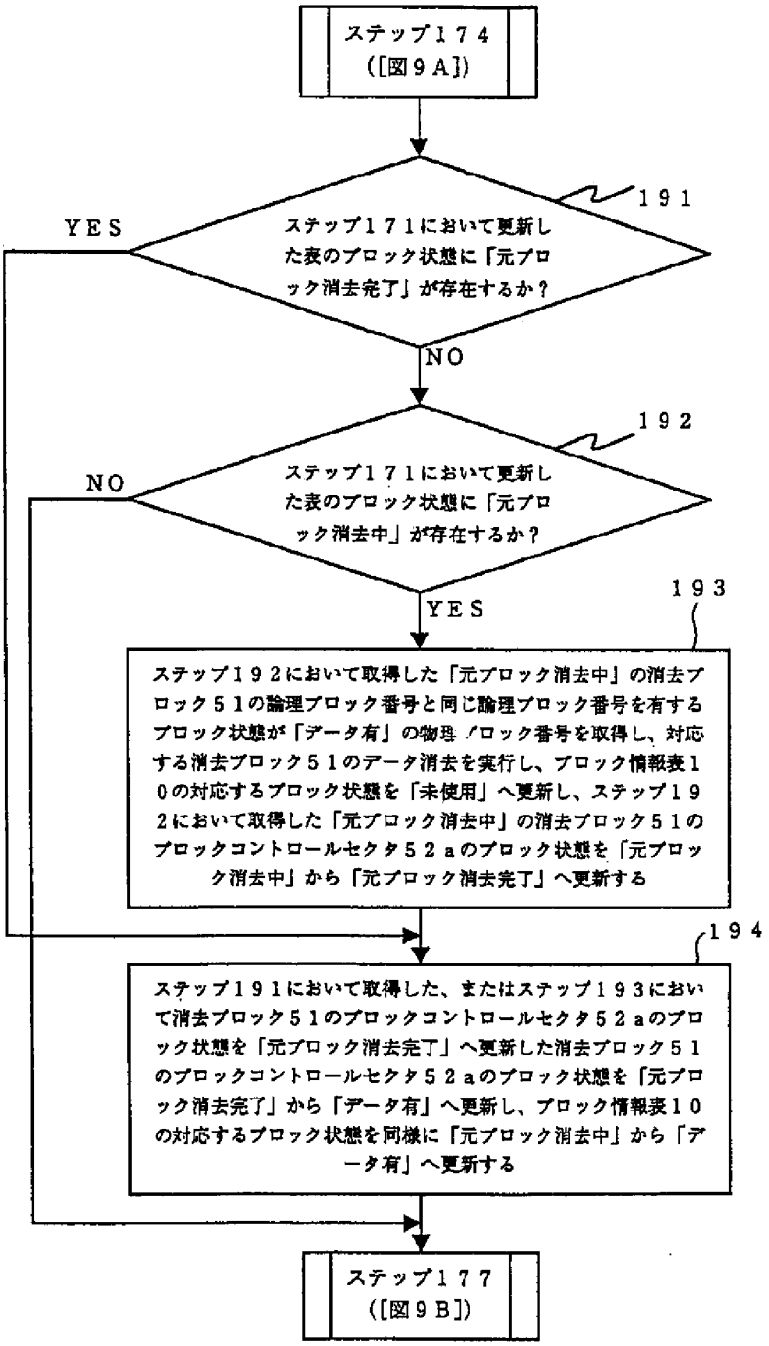
Samsung Ex. 1002, Page 235 of 622

【図１２】



Samsung Ex. 1002, Page 236 of 622

【図１３】



Samsung Ex. 1002, Page 237 of 622

MEMORY CONTROLLER AND SEMICONDUCTOR STORAGE DEVICE

DOCUMENT ID
JP-2011203916-A

DATE PUBLISHED
2011-10-13

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
SUKEGAWA HIROSHI	N/A	N/A	N/A	N/A

DATE FILED
2010-03-25

CPC CURRENT

TYPE	CPC	DATE
CPCI	G06F12/0246	2013-01-01
CPCI	G06F12/0246	2013-01-01
CPCA	G06F2212/7201	2013-01-01
CPCA	G06F2212/7211	2013-01-01
CPCA	G06F2212/7211	2013-01-01
CPCA	G06F2212/7201	2013-01-01

Abstract

PROBLEM TO BE SOLVED: To provide a memory controller performing block erasure frequency averaging with higher accuracy.SOLUTION: The memory controller 10 performing rewriting control of a nonvolatile memory 40 having a plurality of blocks includes: a first management table 31 storing correspondence between logical block addresses which are logical addresses of block unit designated by a host 3 and physical block addresses indicating storage positions in the memory 40 of the blocks; a second management table 31 storing a writing frequency from the host 3 into the memory 40 in each logical block address; third management tables 31, 32 storing an erasure frequency of each physical block address; and a control part 23 selecting a spare block not having a corresponding logical address to a writing request wherein the logical address from the host 3 is designated, and performing writing in the spare block. The memory controller averages the erasure frequency between the plurality of blocks based on the writing frequency stored in each logical block address and the erasure frequency stored in each physical block address.

Description

TECHNICAL-FIELD

The present invention relates to a memory controller and a semiconductor memory device.

BACKGROUND-ART

The NAND flash memory simplifies the structure by collectively erasing data stored in a plurality of memory cells called blocks, and realizes a low cost and a large capacity. Furthermore, NAND flash memories are widely used as storage devices in place of HDDs, host storage devices such as mobile phones or portable

music players because they have no moving parts and low power consumption. However, it is known that the NAND flash memory has a limit on the number of times of writing / erasing.

CITATION-LIST

PATENT-LITERATURE

[1] JP 2004-326523 A

DESCRIPTION-OF-EMBODIMENTS

The limitation on the number of times of writing / erasing to the memory cell of the NAND flash memory is that a high voltage is applied to the gate to the substrate and electrons are injected to the floating gate (writing), and a high voltage is applied to the source relative to the gate. This is caused by the fact that electrons are extracted from the floating gate (erasing). That is, if the write / erase process is performed many times on a specific memory cell, the oxide film around the floating gate may deteriorate and the data may be destroyed.

In order to avoid such a write / erase process from being concentrated on a specific memory cell, the memory controller attempts to average the number of write / erase processes. This is because the memory controller counts the number of erasures of physical blocks, which are units of erasure processing, and replaces the physical blocks with a large number of erasure processing times with the physical blocks with a small number of erasing operations to average the number of write / erase processing times. It is implemented by leveling.

However, in conventional wear leveling, only the number of times of writing / erasing as an attribute of the physical block is referred to (for example, see Patent Document 1). In Patent Document 1, an attempt is made to average the number of times of writing by providing a block exchanging means for exchanging a physical block with the smallest number of times of writing and a physical block whose number of times of writing reaches a predetermined number of checks.

2. Description of the Related Art A semiconductor memory device that performs wear leveling generally uses a logical address / physical address conversion (hereinafter referred to as “logical / physical conversion”) table for converting a logical address into a physical address. By using this logical-physical conversion table, it is possible to distribute physical storage positions inside the semiconductor memory device even when there is a bias in the update position specified by the logical address by the host. In a semiconductor memory device that performs wear leveling, the logical / physical address correspondence changes each time rewriting is performed.

Unbalanced rewrite frequency as an external request from the host when the correspondence between logical address / physical address changes each time rewrite is performed, such as a solid state drive (SSD) or other wear leveling semiconductor memory device Is not an attribute of a physical block but an attribute of a logical address. However, in the conventional wear leveling, only the number of times of writing / erasing as the physical block attribute is referred to as described above, and the bias characteristic of the rewrite frequency as the logical address attribute is not considered at all.

Therefore, in conventional wear leveling, for example, a physical address of a physical block with a small number of write / erase times may be assigned to a logical address with a low rewrite frequency as an external request from the host. In this case, the physical block is unlikely to be updated in relation to the new logical-physical conversion after the allocation, and it is highly likely that the number of times of writing / erasing remains small. There was a problem that it was not done properly.

Exemplary embodiments of a memory controller and a semiconductor memory device will be explained below in detail with reference to the accompanying drawings. Note that the present invention is not limited to these embodiments.

(First embodiment)

FIG. 1 is a block diagram showing a configuration of a semiconductor memory device 2 as a memory system according to the first embodiment of the present invention. The semiconductor storage device 2 is a memory card that is detachably connected to a host 3 such as a personal computer or a digital camera, or an embedded type memory that is housed inside the host 3 and functions as an external storage device of the host 3. .

The semiconductor memory device 2 includes a nonvolatile semiconductor memory (hereinafter also simply referred to as “memory unit”) 40 and a memory controller 10. The memory unit 40 is, for example, a NAND

flash memory, and has a structure in which a large number of memory cells 41 as unit cells are arranged in a matrix at intersections of bit lines (not shown) and word lines 42. Data erasure in the memory unit 40 is performed in units of physical blocks including a plurality of unit cells, and the memory unit 40 includes a plurality of physical blocks. Writing / reading to / from the memory unit 40 is performed in units of physical pages. Since one physical block is composed of a plurality of physical pages, the size of the physical page is smaller than the size of the physical block.

The memory controller 10 includes a CPU 20, which is a control unit, a RAM 30, a host I / F (interface) 12, a ROM 13, and error correction (ECC) that performs encoding processing of stored data and decoding processing of stored data. : Error Correcting Code) circuit 14 and a NAND I / F (interface) 15, which are connected via a bus 11.

The memory controller 10 performs data transmission / reception between the host 3 and the RAM 30 via the host I / F 12 and performs data transmission / reception between the memory unit 40 and the RAM 30 via the NAND I / F 15 under the control of the CPU 20. . In addition to the normal control unit that performs data transmission / reception control between the host 3 and the RAM 30 and data transmission / reception control between the memory unit 40 and the RAM 30, the CPU 20 determines the number of times data is written to the memory unit 40 for each logical address. The number-of-writes counting unit 21 that counts, the number-of-erasing-counting unit 22 that counts the number of data erasures as the number of data rewrites for each physical block that is a unit of data erasure in the memory unit 40, A spare block selection processing unit 23 for selecting a certain spare block is provided. A spare block (free block) is a physical block that does not contain valid data and is not allocated for use.

When a data write request is generated from the host 3, the spare block selection processing unit 23 accesses the RAM 30 via the bus 11, and based on management information (for example, a spare block management table described later) held therein. The spare block to which data is written is selected.

FIG. 2 shows the concept of logical-physical conversion in the present embodiment, and the premise terms will be described. The logical address is an address used by the host, for example, LBA (Logical Block Addressing). LBA is a logical address in which a serial number from 0 is assigned to a sector (size: eg, 512 B). A sector is a unit smaller than the physical page described above. The physical address is an address indicating a storage position in the memory unit 40.

In this embodiment, as a logical address unit having the same partition size as the physical block size that is a data erasing unit, a logical block address that is a higher address of the logical address (for example, LBA) is used. It is associated with the physical block address which is the physical address of. That is, as shown in FIG. 2, logical-physical conversion in which one physical block address is associated with one logical block address is performed. However, as long as it is within the scope of the technical idea included in the gist of the present embodiment, the logical-physical conversion relationship is not necessarily limited.

The RAM 30 includes a logical address management table 31 and a spare block management table 32 in addition to an area functioning as a cache area interposed between the host 3 and the memory unit 40. The logical address management table 31 includes a logical address / physical address conversion table (logical / physical conversion table) 70 which is a first management table for converting a logical block address into a physical block address in the memory unit 40 as shown in FIG. It is out.

As described above, since the spare block selection processing unit 23 selects a spare block to which a corresponding logical address is not assigned in response to a data write instruction from the host 3, in this embodiment, data rewriting is performed. Each time, the relationship between logical and physical conversion will change. A write instruction to the memory unit 40 from the host 3 usually includes a head logical address and a data size.

Further, as shown in FIG. 3, the logical address management table 31 is a host for each physical block data erase count (physical block erase count) indicated by the physical block address registered in the logical-physical conversion table 70 and for each logical block address. 3 stores the number of times of writing (number of times of logical address writing) designated by 3.

Accordingly, the logical address management table 31 also includes a logical write count table which is a second management table for storing the write count from the host 3 for each logical block address. The logical write count table reflects the rewrite frequency bias characteristic as an attribute of the logical address.

As shown in FIG. 4, the spare block management table 32 has a physical block address of a spare block,

which is a physical block to which a corresponding logical address is not assigned, and a data erase count (physical block erase count) of the corresponding physical block. Is remembered. In this table, for example, as shown in FIG. 4, physical block addresses may be sorted according to the number of physical block erase times.

Therefore, the information on the number of times of data erasure of the physical block registered for each physical block address registered in the logical address management table 31 and the information on the number of times of data erasure of the physical block registered for each physical block address registered in the spare block management table 32 are obtained. In addition, a physical erase count table, which is a third management table for storing the data erase count for each physical block address, is configured. The physical erase count table reflects the frequency of erase (and write) for each physical block, that is, the physical fatigue level of the physical block.

The CPU 20 executes an operation as an address conversion unit (not shown) that performs conversion between a logical address and a physical address based on the logical-physical conversion table 70 included in the logical address management table 31 by firmware (FW). Control of the entire semiconductor memory device 2 in response to a command input from the host 3 is also executed in the CPU 20 by firmware.

The ROM 13 is a storage unit in which a boot program or the like of the semiconductor storage device 2 is stored, and the memory controller 10 controls the semiconductor storage device 2 in a part of the semiconductor memory unit 30 or a non-illustrated nonvolatile storage unit. Information is stored.

In the present embodiment, the CPU 20 included in the memory controller 10 has a logical address that the write count counter 21 counts for each logical block address when the host 3 receives an instruction to write data to an arbitrary logical address. Increment the number of writes. The result is stored in the logical address write count column of the logical address management table 31 in the RAM 30 via the bus 11.

Further, the CPU 20 increments the physical block erase count counted by the erase count counting unit 22 for each physical block address even when data is erased in each physical block of the memory unit 40. As a result, if the physical block has a corresponding logical address (that is, logical block address), the physical block address of the physical block in the logical address management table 31 in the RAM 30 is registered via the bus 11. Stored in the column of the physical block erase count in the row (that is, the row in which the logical block address is registered).

If the physical block from which data has been erased is a spare block to which a corresponding logical address is not assigned, the physical block address of the physical block in the spare block management table 32 in the RAM 30 is stored via the bus 11. It is stored in the physical block erase count column of the stored row.

Conventionally, wear leveling (hereinafter referred to as passive wear leveling) performed at the time of a write request from the host 3 is the number of past write (rewrite) requests to the logical address designated by the write request from the host 3. It was done without any consideration. That is, in the conventional passive wear leveling, for example, depending on only the number of erasures of the spare block, for example, a spare block with a small number of physical block erasures is selected in the spare block management table 32 of FIG. Block erasing and writing) and assigning a logical address designated by the host to the physical block.

However, in such passive wear leveling, there is a possibility that the purpose of wear leveling, that is, averaging the number of erasures, cannot be realized properly. For example, when a rewrite request happens to occur at a logical address that is less frequently rewritten as an external request from the host, there is a possibility that data is written by selecting a physical block with a small number of erases from the spare block. In this case, there is a high possibility that the physical block has a small number of erasures even in relation to the new logical-physical conversion after the logical address assignment, resulting in a bias in the number of physical block rewrites and averaging. There was a problem that it was not done properly.

Therefore, in the present embodiment, when the write frequency of the logical address for which data is instructed (rewritten) from the host is low, a spare block having a large erase count is selected for the logical address. The logical address is assigned and the data is written to the selected spare block. On the other hand, if the write frequency of the logical address that is instructed by the host is high, the spare address with a small number of erases is selected and assigned to the logical address, and the logical address is

assigned to the selected spare block. Write data. Thereby, it is possible to solve the problem that the above-described averaging is not appropriately performed.

Specifically, the spare block selection processing unit 23 (write control unit) responds to a write request specifying a logical block address with a relatively large number of writes stored in the logical address management table 31 with a spare block. A spare block with a relatively small number of erases stored in the management table 32 is selected, and a spare block with a relatively large number of erases is selected in response to a write request specifying a logical address with a relatively small number of writes. select.

For example, when a write request specifying the logical block address "n-1" in FIG. 3 is received from the host 3, it is determined that the logical address write count "150" in the logical address management table 31 is relatively small. Thus, the spare block of the physical block address "WWW" in FIG. 4 having a relatively large number of physical block erases in the spare block management table 32 is selected and data is written. Conversely, when a write request specifying the logical block address "n" in FIG. 3 is received, it is determined that the logical address write count "400" in the logical address management table 31 is relatively large, and the spare block The spare block with the physical block address "aaa" in FIG. 4 having a relatively small number of physical block erases in the management table 32 is selected and data is written.

More specifically, the logical block addresses sorted in the logical address write count size relationship in the logical address management table 31 in FIG. 3 and the spare block sorted in the physical block erase count size in the spare block management table 32 in FIG. The physical block addresses of the blocks may be associated with each other so that the magnitude relationship between them is reversed.

Further, as a method of associating the magnitude relationship in reverse order, it is possible to use a method that does not use the raw count value as described above. For example, a "write number relative value" column is further provided in the logical address management table 31 of FIG. 3, and a value (0 to 1) obtained by dividing each logical address write number by the maximum value of the logical address write number at that time. Stored for each logical block address. The spare block management table 32 is also provided with a column "relative value of erase count", and a value obtained by dividing each physical block erase count by the maximum value of the physical block erase count in the spare block at that time (0 to 0). 1) is held for each spare block.

Then, when the "write count relative value" of the logical block address in FIG. For example, according to the following (Equation 1), the "preferable erase count relative value" Y is calculated.

Y = 1-X (Formula 1)

The spare block selection processing unit 23 selects the spare block having the "erasing number relative value" closest to Y obtained here from the spare block management table 32 and writes the data. For example, for a write request specifying a logical block address with a relatively small number of writes, such as "relative number of write times" X = 0.2, the "relative value of preferred erasure number" Y = 0.8 is the most. A spare block having a relatively large number of erases is selected from the spare blocks.

The above (formula 1) is not limited to the above formula as long as X and Y are in a reverse relationship. Further, the correspondence is made more uniform by using a non-linear expression that reflects the deviation of the write frequency for each logical address from the host shown in FIG. 7 and the deviation of the block erase frequency for each spare block shown in FIG. You may devise as follows.

A specific procedure of passive wear leveling performed by the memory controller 10 of the present embodiment will be described below using the flowchart of FIG.

<Step S101> Data Write Instruction A data write request specifying, for example, the logical block address LA1 comes from the host 3 to the memory controller 10.

<Step S102> Selection of Spare Block The spare block selection processing unit 23 selects the spare block SB1 based on the above (Formula 1) for the designated logical block address LA1.

<Step S103> Block Erasing and Data Writing Block erasing the physical block at the physical block address in the memory unit 40 corresponding to the spare block SB1 selected in step S102, and transferring the data instructed to the physical block from the host 3 Write.

<Step S104> Incrementing the Physical Block Erase Count The erase count counting unit 22 in the CPU 20 counts the block erase in step S103 for each physical block address, and the physical block address in the spare block management table 32 of FIG. Increase the physical block erase count by 1.
<Step S105> Increment of Logical Address Write Count The write count counter 21 in the CPU 20 counts the write instruction to each logical block address from the host 3 in step S101 for each logical block address. The logical address write count of the logical block address LA1 in the address management table 31 is increased by one.
<Step S106> Physical Block Replacement Further, the physical block address and physical block erase count described in the spare block management table 32 of the spare block SB1 selected in step S102, and the physical block of the logical block address LA1 in the logical address management table 31 Exchange address and physical block erase count. At the time of replacement, for example, a predetermined storage area (not shown) in the RAM 30 may be used as a temporary storage area. As a result, the physical block entered in the logical address management table 31 and the physical block entered in the spare block management table 32 are replaced. After the replacement, the spare block management table 32 is sorted according to the size of the physical block erase count (data erase count).
<Step S107> Table Sort The spare block management table 32 is sorted according to the magnitude relationship of the physical block erase count (data erase count).

Steps S101 to S107 are repeated each time an instruction to write data to an arbitrary logical address is received from the host 3 to the memory controller 10.

As described above, in the present embodiment, when the write frequency of the logical address for which data is instructed from the host is low, the spare block with a large erase count is selected and the data is written and the host writes When the frequency of writing the designated logical address is high, a spare block with a small number of erasures is selected and the data is written.

As described above, the semiconductor memory device 2 according to the present embodiment refers to the number of rewrites, which is the attribute of the logical address, that is, the rewrite frequency of the logical address when the relationship of logical-physical conversion changes every time data is rewritten. The number of physical block rewrites is averaged.

In this embodiment, in passive wear leveling, a physical (spare) block having a large number of erases and a large degree of physical fatigue is assigned to a logical block address having a low write frequency, and conversely, a logical block having a high write frequency. By allocating physical (spare) blocks with a small degree of physical fatigue with a small number of erasures to addresses, it is considered possible to improve the accuracy of averaging the number of rewrites of physical blocks.

That is, the phenomenon of selecting a physical block with a small number of write / erase times as a spare block at a logical address with a low rewrite frequency, which has occurred in the conventional passive wear leveling, can be avoided. It is possible to achieve the purpose of wear leveling, which is a higher level of accuracy, and to extend the life of the memory system compared to the conventional one.

On the other hand, even when there is no data rewrite instruction from the host, the memory system voluntarily transfers the data written in the physical block with a small number of erases to the spare block with a large number of erases. There is a method of wear leveling (hereinafter referred to as active wear leveling) in which a physical block with a small number is a spare block.

In the above-described embodiment, it is considered that the required number of active wear leveling is reduced by increasing the accuracy of the number of times of physical block erase for passive wear leveling. Therefore, it is possible to reduce the number of physical block erasures that accompany voluntary data rewriting of the memory system in active wear leveling, and this can be expected to increase the life of the memory system.

(Second Embodiment)

The block diagram showing the configuration of the semiconductor memory device 2 as the memory system according to the second embodiment of the present invention is the same as FIG. Description of the same components is omitted. Also in the present embodiment, as shown in FIG. 2 as an example, logical-physical conversion in which one physical block address, that is, one physical block of the memory unit 40 is associated with one logical block address is performed. To be done. However, the logical-physical conversion relationship is not limited as long as it does not depart from the gist of the present embodiment.

Further, in response to a data write instruction from the host 3 to an arbitrary logical address, a spare block

(with no logical address assigned) is selected, data is written, and associated with the logical address. That is, the place where the relation of logical-physical conversion changes each time data is rewritten is the same as in the first embodiment.

In the present embodiment, the processing of the spare block selection processing unit 23 and the configurations of the logical address management table 31 and the spare block management table 32 provided in the RAM 30 are different from those of the first embodiment.

As in the first embodiment, the logical address management table 31 in this embodiment shown in FIG. 6 is a logical block that converts a logical block address into a physical block address that is a physical address in the memory unit 40 of the corresponding physical block. An address / physical address conversion table (logical / physical conversion table) 70 is included. Stores the data erase count (physical block erase count) of the physical block address described in the logical-physical conversion table 70, stores the write count (logical address write count) for each logical block address, and includes a logical write count table. It is also the same as in the first embodiment.

However, as shown in FIG. 6, the logical address management table 31 of this embodiment further stores a logical address group index for each logical block address. In general, there is a bias in the frequency of writing from the host 3 for each logical address as shown in FIG. Therefore, it is possible to classify all logical block addresses into one of a plurality of logical address groups based on the magnitude relationship of the number of logical address writes in FIG. For example, as shown in FIG. 8, it can be classified into four logical address groups each having indexes A to D.

Specifically, for example, the ratio (0 to 1) of each logical block address write count to the maximum logical address write count for all the logical block addresses shown in FIG. , 0.5, and 0.75, and the four logical address groups having indexes A, B, C, and D from the one with the larger ratio as shown in FIG. Can be classified.

The number of logical address groups is not limited to four as long as it is plural. Also, based on the magnitude relationship of the logical address write count, all the logical block addresses may be classified into a plurality of logical address groups having approximately the same number of logical block addresses, and based on the magnitude relationship of the logical address write count. The classification is not limited to the above classification method including the threshold interval. The index determined in this way is stored for each logical block address as an attribute of each logical block address in the logical address group column of the logical address management table 31 of FIG.

On the other hand, the spare block management table 32 in the present embodiment shown in FIG. 9 also corresponds to the physical block address of a spare block that is a physical block to which the corresponding logical address is not assigned, as in the first embodiment. The physical block erase count (data erase count) of the physical block is stored. The physical block addresses are sorted according to the number of physical block erase times.

However, as shown in FIG. 9, the spare block management table 32 of this embodiment further stores a spare block group index for each physical block address of the spare block.

In general, there is a bias in the frequency of block erases for each spare block as shown in FIG. Accordingly, it is possible to classify all spare blocks into one of the number of logical block groups, for example, the same number of spare block groups based on the magnitude relationship of the physical block erase count in FIG. For example, as shown in FIG. 11, it can be classified into four spare block groups each having indexes A to D.

Specifically, for example, the ratio (0 to 1) of each physical block erase count to the maximum value of the physical block erase count for all the spare blocks shown in FIG. 9 is set to three threshold values, for example, 0.25, 0 .5 and 0.75 are classified into four, and from the smaller ratio as shown in FIG. 11, the blocks are classified into spare block groups having indexes A, B, C, and D, respectively. it can.

The number of spare block groups may be the same as the number of logical address groups, and is not limited to four. Also, based on the magnitude relationship of the physical block erase count, all spare blocks may be classified into a plurality of spare block groups having approximately the same number of spare blocks. If the classification is based on the physical block erase count magnitude relationship For example, the classification method including the threshold interval is not limited. This index is stored in the spare block group column of the spare block management table 32 of FIG. 9 for each physical block address of the spare block.

In the present embodiment, when a logical block address for which a data write (rewrite) instruction has been issued from the host belongs to a logical address group with a low write frequency, the frequency of erasures for that logical block address. A spare block belonging to a larger spare block group is selected, the logical block address is assigned, and the data is written to a physical block in the memory unit 40 corresponding to the selected spare block.

On the other hand, if the logical block address for which a write instruction has been issued from the host belongs to a logical address group with a high write frequency, a spare block belonging to a spare block group with a low frequency of erasure is assigned to the logical block address. Is selected, the logical block address is assigned, and the data is written to the physical block in the memory unit 40 corresponding to the selected spare block.

In the example of FIGS. 8 and 11 described above, each logical address group has a spare block group of the same index, that is, the relationship between the logical address write count and the physical block erase count is reversed. Combine A's, B's, C's, and D's. From this combination, a spare block is selected for a logical address for which a write instruction has been issued.

A specific procedure of passive wear leveling performed by the memory controller 10 of the present embodiment will be described below using the flowchart of FIG.

<Step S201> Data Write Instruction A data write request specifying, for example, the logical block address LA1 comes from the host 3 to the memory controller 10.

<Step S202> Determination of Logical Address Group The spare block selection processing unit 23 accesses the logical address management table 31 in FIG. 6 and determines a logical address group based on the logical address group index of the logical block address LA1.

<Step S203> Spare Block Selection The spare block selection processing unit 23 accesses the spare block management table 32 of FIG. 9, and selects a spare block SB1 belonging to a spare block group having the same index as the logical address group determined in step S202. select.

As described above, a spare block group is associated with each logical address group so that the magnitude relationship between the number of logical address writes and the magnitude relationship between the physical block erase counts are reversed. Accordingly, any spare block belonging to the spare block group corresponding to the logical address group determined in step S202 is selected, for example, one spare block belonging to the spare block group is selected at random. It doesn't matter.

However, the same idea is applied according to the relative magnitude of the logical address write count of the logical block address LA1 in the logical address group determined in step S202, reflecting the idea of the first embodiment. If the spare block SB1 is selected from the index spare block group, the number of physical block erasures can be averaged more effectively. That is, when the logical address write count of the logical address LA1 is relatively large in the logical address group determined in step S202, the physical block erase count is relatively small among the spare blocks belonging to the corresponding spare block group. If a spare block is selected and the logical address write count is relatively large, a spare block with a relatively small physical block erase count may be selected.

Such a mechanism can be realized, for example, by using (Equation 1) described in the first embodiment. In this case, the logical address group column of the logical address management table 31 in FIG. 6 is added to the logical address group index, and the logical address write count of the row is the maximum write count in the logical address group of the index. Further, a “write number relative value” for each logical address group, which is a value obtained by dividing the number, is further stored. For example, “A: 0.3”. Similarly, in the spare block group column of the spare block management table 32 of FIG. 9, the value obtained by dividing the physical block erase count of the row by the maximum erase count within each spare block group is “Erase” for each spare block group. The relative number of times ”is additionally stored as“ D: 0.4 ”. Then, after selecting a spare block group having the same index as the logical address group determined in step S202, the spare block group SB1 closest to the “preferred erase count relative value” is selected according to (Equation 1) and the like.

<Step S204> Block Erase and Data Write Block erase the physical block at the physical block address in the memory unit 40 corresponding to the spare block SB1 selected in step S203, and store the data instructed to the physical block from the host 3 Write.

<Step S205> Incrementing the Physical Block Erase Count The erase count counting unit 22 in the CPU 20 counts the block erase in step S204 for each physical block address, and the physical block address

in the spare block management table 32 of FIG. Increase the physical block erase count by 1.
<Step S206> Increment of Logical Address Write Count The write count counter 21 in the CPU 20 counts the write instruction to each logical block address from the host 3 in step S201 for each logical block address. The logical address write count of the logical block address LA1 in the address management table 31 is increased by one.
<Step S207> Physical Block Replacement Further, the physical block address and physical block erase count described in the spare block management table 32 of the spare block SB1 selected in step S203, and the physical block of the logical block address LA1 in the logical address management table 31 Exchange address and physical block erase count. At the time of replacement, for example, a predetermined storage area (not shown) in the RAM 30 may be used as a temporary storage area. As a result, the physical block entered in the logical address management table 31 and the physical block entered in the spare block management table 32 are replaced. After the replacement, the spare block management table 32 is sorted according to the size of the physical block erase count (data erase count).
<Step S208> Update of Logical Address Group Index Since the number of logical address writes in the logical address management table 31 is incremented in Step S206, it may be necessary to update the logical address group index. Specifically, the ratio (0 to 1) between the maximum number of logical address writes and the number of logical address writes for all logical block addresses is calculated, for example, 0.25, 0.5, 0.75 The four indexes A, B, C, and D are classified again with three threshold values. If there is a logical block address with an index different from the previous one, the logical address group column of the logical block address in FIG. 6 is updated to that index.
<Step S209> Updating Spare Block Group Index In step S207, the physical block address and physical block erase count described in the spare block management table 32 are rewritten and sorted according to the size of the physical block erase count. The block group index needs to be updated. Specifically, a ratio (0 to 1) between the maximum number of physical block erase times of all spare blocks and the number of physical block erase times is calculated, for example, 3 of 0.25, 0.5, and 0.75. The four indexes A, B, C, and D are classified again with one threshold. Then, the spare block group column of all physical block addresses in the spare block management table 34 is updated to the index.

Steps S201 to S209 are repeated each time an instruction to write data to an arbitrary logical address is received from the host 3 to the memory controller 10.

As described above, in this embodiment, in the memory system in which the relationship of logical-physical conversion changes every time data is rewritten, the frequency of writing is counted as an attribute of the logical block address, and the relative magnitude relationship of the frequency is calculated. The logical block addresses are grouped, and the physical blocks constituting the spare block are also grouped according to the relative size of the erase count.

When rewriting data, the logical address group with a low write frequency is combined with a spare block group with a high erase frequency, and a logical address group with a high write frequency is combined with a spare block group with a low erase frequency. A spare block is selected for the logical address requested to be rewritten in the combination.

As a result, the phenomenon of selecting a physical block with a small number of write / erase times as a spare block at a logical address with a low rewrite frequency, which has occurred in the conventional passive wear leveling, can be avoided. The purpose of wear leveling, which is averaging, can be realized with higher accuracy, and the life of the memory system can be extended compared to the conventional case. Further, as in the first embodiment, it is possible to expect a longer life of the memory system by reducing the required number of active wear leveling.

In the present embodiment, the number of spare block groups has been described as being the same as the number of logical address groups. However, the relationship between the number of times of writing and the size of the number of times of erasing the spare block are reversed as logical address attributes. The number is not necessarily the same as long as the purpose of group combination can be realized.

Further, a simpler implementation example of the present embodiment may be as follows. That is, the ratio of the number of writes for each logical block address to the maximum value of the number of logical address writes is classified into A, B, and C logical address groups from the larger, for example, in the magnitude relationship between the threshold values 0.1 and 0.9. The ratio of the number of physical block erases to the maximum number of physical block erases for all spare blocks, for example, the spare block

groups of A, B, and C from the smaller one in the magnitude relationship between the threshold values 0.1 and 0.9. And groups with the same index may be associated with each other.
(Third embodiment)
The block diagram showing the configuration of the memory system according to the third embodiment of the present invention is the same as that in FIG. 1, but this embodiment relates to active wear leveling. The RAM 30 includes a logical address management table 31 shown in FIG. 3 and a spare block management table 32 shown in FIG. The description of the operations common to the first embodiment of each component is omitted.

Also in the present embodiment, as shown in FIG. 2 as an example, logical-physical conversion in which one physical block address, that is, one physical block of the memory unit 40 is associated with one logical block address is performed. To be done. However, the logical-physical conversion relationship is not limited as long as it does not depart from the gist of the present embodiment. Further, in response to a data write instruction from the host 3 to an arbitrary logical address, a spare block (having no logical address) is selected, data is written, and associated with the logical address. That is, the place where the relation of logical-physical conversion changes each time data is rewritten is the same as in the first and second embodiments.

In the present embodiment, in active wear leveling, which is the physical block rewrite frequency averaging control when there is no rewrite instruction from the host 3, the physical block rewrite frequency is small and the logical block address write frequency is small. Replace the physical block with a spare block with high fatigue level. However, active wear leveling is not executed for a physical block with a large number of writes to a logical block address even if the number of rewrites of the physical block is small. As a result, the accuracy of averaging the number of rewrites of the physical block can be improved as compared with the conventional case.

That is, in the active wear leveling, the rewrite frequency of the logical block address of the physical block in which valid data is written is considered. Specifically, the memory controller 10 performs active wear leveling in consideration of the number of logical address writes in the logical address management table 31 of FIG.

A specific procedure of active wear leveling performed by the memory controller 10 of the present embodiment will be described below using the flowchart of FIG.
<Step S301> Start Condition Determination When there is no instruction to write data to the logical address from the host 3, the CPU 20 performs the physical block erase count of the logical address management table 31 in the RAM 30 and the physical block erase count of the spare block management table 32. Check whether or not the physical block erase count imbalance exceeds a predetermined condition, such as when the difference between the maximum and minimum erase counts in all physical blocks exceeds a predetermined threshold. Judge as the start condition. Moreover, it is good also as starting periodically or determining start conditions for every fixed time. If the start condition is satisfied, the process proceeds to step S302. If not, step S301 is repeated.
<Step S302> Determination of threshold value for logical address write count The CPU 20 accesses the logical address management table 31 in the RAM 30, and satisfies the condition that the logical address write count in the logical address management table 31 is smaller than the first threshold value. Determine if there is any. The first threshold only needs to be held in the RAM 30 or the like. As the value of the first threshold, for example, the maximum value of the number of logical address writes for all the logical block addresses shown in FIG. You may enable it to select the thing with a small logical address write count.

However, since the purpose of the first threshold is to select a logical block address that is predicted to have few future write requests from the host 3, the present invention is not limited to the above example and coefficient, and is not a relative value but a specific value, for example. It may be a numerical value of the number of times of writing. If there is no logical block address satisfying the condition, the process returns to step S301. If there is a logical block address satisfying the condition, the process proceeds to step S303.
<Step S303> Threshold Determination of Physical Block Erase Count The CPU 20 is smaller than the second threshold value among the physical block erase counts of the logical block address row satisfying the condition of step S302 in the logical address management table 31 in the RAM 30. It is determined whether there is one satisfying the condition. The second threshold only needs to be held in the RAM 30 or the like. As the value, for example, the physical block in which valid data is written by setting the maximum

value of the physical block erase count for all the logical block addresses in FIG. 3 to a coefficient such as 0.1 to 0.3. A block that can select a block having a smaller number of physical block erases may be selected.

Since the purpose is to select a physical block having a lower physical fatigue level from the physical blocks corresponding to the logical block address satisfying the condition of step S302, the second threshold is not limited to the above example or coefficient. A specific physical block erase count may be used. If there is no physical block that satisfies the condition, the process returns to step S301. If there is a physical block that satisfies the condition, the physical block is a vacant physical block, and the corresponding logical block address is a vacant logical address. Then, the process proceeds to step S304.

<Step S304> Fatigue Spare Block Block Erasure and Data Write The spare block selection processing unit 23 determines that the physical block erase count in the spare block shown in the spare block management table 32 of FIG. Select the fatigue spare block that is larger or the maximum value. Then, after the block of the fatigue spare block is erased, the data written in the vacant physical block is written.

The third threshold is basically a condition that it is larger than the second threshold, but a constant may be selected as long as it is based on the physical absolute fatigue level of the physical block. However, a relative value may be selected, for example, a value obtained by multiplying the maximum value of the physical block erase count for all spare blocks shown in FIG. 4 by a coefficient such as 0.9. As long as a spare block having a high degree of fatigue can be selected, the third threshold value is not limited to the above example or coefficient.

Further, if there is one off-the-shelf physical block that satisfies the condition of step S303, a fatigue spare block having the maximum value in the spare block management table 32 for the physical block erase count may be selected. If there are multiple free physical blocks, the smaller the number of corresponding logical address writes, the more effective it is to select the reverse order so that data is written to a fatigue spare block with a larger physical block erase count. Wear leveling.

Further, if the spare block having the physical block erase count larger than the second threshold is selected as the fatigue spare block and the data written in the off-use physical block is written, the active wear leveling is realized. The fatigue spare block may not be selected in consideration of the threshold value. Next, the process proceeds to step S305.

<Step S305> Incrementing the number of erasures of the fatigue spare block The erasure number counting unit 22 in the CPU 20 counts the block erasure in step S304 for each physical block address, and the physical block in the spare block management table 32 of FIG. Increase the physical block erase count of the address by one.

<Step S306> Exchange of Leisure Physical Block and Fatigue Spare Block The spare block selection processing unit 23 includes the physical block address and the number of physical block erasures of the fatigue spare block in which the data written in the vacant physical block in step S304 is written, The physical block address and physical block erasure count of the relevant physical block in the logical address management table 31 are exchanged. At the time of replacement, for example, a predetermined storage area (not shown) in the RAM 30 may be used as a temporary storage area.

As a result, the physical block address of the unoccupied physical block is erased from the logical address management table 31 and newly entered in the spare block management table 32, and the unoccupied physical block becomes a new spare block. After the replacement, the spare block management table 32 is sorted according to the size of the physical block erase count (data erase count).

When a data write instruction is not received from the host 3 to the memory controller 10, steps 301 to 306 are performed.

As described above, in the present embodiment, in active wear leveling when there is no rewrite instruction from the host, the physical block of the physical block is limited to the physical block of the logical block address with a low write request frequency from the host. When the number of times of erasure is small, replace with a spare block with high fatigue. As a result, the active replacement of a physical block with a logical block address, which frequently requires write requests from the host, with a spare block having a high degree of fatigue based only on the condition that the number of erasures of the physical block is low. Wear leveling can be avoided.

In other words, a logical address with a high data rewrite frequency other than the above-described busy logical address is not subjected to active wear leveling even if the number of rewrites of the physical block

corresponding to the address is small. This is because even if such physical block is not intended for active wear leveling, write requests from the host should come frequently, and at that time, the spare block and physical block are forcibly exchanged. It is.

Therefore, according to the present embodiment, it is possible to reduce the required number of active wear leveling while maintaining the accuracy of the physical block rewrite frequency averaging by the conventional active wear leveling. Therefore, it is possible to reduce the number of physical block erasures associated with voluntary data rewriting of active wear leveling, so that the life of the memory system can be expected to be extended.

In addition, as described in step S304 above, when there are a plurality of vacant physical blocks, data is written to a fatigue spare block having a smaller physical block erase count as the corresponding logical address write count is smaller. By selecting in reverse order, it is possible to improve the accuracy of the physical block rewrite frequency averaging for the same reason as described in the first and second embodiments. That is, according to the present embodiment, it is possible to increase the accuracy of averaging the number of physical block rewrites by active wear leveling, and it is possible to further extend the life of the memory system.

Furthermore, since this embodiment improves active wear leveling, it can be used in combination with the improved passive wear leveling described in the first and second embodiments, and further effects are expected by the combined use. it can. That is, when there is a rewrite instruction from the host, the memory controller 10 may operate according to the flowchart of FIG. 5 or FIG. 12, and when there is no rewrite instruction from the host, according to the flowchart of FIG. As a result, it is expected that the accuracy of erasing the physical block is more accurately averaged than in the first to third embodiments, and the life of the memory system can be further extended.

In any of the above-described embodiments, the number of writes to a logical address in a storage device in which a plurality of storage elements having deterioration characteristics depending on the number of physical rewrites are mounted and the correspondence of logical-physical conversion changes with each rewrite Is based on the idea that this will be a predicted value of the frequency of writing to the logical address in the future. In other words, spare blocks with a high number of erasures and a high degree of fatigue are assigned to those with a low number of writes to logical addresses, and storage is performed so that the physical rewrite times of storage elements can be averaged more accurately in the future. The device has a longer life.

Therefore, the present invention is not limited to the above-described embodiment, and various modifications can be made without departing from the scope of the invention in the implementation stage. Further, the above embodiments include inventions at various stages, and various inventions can be extracted by appropriately combining a plurality of disclosed constituent requirements. For example, even if some constituent requirements are deleted from all the constituent requirements shown in the embodiment, the problem described in the column of the problem to be solved by the invention can be solved, and is described in the column of the effect of the invention. When an effect is obtained, a configuration from which this configuration requirement is deleted can be extracted as an invention. Furthermore, the constituent elements over different embodiments may be appropriately combined.

REFERENCE-SIGNS-LIST

2 Semiconductor memory device, 3 host, 10 memory controller, 11 bus, 12 host I / F (interface), 13 ROM, 14 error correction section, 15 NAND I / F, 20 CPU, 21 write count coefficient section, 22 erase count Counting unit, 23 Spare block selection processing unit, 30 RAM, 31 Logical address management table, 32 Spare block management table, 40 Non-volatile semiconductor memory unit, 41 Memory cell, 42 Word line, 50, 51 Logical block address, 60, 61 Physical block address, 70 logical-physical conversion table, S101 to S107, S201 to S209, S301 to S306.

Claims

A memory controller that performs control for rewriting data in a nonvolatile semiconductor memory having a plurality of blocks as data erasing units,

A first management table for storing a correspondence between a logical block address as a logical address

of the block unit designated by a host and a physical block address indicating a storage position of the block in the nonvolatile semiconductor memory;
A second management table for storing the number of times data is written from the host to the nonvolatile semiconductor memory for each logical block address;
A third management table for storing the number of times data is erased for each physical block address;
A write control unit that selects a spare block that is the block that does not have a corresponding logical address in response to a write request designating a logical address from the host, and writes data to the spare block;
A memory that averages the number of data erasures among the plurality of blocks based on the number of times of writing data stored for each logical block address and the number of data erasures stored for each physical block address controller.
The write control unit
In response to a write request designating a logical block address that has a relatively large number of writes stored in the second management table, the number of erases stored in the third management table is within the spare block. Select relatively few spare blocks,
In response to a write request specifying a logical block address with a relatively small number of writes stored in the second management table, the number of erasures stored in the third management table is within the spare block. The memory controller according to claim 1, wherein a relatively large number of spare blocks are selected.
Based on the second management table and the third management table, the logical block address and the spare block are associated with each other so that the magnitude relation of the number of write times and the magnitude relation of the erase number are reversed.
The memory controller according to claim 1, wherein the write control unit selects a spare block based on the association with a write request designating a logical address from the host.
The logical block addresses are classified into a plurality of logical address groups based on the magnitude relationship of the number of writes stored in the second management table, and based on the magnitude relationship of the number of erases stored in the third management table. The spare block is classified into a plurality of spare block groups, and the logical address group and the spare block group are associated with each other so that the magnitude relationship between the number of write times and the magnitude relationship between the erase times are reversed.
In response to a write request designating a logical address from the host, the write control unit is a spare block classified into the spare block group associated with the logical address group into which the logical address is classified. The memory controller according to claim 1, wherein the memory controller is selected.
Classification of the logical block address based on the magnitude relationship of the number of writing is performed by comparing a ratio of the number of writing to the maximum value of the number of writing stored in the second management table with a threshold value,
The spare block classification based on the size relationship of the number of erases compares the ratio of the number of erases to the maximum value of the number of erases stored in the third management table in the spare block with a threshold value. The memory controller according to claim 4, which is performed by:
The write control unit, when there is no write request from the host, a physical block corresponding in the first management table to a logical block address for which the number of writes stored in the second management table is smaller than a first threshold When the number of times of erasure stored in the third management table of the address satisfies a condition smaller than the second threshold value, the vacant physical block indicated by the physical block address corresponding to the vacant logical address that is the logical block address satisfying the condition Write the written data by selecting a fatigue spare block having a number of erases greater than a third threshold stored in the third management table from the spare block,

The physical block address of the fatigue spare block is written in the first management table so as to correspond to the busy logical address and the physical block address of the busy physical block is erased to make the busy physical block the spare block. The memory controller according to claim 1.
The memory controller according to claim 6, wherein the first threshold value is a value obtained by multiplying a maximum value of the number of times of writing stored in the second management table by a predetermined constant.

8. A semiconductor memory device comprising: the memory controller according to claim 1; and the nonvolatile semiconductor memory.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2011-203916
(P2011-203916A)
(43)公開日　平成23年10月13日(2011.10.13)

(51)Int.Cl.		F I		テーマコード（参考）
G06F 12/16	(2006.01)	G06F 12/16	310A	5B018

審査請求 未請求　請求項の数 8　OL　（全 21 頁）

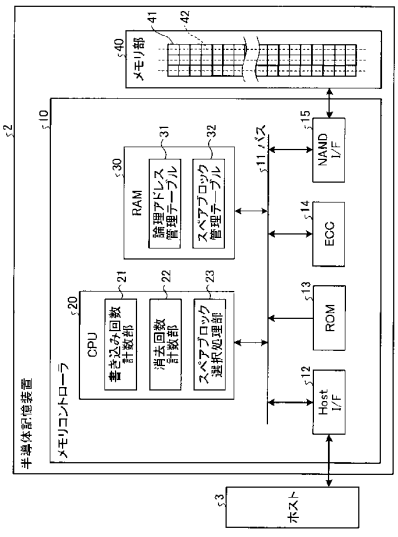
(21)出願番号　特願2010-69418(P2010-69418)
(22)出願日　平成22年3月25日(2010.3.25)

(71)出願人　000003078
株式会社東芝
東京都港区芝浦一丁目１番１号
(74)代理人　100089118
弁理士　酒井　宏明
(72)発明者　助川　博
東京都港区芝浦一丁目１番１号　株式会社東芝内

Fターム(参考)　5B018 GA04　HA23　NA06　QA15

(54)【発明の名称】メモリコントローラ、および半導体記憶装置

(57)【要約】
【課題】ブロック消去回数平均化をより高精度に行うメモリコントローラを提供する。
【解決手段】ブロックを複数有する不揮発性メモリ４０の書換え制御を行うメモリコントローラ１０であって、ホスト３が指定するブロック単位の論理アドレスである論理ブロックアドレスとブロックのメモリ４０での記憶位置を示す物理ブロックアドレスとの対応を記憶する第１管理テーブル３１と、ホスト３からメモリ４０への書き込み回数を論理ブロックアドレス毎に記憶する第２管理テーブル３１と、物理ブロックアドレス毎の消去回数を記憶する第３管理テーブル３１、３２と、ホスト３からの論理アドレスを指定した書き込み要求に対して、対応する論理アドレスをもたないスペアブロックを選択して書き込む制御部２３とを備え、論理ブロックアドレス毎に記憶される書き込み回数と物理ブロックアドレス毎に記憶される消去回数に基づいて、消去回数を複数ブロック間で平均化する。
【選択図】図１



【特許請求の範囲】
【請求項１】
データ消去の単位であるブロックを複数個有する不揮発性半導体メモリのデータを書き換えるための制御を行うメモリコントローラであって、
ホストが指定する前記ブロック単位の論理アドレスとしての論理ブロックアドレスと、前記ブロックの前記不揮発性半導体メモリでの記憶位置を示す物理ブロックアドレスとの対応を記憶する第１の管理テーブルと、
前記ホストから前記不揮発性半導体メモリへのデータの書き込み回数を前記論理ブロックアドレス毎に記憶する第２の管理テーブルと、
前記物理ブロックアドレス毎のデータ消去回数を記憶する第３の管理テーブルと、
前記ホストからの論理アドレスを指定した書き込み要求に対して、対応する論理アドレスをもたない前記ブロックであるスペアブロックを選択し、当該スペアブロックにデータを書き込む書き込み制御部とを備え、
前記論理ブロックアドレス毎に記憶されるデータの書き込み回数と前記物理ブロックアドレス毎に記憶されるデータ消去回数に基づいて、データ消去回数を複数の前記ブロックの間で平均化することを特徴とするメモリコントローラ。
【請求項２】
前記書き込み制御部は、
前記第２の管理テーブルに記憶されている書き込み回数が相対的に多い論理ブロックアドレスを指定した書き込み要求に対しては前記第３の管理テーブルに記憶されている消去回数が前記スペアブロックの中で相対的に少ないスペアブロックを選択し、
前記第２の管理テーブルに記憶されている書き込み回数が相対的に少ない論理ブロックアドレスを指定した書き込み要求に対しては前記第３の管理テーブルに記憶されている消去回数が前記スペアブロックの中で相対的に多いスペアブロックを選択する
ことを特徴とする請求項１に記載のメモリコントローラ。
【請求項３】
前記第２の管理テーブルおよび前記第３の管理テーブルに基づき、前記書き込み回数の大小関係と前記消去回数の大小関係が逆になるように前記論理ブロックアドレスと前記スペアブロックを対応づけ、
前記書き込み制御部は、前記ホストからの論理アドレスを指定した書き込み要求に対して、前記対応づけに基づいてスペアブロックを選択する
ことを特徴とする請求項１に記載のメモリコントローラ。
【請求項４】
前記第２の管理テーブルに記憶された書き込み回数の大小関係に基づいて前記論理ブロックアドレスを複数の論理アドレスグループに分類し、前記第３の管理テーブルに記憶された消去回数の大小関係に基づいて前記スペアブロックを複数のスペアブロックグループに分類し、前記書き込み回数の大小関係と前記消去回数の大小関係が逆になるように前記論理アドレスグループと前記スペアブロックグループを対応づけ、
前記書き込み制御部は、前記ホストからの論理アドレスを指定した書き込み要求に対して、当該論理アドレスが分類されている前記論理アドレスグループに対応づけられている前記スペアブロックグループに分類されているスペアブロックを選択する
ことを特徴とする請求項１に記載のメモリコントローラ。
【請求項５】
前記書き込み回数の大小関係に基づく前記論理ブロックアドレスの分類は、前記第２の管理テーブルに記憶されている前記書き込み回数の最大値に対する前記書き込み回数の比を閾値と比較することにより行い、
前記消去回数の大小関係に基づく前記スペアブロックの分類は、前記第３の管理テーブルに記憶されている前記消去回数の前記スペアブロックの中での最大値に対する前記消去回数の比を閾値と比較することにより行う
ことを特徴とする請求項４に記載のメモリコントローラ。

Samsung Ex. 1002, Page 253 of 622

【請求項６】
前記書き込み制御部は、前記ホストからの書き込み要求が無いときに、前記第２の管理テーブルに記憶された書き込み回数が第１閾値より小さい論理ブロックアドレスに前記第１の管理テーブルで対応する物理ブロックアドレスの前記第３の管理テーブルに記憶された消去回数が第２閾値より小さい条件を満たす場合に、当該条件を満たす論理ブロックアドレスである有閑論理アドレスに対応する物理ブロックアドレスが示す有閑物理ブロックに書き込まれているデータを、前記スペアブロックの中から前記第３の管理テーブルに記憶された消去回数が第３閾値よりも大きい疲労スペアブロックを選択して書き込み、
前記疲労スペアブロックの物理ブロックアドレスを前記有閑論理アドレスに対応するように前記第１の管理テーブルに書き込んで前記有閑物理ブロックの物理ブロックアドレスを消去することにより当該有閑物理ブロックを前記スペアブロックにする
ことを特徴とする請求項１に記載のメモリコントローラ。
【請求項７】
前記第１閾値は、前記第２の管理テーブルに記憶された書き込み回数の最大値に所定の定数を乗じた値である
ことを特徴とする請求項６に記載のメモリコントローラ。
【請求項８】
請求項１から請求項７のいずれか１項に記載のメモリコントローラと、前記不揮発性半導体メモリとを具備することを特徴とする半導体記憶装置。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、メモリコントローラ、および半導体記憶装置に関する。
【背景技術】
【０００２】
ＮＡＮＤ型フラッシュメモリは、ブロックと呼ばれる複数のメモリセルに記憶されたデータを一括して消去処理を行うことにより構造を簡略化し、安価でかつ大容量を実現している。そしてさらに、ＮＡＮＤ型フラッシュメモリは、可動部がなく低消費電力であることから、ＨＤＤに代わる記憶装置、携帯電話または携帯音楽プレーヤ等のホストの記憶装置として広く利用されている。しかし、ＮＡＮＤ型フラッシュメモリには書き込み／消去回数に制限があることが知られている。
【先行技術文献】
【特許文献】
【０００３】
【特許文献１】特開２００４－３２６５２３号公報
【発明の概要】
【発明が解決しようとする課題】
【０００４】
本発明は、ブロックの消去回数の平均化を高い精度で行うことができるメモリコントローラ、および半導体記憶装置を提供する。
【課題を解決するための手段】
【０００５】
本願発明の一態様によれば、データ消去の単位であるブロックを複数個有する不揮発性半導体メモリのデータを書き換えるための制御を行うメモリコントローラであって、ホストが指定する前記ブロック単位の論理アドレスとしての論理ブロックアドレスと、前記ブロックの前記不揮発性半導体メモリでの記憶位置を示す物理ブロックアドレスとの対応を記憶する第１の管理テーブルと、前記ホストから前記不揮発性半導体メモリへのデータの書き込み回数を前記論理ブロックアドレス毎に記憶する第２の管理テーブルと、前記物理ブロックアドレス毎のデータ消去回数を記憶する第３の管理テーブルと、前記ホストからの論理アドレスを指定した書き込み要求に対して、対応する論理アドレスをもたない前記

ブロックであるスペアブロックを選択し、当該スペアブロックにデータを書き込む書き込み制御部とを備え、前記論理ブロックアドレス毎に記憶されるデータの書き込み回数と前記物理ブロックアドレス毎に記憶されるデータ消去回数に基づいて、データ消去回数を複数の前記ブロックの間で平均化することを特徴とするメモリコントローラが提供される。

【発明の効果】

【０００６】

本発明によれば、ブロックの消去回数の平均化を高い精度で行うことができるメモリコントローラ、および半導体記憶装置を提供できる。

【図面の簡単な説明】

【０００７】

【図１】図１は、第１乃至３の実施の形態のメモリシステムの構成を示すブロック図である。

【図２】図２は、第１乃至３の実施の形態の半導体記憶装置における物理アドレスと論理アドレスとの関係を示す図である。

【図３】図３は、第１及び第３の実施の形態の論理アドレス管理テーブルを示す図である。

【図４】図４は、第１及び第３の実施の形態のスペアブロック管理テーブルを示す図である。

【図５】図５は、第１の実施の形態のメモリコントローラが行う受動的ウェアレベリングの具体的な手順を示すフローチャートである。

【図６】図６は、第２の実施の形態の論理アドレス管理テーブルを示す図である。

【図７】図７は、論理アドレス毎の書き込み回数の分布を示す図である。

【図８】図８は、図７を書き込み回数の大小関係でグループ分けした図である。

【図９】図９は、第２の実施の形態のスペアブロック管理テーブルを示す図である。

【図１０】図１０は、スペアブロック毎のブロック消去の回数の分布を示す図である。

【図１１】図１１は、図１０をブロック消去の回数の大小関係でグループ分けした図である。

【図１２】図１２は、第２の実施の形態のメモリコントローラが行う受動的ウェアレベリングの具体的な手順を示すフローチャートである。

【図１３】図１３は、第３の実施の形態のメモリコントローラが行う能動的ウェアレベリングの具体的な手順を示すフローチャートである。

【発明を実施するための形態】

【０００８】

ＮＡＮＤ型フラッシュメモリのメモリセルに対する書き込み／消去回数の制限は、基板に対してゲートに高電圧がかけられフローティングゲートに電子が注入されること（書き込み）、及び、ゲートに対してソースに高電圧がかけられフローティングゲートから電子が引き抜かれること（消去）に起因する。即ち、書き込み／消去処理が特定のメモリセルで何回も実行されるとフローティングゲート周りの酸化膜が劣化し、データが破壊されてしまうことがある。

【０００９】

このような書き込み／消去処理が特定のメモリセルに集中することを回避するために、メモリコントローラが書き込み／消去処理回数の平均化をはかっている。これは、メモリコントローラが、消去処理の単位である物理ブロックの消去回数をカウントし消去処理回数の多い物理ブロックと少ない物理ブロックとを入れ替えて、書き込み／消去処理回数の平均化をはかる、いわゆるウェアレベリングによって実施されている。

【００１０】

しかしながら、従来のウェアレベリングにおいては、物理ブロックの属性としての書き込み／消去回数しか参照していなかった（例えば、特許文献１参照。）。特許文献１では、最も書き込み回数が少ない物理ブロックと書き込み回数が所定のチェック回数に達した物理ブロックとを交換するブロック交換手段を備えることにより書き込み回数の平均化を

試みている。

【００１１】

ウェアレベリングを行う半導体記憶装置では一般に、論理アドレスを物理アドレスに変換するための論理アドレス／物理アドレス変換（以下、「論物変換」という。）テーブルが用いられている。この論物変換テーブルを用いることによって、ホストが論理アドレスで指定する更新位置に偏りがあった場合でも、半導体記憶装置内部における物理的な記憶位置を分散させることが可能となる。ウェアレベリングを行う半導体記憶装置では、このように書き換えの都度、論理アドレス／物理アドレスの対応関係が変化する。

【００１２】

ＳＳＤ（Solid State Drive）等のウェアレベリングが行われている半導体記憶装置のように、書き換えの都度、論理アドレス／物理アドレスの対応関係が変わる場合において、ホストからの外部要求としての書き換え頻度の偏りは、物理ブロックの属性ではなく、本来論理アドレスの属性である。しかしながら従来のウェアレベリングにおいては、上記したように物理ブロックの属性としての書き込み／消去回数しか参照されておらず、論理アドレスの属性としての書き換え頻度の偏り特性が全く考慮されていなかった。

【００１３】

従って、従来のウェアレベリングにおいては、例えば、ホストからの外部要求としての書き換え頻度が少ない論理アドレスに対して、書き込み／消去回数が少ない物理ブロックの物理アドレスを割り当てる可能性があった。この場合、その物理ブロックは割り当て後の新しい論物変換の関係において更新される可能性が低く、書き込み／消去回数が少ないままである可能性が高く、結果として物理ブロックの書き換え回数の平均化が適切に行われないという問題があった。

【００１４】

以下に添付図面を参照して、本発明の実施の形態にかかるメモリコントローラ、半導体記憶装置を詳細に説明する。なお、これらの実施の形態により本発明が限定されるものではない。

【００１５】

（第１の実施の形態）

図１は、本発明の第１の実施の形態にかかるメモリシステムとしての半導体記憶装置２の構成を示すブロック図である。半導体記憶装置２は、パソコンまたはデジタルカメラ等のホスト３と着脱可能に接続されるメモリカード、または、ホスト３の内部に収納され、ホスト３の外部記憶装置として機能するエンベデッドタイプのメモリ等である。

【００１６】

半導体記憶装置２は、不揮発性半導体メモリ（以下、単に「メモリ部」ともいう。）４０と、メモリコントローラ１０とを有する。メモリ部４０は例えばＮＡＮＤ型フラッシュメモリであり、単位セルである多数のメモリセル４１が、ビット線（図示せず）およびワード線４２の交点にマトリクス状に配置された構造を有している。メモリ部４０におけるデータ消去は、単位セル複数個からなる物理ブロック単位で行われ、メモリ部４０は複数の物理ブロックを備えている。メモリ部４０に対する書き込み／読み出しは、物理ページ単位に行われる。１つの物理ブロックは複数の物理ページから構成されるので、物理ページのサイズは物理ブロックのサイズより小さい。

【００１７】

メモリコントローラ１０は、制御部であるＣＰＵ２０と、ＲＡＭ３０と、ホスト　Ｉ／Ｆ（インターフェイス）１２と、ＲＯＭ１３と、記憶するデータの符号化処理および記憶されたデータの復号化処理を行う誤り訂正（ＥＣＣ：Error Correcting Code）回路１４と、ＮＡＮＤ　Ｉ／Ｆ（インターフェイス）１５とを備え、これらがバス１１を介して接続されている。

【００１８】

メモリコントローラ１０は、ＣＰＵ２０による制御に従って、ホストＩ／Ｆ１２を介してホスト３とＲＡＭ３０との間でデータ送受信を行い、ＮＡＮＤ　Ｉ／Ｆ１５を介してメ

モリ部４０とＲＡＭ３０との間でデータ送受信を行う。ＣＰＵ２０は、ホスト３とＲＡＭ３０との間のデータ送受信制御、メモリ部４０とＲＡＭ３０との間のデータ送受信制御を行う通常制御部の他に、メモリ部４０へのデータの書き込み回数を論理アドレス毎に計数する書き込み回数計数部２１、メモリ部４０のデータ消去の単位である物理ブロック毎のデータ書き換え回数としてデータ消去回数を計数する消去回数計数部２２、対応する論理アドレスが割り当てられていない物理ブロックであるスペアブロックの選択を行うスペアブロック選択処理部２３を備える。スペアブロック（フリーブロック）は、内部に有効データを含まない、用途未割り当ての物理ブロックである。

【００１９】

スペアブロック選択処理部２３は、ホスト３からデータの書き込み要求が発生すると、バス１１を介してＲＡＭ３０にアクセスして、そこに保持された管理情報（例えば、後述するスペアブロック管理テーブル）などに基づいて、データを書き込むスペアブロックの選択を行う。

【００２０】

図２は、本実施の形態における論物変換の概念を示すものであるが、その前提となる用語を説明する。論理アドレスとは、ホストが使用するアドレスであり、例えばＬＢＡ（Logical Block Addressing）である。ＬＢＡは、セクタ（サイズ：例えば５１２Ｂ）に対して０からの通し番号をつけた論理アドレスである。セクタは前述した物理ページよりも小さい単位である。物理アドレスとは、メモリ部４０での記憶位置を示すアドレスである。

【００２１】

本実施の形態では、データ消去の単位である物理ブロックのサイズと同一の区分サイズとなる論理アドレスの単位として、上記論理アドレス（例えばＬＢＡ）の上位アドレスである論理ブロックアドレスを用いて、物理ブロックの物理アドレスである物理ブロックアドレスに対応付ける。即ち図２に示すように、１個の論理ブロックアドレスに対して１個の物理ブロックアドレスが対応付けられる論物変換が行われることとする。しかし、本実施の形態の要旨に含まれる技術思想の範囲であるならば、この論物変換関係に必ずしも限定されるわけではない。

【００２２】

ＲＡＭ３０は、ホスト３とメモリ部４０との間に介在するキャッシュ領域として機能する領域の他に、論理アドレス管理テーブル３１及びスペアブロック管理テーブル３２を備えている。論理アドレス管理テーブル３１は、図３に示すように論理ブロックアドレスをメモリ部４０における物理ブロックアドレスに変換する第１の管理テーブルである論理アドレス／物理アドレス変換テーブル（論物変換テーブル）７０を含んでいる。

【００２３】

上述したように、スペアブロック選択処理部２３はホスト３からのデータの書き込み指示に対して、対応する論理アドレスが割り当てられていないスペアブロックを選択するので、本実施の形態においては、データ書き換えの都度、論物変換の関係が変わることになる。ホスト３からのメモリ部４０に対する書き込み指示には、通常、先頭論理アドレス、データサイズが含まれている。

【００２４】

さらに、論理アドレス管理テーブル３１は、図３に示すように上記論物変換テーブル７０に登録された物理ブロックアドレスが示す物理ブロックのデータ消去回数（物理ブロック消去回数）、及び論理ブロックアドレス毎にホスト３から指示された書き込み回数（論理アドレス書き込み回数）を記憶している。

【００２５】

従って、論理アドレス管理テーブル３１は論理ブロックアドレス毎のホスト３からの書き込み回数を記憶する第２の管理テーブルである論理書き込み回数テーブルも含んでいることになる。論理書き込み回数テーブルは、論理アドレスの属性としての書き換え頻度の偏り特性を反映したものとなる。

【００２６】

スペアブロック管理テーブル３２は、図４に示すように、対応する論理アドレスが割り当てられていない物理ブロックであるスペアブロックの物理ブロックアドレスと、それに対応する物理ブロックのデータ消去回数（物理ブロック消去回数）とを記憶している。このテーブルは、例えば図４に示すように、物理ブロック消去回数の大小に従って物理ブロックアドレスがソートされていてもよい。

【００２７】

従って、論理アドレス管理テーブル３１に登録された物理ブロックアドレス毎の物理ブロックのデータ消去回数の情報と、スペアブロック管理テーブル３２に登録された物理ブロックアドレス毎の物理ブロックのデータ消去回数の情報とを合わせると、物理ブロックアドレス毎のデータ消去回数を記憶する第３の管理テーブルである物理消去回数テーブルが構成される。物理消去回数テーブルは、物理ブロック毎の消去（及び書き込み）回数の頻度、即ち当該物理ブロックの物理的な疲労度を反映したものとなる。

【００２８】

ＣＰＵ２０は、論理アドレス管理テーブル３１に含まれる論物変換テーブル７０にもとづいて論理アドレスと物理アドレスとの変換を行うアドレス変換部（図示せず）としての動作をファームウエア（ＦＷ）により実行する。ホスト３からのコマンド入力に応じた半導体記憶装置２全体の制御もＣＰＵ２０においてファームウエアで実行される。

【００２９】

ＲＯＭ１３は、半導体記憶装置２のブートプログラム等が格納されている記憶部であり、半導体メモリ部３０の一部または図示しない不揮発性記憶部にもメモリコントローラ１０が半導体記憶装置２を制御するための情報が記憶される。

【００３０】

本実施の形態においては、メモリコントローラ１０が備えるＣＰＵ２０は、ホスト３から任意の論理アドレスへのデータの書き込み指示があった場合、書き込み回数計数部２１が論理ブロックアドレス毎に計数している論理アドレス書き込み回数をインクリメントする。その結果は、バス１１を介してＲＡＭ３０内の論理アドレス管理テーブル３１の論理アドレス書き込み回数の欄に記憶される。

【００３１】

また、ＣＰＵ２０は、メモリ部４０の各物理ブロックにおいてデータの消去があった場合も、消去回数計数部２２が物理ブロックアドレス毎に計数している物理ブロック消去回数をインクリメントする。その結果は、当該物理ブロックが対応する論理アドレス（即ち、論理ブロックアドレス）を有している場合は、バス１１を介してＲＡＭ３０内の論理アドレス管理テーブル３１の当該物理ブロックの物理ブロックアドレスが登録された行（即ち、上記論理ブロックアドレスが登録された行）の物理ブロック消去回数の欄に記憶される。

【００３２】

データ消去のあった物理ブロックが、対応する論理アドレスが割り当てられていない物理ブロックであるスペアブロックの場合は、バス１１を介してＲＡＭ３０内のスペアブロック管理テーブル３２の当該物理ブロックの物理ブロックアドレスが記憶された行の物理ブロック消去回数の欄に記憶される。

【００３３】

従来、ホスト３からの書き込み要求時に行われていたウェアレベリング（以下、受動的ウェアレベリングと呼ぶ）は、ホスト３からの書き込み要求で指定された論理アドレスへの過去の書き込み（書き換え）要求回数は全く考慮されずに行われていた。即ち、従来の受動的ウェアレベリングにおいては、スペアブロックの消去回数のみに依存して、例えば、図４のスペアブロック管理テーブル３２において物理ブロック消去回数が少ないスペアブロックを選択して、書き込み（または、ブロック消去及び書き込み）を行い、当該物理ブロックにホストから指定された論理アドレスを割り当てていた。

【００３４】

しかし、このような受動的ウェアレベリングでは、消去回数の平均化というウェアレベ

リングの目的が適切に実現出来ない可能性があった。例えば、ホストからの外部要求としての書き換え頻度が少ない論理アドレスに偶々書き換え要求が来た場合に、消去回数が少ない物理ブロックをスペアブロックから選択してデータを書き込む可能性があった。この場合、その物理ブロックは当該論理アドレス割り当て後の新しい論物変換の関係においても消去回数が少ないままである可能性が高く、結果として物理ブロックの書き換え回数の偏りが生じてしまい、平均化が適切に行われないという問題があった。

【００３５】

そこで、本実施の形態においては、ホストからデータの書き込み（書き換え）指示があった論理アドレスの書き込み頻度が少ない場合は、当該論理アドレスに対しては、あえて消去回数が大きいスペアブロックを選択して当該論理アドレスを割り当て、選択したスペアブロックに当該データを書き込む。逆に、ホストから書き込み指示があった論理アドレスの書き込み頻度が大きい場合は、当該論理アドレスに対しては、消去回数が少ないスペアブロックを選択して当該論理アドレスを割り当て、選択したスペアブロックに当該データを書き込む。これにより、上述した平均化が適切に行われないという問題を解決することが可能となる。

【００３６】

具体的には、スペアブロック選択処理部２３（書き込み制御部）が、論理アドレス管理テーブル３１に記憶されている書き込み回数が相対的に多い論理ブロックアドレスを指定した書き込み要求に対しては、スペアブロック管理テーブル３２に記憶されている消去回数が相対的に少ないスペアブロックを選択し、書き込み回数が相対的に少ない論理アドレスを指定した書き込み要求に対しては、消去回数が相対的に多いスペアブロックを選択する。

【００３７】

例えば、ホスト３から図３の論理ブロックアドレス「ｎ－１」を指定した書き込み要求が来た場合は、論理アドレス管理テーブル３１内での論理アドレス書き込み回数「１５０」が相対的に少ないと判断して、スペアブロック管理テーブル３２内での物理ブロック消去回数が相対的に多い図４の物理ブロックアドレス「ＷＷＷ」のスペアブロックを選択してデータを書き込む。逆に、図３の論理ブロックアドレス「ｎ」を指定した書き込み要求が来た場合は、論理アドレス管理テーブル３１内での論理アドレス書き込み回数「４００」が相対的に多いと判断して、スペアブロック管理テーブル３２内での物理ブロック消去回数が相対的に少ない図４の物理ブロックアドレス「ａａａ」のスペアブロックを選択してデータを書き込む。

【００３８】

更に具体的には、図３の論理アドレス管理テーブル３１の論理アドレス書き込み回数の大小関係でソートした論理ブロックアドレスと、図４のスペアブロック管理テーブル３２の物理ブロック消去回数の大小関係でソートしたスペアブロックの物理ブロックアドレスを、両者の大小関係が逆になるように対応づけておいてもよい。

【００３９】

また、大小関係を逆順に対応づける方法としては、上記のように生の回数値を用いるのでない方法で行うこともできる。例えば、図３の論理アドレス管理テーブル３１にさらに「書き込み回数相対値」の欄を設け、各論理アドレス書き込み回数をその時点での論理アドレス書き込み回数の最大値で割った値（０～１）を論理ブロックアドレス毎に保持しておく。さらに、スペアブロック管理テーブル３２にも「消去回数相対値」の欄を設け、各物理ブロック消去回数をその時点でのスペアブロックの中での物理ブロック消去回数の最大値で割った値（０～１）をスペアブロック毎に保持しておく。

【００４０】

そして、ホスト３からの論理ブロックアドレスを指定した書き込み要求に対して、図３の当該論理ブロックアドレスの「書き込み回数相対値」がＸであった場合。例えば、以下の（式１）に従い、「好適消去回数相対値」Ｙを算出する。

【００４１】

Ｙ＝１－Ｘ　（式１）

【００４２】

スペアブロック選択処理部２３は、ここで求まったＹに最も近い「消去回数相対値」を有するスペアブロックをスペアブロック管理テーブル３２から選択してデータを書き込む。例えば、「書き込み回数相対値」Ｘ＝０．２のように書き込み回数が相対的に小さい論理ブロックアドレスを指定した書き込み要求に対しては、「好適消去回数相対値」Ｙ＝０．８に最も近い消去回数がスペアブロックの中で相対的に大きいスペアブロックを選択する。

【００４３】

上記（式１）はＸとＹが逆順の関係になるような式であれば、上記の式に限定されない。また、図７に示すホストからの論理アドレス毎の書き込み回数頻度の偏りと図１０に示すスペアブロック毎のブロック消去回数頻度の偏りを反映させた非線形な式を用いて対応づけがより均等になるように工夫しても構わない。

【００４４】

以下に、図５のフローチャートを用いて、本実施の形態のメモリコントローラ１０が行う受動的ウェアレベリングの具体的な手順を示す。

【００４５】

＜ステップＳ１０１＞　データ書き込み指示

ホスト３からメモリコントローラ１０に、例えば論理ブロックアドレスＬＡ１を指定したデータ書き込み要求が来る。

【００４６】

＜ステップＳ１０２＞　スペアブロックの選択

スペアブロック選択処理部２３は、指定された論理ブロックアドレスＬＡ１に対して、上記（式１）に基づいてスペアブロックＳＢ１を選択する。

【００４７】

＜ステップＳ１０３＞　ブロック消去及びデータの書き込み

ステップＳ１０２で選択したスペアブロックＳＢ１に相当するメモリ部４０内の物理ブロックアドレスの物理ブロックをブロック消去し、その物理ブロックにホスト３から指示のあったデータの書き込みを行う。

【００４８】

＜ステップＳ１０４＞　物理ブロック消去回数のインクリメント

ＣＰＵ２０内の消去回数計数部２２は、ステップＳ１０３におけるブロック消去を物理ブロックアドレス毎に計数しており、図４のスペアブロック管理テーブル３２の当該物理ブロックアドレスの物理ブロック消去回数を１増やす。

【００４９】

＜ステップＳ１０５＞　論理アドレス書き込み回数のインクリメント

ＣＰＵ２０内の書き込み回数計数部２１は、ステップＳ１０１におけるホスト３からの各論理ブロックアドレスへの書き込み指示を論理ブロックアドレス毎に計数しており、図３の論理アドレス管理テーブル３１の論理ブロックアドレスＬＡ１の論理アドレス書き込み回数を１増やす。

【００５０】

＜ステップＳ１０６＞　物理ブロックの交換

さらにステップＳ１０２で選択したスペアブロックＳＢ１のスペアブロック管理テーブル３２に記載された物理ブロックアドレス及び物理ブロック消去回数と、論理アドレス管理テーブル３１の論理ブロックアドレスＬＡ１の物理ブロックアドレス及び物理ブロック消去回数とを交換する。交換時に例えば、ＲＡＭ３０内の所定の記憶領域（図示せず）等を一時記憶領域として利用しても構わない。これにより論理アドレス管理テーブル３１にエントリされていた物理ブロックとスペアブロック管理テーブル３２にエントリされていた物理ブロックが入れ替えられたことになる。交換後、スペアブロック管理テーブル３２を物理ブロック消去回数（データ消去回数）の大小に従ってソートする。

【００５１】
＜ステップＳ１０７＞　テーブルソート
スペアブロック管理テーブル３２を物理ブロック消去回数（データ消去回数）の大小関係に従ってソートする。
【００５２】
ホスト３からメモリコントローラ１０に、任意の論理アドレスへのデータ書き込み指示が来る度にステップＳ１０１～Ｓ１０７を繰り返す。
【００５３】
以上説明したように本実施の形態においては、ホストからデータの書き込み指示があった論理アドレスの書き込み頻度が少ない場合は、あえて消去回数が大きいスペアブロックを選択して当該データを書き込み、ホストから書き込み指示があった論理アドレスの書き込み頻度が大きい場合は、消去回数が少ないスペアブロックを選択して当該データを書き込むようにする。
【００５４】
このように、本実施の形態に係る半導体記憶装置２は、データ書き換えの都度、論物変換の関係が変わる場合において、論理アドレスの属性である書き換え回数、即ち論理アドレスの書き換え頻度を参照して、物理ブロックの書き換え回数の平均化を行う。
【００５５】
本実施の形態では、受動的ウェアレベリングにおいて、書き込み頻度が少ない論理ブロックアドレスには消去回数が大きく物理的な疲労の程度が大きい物理（スペア）ブロックを割り当て、逆に、書き込み頻度が大きい論理ブロックアドレスには消去回数が少ない物理的な疲労の程度が小さい物理（スペア）ブロックを割り当てることによって、物理ブロックの書き換え回数の平均化の精度を高めることが可能になると考えられる。
【００５６】
即ち、従来の受動的ウェアレベリングにおいて生じていた、書き換え頻度が少ない論理アドレスに書き込み／消去回数が少ない物理ブロックをスペアブロックとして選択する現象を回避することができるので、物理ブロックの書き換え回数の平均化というウェアレベリングの目的をより精度を高めて実現することが可能となり、メモリシステムの寿命を従来に比べて延ばすことが可能となる。
【００５７】
一方、ホストからのデータ書き換え指示が無いときであっても、消去回数が少ない物理ブロックに書き込まれているデータを消去回数が多いスペアブロックにメモリシステムが自主的に書き移すことにより、当該消去回数が少ない物理ブロックをスペアブロックにするウェアレベリング（以下、能動的ウェアレベリングと呼ぶ）の方法がある。
【００５８】
上記実施の形態においては、受動的ウェアレベリングの物理ブロック消去回数平均化が高精度になることによって、能動的ウェアレベリングの必要回数も低減されると考えられる。従って、能動的ウェアレベリングにおけるメモリシステムの自主的なデータ書き換えに伴う物理ブロックの消去の回数を減らすことも可能になるので、これによるメモリシステムの長寿命化も期待できる。
【００５９】
（第２の実施の形態）
本発明の第２の実施の形態にかかるメモリシステムとしての半導体記憶装置２の構成を示すブロック図は図１と同様である。同じ構成要素の説明は省略する。本実施の形態においても、例として図２に示すように１個の論理ブロックアドレスに対して１個の物理ブロックアドレス、即ち、メモリ部４０の１つの物理ブロックが対応付けられた論物変換が行われることとする。しかし、本実施の形態の要旨を逸脱しない範囲であるならば、この論物変換関係に限定されるわけではない。
【００６０】
また、ホスト３から任意の論理アドレスへのデータの書き込み指示に対して（論理アド

レスが割り当てられていない）スペアブロックを選択してデータを書き込んで当該論理アドレスと対応づける。即ち、データ書き換えの都度、論物変換の関係が変わるところも、第１の実施の形態と同様である。

【００６１】

本実施の形態においては、スペアブロック選択処理部２３の処理と、ＲＡＭ３０内に備えられる論理アドレス管理テーブル３１及びスペアブロック管理テーブル３２の構成が第１の実施の形態とは異なる。

【００６２】

図６に示す本実施の形態における論理アドレス管理テーブル３１は、第１の実施の形態と同様に、論理ブロックアドレスを対応する物理ブロックのメモリ部４０における物理アドレスである物理ブロックアドレスに変換する論理アドレス／物理アドレス変換テーブル（論物変換テーブル）７０を含んでいる。論物変換テーブル７０に記載された物理ブロックアドレスのデータ消去回数（物理ブロック消去回数）を記憶し、論理ブロックアドレス毎の書き込み回数（論理アドレス書き込み回数）を記憶して論理書き込み回数テーブルを含んでいることも第１の実施の形態と同様である。

【００６３】

しかし図６に示すように、本実施の形態の論理アドレス管理テーブル３１には、論理ブロックアドレス毎に論理アドレスグループのインデックスがさらに記憶されている。一般に、ホスト３からの論理アドレス毎の書き込み回数頻度には図７に示すように偏りが存在する。従って、図６の論理アドレス書き込み回数の大小関係に基づいて論理ブロックアドレス全てを複数の論理アドレスグループのいずれかに分類することが可能である。例えば、図８に示すようにＡ～Ｄのインデックスをそれぞれ持った４つの論理アドレスグループに分類することができる。

【００６４】

具体的には、例えば図６に示された全ての論理ブロックアドレスに対する論理アドレス書き込み回数の最大値に対する各論理ブロックアドレス書き込み回数の比（０～１）を、３つの閾値、例えば、０．２５、０．５、０．７５との大小関係で４つに分類し、図８に示すように比が大きい方から、Ａ、Ｂ、Ｃ、Ｄのインデックスをそれぞれ持った４つ論理アドレスグループに分類することができる。

【００６５】

論理アドレスグループの数は複数であればよく４に限定されるわけではない。また、論理アドレス書き込み回数の大小関係に基づいて、全ての論理ブロックアドレスをほぼ同数ずつの論理ブロックアドレスを有する複数の論理アドレスグループに分類してもよく、論理アドレス書き込み回数の大小関係に基づいた分類ならば閾値の間隔を含め上記分類方法に限定されない。このように定めたインデックスを図６の論理アドレス管理テーブル３１の論理アドレスグループの欄に各論理ブロックアドレスの属性として、論理ブロックアドレス毎に記憶させる。

【００６６】

一方図９に示す本実施の形態におけるスペアブロック管理テーブル３２も、第１の実施の形態と同様に、対応する論理アドレスが割り当てられていない物理ブロックであるスペアブロックの物理ブロックアドレスとそれに対応する物理ブロックの物理ブロック消去回数（データ消去回数）を記憶している。そして、物理ブロック消去回数の大小に従って物理ブロックアドレスがソートされている。

【００６７】

しかし図９に示すように、本実施の形態のスペアブロック管理テーブル３２には、スペアブロックの物理ブロックアドレス毎にスペアブロックグループのインデックスがさらに記憶されている。

【００６８】

一般に、スペアブロック毎のブロック消去の回数頻度には図１０に示すように偏りが存在する。従って、図９の物理ブロック消去回数の大小関係に基づいてスペアブロック全て

を、上記論理アドレスグループの数と例えば同数のスペアブロックグループのいずれかに分類することが可能である。例えば、図１１に示すようにＡ～Ｄのインデックスをそれぞれ持った４つのスペアブロックグループに分類することができる。

【００６９】

具体的には、例えば図９に示された全てのスペアブロックに対する物理ブロック消去回数の最大値に対する各物理ブロック消去回数の比（０～１）を、３つの閾値、例えば、０．２５、０．５、０．７５との大小関係で４つに分類し、図１１に示すように比が小さい方から、Ａ、Ｂ、Ｃ、Ｄのインデックスをそれぞれ持ったスペアブロックグループに分類することができる。

【００７０】

スペアブロックグループの数は上記論理アドレスグループの数と、例えば同数であればよく４に限定されるわけではない。また、物理ブロック消去回数の大小関係に基づいて、全てのスペアブロックをほぼ同数ずつのスペアブロックを有する複数のスペアブロックグループに分類してもよく、物理ブロック消去回数の大小関係に基づいた分類ならば閾値の間隔を含め上記分類方法に限定されない。このインデックスを図９のスペアブロック管理テーブル３２のスペアブロックグループの欄に、スペアブロックの物理ブロックアドレス毎に記憶させる。

【００７１】

本実施の形態においては、ホストからデータの書き込み（書き換え）指示があった論理ブロックアドレスが書き込み頻度の少ない論理アドレスグループに属している場合は、当該論理ブロックアドレスに対しては、消去回数の頻度が大きいスペアブロックグループに属するスペアブロックを選択して当該論理ブロックアドレスを割り当て、選択したスペアブロックに相当するメモリ部４０内の物理ブロックに当該データを書き込む。

【００７２】

逆に、ホストから書き込み指示があった論理ブロックアドレスが書き込み頻度の大きい論理アドレスグループに属している場合は、当該論理ブロックアドレスに対しては、消去回数の頻度が少ないスペアブロックグループに属するスペアブロックを選択して当該論理ブロックアドレスを割り当て、選択したスペアブロックに相当するメモリ部４０内の物理ブロックに当該データを書き込む。

【００７３】

上述した図８及び図１１の例の場合、論理アドレス書き込み回数の大小関係と物理ブロック消去回数の大小関係が逆になるように、それぞれの論理アドレスグループには同じインデックスのスペアブロックグループ、すなわち、Ａ同士、Ｂ同士、Ｃ同士、Ｄ同士を組み合わせる。この組み合わせの中から、書き込み指示があった論理アドレスに対してスペアブロックの選択を行う。

【００７４】

以下に、図１２のフローチャートを用いて、本実施の形態のメモリコントローラ１０が行う受動的ウェアレベリングの具体的な手順を示す。

【００７５】

＜ステップＳ２０１＞　データ書き込み指示

ホスト３からメモリコントローラ１０に、例えば論理ブロックアドレスＬＡ１を指定したデータ書き込み要求が来る。

【００７６】

＜ステップＳ２０２＞　論理アドレスグループの判定

スペアブロック選択処理部２３は、図６の論理アドレス管理テーブル３１にアクセスし論理ブロックアドレスＬＡ１の論理アドレスグループのインデックスに基づき、論理アドレスグループを判定する。

【００７７】

＜ステップＳ２０３＞　スペアブロックの選択

スペアブロック選択処理部２３は、図９のスペアブロック管理テーブル３２にアクセス

して、ステップＳ２０２で判定した論理アドレスグループと同じインデックスのスペアブロックグループに属するスペアブロックＳＢ１を選択する。

【００７８】

上述したように、それぞれの論理アドレスグループには論理アドレス書き込み回数の大小関係と物理ブロック消去回数の大小関係が逆になるようにスペアブロックグループが対応付けられている。従って、ステップＳ２０２で判定した論理アドレスグループに対応するスペアブロックグループに属するスペアブロックであれば、どれを選択しても、例えば当該スペアブロックグループに属するスペアブロックをランダムに１つ選択するようにしても構わない。

【００７９】

しかし、第１の実施の形態の考えをここでも反映させて、ステップＳ２０２で判定した論理アドレスグループの中での論理ブロックアドレスＬＡ１の論理アドレス書き込み回数の相対的な大小の程度に応じて、同一インデックスのスペアブロックグループからスペアブロックＳＢ１を選択するようにした方が、物理ブロックの消去回数の平均化がさらに効果的に行える。即ち、ステップＳ２０２で判定した論理アドレスグループの中で論理アドレスＬＡ１の論理アドレス書き込み回数が相対的に大きい場合は、対応するスペアブロックグループに属するスペアブロックの中で物理ブロック消去回数が相対的に小さいスペアブロックを選択し、逆に論理アドレス書き込み回数が相対的に大きい場合は、物理ブロック消去回数が相対的に小さいスペアブロックを選択してもよい。

【００８０】

このような機構は、例えば第１の実施の形態で説明した（式１）等を用いれば実現できる。この場合、図６の論理アドレス管理テーブル３１の論理アドレスグループの欄に論理アドレスグループのインデックスに付加して、当該インデックスの論理アドレスグループ内での書き込み回数の最大値でその行の論理アドレス書き込み回数を割った値である論理アドレスグループ毎の「書き込み回数相対値」をさらに記憶させる。例えば「Ａ：０．３」等である。同様に図９のスペアブロック管理テーブル３２のスペアブロックグループの欄にも、各スペアブロックグループ内での消去回数の最大値でその行の物理ブロック消去回数を割った値スペアブロックグループ毎の「消去回数相対値」を「Ｄ：０．４」のように付加記憶させる。そしてステップＳ２０２で判定した論理アドレスグループと同じインデックスのスペアブロックグループを選択した後、その中で（式１）等に従って「好適消去回数相対値」に最も近いスペアブロックグループＳＢ１を選択する。

【００８１】

＜ステップＳ２０４＞　ブロック消去及びデータの書き込み

ステップＳ２０３で選択したスペアブロックＳＢ１に相当するメモリ部４０内の物理ブロックアドレスの物理ブロックをブロック消去し、その物理ブロックにホスト３から指示のあったデータの書き込みを行う。

【００８２】

＜ステップＳ２０５＞　物理ブロック消去回数のインクリメント

ＣＰＵ２０内の消去回数計数部２２は、ステップＳ２０４におけるブロック消去を物理ブロックアドレス毎に計数しており、図９のスペアブロック管理テーブル３２の当該物理ブロックアドレスの物理ブロック消去回数を１増やす。

【００８３】

＜ステップＳ２０６＞　論理アドレス書き込み回数のインクリメント

ＣＰＵ２０内の書き込み回数計数部２１は、ステップＳ２０１におけるホスト３からの各論理ブロックアドレスへの書き込み指示を論理ブロックアドレス毎に計数しており、図６の論理アドレス管理テーブル３１の論理ブロックアドレスＬＡ１の論理アドレス書き込み回数を１増やす。

【００８４】

＜ステップＳ２０７＞　物理ブロックの交換

さらにステップＳ２０３で選択したスペアブロックＳＢ１のスペアブロック管理テーブ

ル３２に記載された物理ブロックアドレス及び物理ブロック消去回数と、論理アドレス管理テーブル３１の論理ブロックアドレスＬＡ１の物理ブロックアドレス及び物理ブロック消去回数を交換する。交換時に例えば、ＲＡＭ３０内の所定の記憶領域（図示せず）等を一時記憶領域として利用しても構わない。これにより論理アドレス管理テーブル３１にエントリされていた物理ブロックとスペアブロック管理テーブル３２にエントリされていた物理ブロックが入れ替えられたことになる。交換後、スペアブロック管理テーブル３２を物理ブロック消去回数（データ消去回数）の大小に従ってソートする。

【００８５】

＜ステップＳ２０８＞　論理アドレスグループのインデックスの更新

ステップＳ２０６で論理アドレス管理テーブル３１の論理アドレス書き込み回数がインクリメントされているので、論理アドレスグループのインデックスの更新の必要が生じる場合がある。具体的には、全ての論理ブロックアドレスに対する論理アドレス書き込み回数の最大値と各論理アドレス書き込み回数との比（０～１）を計算し、例えば、０．２５、０．５、０．７５の３つの閾値で再度４つのインデックスＡ、Ｂ、Ｃ、Ｄに分類する。以前と異なるインデックスになった論理ブロックアドレスがあった場合は、図６の当該論理ブロックアドレスの論理アドレスグループの欄をそのインデックスに更新する。

【００８６】

＜ステップＳ２０９＞　スペアブロックグループのインデックスの更新

ステップＳ２０７で、スペアブロック管理テーブル３２に記載された物理ブロックアドレス及び物理ブロック消去回数が書き換えられ、物理ブロック消去回数の大小に従ってソートされているので、スペアブロックグループのインデックスの更新の必要が生じる。具体的には、全てのスペアブロックの物理ブロック消去回数の最大値と各物理ブロック消去回数との比（０～１）を計算し、例えば、０．２５、０．５、０．７５の３つの閾値で再度４つのインデックスＡ、Ｂ、Ｃ、Ｄに分類する。そして、スペアブロック管理テーブル３４の全ての物理ブロックアドレスのスペアブロックグループの欄をそのインデックスに更新する。

【００８７】

ホスト３からメモリコントローラ１０に、任意の論理アドレスへのデータ書き込み指示が来る度にステップＳ２０１～Ｓ２０９を繰り返す。

【００８８】

以上説明したように本実施の形態においては、データ書き換えの都度、論物変換の関係が変わるメモリシステムにおいて、論理ブロックアドレスの属性として書き込み回数頻度を計数してその頻度の相対的な大小関係で論理ブロックアドレスをグループ分けし、スペアブロックを構成する物理ブロックも消去回数の相対的な大小関係でグループ分けしておく。

【００８９】

論物変換の関係が変わるデータ書き換え時に、書き込み回数頻度が小さい論理アドレスグループは消去回数が大きいスペアブロックグループと、書き込み回数頻度が大きい論理アドレスグループは消去回数が小さいスペアブロックグループと組み合わせて、その組み合わせの中で書き換え要請があった論理アドレスにスペアブロックを選択する。

【００９０】

これにより、従来の受動的ウェアレベリングにおいて生じていた、書き換え頻度が少ない論理アドレスに書き込み／消去回数が少ない物理ブロックをスペアブロックとして選択する現象を回避することができるので、物理ブロックの書き換え回数の平均化というウェアレベリングの目的をより精度を高めて実現することが可能となり、メモリシステムの寿命を従来に比べて延ばすことが可能となる。また、第１の実施の形態同様、能動的ウェアレベリングの必要回数の低減によるメモリシステムの長寿命化も期待できる。

【００９１】

なお、本実施の形態においては、スペアブロックグループの数は論理アドレスグループの数と同数として説明したが、論理アドレスの属性として書き込み回数の大小関係とスペ

アブロックの消去回数の大小関係が逆順となるグループの組み合わせにする趣旨が実現できるのであれば、必ずしも同数でなくてもかまわない。

【００９２】

また、本実施の形態のより簡易な実現例として以下のようにしてもよい。即ち、論理アドレス書き込み回数の最大値に対する各論理ブロックアドレス毎の書き込み回数の比を、例えば閾値０．１と０．９との大小関係により大きい方からＡ、Ｂ、Ｃの論理アドレスグループに分類し、全てのスペアブロックに対する物理ブロック消去回数の最大値に対する各物理ブロック消去回数の比も、例えば閾値０．１と０．９との大小関係により小さい方からＡ、Ｂ、Ｃのスペアブロックグループに分類して同一インデックスのグループ同士を対応づけてもよい。

【００９３】

（第３の実施の形態）

本発明の第３の実施の形態にかかるメモリシステムの構成を示すブロック図は、図１と同様であるが、本実施の形態は能動的ウェアレベリングに関するものである。ＲＡＭ３０は、図３に示す論理アドレス管理テーブル３１と、図４に示すスペアブロック管理テーブル３２を備えている。各構成要素の第１の実施の形態と共通な動作の説明は省略する。

【００９４】

本実施の形態においても、例として図２に示すように１個の論理ブロックアドレスに対して１個の物理ブロックアドレス、即ち、メモリ部４０の１つの物理ブロックが対応付けられた論物変換が行われることとする。しかし、本実施の形態の要旨を逸脱しない範囲であるならば、この論物変換関係に限定されるわけではない。また、ホスト３から任意の論理アドレスへのデータの書き込み指示に対して（論理アドレスを持たない）スペアブロックを選択してデータを書き込んで当該論理アドレスと対応づける。即ち、データ書き換えの都度、論物変換の関係が変わるところも、第１及び第２の実施の形態と同様である。

【００９５】

本実施の形態においては、ホスト３からの書き換え指示が無いときの物理ブロックの書き換え回数平均化制御である能動的ウェアレベリングにおいて、物理ブロックの書き換え回数が少なく且つ論理ブロックアドレスへの書き込み回数が少ない物理ブロックを疲労度が高いスペアブロックと交換する。しかし、物理ブロックの書き換え回数が少なくても論理ブロックアドレスへの書き込み回数が多い物理ブロックに対しては能動的ウェアレベリングを実行しない。これにより、物理ブロックの書き換え回数平均化の精度を従来に比べて高めることが可能となる。

【００９６】

即ち、能動的ウェアレベリングにおいて、有効なデータが書き込まれている物理ブロックの論理ブロックアドレスの書き換え頻度を考慮する。具体的には、メモリコントローラ１０が図３の論理アドレス管理テーブル３１の論理アドレス書き込み回数を考慮に入れた上で能動的ウェアレベリングを行う。

【００９７】

以下に、図１３のフローチャートを用いて、本実施の形態のメモリコントローラ１０が行う能動的ウェアレベリングの具体的な手順を示す。

【００９８】

＜ステップＳ３０１＞　開始条件判定

ホスト３から論理アドレスへのデータの書き込み指示が無いときに、ＣＰＵ２０が、ＲＡＭ３０内の論理アドレス管理テーブル３１の物理ブロック消去回数とスペアブロック管理テーブル３２の物理ブロック消去回数をチェックし、全ての物理ブロックの中での消去回数の最大値と最小値の差が所定の閾値以上になったとき等、物理ブロックの消去回数の不均衡が所定の条件を超えたかどうかを開始条件として判定する。また、一定時間毎に周期的に開始する、或いは開始条件を判定するとしてもよい。開始条件を満たす場合にステップＳ３０２に進む。満たさない場合はステップＳ３０１を繰り返す。

【００９９】

＜ステップＳ３０２＞　論理アドレス書き込み回数の閾値判定
ＣＰＵ２０は、ＲＡＭ３０内の論理アドレス管理テーブル３１にアクセスして、論理アドレス管理テーブル３１の論理アドレス書き込み回数の中で第１閾値より小さいという条件を満たすものがあるかどうかを判定する。第１閾値はＲＡＭ３０等に保持されていればよい。第１閾値の値としては、例えば図３に示された全ての論理ブロックアドレスに対する論理アドレス書き込み回数の最大値に例えば０．１～０．３等の係数を掛けた値にして、相対的に論理アドレス書き込み回数が小さいものを選別出来るようにしてもよい。
【０１００】
しかし、ホスト３からの今後の書き込み要求が少ないことが予測される論理ブロックアドレスを選択するのが第１閾値の目的であるので、上記例や係数に限定されず、相対値ではなく例えば具体的な書き込み回数の数値にしても構わない。条件を満たす論理ブロックアドレスが全く無い場合は、ステップＳ３０１に戻り、条件を満たす論理ブロックアドレスがある場合は、ステップＳ３０３に進む。
【０１０１】
＜ステップＳ３０３＞　物理ブロック消去回数の閾値判定
ＣＰＵ２０は、ＲＡＭ３０内の論理アドレス管理テーブル３１の中でステップＳ３０２の条件を満たした論理ブロックアドレスの行の物理ブロック消去回数の中で第２閾値より小さいという条件を満たすものがあるかどうかを判定する。第２閾値はＲＡＭ３０等に保持されていればよい。その値としては、例えば図３の全ての論理ブロックアドレスに対する物理ブロック消去回数の最大値に例えば０．１～０．３等の係数を掛けた値にして、有効なデータが書き込まれている物理ブロックの中からさらに物理ブロック消去回数が小さいものを選別出来るようなものを選べばよい。
【０１０２】
ステップＳ３０２の条件を満たした論理ブロックアドレスに対応する物理ブロックの中からさらに物理的な疲労度が少ない物理ブロックを選択するのが目的であるので、第２閾値も上記例や係数に限定されず、具体的な物理ブロック消去回数でもかまわない。条件を満たす物理ブロックが全く無い場合は、ステップＳ３０１に戻る。条件を満たす物理ブロックがある場合は、当該物理ブロックを有閑物理ブロックとし、それに対応する論理ブロックアドレスを有閑論理アドレスとする。そしてステップＳ３０４に進む。
【０１０３】
＜ステップＳ３０４＞　疲労スペアブロックのブロック消去及びデータの書き込み
スペアブロック選択処理部２３が、図４のスペアブロック管理テーブル３２に示されたスペアブロックの中での物理ブロック消去回数が第３閾値よりも大きいか或いは最大値である疲労スペアブロックを選択する。そして、当該疲労スペアブロックをブロック消去した上で、上記有閑物理ブロックに書き込まれているデータを書き込む。
【０１０４】
第３閾値としては、基本的に第２閾値より大きいことが条件であるが、物理ブロックの物理的な絶対的な疲労度を基準にするのであれば定数を選択してもよい。しかし、例えば図４に示された全てのスペアブロックに対する物理ブロック消去回数の最大値に例えば０．９等の係数を掛けた値のように相対的な値を選択してもよい。疲労度が高いスペアブロックを選択できるのであれば、第３閾値は上記例や係数に限定されるものではない。
【０１０５】
また、ステップＳ３０３の条件を満たす有閑物理ブロックが１つであれば、物理ブロック消去回数がスペアブロック管理テーブル３２の中で最大値をとる疲労スペアブロックを選択してもよい。有閑物理ブロックが複数ある場合は、その中で対応する論理アドレス書き込み回数がより小さいもの程、物理ブロック消去回数がより大きい疲労スペアブロックにデータを書き込むように逆順に選択すればより効果的な能動的ウェアレベリングとなる。
【０１０６】
また、第２閾値より大きい物理ブロック消去回数のスペアブロックを疲労スペアブロッ

クとして選択して有閑物理ブロックに書き込まれているデータを書き込むのであれば、能動的ウェアレベリングは実現されるので、必ずしも第３閾値を考慮して疲労スペアブロックを選択しなくてもよい。次にステップＳ３０５に進む。

【０１０７】

＜ステップＳ３０５＞　疲労スペアブロックの消去回数のインクリメント

ＣＰＵ２０内の消去回数計数部２２は、ステップＳ３０４におけるブロック消去を物理ブロックアドレス毎に計数しており、図４のスペアブロック管理テーブル３２の当該物理ブロックアドレスの物理ブロック消去回数を１増やす。

【０１０８】

＜ステップＳ３０６＞　有閑物理ブロックと疲労スペアブロックの交換

スペアブロック選択処理部２３は、ステップＳ３０４で有閑物理ブロックに書き込まれているデータを書き込んだ疲労スペアブロックの物理ブロックアドレス及び物理ブロック消去回数と、論理アドレス管理テーブル３１の当該有閑物理ブロックの物理ブロックアドレス及び物理ブロック消去回数を交換する。交換時に例えば、ＲＡＭ３０内の所定の記憶領域（図示せず）等を一時記憶領域として利用しても構わない。

【０１０９】

これにより当該有閑物理ブロックの物理ブロックアドレスが論理アドレス管理テーブル３１から消去されて、スペアブロック管理テーブル３２に新たにエントリされ、当該有閑物理ブロックは新たにスペアブロックになる。交換後、スペアブロック管理テーブル３２を物理ブロック消去回数（データ消去回数）の大小に従ってソートする。

【０１１０】

ホスト３からメモリコントローラ１０に、データ書き込み指示が来ないときに、ステップ３０１～ステップ３０６を行う。

【０１１１】

以上説明したように本実施の形態においては、ホストからの書き換え指示が無いときの能動的ウェアレベリングにおいて、ホストからの書き込み要求頻度が少ない論理ブロックアドレスの物理ブロックに限定して、当該物理ブロックの消去回数が少ない場合に、疲労度が高いスペアブロックと交換する。これにより、従来、ホストからの書き込み要求頻度が多い論理ブロックアドレスの物理ブロックに対しても当該物理ブロックの消去回数が少ないという条件のみに基づいて疲労度が高いスペアブロックと交換していた能動的ウェアレベリングを回避できる。

【０１１２】

即ち、上記有閑論理アドレス以外のデータ書き換え頻度が高い論理アドレスに対しては、例え当該アドレスに対応する物理ブロックの書き換え回数が少なくても能動的ウェアレベリングの対象とはしない。これは、そのような物理ブロックはあえて能動的ウェアレベリングの対象としなくても、ホストからの書き込み要求が頻繁に来るはずなので、そのときに、強制的にスペアブロックと物理ブロックが交換されるからである。

【０１１３】

従って、本実施の形態によって、従来の能動的ウェアレベリングによる物理ブロックの書き換え回数平均化の精度を維持しつつ能動的ウェアレベリングの必要回数を減らすことが可能になる。よって、能動的ウェアレベリングの自主的なデータ書き換えに伴う物理ブロックの消去回数を減らすことが可能になるので、メモリシステムの長寿命化が期待できる。

【０１１４】

また、上記ステップＳ３０４で述べたように、有閑物理ブロックが複数ある場合は、その中で対応する論理アドレス書き込み回数がより小さいもの程、物理ブロック消去回数がより大きい疲労スペアブロックにデータを書き込むように逆順に選択することにより、第１及び２の実施の形態で説明したのと同様な理由により物理ブロックの書き換え回数平均化の精度を高めることが可能となる。即ち、本実施の形態によって能動的ウェアレベリングによる物理ブロックの書き換え回数平均化の精度を高めることも可能であり、メモリシ

ステムのさらなる長寿命化が可能となる。

【０１１５】

さらに、本実施の形態は能動的ウェアレベリングを改良するものであるので、第１及び第２の実施の形態で説明した改良した受動的ウェアレベリングと併用可能であり、併用により更なる効果が期待できる。即ち、ホストからの書き換え指示があったときは、図５或いは図１２のフローチャートに従い、ホストからの書き換え指示がないときは、図１３のフローチャートに従って、メモリコントローラ１０が動作すればよい。これにより、上記した第１乃至３の実施の形態より、物理ブロックの消去回数平均化の精度をさらに高めることが期待でき、メモリシステムの長寿命化がさらに図れることになる。

【０１１６】

上記したいずれの実施の形態も、物理的な書き換え回数に依存した劣化特性を有する記憶素子を複数搭載し、論物変換の対応関係が書き換えの都度変化するストレージデバイスにおいて、論理アドレスへの書き込み回数が、今後の当該論理アドレスへの書き込み頻度の予測値になるという考えに基づいている。即ち、論理アドレスへの書き込み回数が少ないものには消去回数が多く疲労度が高いスペアブロックをあえて割り当て、記憶素子の物理的な書き換え回数の平均化を将来にわたってより高精度に行えるようにしてストレージデバイスの長寿命化を図っている。

【０１１７】

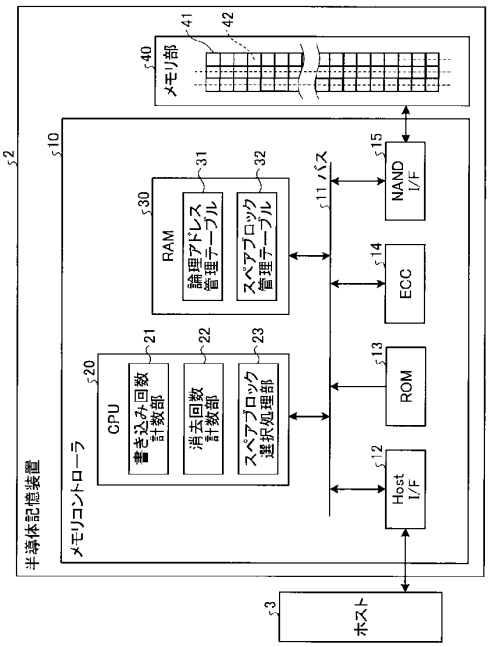
従って、本願発明は上記実施の形態に限定されるものではなく、実施段階ではその要旨を逸脱しない範囲で種々に変形することが可能である。また、上記実施の形態には種々の段階の発明が含まれており、開示される複数の構成要件における適宜な組み合わせにより種々の発明が抽出されうる。例えば、実施の形態に示される全構成要件からいくつかの構成要件が削除されても、発明が解決しようとする課題の欄で述べた課題が解決でき、発明の効果の欄で述べられている効果が得られる場合には、この構成要件が削除された構成が発明として抽出されうる。更に、異なる実施の形態にわたる構成要素を適宜組み合わせてもよい。

【符号の説明】

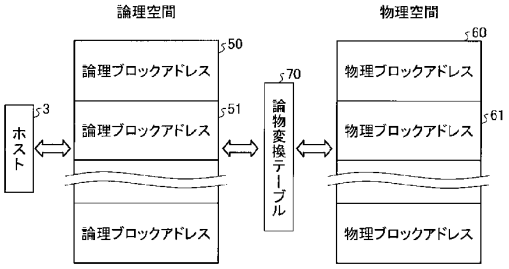
【０１１８】

２　半導体記憶装置、３　ホスト、１０　メモリコントローラ、１１　バス、１２　ホスト　Ｉ／Ｆ（インターフェイス）、１３　ＲＯＭ、１４　誤り訂正部、１５　ＮＡＮＤ　Ｉ／Ｆ、２０　ＣＰＵ、２１　書き込み回数係数部、２２　消去回数計数部、２３　スペアブロック選択処理部、３０　ＲＡＭ、３１　論理アドレス管理テーブル、３２　スペアブロック管理テーブル、４０　不揮発性半導体メモリ部、４１　メモリセル、４２　ワード線、５０、５１　論理ブロックアドレス、６０、６１　物理ブロックアドレス、７０　論物変換テーブル、Ｓ１０１～Ｓ１０７、Ｓ２０１～Ｓ２０９、Ｓ３０１～Ｓ３０６　ステップ。

【図１】



【図２】



【図３】

論理アドレス管理テーブル

論理ブロックアドレス	物理ブロックアドレス	物理ブロック消去回数	論理アドレス書き込み回数
0	XXX	450	200
1	YYY	400	300
.....	.....	.....	.....
n-1	PPP	600	150
n	ZZZ	300	400

70
論物変換テーブル

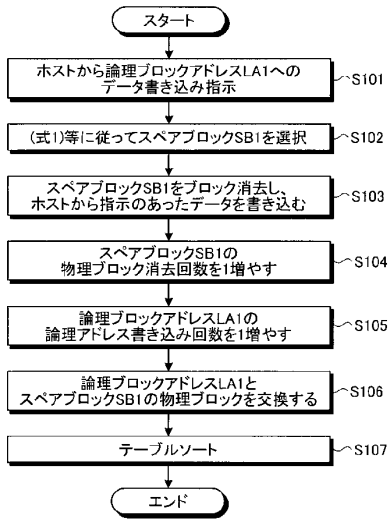
【図４】

スペアブロック管理テーブル

物理ブロックアドレス	物理ブロック消去回数
aaa	300
bbb	350
.....	.....
vvv	700
www	800

小
物理ブロック消去回数
大

【図５】



【図６】

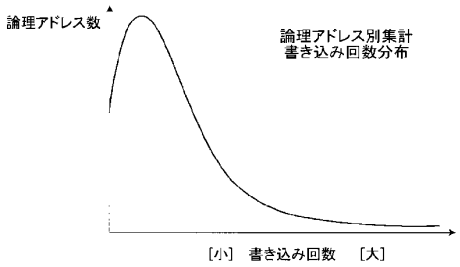
論理アドレス管理テーブル

31

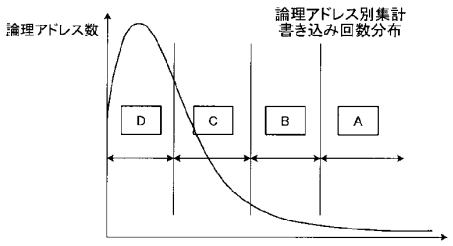
論理ブロックアドレス	物理ブロックアドレス	物理ブロック消去回数	論理アドレス書き込み回数	論理アドレスグループ
0	XXX	450	200	B
1	YYY	400	300	C
⋮	⋮	⋮	⋮	⋮
n-1	PPP	600	150	D
n	ZZZ	300	400	A

70
論物変換テーブル

【図７】



【図８】



【図９】

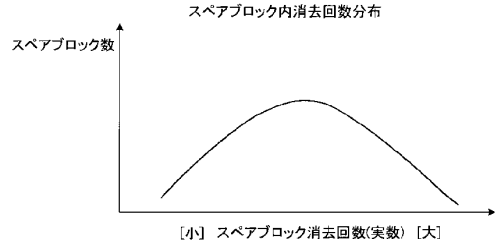
スペアブロック管理テーブル

32

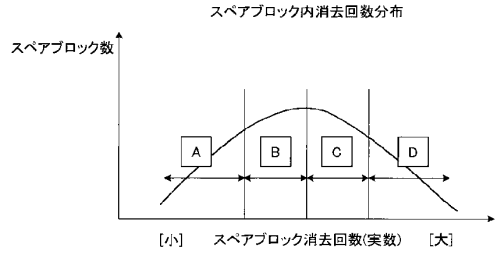
物理ブロックアドレス	物理ブロック消去回数	スペアブロックグループ
aaa	300	A
bbb	350	A
⋮	⋮	⋮
vvv	700	D
www	800	D

A
B
C
D
小
物理ブロック消去回数
大

【図１０】



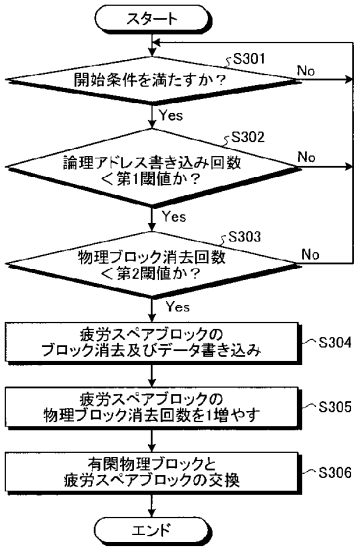
【図１１】



【図１２】



【図１３】


スタート
S301
開始条件を満たすか？
No
Yes
S302
論理アドレス書き込み回数
＜第1閾値か？
No
Yes
S303
物理ブロック消去回数
＜第2閾値か？
No
Yes
疲労スペアブロックの
ブロック消去及びデータ書き込み
S304
疲労スペアブロックの
物理ブロック消去回数を1増やす
S305
有閑物理ブロックと
疲労スペアブロックの交換
S306
エンド

Empty/full status tracking method for random access read/write memory cell, involves generating enabled signals to allow write and read flag status registers to be updated by using write and read flag enable logics, respectively

---

DOCUMENT ID
RD-527033-A

DATE PUBLISHED
2008-03-10

APPLICATION NO
2008RD-527033

DATE FILED
2008-03-10

FOREIGN APPLICATION PRIORITY DATA

COUNTRY	APPLICATION NO	APPLICATION DATE
RD	527033	

DERWENT ACCOUNT NO
RD-527033-A

DERWENT WEEK
200839

PATENT FAMILY

PUB NO	PUB DATE	LANGUAGE
RD 527033 A	2008-03-10	EN

ABSTRACTED PUB NO	CHOSEN DRAWING	DERWENT CLASS
RD 527033 A	Dwg.1/1	U14

TITLE TERMS
EMPTY FULL STATUS TRACK METHOD RANDOM ACCESS READ WRITING MEMORY CELL GENERATE ENABLE SIGNAL ALLOW FLAG REGISTER UPDATE LOGIC RESPECTIVE

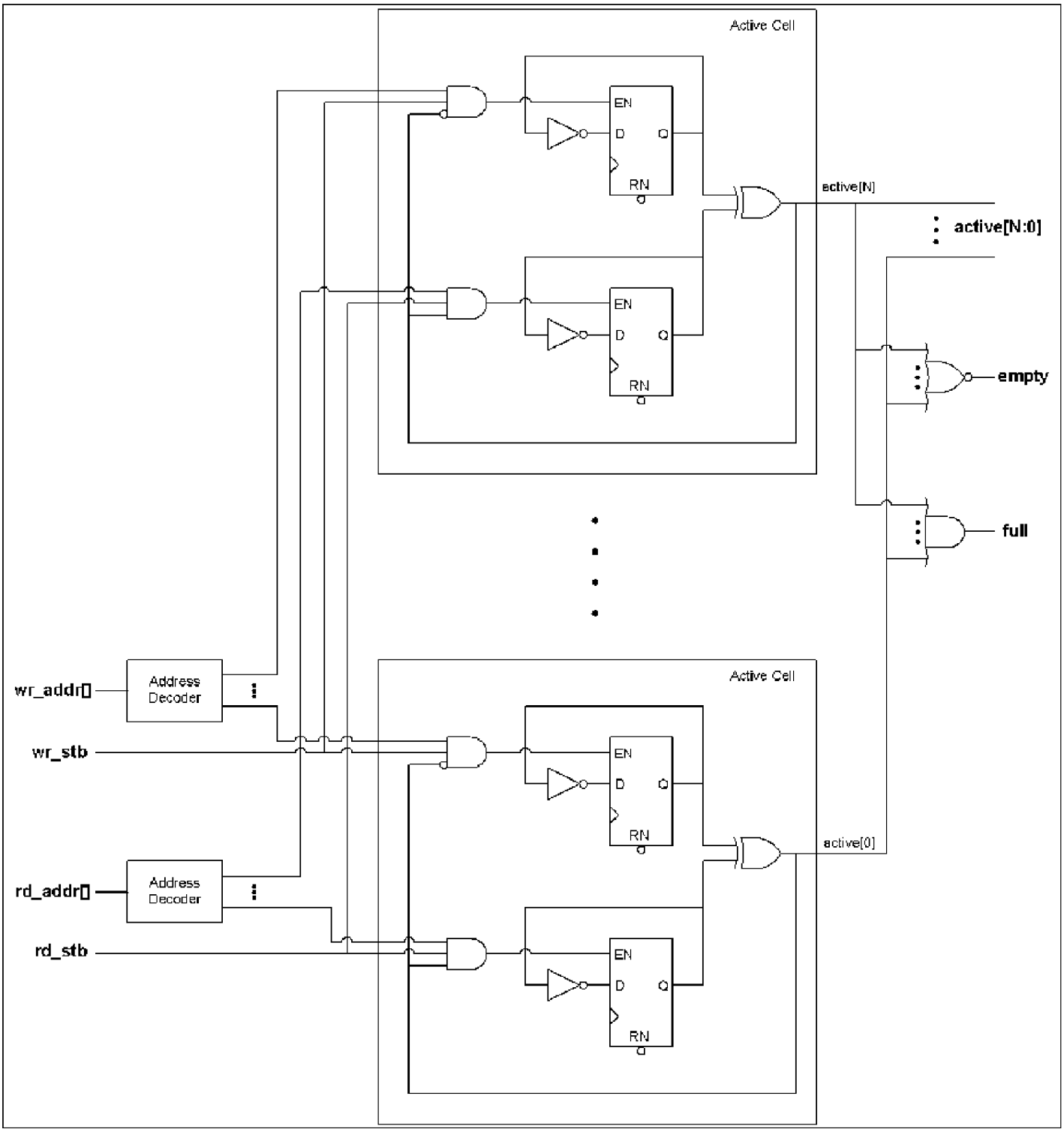
APPL PUB NO	APPL DESCRIPTOR	APPL NO	APPL DATE
RD 527033A	N/A	2008RD-527033	Sun Mar 09 20:00:00 EDT 2008

EPI CODES
U14-A03B4

Abstract

NOVELTY - The method involves selecting an appropriate active cell based on an input read address and a write address by using a read and write address decoder. Enabled signals are generated to allow write and read flag status registers to be updated by using write and read flag enable logics. The read or write flag status register is updated with an inverted version of a current state value of the read or write flag status register, when the read or write enable signal is asserted.
USE - Method for tracking empty/full status for a random access read/write memory cell by using an XOR gate in a verilog hardware description language.
ADVANTAGE - The method tracks the status of a memory cell in an efficient way without the need for arbitration, thus allowing simultaneous read and write operations to be taken place in a RAM without the need for arbiter logic to update the cell status.
DESCRIPTION OF DRAWING(S) - The drawing shows a circuit diagram of a random access read/write memory cell.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.


Active Cell
EN
D
Q
RN
active[N]
active[N:0]
empty
full
Active Cell
wr_addr[]
Address Decoder
wr_stb
rd_addr[]
Address Decoder
rd_stb
active[0]

FLASH MEMORY STORAGE SYSTEM FOR SIMULATING REWRITABLE DISK DRIVE, FLASH MEMORY CONTROLLER, COMPUTER SYSTEM, AND METHOD THEREOF

DOCUMENT ID
JP-2011123863-A

DATE PUBLISHED
2011-06-23

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
NG HON-WAI	N/A	N/A	N/A	N/A
HUANG YI-HSIANG	N/A	N/A	N/A	N/A
HSU SHIH-HSIEN	N/A	N/A	N/A	N/A
YU HSIANG-HSIUNG	N/A	N/A	N/A	N/A

DATE FILED
2010-05-21

CPC CURRENT

TYPE	CPC	DATE
CPCI	G06F3/0664	2013-01-01
CPCI	G06F3/0661	2013-01-01
CPCI	G06F3/0661	2013-01-01
CPCI	G06F3/0679	2013-01-01
CPCI	G06F3/0679	2013-01-01
CPCI	G06F3/0664	2013-01-01
CPCI	G06F3/061	2013-01-01
CPCI	G06F3/0626	2013-01-01
CPCI	G06F3/0626	2013-01-01
CPCI	G06F3/061	2013-01-01
CPCA	G06F12/0246	2013-01-01
CPCA	G06F12/0246	2013-01-01

Abstract

<P>PROBLEM TO BE SOLVED: To provide a flash memory storage system for simulating rewritable disk drive. <P>SOLUTION: There is provided the flash memory storage system including a flash memory chip, a connector, and a controller. The flash memory chip has a plurality of physical blocks. The connector is arranged to be coupled to a host system. The controller is coupled to the flash memory chip and the connector. The controller configures a plurality of logical blocks and maps the logical blocks to a portion of the physical blocks. In addition, the controller identifies at least one rewritable disk command from the host system and writes data from the host system into the physical blocks mapped to at least one of the logical blocks according to the rewritable disc command. Thereby, a rewritable disc device is simulated by using the flash memory storage system. <P>COPYRIGHT: (C)2011,JPO&INPIT

Description

TECHNICAL-FIELD

The present invention generally relates to a flash memory storage system, and more particularly to a flash memory storage system for simulating a rewritable disk device, a flash memory controller, a computer system equipped with the flash memory storage system, and The present invention relates to a method for simulating a rewritable disc.

BACKGROUND-ART

A rewritable digital versatile disc (DVD = Digital Versatile Disc) drive is an optical disc storage device. Due to its large capacity, DVDs are gradually being replaced by conventional CDs (Compact Discs) in order to store high-quality video data, high-quality audio data, and large amounts of data. DVDs must be accessed via an optical pickup in the DVD drive. However, due to the large volume and high power consumption of DVD drive 2002 (shown in FIG. 1A), small computers (eg, netbooks) typically cannot be equipped with any DVD drive, which is very inconvenient for the user. In addition, since the DVD drive 2002 uses a mechanical optical pickup to read data from or burn data to the DVD 2004, the access speed of the DVD drive 2002 is very slow. Furthermore, the volume of DVD 2004 is large and it is inconvenient to carry.

The above problem is that if the DVD drive 2002 is replaced with a flash drive 2006 having a flash memory, the flash memory data non-volatility, low power consumption, small volume and non-mechanical structure can be stored in the storage medium (storage medium).

A Universal Serial Bus (USB) storage device that uses a flash memory as its storage medium is disclosed in Patent Document 1, among which a USB storage device simulates a CD-ROM device and a USB storage device Data stored in can be prevented from being deleted by an accident. However, the user cannot satisfy the requirement of data rewritability because the data cannot be rewritten by using the CD-ROM device simulated by the USB storage device.

CITATION-LIST

PATENT-LITERATURE

[1] US Pat. No. 7,111,121

DESCRIPTION-OF-EMBODIMENTS

Hereinafter, embodiments for carrying out the present invention will be described with reference to the drawings.

Generally speaking, a flash memory storage device (also called a flash memory storage system) includes a flash memory chip and a controller (also called a control circuit). The flash memory storage device is used together with the host system so that the host system can write data to and read data from the flash memory storage device. In addition, the flash memory storage device may also include embedded flash memory and software that is substantially executed on the host system and serves as a controller for the embedded flash memory.

FIG. 1B is a block diagram showing a host system and a flash memory storage device according to the first embodiment of the present invention.

In FIG. 1B, a host system 1000 includes a computer 1100 and an input / output (I / O) device 1106.

A computer 1100 includes a central processing unit (CPU) 1102, a main memory 1104, a system bus 1108, a data transmission interface 1110, and a main disk drive 1112. The computer 1100 can be a desktop computer or a notebook computer.

CPU 1102 interprets the mand and performs the main operations. Main memory 1104 is arranged for temporarily storing data and programs. A system bus 1108 is arranged for connection to various components of the host system 1000. A data transmission interface 1110 is arranged for connection to various peripheral devices. A main disk drive 1112 is arranged for installing an operation system (OS) 1122 and a burning program 1124. An OS 1122 is arranged to manage the host system 1000 hardware and software. In this embodiment, the OS 1122 may be Microsoft Windows® 7. However, the present invention is not limited to this. The burn program 1124 is arranged to provide a user interface that allows a user to burn data to the flash memory storage device 100 or erase data from the same.

As shown in FIG. 1C, the I / O device 1106 includes a mouse 1202, a keyboard 1204, a display 1206, and a printer 1208. It should be noted that the device shown in FIG. 1C is not intended to limit the I / O device

1106, and that the I / O device 1106 may further include other devices.

In this embodiment, the flash memory storage device 100 is coupled to other components of the host system 1000 via a data transmission interface 1110. The host system 1000 can write data to or read data from the flash memory storage device 100 through operations of the CPU 1102, the main memory 1104, the I / O device 1106, and the main disk drive 1112. . The flash memory storage device 100 can be a flash drive 1212, a memory card 1214, a solid state driver (SSD) 1216 or a plug-in storage device 1218, as shown in FIG. 1C. It should be noted that if the flash memory storage device 100 is implemented as a pluggable storage device (ie, a pluggable storage device 1218), the pluggable flash memory storage device 100 may be the motherboard (not shown) of the host system 1000. N) directly.

The flash memory storage device 100 includes a connector 102, a flash memory controller 104, and a flash memory chip 106.

A connector 102 is coupled to the flash memory controller 104 and arranged to be coupled to the host system 1000. In this embodiment, the connector 102 is a universal serial bus (USB) connector. However, the present invention is not limited thereto, and the connector 102 can also be a parallel advanced technology attachment (Parallel Advanced Technology Attachment = PATA) connector, a serial advanced technology attachment (Serial Advanced Technology Attachment = SATA) connector, IEEE (Institute of Electrical and Electronics Engineers) 1394 connector, Peripheral Component Interconnect (PCI) high-speed connector, Secure Digital interface connector, Memory Stick (MS) interface connector, Multimedia card (Multi Media Card = MMC) interface connector, Compact Flash (registered trademark) (Compact Flash = CF) interface connector, Integrated Device Electronics (Integrated Device) Electronic = IDE) connector or other suitable connector.

The flash memory controller 104 is arranged in order to execute a number of logic gates or control commands implemented in the form of hardware or firmware, and variously in the flash memory chip 106 according to the commands of the host system 1000. Data manipulation. The flash memory controller 104 includes a microprocessor unit 202, a host interface unit 206 and a flash memory interface unit 208.

The microprocessor unit 202 is a main control unit of the flash memory controller 104, and it cooperates with the memory management unit 204, the host interface unit 206, and the flash memory interface unit 208 to perform various operations of the flash memory storage device 100. Execute.

A memory management unit 204 is coupled to the microprocessor unit 202 and arranged to perform a data write mechanism and a block management mechanism according to this embodiment. In particular, the memory management unit 204 declares to the host system 1000 that the flash memory storage device 100 is a rewritable digital versatile disc (DVD) device, and establishes management information for the rewritable DVD device. At the same time, it responds to the host system 1000 to access the flash memory storage device 100 based on the management information.

For example, the memory management unit 204 may notify the host system 1000 that the flash memory storage device 100 is a disk device that conforms to the DVD + RW standard, DVD-RW standard, DVD-RAM standard, or Blu-Ray Disc (BD) standard. Declare. Thereafter, the memory management unit 204 identifies the rewritable DVD command received from the host system 1000, and operates the flash memory chip 106 according to the rewritable DVD command. For example, the flash memory chip 106 burns data received from the host system 1000 to the flash memory chip 106 in accordance with a rewritable DVD command. That is, when the user of the host system 1000 operates the flash memory storage device 100 via a rewritable DVD device burning program, the memory management unit 204 identifies the burning command and performs the operation on the flash memory chip 106 according to the burning command. Perform various data operations. Hereinafter, the operation of the memory management unit 204 will be described in detail with reference to the accompanying drawings.

In this embodiment, the memory management unit 204 is implemented in the flash memory controller 104 in firmware form. For example, the memory management unit 204 includes a number of control commands burned into a program memory (for example, Read Only Memory = ROM), and the program memory is fitted into the memory management unit 204. When the flash memory storage device 100 is in operation, the control commands of the memory management unit 204 are executed by the microprocessor unit 202, and according to an embodiment of the present invention, a burn mechanism (eg, data read, write, erase) and flash Achieve a memory management mechanism.

In another embodiment of the present invention, the control command of the memory management unit 204

is stored as a program code in a specific area (for example, a system area in a flash memory chip for storing system data in Xiamen). Similarly, the control commands of the memory management unit 204 are executed by the microprocessor unit 202 when the flash memory storage device 100 is in operation. In yet another embodiment, the memory management unit 204 is also implemented in the flash memory controller 104 as firmware.

A host interface unit 206 is coupled to the microprocessor unit 202 and receives and identifies commands and data from the host system 1000. That is, commands and data received from the host system 1000 are transmitted to the microprocessor unit 202 via the host interface unit 206. In this embodiment, the host interface unit 206 is a USB interface corresponding to the connector 102. However, the present invention is not limited thereto, and the host interface unit 206 may be a PATA interface, SATA interface, IEEE 1394 interface, PCI high-speed interface, SD interface, MS interface, MMC interface, CF interface, IED interface or There can be other suitable transmission interfaces.

A flash memory interface unit 208 is coupled to the microprocessor unit 202 and arranged to access the flash memory chip 106. That is, data written to the flash memory chip 106 is converted into a format that can be accepted by the flash memory chip 106 by the flash memory interface unit 208.

In the embodiment of the present invention, the flash memory controller 104 further includes a buffer memory 252, a power management unit 254, and an error correction unit 256.

A buffer memory 252 is coupled to the microprocessor unit 202 and temporarily stores data and commands received from the host system 1000 or data received from the flash memory chip 106.

A power management unit 254 is coupled to the microprocessor unit 202 and is arranged to control the power supply of the flash memory storage device 100.

An error correction unit 256 is coupled to the microprocessor unit 202 and arranged to perform an error correction procedure to ensure data accuracy. Specifically, when the memory management unit 204 receives a write command from the host system 1000, the error correction unit 256 generates an error checking and correcting (ECC) code for data corresponding to the write command. At the same time, the memory management unit 204 writes both the data and the corresponding ECC code to the flash memory chip 106. Subsequently, when the memory management unit 204 reads data from the flash memory chip 106, it also reads the ECC code corresponding to the data, and the error correction unit 256 performs an error correction procedure according to the ECC code.

A flash memory chip 106 is coupled to the flash memory controller 104 and arranged to store data. The flash memory chip 106 has a large number of substantial blocks. In the flash memory, each actual block is a minimum erase unit. That is, each actual block includes a minimum number of memory cells that are erased together. Each entity block has a number of entity addresses. In this embodiment, the entity address is an entity page. However, the present invention is not limited to this. Each entity page is a minimum program unit. That is, each physical page is a minimum unit for writing or reading data. Each physical page typically includes a user data area and a redundant area. The user data area stores user data, while the redundant area stores system data (eg, ECC code). In this embodiment, the flash memory chip 106 is a multi-level cell (Multi Level Cell = MLC) NAND flash memory chip. However, the present invention is not limited thereto, and the flash memory chip 106 may be a single level cell (Single Level Cell = SLC) NAND flash memory chip.

FIG. 2 is a block diagram showing a flash memory chip according to the embodiment of the present invention.

In FIG. 2, the flash memory chip 106 has actual blocks PBA (0) -PBA (N). Each entity block is a minimum erase unit. That is, each actual block includes a minimum number of memory cells that are erased together. Each entity block has 128 entity addresses (ie, entity pages). However, the invention is not so limited, and in other embodiments of the invention, each entity block may have 64, 256 or other suitable quantity of entity addresses.

The memory management unit 204 logically classifies the actual blocks PBA (0) -PBA (N) into a system area 302, a storage area 304, and a replacement area 306.

The logical blocks PBA (0) -PBA (S) logically belonging to the system area 302 are used to record system data such as the flash memory manufacturer and model number, and the physical address quantity corresponding to each physical block. used.

The entity blocks PBA (S + 1) -PBA (A) logically belonging to the storage area 304 (referenced in the center of FIG. 2) are used for storing data written by the host system 1000. That is, the flash memory storage

device 100 actually stores the data received from the host system 1000 using the actual block in the storage area 304. Specifically, the entity block PBA (S + 1) -PBA (A) in the storage area 304 is further classified into a data area 304a and a spare area 304b, of which the entity block PBA (S + 1) -PBA (D) is an entity block that includes data, and an entity block PBA (D + 1) -PBA (A) in the spare area 304b is an entity block that has not yet been used for data storage. More specifically, when the memory management unit 204 obtains a real block from the spare area 304b and writes the data to the obtained real block, the obtained real block is associated with the data area 304a and associated with the data area 304a. When the actual block is erased, the erased actual block is associated with the spare area 304b.

The entity blocks PBA (A + 1) -PBA (N) logically belonging to the replacement area 306 are replacement entity blocks. For example, when the flash memory chip 106 is manufactured, 4% of the actual blocks are reserved for the purpose of replacement. That is, when the entity block in the system area 302 and the storage area 304 is damaged, the entity block reserved in the replacement area 306 can be used to replace the damaged entity block (ie, bad block). Thus, if there is still a normal entity block in the replacement area 306 and the entity block is damaged, the memory management unit 204 obtains a normal entity block from the replacement area 306 and replace. If there is no normal entity block in the replacement area 306 and the entity block is damaged, the flash memory storage device 100 is declared as a write protected state and cannot be used as write data. .

It should be noted that during the operation of the flash memory storage device 100, the classification relationship that the entity blocks PBA (0) -PBA (N) are classified into the system area 302, the data area 304a, the spare area 304b, and the replacement area 306, It is changing dynamically. That is, when the memory management unit 204 writes data to one entity block (for example, the entity block PBA (D + 2)) originally belonging to the spare area 304b, the entity block is associated with the data area 304a. Alternatively, since a substantial block in the data area 304a (or spare area 304b) is damaged and replaced with a substantial block obtained from the replacement area 306, the original block belonging to the replacement area 306 is replaced with the data area 304a (or spare area). Associated with region 304b).

The memory management unit 204 provides logical blocks LBA (0) -LBA (L) because the flash memory storage device 100 stores the data received from the host system 1000 by using the physical block in the storage area 304 instead of using it. To be accessed by the host system 1000. Specifically, each logical block includes a large number of logical addresses, and each logical address is mapped to an actual address of the actual block. Accordingly, the OS 1122 of the host system 1000 simply accesses the data according to the logical address, while the memory management unit 204 actually accesses the data in the corresponding physical address. In this embodiment, the memory management unit 204 maps the logical blocks LBA (0) -LBA (L) to the physical blocks in the storage area 304, and also creates a logical block-physical block mapping table. Record the map relationship between logical blocks and entity blocks. In this embodiment, the data written by the host system 1000 is written to the actual blocks belonging to the storage area 304, and the host system 1000 cannot access the actual blocks in the system area 302 and the replacement area 306.

FIG. 3 is an explanatory diagram showing the configuration of the logic block according to the embodiment of the present invention.

In FIG. 3, the memory management unit 204 classifies the logical blocks LBA (0) -LBA (L) into a first logical unit 360 and a second logical unit 380, and the first logical unit 360 is logical block LBA (0)-. The second logical unit 380 includes LBA (G) and the logical block LBA (G + 1) -LBA (L).

In this embodiment, the memory management unit 204 conforms to a DVD disc (for example, a DVD + RW standard, a DVD-RW standard, a DVD-RAM standard, or a Blue-Ray Disc (BD) standard) in which the first logical unit 360 can be rewritten. The second logical unit 380 is arranged as a mass storage device (for example, a mass storage device conforming to the File Allocation Table 32 (FAT32) format or the NTFS format).

In particular, in this embodiment, the flash memory storage device 100 is connected to the host system 1000, and the CPU 1102 of the host system 1000 identifies that the flash memory storage device 100 is composed of a rewritable DVD device and a mass storage device. .

Specifically, when the CPU 1102 of the host system 1000 requests a device feature from the flash memory storage device 100, the memory management unit 204 determines that the flash memory storage device 100 is in the mass storage category and the first logical unit. Declare to host system 1000 that it has 360 and

second logical unit 380. Thereafter, when the CPU 1102 requests the device characteristics of the first logical unit 360 from the flash memory storage device 100 (for example, the CPU 1102 inquires about the device characteristics of the first logic unit 360 via the 0 × 12 command), the memory The management unit 204 declares to the host system 1000 that the first logical unit 360 is a rewritable DVD device (for example, the memory management unit 204 responds to the host system 1000 with a 0 × 05 message), and the CPU 1102 When requesting device features of the second logical unit 380 from the flash memory storage device 100, the memory management unit 204 declares that the second logical unit 380 is a mass storage device (eg, the memory management unit 204 is a host system). Respond to 1000 with a 0x00 message). Therefore, when the host system 1000 tries to burn to a rewritable DVD device (that is, the first logical unit 360) of the flash memory storage device 100, the memory management unit 204 performs the host command according to the disk command received from the host system 1000. Data from the system 1000 is written to the physical block mapped to the logical blocks LBA (0) -LBA (G) of the first logical unit 360. In addition, when the host system 1000 tries to save in the mass storage device (that is, the second logical unit 380) of the flash memory storage device 100, the data from the host system 1000 is stored in accordance with the disk command received from the host system 1000. Write to a physical block mapped to logical blocks LBA (G + 1) -LBA (L) of two logical units 380.

It should be noted that in this embodiment, the first logical unit 360 having a DVD format and the second logical unit 380 having a disk format, even if the memory management unit 204 can rewrite the logical blocks LBA (0) -LBA (L). However, the present invention is not limited to this. For example, in another embodiment of the present invention, the second logic unit 380 can also be arranged as a Compact Disk-Read Only Memory (DC-ROM). Alternatively, in yet another embodiment of the present invention, the second logical unit 380 can also be another rewritable DVD device, and the memory management unit 204 can be a rewritable DVD device by the second logical unit 380. (For example, a disk device conforming to the DVD + RW standard, the DVD-RW standard, the DVD-RAM standard, or the BD standard) is declared to the host system 1000, and the second logical unit 380 according to the rewritable DVD command. Access the data inside. Furthermore, in yet another embodiment of the present invention, the memory management unit 204 also arranges all the logical blocks LBA (0) -LBA (L) as a rewritable DVD device, or the logical block LBA (0) -LBA (L) is classified into more logical units, and these logical blocks are arranged as a rewritable DVD device, a mass storage device, or a compact disk read-only memory device, respectively.

Here, the memory management unit 204 of the flash memory controller 104 executes disk commands from the host system 1000 according to the normal operation pattern, and performs various data operations on the second logical unit 380. Also, when the flash memory storage device 100 receives a rewritable DVD command given to the first logical unit 360, the memory management unit 204 uses the first logical unit 360 in response to an appropriate message. Simulate a DVD. Hereinafter, the operation mechanism of the memory management unit 204 will be described in detail.

In the embodiment of the present invention, the CPU 1102 of the host system 1000 transmits a storage medium feature inquiry command (eg, a rewritable DVD command 0 × 46 command) to the flash memory storage device 100 to store the first logical unit 360. When obtaining media characteristics, the memory management unit 204 sends a single layer rewritable DVD message to the host system 1000. Accordingly, the CPU 1102 identifies the storage medium of the first logical unit 360 as a single layer rewritable DVD. It should be noted that in other embodiments of the present invention, the memory management unit 204 may transmit the corresponding storage medium feature according to the capacity of the first logic unit 360. For example, if the capacity of the first logical unit 360 is less than or equal to 4.7 GB, the memory management unit 204 sends a single layer rewritable DVD message to the host system 1000. If the capacity of the first logical unit 360 is greater than or equal to 8.5 GB, the memory management unit 204 sends a double layer rewritable DVD message to the host system 1000.

In the embodiment of the present invention, the CPU 1102 of the host system 1000 sends a capacity inquiry command (for example, a rewritable DVD command 0 × 25 command) to the flash memory storage device 100 to obtain the capacity of the first logical unit 360. When doing so, the memory management unit 204 sends the capacity value of the first logical unit 360 back to the host system 1000. For example, if the total capacity is 15 GB, the memory management unit 204 declares to the host system 1000 that the capacity of the first logical unit 360 is 15 GB. It should be noted that in other embodiments of the present invention, the memory management unit 204 may respond to the host system 1000 according to the

rewritable DVD design specification when the total capacity is supported. That is. For example, if the total capacity is 15 GB, the memory management unit 204 declares to the host system 1000 that the capacity of the first logical unit 360 is 4.7 GB, and the memory management unit 204 Declare it as a single layer rewritable DVD. Further, if the total capacity is 15 GB, the memory management unit 204 declares to the host system 1000 that the capacity of the first logical unit 360 is 8.5 GB. Declares that it is a dual layer rewritable DVD.

In an embodiment of the present invention, the memory management unit 204 establishes a rewritable DVD blank flag. Here, the rewritable DVD blank flag is one piece of management information of the rewritable DVD device, and records the accumulation state of the first logical unit 360. For example, if the flash memory storage device 100 was just manufactured and no data was written, the rewritable DVD blank flag is labeled as “blank status”. If data has already been written to the first logical unit 360 of the flash memory storage device 100, the rewritable DVD blank flag is labeled as “non-blank status”. In this embodiment, “0” in the rewritable DVD blank flag represents “blank state”, while “1” represents “non-blank state”. Specifically, when data is re-burned to the same storage address of a rewritable DVD, the entire rewritable DVD must first be erased. Thus, the host system 1000 clears the first logical unit 360 using a clear command (for example, a rewritable DVD command 0 × A1 command). However, since data is erased in the unit of the substantial block in the flash memory chip 106, the substantial block is erased only when the data is written into the substantial block. When the CPU 1102 sends a clear command to the flash memory storage device 100, the memory management unit 204 displays a rewritable DVD blank flag as “blank state” and responds to the host system 1000 that the clear command has already been executed. . It should be noted that the memory management unit 204 does not actually erase the physical block mapped to the first logical unit 360, so the speed at which the flash memory storage device 100 executes the clear command is a general rewrite. Possible DVD devices are much faster than the speed at which they do the same thing.

In addition, when the CPU 1102 sends a format command (eg, a rewritable DVD command 0 × 04 command) or a section closure command (eg, a rewritable DVD command 0 × 5B command) to the flash memory storage device 100, memory management Unit 204 marks the rewritable DVD blank flag as “non-blank”.

In particular, the memory management unit 204 is rewritable when the CPU 1102 sends a read command (eg, a rewritable DVD command 0 × 28 command) to the flash memory storage device 100 to read data from the first logical unit 360. It is determined whether the DVD blank flag is “blank”. If the rewritable DVD blank flag indicates “blank state”, the memory management unit 204 sends a null value directly to the host system 1000. If the rewritable DVD blank flag indicates “non-blank”, the memory management unit 204 is mapped to the first logical unit 360 for the host system 1000 according to the logical address indicated by the read command. Read data from entity block.

For example, in this embodiment, the memory management unit 204 stores the rewritable DVD blank flag in the actual block in the system area 302 (upper right of FIG. 2). However, the invention is not so limited, and in other embodiments of the invention, the rewritable DVD blank flag may also be added to other non-volatile memory (eg, flash memory controller 104). Can also be stored in a non-volatile memory located in the memory.

In an embodiment of the invention, the memory management unit 204 establishes a write capacity parameter. Here, the write capacity parameter is one of the management information of the rewritable DVD device, and the write capacity of the first logical unit 360 is recorded. Specifically, in the rewritable DVD device, data is sequentially burned to the storage address of the rewritable DVD. Accordingly, the host system 1000 uses the next writable inquiry command in the first logical unit 360 (for example, the 0 × 52 command of the rewritable DVD command) to write the next writable address in the first logical unit 360. Inquire about. When the CPU 1102 sends a write command (for example, a rewritable DVD command 0 × 2A command) to the flash memory storage device 100 and burns the data to the first logical unit 360, the memory management unit 204 sets the logical address to be written. The data is written to the physical block mapped to the corresponding logical block, and the write capacity parameter is updated according to the write data capacity. Also, when the CPU 1102 sends a format command or a clear command to the flash memory storage device 100, the memory management unit 204 initializes the write capacity parameter (for example, the write capacity parameter is reset to zero). In particular, when the CPU 1102 sends the next writable address inquiry command to the flash memory storage device 100, the memory management unit

204 responds to the host system 1000 with the next writable address according to the write capacity parameter.

It should be noted that when a general rewritable DVD device executes a section closing command on a rewritable DVD, a section closing occurs at the storage address of the rewritable DVD, and subsequent data is not burned after the section is closed. It is not to be. Thus, in this embodiment, when the CPU 1102 sends a section close command to the flash memory storage device 100, the memory management unit 204 still follows the size of the closed section, even if the host system 1000 does not write any data. Update the write capacity parameter.

Similarly, in this embodiment, the memory management unit 204 stores the write capacity parameter in the actual block in the system area 302 (upper right of FIG. 2). However, the present invention is not so limited, and in other embodiments of the present invention, the write capacity parameter may also be additionally placed in other non-volatile memory (eg, flash memory controller 104). Stored in a non-volatile memory).

In another embodiment of the present invention, the CPU 1102 of the host system 1000 sends a read command or a write command to the flash memory storage device 100 to read data from the first logical unit 360 or to send data to the first logical unit 360. When burning, the memory management unit 204 uses the conversion parameters to calculate the logical address and data capacity indicated by the read or write command. Specifically, the logical access unit has 2048 bytes in the DVD system, while the logical access unit has 512 bytes in the flash memory system. Therefore, the conversion parameter is set to 4 (= 2048/512). For example, when the CPU 1102 tries to read 512 units of data starting from the logical address "512", the memory management unit 204 starts from the physical address mapped to the logical address "2048". Read data.

In the embodiment of the present invention, when the CPU 1102 of the host system 1000 sends an error correction data inquiry command to the flash memory storage device 100, the memory management unit 204 sends a non-support message to the host system 1000. Send back. Specifically, the DVD system uses an appropriate error correction algorithm to ensure the accuracy of the data stored on the DVD. In this embodiment, the DVD system is not suitable for a flash memory system. Thus, the error correction unit 256 (FIG. 1B) ensures data accuracy using an Error Checking and Correcting (ECC) code suitable for flash memory systems. Therefore, when the CPU 1102 attempts to read error correction data from the first logical unit 360, it rejects the error correction data inquiry command from the host system 1000.

In the DVD system, the host system 1000 uses a minute / second / frame (Minute / Second / Frame = MSF) read command (for example, the 0 × B9 command of the rewritable DVD command) to set the parameter of minute / second / frame Can read data. In the embodiment of the present invention, when the CPU 1102 of the host system 1000 transmits the MSF read command to the flash memory storage device 100, the memory management unit 204 is indicated by the MSF read command through the following formula (1). Calculate logical address.

$$LBA = ((M \times 60 + S-2) \times 75 + F) \times 4 \quad (1)$$

Where LBA represents the logical address corresponding to the MSF read command in the flash memory, M represents the minute parameter of the MSF read command, S represents the second parameter of the MSF read command, and F represents the frame of the MSF read command. Represents a parameter. That is, one minute of data is stored in a 18000 (= 60 × 75 × 4) entity block, each entity address has 512 bytes, and one frame of data is stored in four entity addresses. In addition, each entity address has 512 bytes.

In the DVD system, the host system 1000 ejects a rewritable DVD in a rewritable DVD device using a disc eject command. In particular, after the rewritable DVD has been burned in the DVD system, the information stored in the DVD has changed, so the host system 1000 requests that the DVD be ejected, and the DVD device is rewritable again. After the insertion, the stored information on the DVD is read again. However, in the flash memory storage system, all stored information (eg, logical block-substance block map table) is temporarily stored in the buffer memory 252 (FIG. 1B). Thereby, the memory management unit 204 can obtain the latest information of the first logical unit 360 without having to unplug and re-plug the flash memory storage device 100. In an embodiment of the present invention, when the CPU 1102 of the host system 1000 sends a disk eject command to the flash memory storage device 100, the memory management unit 204 responds to the host system 1000 with a received message, but no action is taken. Absent. Subsequently, when the CPU 1102 queries the first logical unit 360, the memory management unit 204 returns a ready message directly to the host system 1000.

In an embodiment of the present invention, the memory management unit 204 sets a write protection flag to record whether the first logic unit 360 is in a “write protected state” or a “non-write protected state”. Specifically, the user can set the non-write protection flag of the first logical unit 360 by using a vendor (vendor manufacturer) command through an application program. For example, if the user does not want to store data in the first logical unit 360 and deletes it by mistake, the user can issue a vendor command via an application program installed in the host system 1000 to 204 can be requested to set the write protection flag as “write protected state”.

In another embodiment of the invention, when changing the write protection flag, the application program asks the user for an authentication password and sends the authentication password and the data to be set to the memory management unit 204. The memory management unit 204 then compares the authentication password entered by the user with the vendor password. The memory management unit 204 changes the write protection flag according to the user's request only when the authentication password entered by the user matches the vendor password. Here, the memory management unit 204 stores the write protection flag and the vendor password in the system area 302 (upper right of FIG. 2). However, the invention is not so limited, and in other embodiments of the invention, the write protection flag and the vendor password may also be added to other non-volatile memory (eg, flash memory controller 104). Can also be stored in a non-volatile memory that is allocated to the system.

In the embodiment of the present invention, when the CPU 1102 of the host system 1000 sends a clear command, a format command or a section close command to the flash memory storage device 100, the memory management unit 204 indicates that the write protection flag is “write protected state”. Determine if there is. When the write protection flag indicates “write protected state”, the memory management unit 204 sends an error message back to the host system 1000 or does not take any action. The error messages are “write protected (ie ASC code 0 × 27 and SenseKey code 0 × 07)”, “medium not present (ie ASC code 0 × 3A and SenseKey code 0 × 02)”, “ Medium is changing (ie ASC code 0 × 28 and SenseKey code 0 × 06) ”,“ Command not supported (ie ASC code 0 × 20 and SenseKey code 0 × 05) ”,“ Command parameter error (That is, ASC code 0 × 24 and SenseKey code 0 × 05) ”,“ Data write error (ie, ASC code 0 × 03 and SenseKey code 0 × 03) ”or“ Data write error (ie, ASC code 0 × 11 and SenseKey code 0 × 03) ”.

In addition to responding to the host system 1000 with one of the error messages described above, if the write protection flag indicates “write protection status”, in another embodiment of the present invention, the memory management unit 204 also The storage system characteristics can be changed directly to prevent the host system 1000 from issuing any write or clear command. For example, when the CPU 1102 of the host system 1000 sends a storage medium feature inquiry command to the flash memory storage device 100, the memory management unit 204 determines whether the write protection flag indicates “write protected status”. If the write protection flag indicates “write protected state”, the memory management unit 204 returns a single layer read-only DVD message or a dual layer read-only DVD message to the host system 1000 for the first logical unit 360. Notifies the host system 1000 that is a read-only DVD.

FIG. 4 is a flowchart (flow chart) showing a method for simulating a rewritable digital versatile disc (DVD) according to an embodiment of the present invention.

In FIG. 4, during step S401, the memory management unit 204 arranges a large number of logical blocks (for example, LBA (0) -LBA (L)) and maps the logical blocks to a part of the actual blocks of the flash memory chip 106. (FIG. 2).

In step S403, the memory management unit 204 classifies a part of the logical block into one logical unit (for example, the first logical unit 360) (FIG. 3). In step S405, the memory management unit 204 arranges the logical unit as a rewritable DVD storage medium.

Next, in step S407, the memory management unit 204 declares to the host system 1000 that the logical unit is a rewritable DVD device. In step S409, the memory management unit 204 identifies that the rewritable DVD command has been received from the host system 1000, and the data received from the host system 1000 in accordance with the rewritable DVD command is mapped to a logical block. To record. Since the mechanism for identifying the rewritable DVD command by the memory management unit 204 has been described above, it will not be described here.

FIG. 5 is a block diagram showing a flash memory storage system according to the second embodiment of the present invention.

In FIG. 5, a flash memory storage system 500 includes a card reader 502 and a removable memory card

504.

The card reader 502 includes a connector 512, a card reader controller 514, and a memory card connection interface unit 516.

A connector 512 is arranged to connect to the host system 1000 (FIG. 1B). In this embodiment, the connector 512 is a USB connector. However, the present invention is not so limited, and the connector 512 can also be a PATA connector, SATA connector, IEEE 1394 connector, PCI high-speed connector, IED connector or other suitable transmission connector. In this embodiment, a card reader 502 is coupled to the host system 1000 in a detachable manner via a connector 512. However, the invention is not so limited, and in other embodiments of the invention, the card reader 502 can be coupled to the host system 1000 in a fixed manner.

A card reader controller 514 is coupled to the connector 512 and is arranged to control the operation of the card reader 502. In particular, in this embodiment, the card reader controller 514 declares to the host system 1000 that the card reader 502 is a rewritable DVD device. For example, the card reader controller 514 declares to the host system 1000 that the card reader 502 is a disk device that conforms to the DVD + RW standard, the DVD-RW standard, the DVD-RAM standard, or the BD standard. Further, the card reader controller 514 identifies the rewritable DVD command received from the host system 1000 and operates the removable memory card 504 according to the rewritable DVD command. The mechanism for responding to a rewritable DVD command received from the host system 1000 has been described above and will not be described here.

A memory card articulation interface unit 516 is coupled to the card reader controller 514 and is arranged to couple to a removable memory card 504.

A removable memory card 504 is plugged into the card reader 502 in a separable manner. A removable memory card 50 includes a memory card controller 522 and a flash memory chip 524.

Since the memory card controller 522 arranges the logical blocks and maps the logical blocks to the physical blocks, the host system 1000 can access the flash memory chip 524 in the removable memory card 504 via the card reader 502.

Specifically, the card reader controller 514 declares to the host system 1000 that the removable memory card 504 is a rewritable DVD, and identifies a rewritable DVD command from the host system 1000, The memory card controller 522 executes the block management mechanism of the flash memory chip 524, and operates the flash memory chip 524 according to the command of the card reader controller 514.

In this embodiment, since the flash memory chip 524 is the same as the flash memory chip 106, the structure and function of the flash memory chip 524 are not described here.

As described above, the present invention has been disclosed by the embodiments. However, the present invention is not intended to limit the present invention, and is within the scope of the technical idea of the present invention so as to be easily understood by those skilled in the art. Therefore, the scope of patent protection should be defined based on the scope of claims and the equivalent area.

REFERENCE-SIGNS-LIST

100 flash memory storage
102 connector
104 Flash memory controller
106 Flash memory chip
202 microprocessor unit
204 Memory management unit
206 Host interface unit
208 Flash memory interface unit
252 buffer memory
254 Power management unit
256 error correction unit
302 System area
304 Storage area
304a Data area
304b Reserved area
306 Alternating area

360 First logical unit
380 Second logical unit
502 card reader
504 Removable memory card
512 connector
514 Card reader controller
516 Memory card connection interface unit
522 Memory card controller
524 flash memory chip
1000 host system
1100 computer
1102 Central processing unit
1104 Main memory
1106 Input / output devices
1108 System bus
1110 Data transmission interface
1112 Main disk drive
1122 Operation System (OS)
1124 Baking program
1202 mouse
1204 keyboard
1206 display
1208 printer
1212 Flash drive 1214 Memory card
1216 Solid State Driver (SSD)
1218 Insertion type memory device
2002 Digital Versatile Disc (DVD) drive
2004 Digital Versatile Disc (DVD)
2006 Flash drive
PBA (0) -PBA (N) entity block
LBA (0) -LBA (L) logical block

Claims

A flash memory chip having a large number of substance blocks;
A connector arranged to be coupled to a host system;
A controller coupled to the flash memory chip and the connector and arranged with a plurality of logic blocks, and mapping the logic blocks to a part of the physical blocks;
Wherein the controller is further arranged to identify at least one rewritable disk command from the host system and to send data from the host system according to the at least one rewritable disk command at least in the logical block. A flash memory storage system that writes to physical blocks partially mapped.

2. The flash memory according to claim 1, wherein the controller establishes at least one management information of the rewritable disk device and stores at least one management information of the rewritable disk device in one of the entity blocks. Storage system.

3. The flash memory storage according to claim 2, wherein the at least one management information of the rewritable disk device includes accumulation state information, and the accumulation state information records at least a partial accumulation state of the logical block. system.

3. The flash memory storage according to claim 2, wherein the at least one management information of the rewritable disk device includes a write capacity parameter, and the write capacity parameter records a write capacity of at least a part of the logical block. system.

A flash memory storage system,
A flash memory chip having a large number of substance blocks;

A connector arranged to be coupled to a host system;
A controller coupled to the flash memory chip and the connector and arranged with a plurality of logic blocks, and mapping the logic blocks to a part of the physical blocks;
Among them, one of the entity blocks stores at least one management information of a rewritable disk device, and the flash memory storage system arranges at least one management information of the controller and the rewritable disk device. A flash memory storage system that allows the host to identify at least a portion of the logical block as a rewritable storage medium of the rewritable disk device when coupled to the host by Memory storage system.

6. The controller according to claim 5, wherein the controller sends back a single layer rewritable DVD message or a dual layer rewritable DVD message to the host system when the controller receives a storage medium characteristic inquiry command from the host system. Flash memory storage system.
When the controller receives a capacity inquiry command from the host system, the controller sends a capacity value back to the host system;
6. The flash memory storage system according to claim 5, wherein the capacity value is smaller than or equal to the flash memory chip.

6. The flash memory storage system according to claim 5, further comprising: a nonvolatile memory, wherein at least one management information of the rewritable disk device is stored in the nonvolatile memory.

6. The flash memory storage according to claim 5, wherein at least one management information of the rewritable disk device includes storage state information, and the storage state information records at least a part of the storage state of the logical block. system.

10. The flash memory storage system according to claim 9, wherein when the controller receives a clear command from the host system, the controller marks the storage state information as a blank state.

10. The flash memory storage system according to claim 9, wherein when the controller receives a format command or a section close command from the host system, the storage status information is marked as a non-blank status.
When the controller receives a read command from the host system, the controller determines whether or not the accumulated state information indicates a blank state;
10. The flash memory storage system according to claim 9, wherein the controller sends back a null value to the host system when the accumulation state information indicates the blank state.

6. The flash memory storage according to claim 5, wherein the at least one management information of the rewritable disk device includes a write capacity parameter, and the write capacity parameter records at least a part of the write capacity of the logical block. system.

14. The flash memory storage system according to claim 13, wherein when the controller receives a format command or a clear command from the host system, the controller initializes the write capacity parameter.

The controller receives the write command from the host system, and the controller updates the write capacity parameter after writing the data to the physical block mapped to at least a part of the logical block.
13. The flash memory storage system according to 13.

14. The flash according to claim 13, wherein when the controller receives a next writable address inquiry command from the host system, the controller sends back the next writable address to the host system according to the write capacity parameter. Memory storage system.

The controller calculates a logical address and data capacity indicated by the read command or the write command using a conversion parameter when the controller receives a read command or a write command from the host system. 5. The flash memory storage system according to 5.

6. The flash memory storage system according to claim 5, wherein when the controller receives an error correction data inquiry command from the host system, the controller sends a non-support message back to the host system.
When the controller receives an MSF read command from the host system, the controller calculates a logical address corresponding to the MSF read command via Equation (1) below,
$LBA = ((M \times 60 + S-2) \times 75 + F) \times 4$ (1)
Among them, LBA represents the logical address corresponding to the MSF read command in the flash memory chip, M represents a minute parameter of the MSF read command, S represents a second parameter of the MSF read command, and F represents the 6. The flash memory storage system according

to claim 5, which represents a frame parameter of an MSF read command.

6. The flash memory storage system according to claim 5, wherein when the controller receives a disk ejection command from the host system, the controller sends back a received message or a prepared message to the host system.

The host system may store a rewritable storage medium of a disk device, a read-only storage medium of a compact disk read only memory (CD-ROM) device, or a rewritable storage medium of a rewritable disk device as another part of the logical block. 6. The flash memory storage system according to claim 5, wherein the flash memory storage system is identified as

The controller sets the write protection information in accordance with a command and writes the write protection information to one of the entity blocks, of which the write protection information indicates that the part of the logical block is in a write protection state or is not written 6. The flash memory storage system according to claim 5, which records whether it is in a protected state.

When the controller receives a clear command, a format command or a section close command from the host system, the controller determines whether the write protection information indicates the write protection state;

23. The flash memory storage system according to claim 22, wherein when the write protection information indicates the write protection state, the controller sends an error message back to the host system.

When the controller receives a storage medium characteristic inquiry command from the host system, the controller determines whether the write protection information indicates the write protection state;

23. The flash of claim 22, wherein the controller sends back a single layer read only DVD message or a dual layer read only DVD message to the host system when the write protection information indicates the write protection state. Memory storage system.

A flash memory controller for managing a large number of actual blocks of a flash memory chip, wherein the flash memory controller is

A microprocessor unit;

A flash memory interface unit coupled to the microprocessor unit and arranged to be coupled to the flash memory chip;

A host interface unit coupled to the microprocessor unit and arranged to be coupled to a host system;

A memory management unit coupled to the microprocessor unit, arranged in a number of logical blocks, and mapping the number of logical blocks to a portion of the number of physical blocks.

Among them, the memory management unit identifies at least one rewritable disk command from the host system, and the data from the host system is stored in the plurality of logical blocks according to the at least one rewritable disk command. A flash memory controller for writing to the multiple entity blocks mapped at least in part.

The memory management unit establishes at least one management information of a rewritable disk device, and stores the at least one management information of the rewritable disk device in one of the multiple actual blocks. 26. The flash memory controller according to claim 25.

27. The at least one management information of the rewritable disk device includes accumulation state information, and the accumulation state information records at least one accumulation state of the plurality of logical blocks. Flash memory controller.

27. The at least one management information of the rewritable disk device includes a write capacity parameter, and the write capacity parameter records at least one write capacity of the plurality of logical blocks. Flash memory controller.

A computer system,

A central processing unit (CPU),

A main memory coupled to the CPU;

Embedded flash memory storage device coupled to the CPU, of which the embedded flash memory storage device,

A flash memory chip having a large number of substance blocks;

A controller coupled to the flash memory chip, arranged in a plurality of logical blocks, and mapping the plurality of logical blocks to a part of the plurality of physical blocks.

Among them, the controller identifies at least one rewritable disk command from the CPU and, according to the rewritable disk command, maps the plurality of actual blocks to at least a part of the plurality of logical

blocks. A computer system that writes data to a computer.

The controller establishes at least one management information of the rewritable disk device and accumulates the at least one management information of the rewritable disk device in one of the plurality of actual blocks. 30. A computer system according to claim 29.

31. The at least one management information of the rewritable disk device includes accumulation state information, and the accumulation state information records an accumulation state of at least a part of the plurality of logical blocks. Computer system.

31. The computer according to claim 30, wherein the at least one management information of the rewritable disk device includes a write capacity parameter, and the write capacity parameter records a write capacity of at least a part of a plurality of logical blocks. system.

A computer system,

CPU and

A main memory coupled to the CPU;

A card reader coupled to the CPU, wherein the card reader includes a card reader controller, and the card reader controller is arranged to declare to the CPU that the card reader is a rewritable disk device A card reader,

A removable memory card plugged into the card reader, of which the removable memory card is

A flash memory chip having a large number of substance blocks;

A memory card controller coupled to the flash memory chip, arranged in a number of logical blocks, and mapping the number of logical blocks to a portion of the plurality of physical blocks,

Among them, the card reader controller identifies at least one rewritable disk command from the CPU and, according to the rewritable disk command, the plurality of the above-described many mapped to at least a part of the plurality of logical blocks. A computer system that writes data to physical blocks.

A flash memory storage system,

A card reader including a card reader controller, wherein the card reader controller is arranged to declare to the host system that the card reader is a rewritable disk device; and

A removable memory card plugged into the card reader, of which the removable memory card is

A flash memory chip having a large number of substance blocks;

A memory card controller coupled to the flash memory chip and arranged to be arranged in a plurality of logical blocks and mapping the plurality of logical blocks to a part of the plurality of physical blocks.

Among them, the card reader controller identifies at least one rewritable disk command from the host system and, according to the rewritable disk command, maps the at least some of the plurality of logical blocks described above. A flash memory storage system for writing data from the host system to a physical block of

A method of simulating a rewritable disk device by using a flash memory storage device, wherein the flash memory storage device has a flash memory chip and the flash memory chip has a number of substantial blocks The simulation method comprises:

Arranging a large number of logical blocks and mapping the large number of logical blocks to a part of the large number of actual blocks, wherein the large number of logical blocks have a large number of logical addresses; A plurality of entity blocks having a plurality of entity addresses, and mapping a relationship existing between the plurality of logical addresses and the plurality of entity addresses.

Storing at least one management information of the rewritable disk device in one of the multiple actual blocks;

Accessing the flash memory storage device in response to a host system in accordance with the at least one management information of the rewritable disk device;

Calculating a write logical address and an amount of data using conversion parameters according to a write command of at least one of the rewritable disk commands provided by the host system.

In accordance with the rewritable disk command, data is written to one of the many actual blocks, and when a write error occurs, the data is written to another one of the many actual blocks. 36. The simulation method according to claim 35.

36. The at least one management information of the rewritable disk device includes storage state information, and the storage state information records at least a part of the storage states of the plurality of logical blocks. Simulation method.

36. The at least one management information of the rewritable disk device includes a write capacity

parameter, and the write capacity parameter records a write capacity of at least a part of the plurality of logical blocks. Simulation method.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2011-123863
(P2011-123863A)
(43)公開日　平成23年6月23日(2011.6.23)

(51)Int.Cl.		F I			テーマコード（参考）
G06F 12/00	(2006.01)	G06F	12/00	514E	5B035
G06F 12/02	(2006.01)	G06F	12/02	570A	5B058
G06K 19/07	(2006.01)	G06F	12/00	597U	5B060
G06F 3/06	(2006.01)	G06K	19/00	N	5B065
G06F 3/08	(2006.01)	G06F	3/06	301Z	5B082

審査請求 有　請求項の数 38　OL　外国語出願　（全 26 頁）　最終頁に続く

(21)出願番号　特願2010-117729 (P2010-117729)
(22)出願日　平成22年5月21日 (2010.5.21)
(31)優先権主張番号　098142336
(32)優先日　平成21年12月10日 (2009.12.10)
(33)優先権主張国　台湾 (TW)

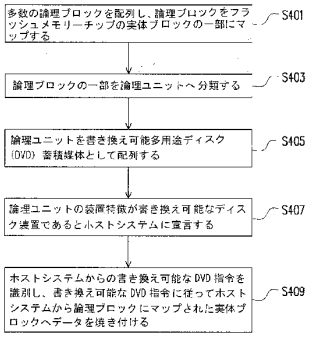
（特許庁注：以下のものは登録商標）
1．MEMORY　STICK

(71)出願人　509285089
群聯電子股▲ふん▼有限公司
台湾３５０苗栗縣竹南鎮群義路１號
(74)代理人　100147485
弁理士　杉村　憲司
(74)代理人　100143568
弁理士　英　貫
(74)代理人　100134577
弁理士　石川　雅章
(72)発明者　伍　漢維
台湾苗栗縣竹南鎮群義路１號
(72)発明者　黄　意翔
台湾新竹市香山区頂埔路７８號

最終頁に続く

(54)【発明の名称】書き換え可能ディスク装置をシミュレーションするためのフラッシュメモリー記憶システム、フラッシュメモリー制御器、コンピューターシステムおよびその方法

(57)【要約】
【課題】書き換え可能ディスク装置をシミュレートするフラッシュメモリー記憶システムを提供する。
【解決手段】フラッシュメモリーチップとコネクターと制御器とを含むフラッシュメモリー記憶システムが提供される。フラッシュメモリーチップが多数の実体ブロックを有する。コネクターがホストシステムに連結されるよう配列される。制御器がフラッシュメモリーチップおよびコネクターに連結される。制御器が多数の論理ブロックを配列するとともに、論理ブロックを実体ブロックの一部にマップする。また、制御器が更にホストシステムからの少なくとも１つの書き換え可能ディスク指令を識別するとともに、書き換え可能ディスク指令に従ってホストシステムからのデータを論理ブロックの少なくとも１つにマップされたへ実体ブロックに書き込む。従って、書き換え可能ディスクがフラッシュメモリー記憶システムを使用することによりシミュレーションされる。
【選択図】図４



【特許請求の範囲】
【請求項１】
多数の実体ブロックを有するフラッシュメモリーチップと、
ホストシステムに連結されるよう配列されるコネクターと、
前記フラッシュメモリーチップおよび前記コネクターに連結され、かつ多数の論理ブロックを配列されるとともに、前記論理ブロックを前記実体ブロックの一部にマップする制御器とを備え、
そのうち、前記制御器が更に前記ホストシステムからの少なくとも１つの書き換え可能ディスク指令を識別するために配列されるとともに、少なくとも１つの前記書き換え可能ディスク指令に従って前記ホストシステムからのデータを前記論理ブロックの少なくとも一部にマップされたへ実体ブロックに書き込む、フラッシュメモリー記憶システム。
【請求項２】
前記制御器が、書き換え可能なディスク装置の少なくとも１つの管理情報を確立するとともに、前記書き換え可能なディスク装置の少なくとも１つの管理情報を前記実体ブロックの１つに保存する請求項１記載のフラッシュメモリー記憶システム。
【請求項３】
前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が、蓄積状態情報を含むとともに、前記蓄積状態情報が、前記論理ブロックの少なくとも一部の蓄積状態を記録する請求項２記載のフラッシュメモリー記憶システム。
【請求項４】
前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が、書き込み容量パラメーターを含むとともに、前記書き込み容量パラメーターが、前記論理ブロックの少なくとも一部の書き込み容量を記録する請求項２記載のフラッシュメモリー記憶システム。
【請求項５】
フラッシュメモリー記憶システムであり、
多数の実体ブロックを有するフラッシュメモリーチップと、
ホストシステムに連結されるよう配列されるコネクターと、
前記フラッシュメモリーチップおよび前記コネクターに連結され、かつ多数の論理ブロックを配列されるとともに、前記論理ブロックを前記実体ブロックの一部にマップする制御器とを備え、
そのうち、前記実体ブロックの１つが書き換え可能なディスク装置の少なくとも１つの管理情報を保存するとともに、前記フラッシュメモリー記憶システムが前記制御器および前記書き換え可能なディスク装置の少なくとも１つの管理情報を配列することにより前記ホストに連結される時、前記フラッシュメモリー記憶システムが、前記ホストが前記論理ブロックの少なくとも一部を前記書き換え可能なディスク装置の書き換え可能な蓄積媒体として識別することを許すものである、フラッシュメモリー記憶システム。
【請求項６】
前記制御器は、前記制御器が前記ホストシステムから蓄積媒体特性照会指令を受信した時、単一層書き換え可能DVDメッセージまたは二重層書き換え可能DVDメッセージを前記ホストシステムへ送り返すものである請求項５記載のフラッシュメモリー記憶システム。
【請求項７】
前記制御器は、前記ホストシステムから容量照会指令を受信した時、容量値を前記ホストシステムへ送り返すものであり、
そのうち、前記容量値が前記フラッシュメモリーチップより小さいか等しいものである請求項５記載のフラッシュメモリー記憶システム。
【請求項８】
さらに、不揮発メモリーを含み、そのうち、前記書き換え可能なディスク装置の少なくとも１つの管理情報が、前記不揮発メモリー中に保存されるものである請求項５記載のフラッシュメモリー記憶システム。

【請求項９】
前記書き換え可能なディスク装置の少なくとも１つの管理情報が、蓄積状態情報を含むとともに、前記蓄積状態情報が前記論理ブロックの少なくとも一部の蓄積状態を記録するものである請求項５記載のフラッシュメモリー記憶システム。
【請求項１０】
前記制御器が前記ホストシステムからクリアー指令を受信する時、前記制御器が前記蓄積状態情報をブランク状態としてマークするものである請求項９記載のフラッシュメモリー記憶システム。
【請求項１１】
前記制御器が前記ホストシステムからフォーマット指令またはセクション閉鎖指令を受信する時、前記蓄積状態情報を非ブランク状態としてマークするものである請求項９記載のフラッシュメモリー記憶システム。
【請求項１２】
前記制御器が前記ホストシステムから読み出し指令を受信する時、前記制御器は、前記蓄積状態情報がブランク状態を指示するか否か判定するものであり、
そのうち、前記蓄積状態情報が前記ブランク状態を指示する時、前記制御器が前記ホストシステムへヌル値を送り返すものである請求項９記載のフラッシュメモリー記憶システム。
【請求項１３】
前記書き換え可能なディスク装置の少なくとも１つの管理情報が、書き込み容量パラメーターを含むとともに、前記書き込み容量パラメーターが前記論理ブロックの少なくとも一部の書き込み容量を記録するものである請求項５記載のフラッシュメモリー記憶システム。
【請求項１４】
前記制御器が前記ホストシステムからフォーマット指令またはクリアー指令を受信する時、前記制御器が前記書き込み容量パラメーターを初期化するものである請求項１３記載のフラッシュメモリー記憶システム。
【請求項１５】
前記制御器が前記ホストシステムから書き込み指令を受信するとともに、前記論理ブロックの少なくとも一部にマップされた前記実体ブロックへ前記データを書き込んだ後、前記制御器が前記書き込み容量パラメーターを更新する請求項１３記載のフラッシュメモリー記憶システム。
【請求項１６】
前記制御器が前記ホストシステムから次の書き込み可能アドレス照会指令を受信した時、前記制御器は、前記書き込み容量パラメーターに従って次の書き込み可能アドレスを前記ホストシステムへ送り返すものである請求項１３記載のフラッシュメモリー記憶システム。
【請求項１７】
前記制御器が前記ホストシステムから読み出し指令または書き込み指令を受信した時、前記制御器が、変換パラメーターを使用して前記読み出し指令または前記書き込み指令により指示された論理アドレスおよびデータ容量を計算する請求項５記載のフラッシュメモリー記憶システム。
【請求項１８】
前記制御器が前記ホストシステムからエラー訂正データ照会指令を受信した時、前記制御器が、非支援メッセージを前記ホストシステムへ送り返すものである請求項５記載のフラッシュメモリー記憶システム。
【請求項１９】
前記制御器が前記ホストシステムからMSF読み出し指令を受信した時、前記制御器が、下記する数式（１）を介して前記MSF読み出し指令に対応する論理アドレスを計算するものであり、

LBA＝（（M× 60＋S-2）× 75＋F）× 4…（1）

そのうち、LBAが前記フラッシュメモリーチップ中の前記MSF読み出し指令に対応する前記論理アドレスを表し、Mが前記MSF読み出し指令の分パラメーターを表し、Sが前記MSF読み出し指令の秒パラメーターを表し、Fが前記MSF読み出し指令のフレームパラメーターを表すものである、請求項５記載のフラッシュメモリー記憶システム。

【請求項２０】

前記制御器が前記ホストシステムからディスク排出指令を受信した時、前記制御器が、受信済みメッセージまたは準備済みメッセージを前記ホストシステムへ送り返すものである請求項５記載のフラッシュメモリー記憶システム。

【請求項２１】

前記ホストシステムが、ディスク装置の書き換え可能な蓄積媒体、コンパクトディスク読み出し専用メモリー（CD-ROM）装置の読み出し専用蓄積媒体または書き換え可能なディスク装置の書き換え可能な蓄積媒体を前記論理ブロックの別な部分として識別するものである請求項５記載のフラッシュメモリー記憶システム。

【請求項２２】

前記制御器が、指令に従って書き込み保護情報を設定するとともに、前記書き込み保護情報を前記実体ブロックの１つに書き込み、そのうち、前記書き込み保護情報が前記論理ブロックの前記一部が書き込み保護状態または非書き込み保護状態にあるかどうかを記録する請求項５記載のフラッシュメモリー記憶システム。

【請求項２３】

前記制御器が前記ホストシステムからクリアー指令、フォーマット指令またはセクション閉鎖指令を受信した時、前記制御器は、前記書き込み保護情報が前記書き込み保護状態を指示しているか否か判定し、

そのうち、前記書き込み保護情報が前記書き込み保護状態を指示している時、前記制御器が前記ホストシステムへエラーメッセージを送り返すものである請求項２２記載のフラッシュメモリー記憶システム。

【請求項２４】

前記制御器が前記ホストシステムから蓄積媒体特性照会指令を受信した時、前記制御器は、前記書き込み保護情報が前記書き込み保護状態を指示しているかどうか判定し、

そのうち、前記書き込み保護情報が前記書き込み保護状態を指示している時、前記制御器が前記ホストシステムへ単一層読み出し専用DVDメッセージまたは二重層読み出し専用DVDメッセージを送り返すものである請求項２２記載のフラッシュメモリー記憶システム。

【請求項２５】

フラッシュメモリー制御器であり、フラッシュメモリーチップの多数の実体ブロックを管理するためのものであって、前記フラッシュメモリー制御器が、

マイクロプロセッサーユニットと、

前記マイクロプロセッサーユニットに連結されるとともに、前記フラッシュメモリーチップに連結されるよう配列されるフラッシュメモリーインターフェースユニットと、

前記マイクロプロセッサーユニットに連結されるとともに、ホストシステムに連結されるよう配列されるホストインターフェースユニットと、

前記マイクロプロセッサーユニット連結されるとともに、多数の論理ブロックに配列され、かつ前記した多数の論理ブロックを前記した多数の実体ブロックの一部にマップするメモリー管理ユニットとを含み、

そのうち、前記メモリー管理ユニットが前記ホストシステムからの少なくとも１つの書き換え可能なディスク指令を識別するとともに、前記した少なくとも１つの書き換え可能なディスク指令に従って前記ホストシステムからのデータを前記した多数の論理ブロックの少なくとも一部にマップされた前記した多数の実体ブロックへ書き込むものである、フラッシュメモリー制御器。

【請求項２６】

前記メモリー管理ユニットが、書き換え可能なディスク装置の少なくとも１つの管理情

報を確立するとともに、前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報を前記した多数の実体ブロックの１つへ保存するものである請求項２５記載のフラッシュメモリー制御器。

【請求項２７】

前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が蓄積状態情報を含むとともに、前記蓄積状態情報が前記した多数の論理ブロックの少なくとも１つの蓄積状態を記録するものである、請求項２６記載のフラッシュメモリー制御器。

【請求項２８】

前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が書き込み容量パラメーターを含むとともに、前記書き込み容量パラメーターが前記した多数の論理ブロックの少なくとも１つの書き込み容量を記録するものである、請求項２６記載のフラッシュメモリー制御器。

【請求項２９】

コンピューターシステムであって、

中央処理ユニット（ CPU）と、

前記CPUに連結されたメインメモリーと、

前記CPUに連結された埋め込み式フラッシュメモリー記憶装置とを含み、そのうち、前記埋め込み式フラッシュメモリー記憶装置が、

多数の実体ブロックを有するフラッシュメモリーチップと、

前記フラッシュメモリーチップに連結されるとともに、多数の論理ブロックに配列され、かつ前記した多数の論理ブロックを前記した複数の実体ブロックの一部にマップする制御器とを含み、

そのうち、前記制御器が前記CPUから少なくとも１つの書き換え可能なディスク指令を識別するとともに、前記書き換え可能なディスク指令に従って、前記した多数の論理ブロックの少なくとも一部にマップされた前記した多数の実体ブロックへデータを書き込むものである、コンピューターシステム。

【請求項３０】

前記制御器が、書き換え可能なディスク装置の少なくとも１つの管理情報を確立するとともに、前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報を前記した多数の実体ブロックの１つに蓄積するものである請求項２９記載のコンピューターシステム。

【請求項３１】

前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が、蓄積状態情報を含むとともに、前記蓄積状態情報が前記した多数の論理ブロックの少なくとも一部分の蓄積状態を記録するものである請求項３０記載のコンピューターシステム。

【請求項３２】

前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が、書き込み容量パラメーターを含むとともに、前記書き込み容量パラメーターが多数の論理ブロックの少なくとも一部分の書き込み容量を記録するものである請求項３０記載のコンピューターシステム。

【請求項３３】

コンピューターシステムであって、

CPUと、

前記CPUに連結されたメインメモリーと、

前記CPUに連結されたカードリーダーであり、そのうち、前記カードリーダーがカードリーダー制御器を含むとともに、前記カードリーダー制御器は、前記カードリーダーが書き換え可能なディスク装置であるとCPUへ宣言するよう配列されるものである、カードリーダーと、

前記カードリーダーにプラグされた取外し可能なメモリーカードとを含み、そのうち、前記取外し可能なメモリーカードが、

Samsung Ex. 1002, Page 295 of 622

多数の実体ブロックを有するフラッシュメモリーチップと、
フラッシュメモリーチップに連結されるとともに、多数の論理ブロックに配列され、かつ前記した多数の論理ブロックを前記した複数の実体ブロックの一部にマップするメモリーカード制御器とを含み、
そのうち、前記カードリーダー制御器が前記CPUから少なくとも１つの書き換え可能なディスク指令を識別するとともに、前記書き換え可能なディスク指令に従って、前記した多数の論理ブロックの少なくとも一部にマップされた前記した多数の実体ブロックへデータを書き込む、コンピューターシステム。
【請求項３４】
フラッシュメモリー記憶システムであって、
カードリーダー制御器を含むカードリーダーであり、そのうち、前記カードリーダー制御器は、前記カードリーダーが書き換え可能なディスク装置であるとホストシステムへ宣言するよう配列されるものであるカードリーダーと、
前記カードリーダーにプラグされた取外し可能なメモリーカードとを含み、そのうち、前記取外し可能なメモリーカードが、
多数の実体ブロックを有するフラッシュメモリーチップと、
フラッシュメモリーチップに連結されるとともに、多数の論理ブロックに配列されるよう配列され、かつ前記した多数の論理ブロックを前記した複数の実体ブロックの一部にマップするメモリーカード制御器とを含み、
そのうち、前記カードリーダー制御器が前記ホストシステムから少なくとも１つの書き換え可能なディスク指令を識別するとともに、前記書き換え可能なディスク指令に従って、前記した多数の論理ブロックの少なくとも一部にマップされた前記した多数の実体ブロックへ前記ホストシステムからのデータを書き込む、フラッシュメモリー記憶システム。
【請求項３５】
フラッシュメモリー記憶装置を使用することにより書き換え可能なディスク装置をシミュレーションする方法であって、そのうち、前記フラッシュメモリー記憶装置が、フラッシュメモリーチップを有するとともに、前記フラッシュメモリーチップが多数の実体ブロックを有し、前記シミュレーション方法が、
多数の論理ブロックを配列するとともに、前記した多数の論理ブロックを前記した多数の実体ブロックの一部にマップするものであり、そのうち、前記した多数の論理ブロックが多数の論理アドレスを有し、前記した多数の実体ブロックが多数の実体アドレスを有するとともに、前記した多数の論理アドレスおよび前記した多数の実体アドレス間に存在する関係をマップすることと、
前記書き換え可能なディスク装置の少なくとも１つの管理情報を前記した多数の実体ブロックの１つに保存することと、
前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報に従って、ホストシステムへ応答して前記フラッシュメモリー記憶装置へアクセスすることと、
前記ホストシステムにより与えられる少なくとも１つの前記書換可能なディスク指令の１つの書き込み指令に従って、変換パラメーターを使用して書き込み論理アドレスおよびデータ量を計算することと、を含むシミュレーション方法。
【請求項３６】
前記書き換え可能なディスク指令に従ってデータが前記した多数の実体ブロックの１つに書き込まれるとともに、書き込みエラーが発生した時、前記データが前記した多数の実体ブロックの別な１つに書き込まれるものである請求項３５記載のシミュレーション方法。
【請求項３７】
前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が蓄積状態情報を含むとともに、前記蓄積状態情報が前記した多数の論理ブロックの少なくとも一部の蓄積状態を記録するものである請求項３５記載のシミュレーション方法。
【請求項３８】

前記書き換え可能なディスク装置の前記した少なくとも１つの管理情報が書き込み容量パラメーターを含むとともに、前記書き込み容量パラメーターが前記した多数の論理ブロックの少なくとも一部の書き込み容量を記録するものである請求項３５記載のシミュレーション方法。

【発明の詳細な説明】

【技術分野】

【０００１】

この発明は、一般的にフラッシュメモリー記憶システムに関し、特に、書き換え可能ディスク装置（Rewritable Disc Device）をシミュレーションするためのフラッシュメモリー記憶システム、フラッシュメモリー制御器、そのフラッシュメモリー記憶システムを装備したコンピューターシステムおよび書き換え可能ディスクをシミュレーションする方法に関する。

【背景技術】

【０００２】

書き換え可能なデジタル多用途ディスク（DVD = Digital Versatile Disc)ドライブは、光学ディスク記憶装置である。DVDは、その大容量のために、高画質ビデオデータ、高品質オーディオデータおよび大量データを保存するため、次第に従来のCD（Compact Disc）に置き換えられてきている。DVDは、DVDドライブの光学ピックアップを介してアクセスされなければならない。しかし、DVDドライブ2002（図１Ａに図示）の大きな体積および高い電力消費の故に、スモールコンピューター（例えば、ネットブック）は、通常、いかなるDVDドライブも装備できず、ユーザーにとって非常に不便である。また、DVDドライブ2002が機械的光学ピックアップを使用して、DVD 2004からデータ読み出し又はDVD 2004へデータを焼き付けるので、DVDドライブ2002のアクセス速度が非常に遅い。さらに、DVD 2004の体積が大きく、持ち運びに不便である。

【０００３】

上記の問題は、もしもDVDドライブ2002がフラッシュメモリー（Flash Memory）を有するフラッシュドライブ2006に置き換えられるならば、フラッシュメモリーのデータ非揮発性、低い電力消費、小さい体積および非機械的構造を蓄積媒体（storage medium）とする故に、解決可能である。

【０００４】

フラッシュメモリーをその蓄積媒体として使用するユニバーサルシリアルバス（Universal Serial Bus = USB）記憶装置が特許文献１に開示されており、そのうち、USB記憶装置がCD-ROM装置をシミュレートして、USB記憶装置に保存されているデータがアクシデントによって削除されることを防止できる。しかし、ユーザーは、USB記憶装置によりシミュレートされたCD-ROM装置を使用することによりデータを書き換えられないので、データ書き換え性の要求（requirement of data rewritability）を満足させることができない。

【先行技術文献】

【特許文献】

【０００５】

【特許文献１】米国特許第７，１１１，１２１号明細書

【発明の概要】

【発明が解決しようとする課題】

【０００６】

そこで、この発明の目的は、書き換え可能ディスク装置をシミュレートできるフラッシュメモリー記憶システムを提供することにある。

【０００７】

この発明の目的は、フラッシュメモリーチップを使用することにより書き換え可能蓄積媒体をシミュレーションできるフラッシュメモリー制御器を提供することにある。

【０００８】

この発明の目的は、書き換え可能ディスク指令に従って、フラッシュメモリー記憶装置にアクセスできるコンピューターシステムを提供することにある。

【０００９】

この発明の目的は、フラッシュメモリー記憶装置を使用することによって、書き換え可能ディスクをシミュレーションできる方法を提供することにある。

【課題を解決するための手段】

【００１０】

この発明の実施形態がフラッシュメモリーチップとコネクターと制御器とを含むフラッシュメモリー記憶システムを提供する。フラッシュメモリーチップが多数の実体ブロックを有する。コネクターがホストシステムに連結されるよう配列される。制御器がフラッシュメモリーチップおよびコネクターに連結される。制御器が多数の論理ブロックを配列するとともに、論理ブロックを実体ブロックの一部にマップする（map 「写像する」とも言う）。また、制御器が更にホストシステムからの少なくとも１つの書き換え可能ディスク指令を識別とともに、少なくとも１つの書き換え可能ディスク指令に従ってホストシステムからのデータを論理ブロックの少なくとも一部にマップされたへ実体ブロックに書き込む。

【００１１】

この発明がフラッシュメモリーチップとコネクターと制御器とを含むフラッシュメモリー記憶システムを提供する。フラッシュメモリーチップが多数の実体ブロックを有する。コネクターがホストシステムに連結されるよう配列される。制御器がフラッシュメモリーチップおよびコネクターに連結される。制御器が多数の論理ブロックを配列するとともに、論理ブロックを実体ブロックの一部にマップする。実体ブロックの１つが書き換え可能なディスク装置の少なくとも１つの管理情報を保存する。制御器および前記書き換え可能なディスク装置の少なくとも１つの管理情報を配列することにより、フラッシュメモリー記憶システムがホストに連結される時、ホストが論理ブロックの少なくとも一部を書き換え可能なディスク装置の書き換え可能な蓄積媒体として識別することを許すものである。

【００１２】

この発明は、フラッシュメモリーチップの多数の実体ブロックを管理するためのフラッシュメモリー制御器を提供する。フラッシュメモリー制御器がマイクロプロセッサーユニットとフラッシュメモリーインターフェースユニットとホストインターフェースユニットとメモリー管理ユニットとを含む。フラッシュメモリーインターフェースユニットがマイクロプロセッサーユニットに連結されるとともに、フラッシュメモリーチップに連結されるよう配列される。ホストインターフェースユニットがマイクロプロセッサーユニットに連結されるとともに、ホストシステムに連結されるよう配列される。メモリー管理ユニットがマイクロプロセッサーユニット連結される。メモリー管理ユニットが多数の論理ブロックに配列するとともに、論理ブロックを実体ブロックの一部にマップする。また、メモリー管理ユニットがホストシステムからの少なくとも１つの書き換え可能なディスク指令を識別するとともに、少なくとも１つの書き換え可能なディスク指令に従ってホストシステムからのデータを論理ブロックの少なくとも一部にマップされた実体ブロックへ書き込むものである。

【００１３】

この発明は、中央処理ユニット（CPU）とメインメモリーと埋め込み式フラッシュメモリー記憶装置とを含むコンピューターシステムを提供する。埋め込み式フラッシュメモリー記憶装置がCPUに連結されるとともに、フラッシュメモリーチップと制御器とを含み、そのうち、フラッシュメモリーチップが多数の実体ブロックを有し、かつ制御器がフラッシュメモリーチップに連結される。制御器が多数の論理ブロックに配列され、かつ論理ブロックを実体ブロックの一部にマップする。また、制御器がCPUから少なくとも１つの書き換え可能なディスク指令を識別するとともに、書き換え可能なディスク指令に従って、論理ブロックの少なくとも一部にマップされた実体ブロックへデータを書き込むものである。

【００１４】
この発明は、CPUとメインメモリーとカードリーダーと取外し可能なメモリーカードとを含むコンピューターシステムを提供する。カードリーダーがCPUに連結されるとともに、カードリーダー制御器を有し、そのうち、カードリーダー制御器がカードリーダーが書き換え可能なディスク装置であるとCPUへ宣言する。取外し可能なメモリーカードがカードリーダーにプラグされる。取外し可能なメモリーカードがフラッシュメモリーチップとメモリーカード制御器とを含み、そのうち、フラッシュメモリーチップが多数の実体ブロックを有し、かつメモリーカード制御器がフラッシュメモリーチップに連結される。メモリーカード制御器が多数の論理ブロックに配列されるとともに、多数の論理ブロックを実体ブロックの一部にマップする。また、メモリーカード制御器がCPUから少なくとも１つの書き換え可能なディスク指令を識別するとともに、書き換え可能なディスク指令に従って、論理ブロックの少なくとも一部にマップされた実体ブロックへデータを書き込む。
【００１５】
この発明は、カードリーダーと前記取外し可能なメモリーカードとを含むフラッシュメモリー記憶システムを提供する。カードリーダーがカードリーダー制御器を含むとともに、カードリーダー制御器がカードリーダーが書き換え可能なディスク装置であるとホストシステムへ宣言するよう配列される。取外し可能なメモリーカードがカードリーダーにプラグされる。取外し可能なメモリーカードがフラッシュメモリーチップとメモリーカード制御器とを含み、そのうち、フラッシュメモリーチップが多数の実体ブロックを有するとともに、メモリーカード制御器がフラッシュメモリーチップに連結される。メモリーカード制御器が多数の論理ブロックを配列するとともに、論理ブロックを実体ブロックの一部にマップする。カードリーダー制御器がホストシステムから少なくとも１つの書き換え可能なディスク指令を識別するとともに、書き換え可能なディスク指令に従って、論理ブロックの少なくとも一部にマップされた実体ブロックへホストシステムからのデータを書き込む。
【００１６】
この発明は、フラッシュメモリー記憶装置を使用することにより書き換え可能なディスク装置をシミュレーションする方法を提供する。フラッシュメモリー記憶装置がフラッシュメモリーチップを有し、そのうち、フラッシュメモリーチップが多数の実体ブロックを有する。この方法は、多数の論理ブロックを配列するとともに、論理ブロックを実体ブロックの一部にマップするものであり、そのうち、論理ブロックが多数の論理アドレスを有し、実体ブロックが多数の実体アドレスを有するとともに、論理アドレスおよび実体アドレス間に存在する関係をマップすることを含む。この発明は、また、書き換え可能なディスク装置の少なくとも１つの管理情報に従って、ホストシステムへ応答してフラッシュメモリー記憶装置へアクセスすることを含む。この発明は、更に、ホストシステムにより与えられる少なくとも１つの前記書換可能なディスク指令の１つの書き込み指令に従って、変換パラメーターを使用して書き込み論理アドレスおよびデータ量を計算することを含み、そのうち、書き込み論理アドレスが論理アドレスの１つである。
【００１７】
上記したように、この発明の実施形態に従ったフラッシュメモリー記憶装置、フラッシュメモリー制御器、コンピューターシステムおよび方法は、フラッシュメモリーチップを使用することにより書き換え可能なディスクをシミュレーションできる。
【発明の効果】
【００１８】
つまり、この発明の実施形態中、書き換え可能なDVD装置は、フラッシュメモリー記憶装置を使用することによりシミュレーションされるとともに、様々なデータ操作が様々な書き換え可能なDVD指令に従って、シミュレーションされた書き換え可能なDVD装置上で実施される。フラッシュメモリー記憶システムのアクセス速度が一般的な書き換え可能なDVD装置のそれよりも速いので、フラッシュメモリー記憶装置を使用することによりシミュレーションされた書き換え可能なDVD装置は、一般的な書き換え可能なDVD装置よりも速い

アクセス速度を有する。また、フラッシュメモリー記憶装置を使用することによりシミュレーションされた書き換え可能なDVD装置は、いかなる機械的な光学ピックアップも必要としない。従って、書き換え可能なDVD装置のサイズが大幅に削減される。それにより、フラッシュメモリー記憶装置を使用することによりシミュレーションされた書き換え可能なDVD装置は、小型ネットブック中に配置できる。

【図面の簡単な説明】

【００１９】

【図１Ａ】従来の書き換え可能ディスク装置およびフラッシュドライブを示す図である。

【図１Ｂ】この発明の第１の実施形態にかかるホストシステムおよびフラッシュメモリー記憶装置を示すブロック図である。

【図１Ｃ】この発明の実施形態にかかるコンピューター、入力／出力装置およびフラッシュメモリー記憶装置を示す説明図である。

【図２】この発明の実施形態にかかるフラッシュメモリーチップを示すブロック説明図である。

【図３】この発明の実施形態にかかる論理ブロックの構成を示す説明図である。

【図４】この発明の実施形態にかかる書き換え可能なデジタル多用途ディスク（ＤＶＤ）をシミュレーションする方法を示すフローチャート（流れ図）である。

【図５】この発明の第２の実施形態にかかるフラッシュメモリー記憶システムを示すブロック図である。

【発明を実施するための形態】

【００２０】

以下、この発明を実施するための形態を図面に基づいて説明する。

一般的に言って、フラッシュメモリー記憶装置（フラッシュメモリー記憶システムとも言う）は、フラッシュメモリーチップと制御器（制御回路とも言う）とを含む。フラッシュメモリー記憶装置は、ホストシステムと一緒に使用されて、ホストシステムがフラッシュメモリー記憶装置へデータを書き込んだりフラッシュメモリー記憶装置からデータを読み出したりできる。加えて、フラッシュメモリー記憶装置は、また、嵌め込み式（embeded）フラッシュメモリーと、ホストシステム上で実質的に実行されて嵌め込み式フラッシュメモリーの制御器として供されるソフトウェアとを含み得る。

【００２１】

図１Ｂは、この発明の第１の実施形態にかかるホストシステムおよびフラッシュメモリー記憶装置を示すブロック図である。

【００２２】

図１Ｂにおいて、ホストシステム1000がコンピューター1100と入力／出力（input/output=I/O）装置1106とを含む。

【００２３】

コンピューター1100が中央処理ユニット（CPU）1102とメインメモリー1104とシステムバス1108とデータ伝送インターフェース1110とメインディスクドライブ1112とを含む。コンピューター1100がディスクトップコンピューターまたはノートブックコンピューターで有り得る。

【００２４】

CPU 1102がマンドを解釈するとともに、主要な操作を実行する。メインメモリー1104が一時的にデータおよびプログラムを保存するために配列される。システムバス1108がホストシステム1000の多様なコンポーネントに接続するために配列される。データ伝送インターフェース1110が様々な周辺機器に接続するために配列される。メインディスクドライブ1112が操作システム（Operation System = OS）1122および焼き付けプログラム1124をインストールするために配置される。OS 1122がホストシステム1000のハードウェアおよびソフトウェアを管理するために配列される。この実施形態中、OS 1122がMicrosoft Windows（登録商標）7であり得る。しかし、この発明は、それに限定されるものではない。焼き付けプログラム1124は、ユーザーがデータをフラッシュメモリー記憶装置100に焼き付

ける又は同者（same）からデータを消去するようなユーザーインターフェースを提供するために配列される。

【００２５】

図１Ｃに示すように、I/O 装置1106がマウス1202とキーボード1204とディスプレイ1206とプリンター1208とを含む。注意すべきは、図１Ｃに示めされたデバイスは、I/O 装置1106を限定しようというものではないとともに、I/O 装置1106が更に他のデバイスを含み得ることである。

【００２６】

この実施形態中、フラッシュメモリー記憶装置100は、データ伝送インターフェース1110を介してホストシステム1000の他のコンポーネントに連結される。ホストシステム1000は、CPU 1102とメインメモリー1104とI/O 装置1106とメインディスクドライブ1112との操作を介してフラッシュメモリー記憶装置100へデータを書き込む又はフラッシュメモリー記憶装置100からデータを読み出すことができる。フラッシュメモリー記憶装置100は、図１Ｃに示すように、フラッシュドライブ1212、メモリーカード1214、固態ドライバー（Solid State Driver = SSD）1216または嵌め込み式記憶装置1218であることができる。注意すべきことは、もしフラッシュメモリー記憶装置100が嵌め込み式記憶装置（即ち、嵌め込み式記憶装置1218）として履行されるならば、嵌め込み式フラッシュメモリー記憶装置100は、ホストシステム1000のマザーボード（図示せず）に直接嵌め込まれることである。

【００２７】

フラッシュメモリー記憶装置100がコネクター102とフラッシュメモリー制御器104とフラッシュメモリーチップ106とを含む。

【００２８】

コネクター102がフラッシュメモリー制御器104に連結されるとともに、ホストシステム1000に連結されるよう配列される。この実施形態中、コネクター102がユニバーサルシリアルバス（Universal Serial Bus = USB）コネクターである。しかし、この発明は、それに限定されるものではないとともに、コネクター102は、また、パラレルアドバンスドテクノロジーアタッチメント（Parallel Advanced Technology Attachment = PATA）コネクター、シリアルアドバンスドテクノロジーアタッチメント（Serial Advanced Technology Attachment = SATA）コネクター、IEEE（Institute of Electrical and Electronics Engineers）1394コネクター、周辺部品相互連絡（Peripheral Component Interconnect = PCI）高速コネクター、安全デジタル（Secure Digital）インターフェースコネクター、メモリースティック（Memory Stick = MS）インターフェースコネクター、マルチメディアカード（Multi Media Card = MMC）インターフェースコネクター、コンパクトフラッシュ（登録商標）（Compact Flash = CF）インターフェースコネクター、総合デバイス電子（Integrated Device Electronic = IDE）コネクターまたは他の適合するコネクターで有り得る。

【００２９】

フラッシュメモリー制御器104は、ハードウェアまたはファームウェア（firmware）の形式で履行された多数の論理ゲート、あるいは制御指令を実行するために配列されるとともに、ホストシステム1000の指令に従ってフラッシュメモリーチップ106で様々なデータ操作を実施する。フラッシュメモリー制御器104がマイクロプロセッサーユニット202とホストインターフェースユニット206とフラッシュメモリーインターフェースユニット208を含む。

【００３０】

マイクロプロセッサーユニット202は、フラッシュメモリー制御器104のメイン制御ユニットであるとともに、それがメモリー管理ユニット204とホストインターフェースユニット206とフラッシュメモリーインターフェースユニット208と合作してフラッシュメモリー記憶装置100の様々な操作を実行する。

【００３１】

メモリー管理ユニット204がマイクロプロセッサーユニット202に連結されるとともに、この実施形態に従って、データ書き込みメカニズムおよびブロック管理メカニズムを実行するために配列される。とりわけ、メモリー管理ユニット204は、フラッシュメモリー記憶装置100が書き換え可能なデジタル多用途ディスク（Digital Versatile Disc = DVD)装置であるとホストシステム1000に宣言し、書き換え可能なDVD装置の管理情報を確立するとともに、管理情報に基づいてフラッシュメモリー記憶装置100へアクセスするようホストシステム1000へ応答する。

【００３２】

例えば、メモリー管理ユニット204は、フラッシュメモリー記憶装置100がDVD+RW規格、DVD-RW規格、DVD-RAM規格またはBlu-Ray Disc(BD)規格に適合するディスク装置であるとホストシステム1000に対して宣言する。その後、メモリー管理ユニット204がホストシステム1000から受信した書き換え可能なDVD指令を識別し、かつ書き換え可能なDVD指令に従ってフラッシュメモリーチップ106を操作する。例えば、フラッシュメモリーチップ106がホストシステム1000から受信したデータを書き換え可能なDVD指令に従ってフラッシュメモリーチップ106へ焼き付ける。つまり、ホストシステム1000のユーザーが書き換え可能なDVD装置の焼き付けプログラムを介してフラッシュメモリー記憶装置100を操作する時、メモリー管理ユニット204が焼き付け指令を識別し、かつ焼き付け指令に従ってフラッシュメモリーチップ106上で様々なデータ操作を実施する。以下、メモリー管理ユニット204の操作が添付図面とともに詳述される。

【００３３】

この実施形態中、メモリー管理ユニット204がファームウェア形式（firmware form)でフラッシュメモリー制御器104中で履行される。例えば、メモリー管理ユニット204がプログラムメモリー（例えば、Read Only Memory = ROM)に焼き付けられた多数の制御指令を含むとともに、プログラムメモリーがメモリー管理ユニット204中に嵌め込まれる。フラッシュメモリー記憶装置100が操作中である時、メモリー管理ユニット204の制御指令がマイクロプロセッサーユニット202によって実行され、この発明の実施形態に従って、焼き付けメカニズム（例えば、データの読み出し、書き込み、消去）およびフラッシュメモリー管理メカニズムを達成する。

【００３４】

この発明の他の実施形態中、メモリー管理ユニット204の制御指令がプログラムコードとして特定領域（例えば、システムデータを専門に保存するためのフラッシュメモリーチップ中のシステム領域）に保存される。同様に、メモリー管理ユニット204の制御指令は、フラッシュメモリー記憶装置100が操作中である時、マイクロプロセッサーユニット202によって実行される。また、更に他の実施形態中、メモリー管理ユニット204がまたファームウェアとしてフラッシュメモリー制御器104中で履行される。

【００３５】

ホストインターフェースユニット206がマイクロプロセッサーユニット202に連結されるとともに、ホストシステム1000からの指令およびデータを受信かつ識別する。つまり、ホストシステム1000から受信した指令およびデータは、ホストインターフェースユニット206を介してマイクロプロセッサーユニット202へ伝送される。この実施形態中、ホストインターフェースユニット206は、コネクター102に対応するUSBインターフェースである。しかし、この発明は、それに限定されるものではないとともに、ホストインターフェースユニット206は、PATAインターフェース、SATAインターフェース、IEEE 1394インターフェース、PCI高速インターフェース、SDインターフェース、MSインターフェース、MMCインターフェース、CFインターフェース、IEDインターフェースまたは他の適合する伝送インターフェースで有り得る。

【００３６】

フラッシュメモリーインターフェースユニット208がマイクロプロセッサーユニット202に連結され、かつフラッシュメモリーチップ106にアクセスするよう配列される。即ち、フラッシュメモリーチップ106へ書き込まれるデータは、フラッシュメモリーインターフ

ェースユニット208によりフラッシュメモリーチップ106が受け入れることのできるフォーマットに変換される。

【００３７】

この発明の実施形態中、フラッシュメモリー制御器104が更にバッファーメモリー252と電源管理ユニット254とエラー訂正ユニット256とを含む。

【００３８】

バッファーメモリー252がマイクロプロセッサーユニット202に連結されるとともに、ホストシステム1000から受信したデータおよび指令またはフラッシュメモリーチップ106から受信したデータを一時的に保存する。

【００３９】

電源管理ユニット254がマイクロプロセッサーユニット202に連結され、かつフラッシュメモリー記憶装置100の電源供給を制御するよう配列される。

【００４０】

エラー訂正ユニット256がマイクロプロセッサーユニット202に連結されるとともに、エラー訂正手続を実行するよう配列されてデータの正確性を確保する。具体的には、メモリー管理ユニット204がホストシステム1000から書き込み指令を受信した時、エラー訂正ユニット256が書き込み指令に対応するデータのためにエラー検査訂正（Error Checking and Correcting = ECC）コードを発生させるとともに、メモリー管理ユニット204がデータおよび対応するECCコードを共にフラッシュメモリーチップ106へ書き込む。続いて、メモリー管理ユニット204がフラッシュメモリーチップ106からデータを読み出す時、それは、データに対応するECCコードも読み出すとともに、エラー訂正ユニット256がECCコードに従ってエラー訂正手続を実行する。

【００４１】

フラッシュメモリーチップ106がフラッシュメモリー制御器104に連結され、かつデータを保存するよう配列される。フラッシュメモリーチップ106が多数の実体ブロックを有する。フラッシュメモリー中、各実体ブロックは、最小消去ユニットである。つまり、各実体ブロックが一緒に消去される最小数量のメモリーセルを含む。各実体ブロックが多数の実体アドレスを有する。この実施形態中、実体アドレスは、実体ページである。しかし、この発明は、それに限定されない。各実体ページは、最小プログラムユニットである。即ち、各実体ページは、データを書き込み又は読み出すための最小ユニットである。各実体ページは、通常、ユーザーデータ領域および冗長領域（redundant area）を含む。ユーザーデータ領域がユーザーデータを保存する一方、冗長領域がシステムデータ（例えば、ECCコード）を保存する。この実施形態中、フラッシュメモリーチップ106は、マルチレベルセル（Multi Level Cell = MLC）NANDフラッシュメモリーチップである。しかし、この発明は、それに限定されないとともに、フラッシュメモリーチップ106がシングルレベルセル（Single Level Cell = SLC）NANDフラッシュメモリーチップでも有り得る。

【００４２】

図２は、この発明の実施形態にかかるフラッシュメモリーチップを示すブロック説明図である。

【００４３】

図２において、フラッシュメモリーチップ106が実体ブロックPBA(0)-PBA(N)を有する。各実体ブロックが最小消去ユニットである。つまり、各実体ブロックが一緒に消去される最小数量のメモリーセルを含む。各実体ブロックが128の実体アドレス（即ち、実体ページ）を有する。しかし、この発明は、それに限定されないとともに、この発明の他の実施形態中、各実体ブロックが64, 256または他の適合する数量の実体アドレスを有することもあり得る。

【００４４】

メモリー管理ユニット204が実体ブロックPBA(0)-PBA(N)を論理的にシステム領域（system area）302と蓄積領域（storage area）304と交替領域（replacement area）306とに分類する。

【００４５】
論理的にシステム領域302に属する実体ブロックPBA(0)-PBA(S)が、フラッシュメモリーの製造業者および型番、各実体ブロックに対応する実体アドレスの数量のような、システムデータを記録するために使用される。
【００４６】
論理的に蓄積領域304（図２中央に符号）に属する実体ブロックPBA(S+1)-PBA(A)が、ホストシステム1000によって書き込まれたデータを保存するために使用される。つまり、フラッシュメモリー記憶装置100は、蓄積領域304中の実体ブロックを使用してホストシステム1000から受信したデータを実際に保存する。具体的には、蓄積領域304中の実体ブロックPBA(S+1)-PBA(A)が更にデータ領域304aと予備領域304bとに分類され、そのうち、データ領域304a中の実体ブロックPBA(S+1)-PBA(D)がデータを包含する実体ブロックであり、予備領域304b中の実体ブロックPBA(D+1)-PBA(A)がまだデータの保存に使用されていない実体ブロックである。さらに詳細には、メモリー管理ユニット204が予備領域304bから実体ブロックを得てデータを得られた実体ブロックへ書き込む時、得られた実体ブロックがデータ領域304aに連合され、かつデータ領域304aに連合された実体ブロックが消去された時、消去された実体ブロックは、予備領域304bに連合される。
【００４７】
論理的に交替領域306に属する実体ブロックPBA(A+1)-PBA(N)は、交替実体ブロックである。例えば、フラッシュメモリーチップ106が製造された時、その４％の実体ブロックが交替の目的で確保（reserved）されている。つまり、システム領域302および蓄積領域304中の実体ブロックが損傷した時、交替領域306中に確保された実体ブロックは、損傷した実体ブロック（即ち、悪いブロック）と置き換えるために使用できる。このように、もし未だ交替領域306中に正常な実体ブロックがあるとともに、実体ブロックが損傷された場合、メモリー管理ユニット204が交替領域306から正常な実体ブロックを手に入れて損傷した実体ブロックと置き換える。もしも交替領域306中に正常な実体ブロックがなく、かつ実体ブロックが損傷された場合、フラッシュメモリー記憶装置100は、書き込み保護（write protected）状態と宣言されるとともに、書き込みデータとして使用することができない。
【００４８】
注意すべきは、フラッシュメモリー記憶装置100の操作中、実体ブロックPBA(0)-PBA(N)がシステム領域302、データ領域304a、予備領域304bおよび交替領域306に分類されるという分類関係は、ダイナミックに変化することである。つまり、メモリー管理ユニット204が元は予備領域304bに属する１つの実体ブロック（例えば、実体ブロックPBA(D+2)）にデータを書き込む時、実体ブロックがデータ領域304aに連合される。あるいは、データ領域304a（又は予備領域304b）中の実体ブロックが損傷されたため、交替領域306から得られた実体ブロックによって置き換えられる時、元は交替領域306に属する実体ブロックがデータ領域304a（又は予備領域304b）に連合される。
【００４９】
フラッシュメモリー記憶装置100が蓄積領域304中の実体ブロックをかわるがわる使用してホストシステム1000から受信したデータを保存するので、メモリー管理ユニット204は、論理ブロックLBA(0)-LBA(L)を提供してホストシステム1000によりアクセスさせる。具体的には、各論理ブロックが多数の論理アドレスを含むとともに、各論理アドレスが実体ブロックの実体アドレスにマップされる。従って、ホストシステム1000のOS 1122が論理アドレスに従って単純にデータにアクセスする一方、メモリー管理ユニット204が対応する実体アドレス中のデータに実際にアクセスする。この実施形態中、メモリー管理ユニット204が論理ブロックLBA(0)-LBA(L)を蓄積領域304中の実体ブロックにマップするとともに、論理ブロック-実体ブロックマップテーブル（logical block-physical block mapping table）中に論理ブロックおよび実体ブロック間のマップ関係を記録する。この実施形態中、ホストシステム1000により書き込まれるデータが蓄積領域304に属する実体ブロックへ書き込まれるとともに、ホストシステム1000は、システム領域302および交替領域306

中の実体ブロックにアクセスできない。

【００５０】

図３は、この発明の実施形態にかかるロジックブロックの構成を示す説明図である。

【００５１】

図３において、メモリー管理ユニット204が論理ブロックLBA(0)-LBA(L)を第１論理ユニット360と第２論理ユニット380とに分類し、第１論理ユニット360が論理ブロックLBA(0)-LBA(G)を含み、第２論理ユニット380が論理ブロックLBA(G+1)-LBA(L)を含む。

【００５２】

この実施形態中、メモリー管理ユニット204が第１論理ユニット360を書き換え可能なDVDディスク（例えば、DVD+RW規格、DVD-RW規格、DVD-RAM規格またはBlue-Ray Disc(BD)規格に適合するディスク装置）として配列するとともに、第２論理ユニット380を大容量記憶装置（例えば、ファイル配置テーブル32（File Allocation Table 32 = FAT32）フォーマットまたはNTFSフォーマットに適合する大容量記憶装置）として配列する。

【００５３】

とりわけ、この実施形態中、フラッシュメモリー記憶装置100がホストシステム1000に連結されるとともに、ホストシステム1000のCPU 1102が、フラッシュメモリー記憶装置100が書き換え可能なDVD装置および大容量記憶装置からなると識別する。

【００５４】

具体的には、ホストシステム1000のCPU 1102がフラッシュメモリー記憶装置100からの装置特徴を要請する時、メモリー管理ユニット204は、フラッシュメモリー記憶装置100が大容量記憶カテゴリーであるとともに、第１論理ユニット360と第２論理ユニット380とを有することをホストシステム1000に宣言する。その後、CPU 1102がフラッシュメモリー記憶装置100から第１論理ユニット360の装置特徴を要請する（例えば、CPU 1102が0× 12指令を介して第１論理ユニット360の装置特徴について質問する）時、メモリー管理ユニット204は、第１論理ユニット360が書き換え可能なDVD装置であるとホストシステム1000に宣言する（例えば、メモリー管理ユニット204がホストシステム1000に0× 05メッセージで応答する）とともに、CPU 1102がフラッシュメモリー記憶装置100から第２論理ユニット380の装置特徴を要請する時、メモリー管理ユニット204は、第２論理ユニット380が大容量記憶装置であると宣言する（例えば、メモリー管理ユニット204がホストシステム1000に0× 00メッセージで応答する）。従って、ホストシステム1000がフラッシュメモリー記憶装置100の書き換え可能なDVD装置（即ち、第１論理ユニット360）に焼き付けようとする時、メモリー管理ユニット204は、ホストシステム1000から受信したディスク指令に従って、ホストシステム1000からのデータを第１論理ユニット360の論理ブロックLBA(0)-LBA(G)にマップされた実体ブロックへ書き込む。また、ホストシステム1000がフラッシュメモリー記憶装置100の大容量記憶装置（即ち、第２論理ユニット380）に保存しょうとする時、ホストシステム1000から受信したディスク指令に従って、ホストシステム1000からのデータを第２論理ユニット380の論理ブロックLBA(G+1)-LBA(L)にマップされた実体ブロックへ書き込む。

【００５５】

注意すべきは、この実施形態中、たとえメモリー管理ユニット204が論理ブロックLBA(0)-LBA(L)を書き換え可能なDVDフォーマットを有する第１論理ユニット360およびディスクフォーマットを有する第２論理ユニット380へ分類するとしても、この発明は、それに限定されるものではないことである。例えば、この発明の他の実施形態中、第２論理ユニット380は、また、コンパクトディスク読み出し専用メモリー（Compact Disk-Read Only Memory = DC-ROM）として配列され得る。あるいは、この発明の更に別の実施形態中、第２論理ユニット380は、また、他の書き換え可能なDVD装置であり得るとともに、メモリー管理ユニット204は、第２論理ユニット380が書き換え可能なDVD装置（例えば、例えば、DVD+RW規格、DVD-RW規格、DVD-RAM規格またはBD規格に適合するディスク装置）であるとホストシステム1000に宣言するとともに、書き換え可能なDVD指令に従って第２論理ユニット380中のデータにアクセスする。更に、この発明のなお別な実施形態中、メモリー管理ユニ

ット204は、また、全ての論理ブロックLBA(0)-LBA(L)を書き換え可能なDVD装置として配列する、あるいは、論理ブロックLBA(0)-LBA(L)をより多くの論理ユニットに分類するとともに、これらの論理ブロックをそれぞれ書き換え可能なDVD装置、大容量記憶装置またはコンパクトディスク読み出し専用メモリー装置として配列する。

【００５６】

ここで、フラッシュメモリー制御器104のメモリー管理ユニット204が通常操作パターンに従って、ホストシステム1000からのディスク指令を実行して、第２論理ユニット380上で様々なデータ操作を実施する。また、フラッシュメモリー記憶装置100が第１論理ユニット360に与えられた書き換え可能なDVD指令を受信する時、メモリー管理ユニット204が適切なメッセージに応答して第１論理ユニット360を使用する書き換え可能なDVDをシミュレートする。以下、メモリー管理ユニット204の操作メカニズムが詳細に記述される。

【００５７】

この発明の実施形態中、ホストシステム1000のCPU 1102が蓄積媒体特徴照会指令（例えば、書き換え可能なDVD指令の0× 46指令）をフラッシュメモリー記憶装置100へ送信して第１論理ユニット360の蓄積媒体特徴を入手する時、メモリー管理ユニット204は、単一層書き換え可能DVDメッセージをホストシステム1000へ送信する。従って、CPU 1102が第１論理ユニット360の蓄積媒体を単一層（single layer）書き換え可能DVDとして識別する。注意すべきことは、この発明の他の実施形態中、メモリー管理ユニット204が第１論理ユニット360の容量に従って対応する蓄積媒体特徴を送信することもあり得ることである。例えば、もし第１論理ユニット360の容量が4.7GBより小さいか等しい場合、メモリー管理ユニット204が単一層書き換え可能DVDメッセージをホストシステム1000へ送信する。もしも第１論理ユニット360の容量が8.5GBより大きいか等しい場合、メモリー管理ユニット204が二重層（double layer）書き換え可能DVDメッセージをホストシステム1000へ送信する。

【００５８】

この発明の実施形態中、ホストシステム1000のCPU 1102が容量照会指令（例えば、書き換え可能なDVD指令の0× 25指令）をフラッシュメモリー記憶装置100へ送信して第１論理ユニット360の容量を入手する時、メモリー管理ユニット204は、第１論理ユニット360の容量値をホストシステム1000へ送り返す。例えば、もし総容量が15GBであれば、メモリー管理ユニット204は、第１論理ユニット360の容量が15GBであるとホストシステム1000に宣言する。注意すべきことは、この発明の他の実施形態中、メモリー管理ユニット204は、総容量がサポート（支持）される時、書き換え可能なDVDの設計仕様に従ってホストシステム1000へ応答することもあり得ることである。例えば、メモリー管理ユニット204は、もし総容量が15GBであれば、第１論理ユニット360の容量が4.7GBであるとホストシステム1000に宣言するとともに、メモリー管理ユニット204は、第１論理ユニット360が単一層書き換え可能DVDであると宣言する。さらに、メモリー管理ユニット204は、もし総容量が15GBであれば、第１論理ユニット360の容量が8.5GBであるとホストシステム1000に宣言するとともに、メモリー管理ユニット204は、第１論理ユニット360が二重層書き換え可能DVDであると宣言する。

【００５９】

この発明の実施形態中、メモリー管理ユニット204は、書き換え可能DVDブランクフラッグ（blank flag）を確立する。ここで、書き換え可能DVDブランクフラッグは、書き換え可能DVD装置の管理情報の１つであり、かつ第１論理ユニット360の蓄積状態を記録する。例えば、もしもフラッシュメモリー記憶装置100が製造されたばかりで何のデータも書き込まれていない場合、書き換え可能DVDブランクフラッグは、「ブランク状態」（blank status）として標示される。もしデータが既にフラッシュメモリー記憶装置100の第１論理ユニット360へ書き込まれていれば、書き換え可能DVDブランクフラッグは、「非ブランク状態」（non-blank status）として標示される。この実施形態中、書き換え可能DVDブランクフラッグ中の「0」が「ブランク状態」を表す一方、「1」が「非ブランク状態」を表す。詳しくは、書き換え可能DVDの同一蓄積アドレスにデータが再焼き付けされる時、書

き換え可能DVD全体が先ず消去されなければならない。かくして、ホストシステム1000がクリアー指令（例えば、書き換え可能DVD 指令の0× A1指令）を使用して第１論理ユニット360をクリアーする。しかし、フラッシュメモリーチップ106中の実体ブロックのユニットにおいてデータが消去されるため、データがその実体ブロックへ書き込まれる時にだけ、実体ブロックが消去される。CPU 1102がクリアー指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204は、「ブランク状態」として書き換え可能DVDブランクフラッグを標示するとともに、クリアー指令が既に実行されたとホストシステム1000へ応答する。注意すべきは、メモリー管理ユニット204が第１論理ユニット360にマップされた実体ブロックを実際に消去する訳ではないので、フラッシュメモリー記憶装置100がクリアー指令を実行するための速度は、一般の書き換え可能DVD装置が同じ事を行う速度より格段に速いことである。

【００６０】

また、CPU 1102がフォーマット指令（例えば、書き換え可能DVD 指令の0× 04指令）またはセクション閉鎖指令（例えば、書き換え可能DVD 指令の0× 5B指令）をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204が書き換え可能DVDブランクフラッグを「非ブランク状態」として標示する。

【００６１】

とりわけ、CPU 1102が読み出し指令（例えば、書き換え可能DVD 指令の0× 28指令）をフラッシュメモリー記憶装置100へ送信して、第１論理ユニット360からデータを読み出す時、メモリー管理ユニット204は、書き換え可能DVDブランクフラッグが「ブランク状態」であるか否か判定する。もし書き換え可能DVDブランクフラッグが「ブランク状態」を標示している場合、メモリー管理ユニット204は、ヌル値（null value）をホストシステム1000へ直接送信する。もしも書き換え可能DVDブランクフラッグが「非ブランク状態」を標示している場合、メモリー管理ユニット204は、ホストシステム1000のために、読み出し指令により標示された論理アドレスに従って第１論理ユニット360にマップされた実体ブロックからデータを読み出す。

【００６２】

例えば、この実施形態中、メモリー管理ユニット204が書き換え可能DVDブランクフラッグをシステム領域302（図２右上）の実体ブロックへ保存する。しかし、この発明は、それに限定されるものではないとともに、この発明の他の実施形態中、書き換え可能DVDブランクフラッグは、また、他の非揮発メモリー（例えば、フラッシュメモリー制御器104中に追加的に配置された非揮発メモリー）へ保存することもできる。

【００６３】

この発明の実施形態中、メモリー管理ユニット204が書き込み容量パラメーターを確立する。ここで、書き込み容量パラメーターは、書き換え可能DVD装置の管理情報の１つであり、第１論理ユニット360の書き込み容量を記録する。詳細には、書き換え可能DVD装置中、データが書き換え可能DVDの蓄積アドレスへ順番に焼き付けられる。従って、ホストシステム1000は、第１論理ユニット360中の次の書き込み可能照会指令（例えば、書き換え可能DVD 指令の0× 52指令）を使用して、第１論理ユニット360中の次の書き込み可能アドレスについて照会する。CPU 1102が書き込み指令（例えば、書き換え可能DVD 指令の0× 2A指令）をフラッシュメモリー記憶装置100へ送信して、第１論理ユニット360へデータを焼き付ける時、メモリー管理ユニット204が書き込むべき論理アドレスに対応する論理ブロックにマップされた実体ブロックへデータを書き込むとともに、書き込みデータの容量に従って書き込み容量パラメーターを更新する。また、CPU 1102がフォーマット指令またはクリアー指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204が書き込み容量パラメーターを初期化する（例えば、書き込み容量パラメーターがゼロにリセットされる）。とりわけ、CPU 1102が次の書き込み可能アドレス照会指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204は、書き込み容量パラメーターに従って次の書き込み可能アドレスをホストシステム1000へ応答する。

【００６４】

特記すべきは、一般の書き換え可能DVD装置が書き換え可能DVD上でセクション閉鎖指令を実行する時、書き換え可能DVDの蓄積アドレスにおいてセクション閉鎖が発生するとともに、セクション閉鎖後に後続するデータが焼き付けられなければならないことである。従って、この実施形態中、CPU 1102がセクション閉鎖指令をフラッシュメモリー記憶装置100へ送信する時、たとえホストシステム1000が何のデータも書き込まなくても、メモリー管理ユニット204は、なお閉鎖セクションのサイズに従って書き込み容量パラメーターを更新する。

【００６５】

同様に、この実施形態中、メモリー管理ユニット204が書き込み容量パラメーターをシステム領域302（図２右上）の実体ブロックへ保存する。しかし、この発明は、それに限定されるものではないとともに、この発明の他の実施形態中、書き込み容量パラメーターは、また、他の非揮発メモリー（例えば、フラッシュメモリー制御器104中に追加的に配置された非揮発メモリー）へ保存することもできる。

【００６６】

この発明の他の実施形態中、ホストシステム1000のCPU 1102が読み出し指令または書き込み指令をフラッシュメモリー記憶装置100へ送信して、第１論理ユニット360からデータを読み出す又は第１論理ユニット360へデータを焼き付ける時、メモリー管理ユニット204が変換パラメーターを使用して、読み出し指令または書き込み指令により標示された論理アドレスおよびデータ容量を計算する。詳しくは、論理アクセスユニットがDVDシステム中に2048バイトを有する一方、論理アクセスユニットがフラッシュメモリーシステム中に512バイトを有する。従って、変換パラメーターが４（=2048/512）に設定される。例えば、CPU 1102が論理アドレス「512」から開始して512ユニットのデータを読み出そうとする時、メモリー管理ユニット204は、論理アドレス「2048」にマップされた実体アドレスから開始して2048ユニットのデータを読み出す。

【００６７】

この発明の実施形態中、ホストシステム1000のCPU 1102がエラー訂正データ照会指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204は、不支持メッセージ（non-support message）をホストシステム1000へ送り返す。詳細には、DVDシステムが適切なエラー訂正アルゴリズムを使用してDVDに保存されたデータの正確性を確保する。この実施形態中、DVDシステムは、フラッシュメモリーシステムに適していない。かくして、エラー訂正ユニット256（図１Ｂ）がフラッシュメモリーシステムには適したエラー検査訂正（Error Checking and Correcting = ECC）コードを使用してデータ正確性を確保する。従って、CPU 1102が第１論理ユニット360からエラー訂正データを読み出そうとする時、ホストシステム1000からのエラー訂正データ照会指令を拒絶する。

【００６８】

DVDシステム中、ホストシステム1000は、分／秒／フレーム（Minute/Second/Frame = MSF）読み出し指令（例えば、書き換え可能DVD 指令の0× B9指令）を使用して、分／秒／フレームのパラメーターでデータを読み出すことができる。この発明の実施形態中、ホストシステム1000のCPU 1102がMSF読み出し指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204は、下記の数式（１）を介してMSF読み出し指令により標示された論理アドレスを計算する。

$$LBA = ((M \times 60 + S - 2) \times 75 + F) \times 4 \quad \ldots (1)$$

【００６９】

式中、LBAが前記フラッシュメモリー中のMSF読み出し指令に対応する論理アドレスを表し、MがMSF読み出し指令の分パラメーターを表し、SがMSF読み出し指令の秒パラメーターを表し、FがMSF読み出し指令のフレームパラメーターを表すものである。つまり、１分のデータが18000（= 60× 75× 4）の実体ブロック中に保存されるとともに、各実体アドレスが512バイトを有し、かつ１フレームのデータが４つの実体アドレス中に保存されるとともに、各実体アドレスが512バイトを有する。

【００７０】

DVDシステム中、ホストシステム1000がディスク排出（eject）指令を使用して書き換え可能なDVD装置中の書き換え可能なDVDを排出する。とりわけ、DVDシステム中、書き換え可能なDVDの焼き付けが完了した後、DVDの蓄積情報が変化しているので、ホストシステム1000がDVDを排出するよう要請するとともに、再びDVDが書き換え可能なDVD装置に挿入された後、DVDの蓄積情報を再読み出しする。しかし、フラッシュメモリー記憶システム中、全ての蓄積情報（例えば、論理ブロック-実体ブロックマップテーブル）が一時的にバッファーメモリー252（図１Ｂ）中に保存される。これにより、メモリー管理ユニット204は、フラッシュメモリー記憶装置100に対してプラグを抜いたり差し込んだりする（unplug and re-plug）必要なしに、第１論理ユニット360の最新情報を入手できる。この発明の実施形態中、ホストシステム1000のCPU 1102がディスク排出指令をフラッシュメモリー記憶装置100へ送信する時、メモリー管理ユニット204は、受信済みメッセージでホストシステム1000に応答するが、いかなる動作も起こさない。続いてCPU 1102が第１論理ユニット360を照会する時、メモリー管理ユニット204は、準備完了（ready）メッセージをホストシステム1000へ直接返送する。

【００７１】

この発明の実施形態中、メモリー管理ユニット204は、第１論理ユニット360が「書き込み保護状態」または「非書き込み保護状態」にあるかどうかを記録するために、書き込み保護フラッグを設定する。具体的には、ユーザーが応用プログラムを介してベンダー（vendor　製造業者）指令を使用することにより第１論理ユニット360の非書き込み保護フラッグを設定できる。例えば、もしもユーザーがデータを第１論理ユニット360に保存したくなくて誤って削除した場合、ユーザーは、ホストシステム1000にインストールされた応用プログラムを介してベンダー指令を発行することにより、メモリー管理ユニット204に「書き込み保護状態」として書き込み保護フラッグを設定するよう要請できる。

【００７２】

この発明の別の実施形態中、書き込み保護フラッグを変更する時、応用プログラムがユーザーに認証パスワードを尋ねるとともに、認証パスワードおよびセットされるデータをメモリー管理ユニット204へ送信する。そして、メモリー管理ユニット204がユーザーにより入力された認証パスワードをベンダーパスワードと比較する。メモリー管理ユニット204は、ユーザーにより入力された認証パスワードがベンダーパスワードと一致する時のみ、ユーザーの要求に従って書き込み保護フラッグを変更する。ここで、メモリー管理ユニット204は、書き込み保護フラッグおよびベンダーパスワードをシステム領域302（図２右上）に保存する。しかし、この発明は、それに限定されるものではないとともに、この発明の他の実施形態中、書き込み保護フラッグおよびベンダーパスワードは、また、他の非揮発メモリー（例えば、フラッシュメモリー制御器104中に追加的に配置された非揮発メモリー）へ保存することもできる。

【００７３】

この発明の実施形態中、ホストシステム1000のCPU 1102がクリアー指令、フォーマット指令またはセクション閉鎖指令をフラッシュメモリー記憶装置100に送信する時、メモリー管理ユニット204は、書き込み保護フラッグが「書き込み保護状態」であるかどうか判定する。書き込み保護フラッグが「書き込み保護状態」を標示している時、メモリー管理ユニット204は、エラーメッセージをホストシステム1000へ送り返すか如何なる動作も起こさない。エラーメッセージは、「書き込み保護されている（即ち、ASCコード0× 27およびSenseKeyコード0× 07）」、「媒体が存在しない（即ち、ASCコード0× 3AおよびSenseKeyコード0× 02）」、「媒体が変更中（即ち、ASCコード0× 28およびSenseKeyコード0× 06）」、「指令が支持されてない（即ち、ASCコード0× 20およびSenseKeyコード0× 05）」、「指令パラメーターエラー（即ち、ASCコード0× 24およびSenseKeyコード0× 05）」、「データ書き込みエラー（即ち、ASCコード0× 03およびSenseKeyコード0× 03）」または「データ書き込みエラー（即ち、ASCコード0× 11およびSenseKeyコード0× 03）」で有り得る。

【００７４】

もし書き込み保護フラッグが「書き込み保護状態」を標示していれば、上記したエラーメッセージの１つでホストシステム1000へ応答する以外に、この発明の別の実施形態中、メモリー管理ユニット204は、また、蓄積媒体特徴を直接に変更してホストシステム1000がいかなる書き込み又はクリアー指令を発行することを防止できる。例えば、ホストシステム1000のCPU 1102が蓄積媒体特徴照会指令をフラッシュメモリー記憶装置100に送る時、メモリー管理ユニット204は、書き込み保護フラッグが「書き込み保護状態」を標示しているかどうか判定する。もし書き込み保護フラッグが「書き込み保護状態」を標示していれば、メモリー管理ユニット204は、単一層読み出し専用DVDメッセージまたは二重層読み出し専用DVDメッセージをホストシステム1000へ返送して、第１論理ユニット360が読み出し専用DVDであることをホストシステム1000に通知する。

【００７５】

図４は、この発明の実施形態にかかる書き換え可能なデジタル多用途ディスク（ＤＶＤ）をシミュレーションする方法を示すフローチャート（流れ図）である。

【００７６】

図４において、ステップS401中、メモリー管理ユニット204が多数の論理ブロック（例えば、LBA(0)-LBA(L)）を配列するとともに、論理ブロックをフラッシュメモリーチップ106の実体ブロックの一部にマップする（図２）。

【００７７】

そして、ステップS403中、メモリー管理ユニット204が論理ブロックの一部を１つの論理ユニット（例えば、第１論理ユニット360）に分類する（図３）。ステップS405中、メモリー管理ユニット204が論理ユニットを書き換え可能なDVD蓄積媒体として配列する。

【００７８】

次に、ステップS407中、メモリー管理ユニット204は、論理ユニットが書き換え可能なDVD装置であるとホストシステム1000に宣言する。ステップS409中、メモリー管理ユニット204がホストシステム1000から書き換え可能なDVD指令を受信したことを識別するとともに、書き換え可能なDVD指令に従ってホストシステム1000から受信したデータを論理ブロックにマップされた実体ブロック中へ記録する。メモリー管理ユニット204が書き換え可能なDVD指令を識別するメカニズムは、上述したから、ここでは記述しない。

【００７９】

図５は、この発明の第２の実施形態にかかるフラッシュメモリー記憶システムを示すブロック図である。

【００８０】

図５において、フラッシュメモリー記憶システム500がカードリーダー502と取外し可能なメモリーカード504とを含む。

【００８１】

カードリーダー502がコネクター512とカードリーダー制御器514とメモリーカード連接インターフェースユニット516とを含む。

【００８２】

コネクター512がホストシステム1000（図１Ｂ）に連結するよう配列される。この実施形態中、コネクター512がUSBコネクターである。しかし、この発明は、それに限定されるものではないとともに、コネクター512は、また、PATAコネクター、SATAコネクター、IEEE 1394コネクター、PCI高速コネクター、IEDコネクターまたは他の適合する伝送コネクターで有り得る。この実施形態中、カードリーダー502がコネクター512を介して分離方式で（in a detachable manner）ホストシステム1000に連結される。しかし、この発明は、それに限定されるものではないとともに、この発明の他の実施形態中、カードリーダー502は、固定方式で（in fixed manner）ホストシステム1000に連結され得る。

【００８３】

カードリーダー制御器514がコネクター512に連結されるとともに、カードリーダー502の操作を制御するよう配列される。とりわけ、この実施形態中、カードリーダー制御器514は、カードリーダー502が書き換え可能なDVD装置であるとホストシステム1000に対して

宣言する。例えば、カードリーダー制御器514は、カードリーダー502がDVD+RW規格、DVD-RW規格、DVD-RAM規格またはBD規格に適合するディスク装置であるとホストシステム1000に対して宣言する。また、カードリーダー制御器514は、ホストシステム1000から受信した書き換え可能なDVD指令を識別し、かつ書き換え可能なDVD指令に従って取外し可能なメモリーカード504を操作する。ホストシステム1000から受信した書き換え可能なDVD指令に応答するメカニズムは、上述したので、ここでは記述しない。

【００８４】

メモリーカード連接インターフェースユニット516がカードリーダー制御器514に連結されるとともに、取外し可能なメモリーカード504に連結するよう配列される。

【００８５】

取外し可能なメモリーカード504が分離可能な方式でカードリーダー502にプラグされる。取外し可能なメモリーカード50がメモリーカード制御器522とフラッシュメモリーチップ524とを含む。

【００８６】

メモリーカード制御器522が論理ブロックを配列するとともに、論理ブロックを実体ブロックにマップするので、ホストシステム1000がカードリーダー502を介して取外し可能なメモリーカード504中のフラッシュメモリーチップ524にアクセスできる。

【００８７】

具体的には、カードリーダー制御器514は、取外し可能なメモリーカード504が書き換え可能なDVDであるとホストシステム1000に対して宣言するとともに、ホストシステム1000からの書き換え可能なDVD指令を識別し、かつメモリーカード制御器522がフラッシュメモリーチップ524のブロック管理メカニズムを実行し、かつカードリーダー制御器514の指令に従ってフラッシュメモリーチップ524を操作する。

【００８８】

この実施形態中、フラッシュメモリーチップ524がフラッシュメモリーチップ106と同一であるから、フラッシュメモリーチップ524の構造および機能は、ここでは記述しない。

【００８９】

以上のごとく、この発明を実施形態により開示したが、もとより、この発明を限定するためのものではなく、当業者であれば容易に理解できるように、この発明の技術思想の範囲内において、適当な変更ならびに修正が当然なされうるものであるから、その特許権保護の範囲は、特許請求の範囲および、それと均等な領域を基準として定めなければならない。

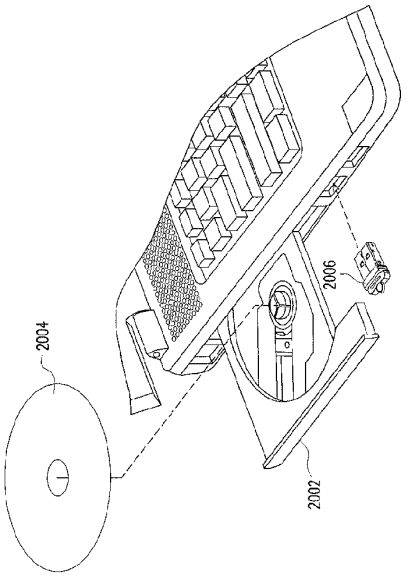
【符号の説明】

【００９０】

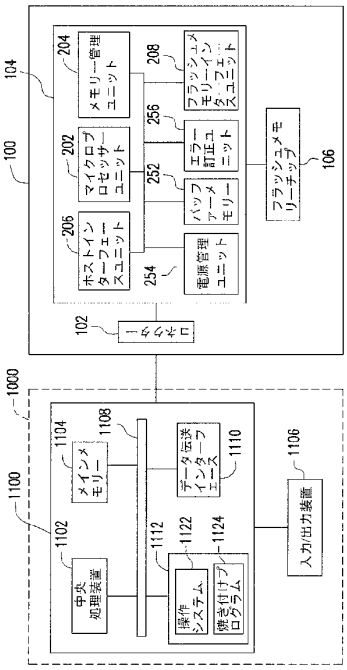
100 フラッシュメモリー記憶装置
102 コネクター
104 フラッシュメモリー制御器
106 フラッシュメモリーチップ
202 マイクロプロセッサーユニット
204 メモリー管理ユニット
206 ホストインターフェースユニット
208 フラッシュメモリーインターフェースユニット
252 バッファーメモリー
254 電源管理ユニット
256 エラー訂正ユニット
302 システム領域
304 蓄積領域
304a データ領域
304b 予備領域
306 交替領域

360 第１論理ユニット
380 第２論理ユニット
502 カードリーダー
504 取外し可能なメモリーカード
512 コネクター
514 カードリーダー制御器
516 メモリーカード連接インターフェースユニット
522 メモリーカード制御器
524 フラッシュメモリーチップ
1000 ホストシステム
1100 コンピューター
1102 中央処理ユニット
1104 メインメモリー
1106 入力／出力装置
1108 システムバス
1110 データ伝送インターフェース
1112 メインディスクドライブ
1122 操作システム（Operation System = OS）
1124 焼き付けプログラム
1202 マウス
1204 キーボード
1206 ディスプレイ
1208 プリンター
1212 フラッシュドライブ1214 メモリーカード
1216 固態ドライバー（Solid State Driver = SSD）
1218 嵌め込み式記憶装置
2002 デジタル多用途ディスク（Digital Versatile Disc = DVD）ドライブ
2004 デジタル多用途ディスク（Digital Versatile Disc = DVD）
2006 フラッシュドライブ
PBA(0)-PBA(N) 実体ブロック
LBA(0)-LBA(L) 論理ブロック

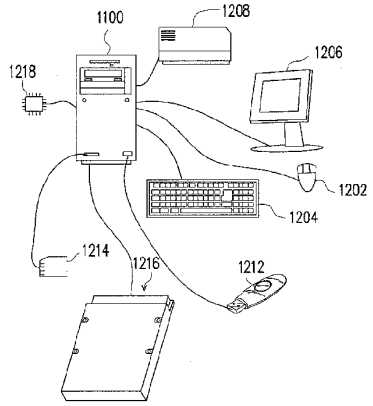
【図１Ａ】



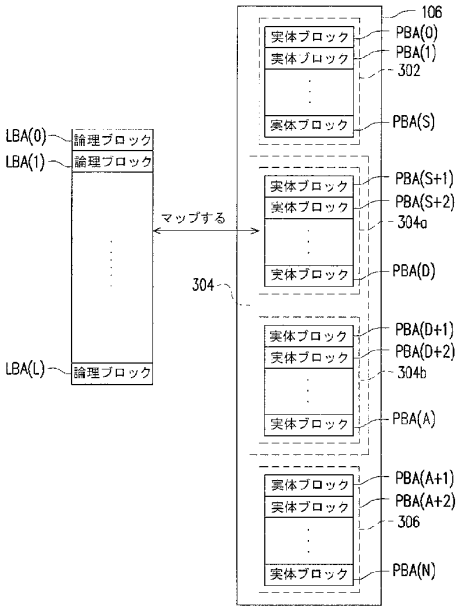
【図１Ｂ】



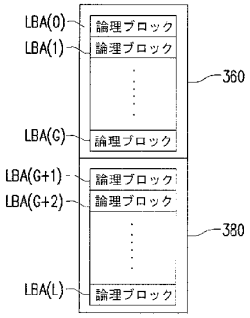
【図１Ｃ】



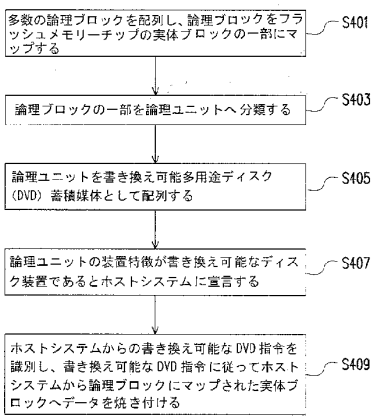
【図２】



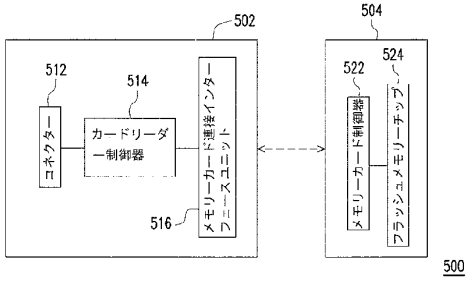
【図３】


LBA(0)
論理ブロック
LBA(1)
論理ブロック
360
LBA(G)
論理ブロック
LBA(G+1)
論理ブロック
LBA(G+2)
論理ブロック
380
LBA(L)
論理ブロック

【図４】


多数の論理ブロックを配列し、論理ブロックをフラッシュメモリーチップの実体ブロックの一部にマップする
S401
論理ブロックの一部を論理ユニットへ分類する
S403
論理ユニットを書き換え可能多用途ディスク（DVD）蓄積媒体として配列する
S405
論理ユニットの装置特徴が書き換え可能なディスク装置であるとホストシステムに宣言する
S407
ホストシステムからの書き換え可能なDVD指令を識別し、書き換え可能なDVD指令に従ってホストシステムから論理ブロックにマップされた実体ブロックへデータを焼き付ける
S409

【図５】


502
504
512
514
524
522
コネクター
カードリーダー制御器
メモリーカード連接インターフェースユニット
メモリーカード制御器
フラッシュメモリーチップ
516
500

フロントページの続き

(51)Int.Cl.		FI		テーマコード(参考)
G06K 17/00	***(2006.01)***	G06F 3/08	H	
		G06K 17/00	L	

(72)発明者　許　世賢
台湾彰化縣芬園鄉中崙村大彰路二段２２１巷七號
(72)発明者　游　祥雄
台湾新竹縣▲弓▼林鄉光復街８９巷１２號

Ｆターム(参考)　5B035 AA06　BB09　CA29
5B058 CA23　CA26
5B060 AB26
5B065 BA01　BA09　CC02　ZA01
5B082 FA05　JA06

【外国語明細書】
2011123863000001. pdf

management unit 204 modifies the write protect flag according to the user's requirement only when the identification password input by the user matches the vendor password. Herein the memory management unit 204 stores the write protect flag and the vendor password in the system area 302. However, the present invention is not limited thereto, and in another exemplary embodiment of the present invention, the write protect flag and the vendor password may also be stored in another non-volatile memory (for example, a non-volatile memory additionally disposed in the flash memory controller 104).

[0083] In an exemplary embodiment of the present invention, when the CPU 1102 of the host system 1000 sends the clear command, the format command, or the close section command to the flash memory storage device 100, the memory management unit 204 determines whether the write protect flag indicates the "write-protected status". When the write protect flag indicates the "write-protected status", the memory management unit 204 sends an error message back to the host system 1000 or does not take any action. The error message may be "write protected (i.e., an ASC code 0×27 and a SenseKey code 0×07)", "medium does not exist (i.e., an ASC code 0×3A and a SenseKey code 0×02)", "medium is being changed (i.e., an ASC code 0×28 and a SenseKey code 0×06)", "command is not supported (i.e., an ASC code 0×20 and a SenseKey code 0×05)", "command parameter error (i.e., an ASC code 0×24 and a SenseKey code 0×05)", " data writing error (i.e., an ASC code 0×03 and a SenseKey code 0×03)", or "data writing error (i.e., an ASC code 0×11 and a SenseKey code 0×03)".

[0084] If the write protect flag indicates the "write-protected status", besides responding the host system 1000 with one of foregoing error messages, in another

exemplary embodiment of the present invention, the memory management unit 204 may also directly modify the storage medium feature to prevent the host system 1000 from issuing any write or clear command. For example, when the CPU 1102 of the host system 1000 sends a storage medium feature inquiry command to the flash memory storage device 100, the memory management unit 204 determines whether the write protect flag indicates the "write-protected status". If the write protect flag indicates the "write-protected status", the memory management unit 204 sends a single layer read only DVD message or a double layers read only DVD message back to the host system 1000 to notify the host system 1000 that the first logical unit 360 is a read only DVD.

[0085] FIG. 4 is a flowchart of a method for simulating a rewritable DVD according to an exemplary embodiment of the present invention.

[0086] Referring to FIG. 4, in step S401, the memory management unit 204 configures a plurality of logical blocks (for example, the logical blocks LBA(0)-LBA(L)) and maps the logical blocks to a portion of the physical blocks of the flash memory chip 106.

[0087] Then, in step S403, the memory management unit 204 groups a portion of the logical blocks into a logical unit (for example, the first logical unit 360). In step S405, the memory management unit 204 configures the logical unit as a rewritable DVD storage medium.

[0088] Next, in step S407, the memory management unit 204 declares to the host system 1000 that the logical unit is a rewritable DVD device. In step S409, the memory management unit 204 identifies rewritable DVD commands received from the host system 1000 and records data received from the host system 1000 into the physical blocks mapped to the logical blocks according to the rewritable DVD commands. The

mechanism for the memory management unit 204 to identify the rewritable DVD command has been described above therefore will not be described herein.

[0089] FIG. 5 is a block diagram of a flash memory storage system according to a second exemplary embodiment of the present invention.

[0090] Referring to FIG. 5, the flash memory storage system 500 includes a card reader 502 and a removable memory card 504.

[0091] The card reader 502 includes a connector 512, a card reader controller 514, and a memory card connection interface unit 516.

[0092] The connector 512 is configured to couple to a host system 1000. In the present exemplary embodiment, the connector 512 is a USB connector. However, the present invention is not limited thereto, and the connector 512 may also be a PATA connector, a SATA connector, an IEEE 1394 connector, a PCI express connector, an IDE connector, or other suitable connector. In the present exemplary embodiment, the card reader 502 is coupled to the host system 1000 through the connector 512 in a detachable manner. However, the present invention is not limited thereto, and in another exemplary embodiment of the present invention, the card reader 502 may also be coupled to the host system 1000 in a fixed manner.

[0093] The card reader controller 514 is coupled to the connector 512 and configured to control the operation of the card reader 502. Particularly, in the present exemplary embodiment, the card reader controller 514 declares to the host system 1000 that the card reader 502 is a rewritable DVD device. For example, the card reader controller 514 declares to the host system 1000 that the card reader 502 is a disc device conforming to the DVD+RW specification, the DVD-RW specification, the DVD-RAM specification, or the BD specification. Besides, the card reader controller 514

identifies a rewritable DVD command received from the host system 1000 and operates the removable memory card 504 according to the rewritable DVD command. The mechanism for responding to the rewritable DVD command received from the host system 1000 has been described above therefore will not be described herein.

[0094] The memory card connection interface unit 516 is coupled to the card reader controller 514 and configured to couple to the removable memory card 504.

[0095] The removable memory card 504 is plugged into the card reader 502 in a detachable manner. The removable memory card 504 includes a memory card controller 522 and a flash memory chip 524.

[0096] The memory card controller 522 configures logical blocks and maps the logical blocks to the physical blocks so that the host system 1000 can access the flash memory chip 524 in the removable memory card 504 through the card reader 502.

[0097] To be specific, the card reader controller 514 declares to the host system 1000 that the removable memory card 504 is a rewritable DVD and identifies a rewritable DVD command from the host system 1000, and the memory card controller 522 executes a block management mechanism of the flash memory chip 524 and operates the flash memory chip 524 according to instructions of the card reader controller 514.

[0098] In the present exemplary embodiment, the flash memory chip 524 is the same as the flash memory chip 106, and therefore the structure and function of the flash memory chip 524 will not be described herein.

[0099] As described above, in exemplary embodiments of the present invention, a rewritable DVD device is simulated by using a flash memory storage device, and various data operations are performed on the simulated rewritable DVD device

according to various rewritable DVD commands. Because the access speed of a flash memory storage system is faster than that of a general rewritable DVD system, the rewritable DVD device simulated by using the flash memory storage device has a faster access speed than a general rewritable DVD device. In addition, a rewritable DVD device simulated by using a flash memory storage device does not require any mechanical optical pickup. Accordingly the size of the rewritable DVD device is greatly reduced. Thereby, the rewritable DVD device simulated by using the flash memory storage device can be disposed in a small-sized netbook. The previously described exemplary embodiments of the present invention have the advantages aforementioned, wherein the advantages aforementioned not required in all versions of the invention.

[00100] It will be apparent to those skilled in the art that various modifications and variations can be made to the structure of the present invention without departing from the scope or spirit of the invention. In view of the foregoing, it is intended that the present invention cover modifications and variations of this invention provided they fall within the scope of the following claims and their equivalents.

WHAT IS CLAIMED IS:

1. A flash memory storage system, comprising:

a flash memory chip, having a plurality of physical blocks;

a connector, configured to couple to a host system; and

a controller, coupled to the flash memory chip and the connector and configured to configure a plurality of logical blocks and map the logical blocks to a portion of the physical blocks,

wherein the controller further is configured to identify at least one rewritable disc command from the host system and writes data from the host system into the physical blocks mapped to at least a portion of the logical blocks according to the at least one rewritable disc command.

2. The flash memory storage system according to claim 1, wherein the controller establishes at least one management information of a rewritable disc device and stores the at least one management information of the rewritable disc device into one of the physical blocks.

3. The flash memory storage system according to claim 2, wherein the at least one management information of the rewritable disc device comprises storage status information, and the storage status information records a storage status of at least a portion of the logical blocks.

4. The flash memory storage system according to claim 2, wherein the at least one management information of the rewritable disc device comprises a written capacity parameter, and the written capacity parameter records a written capacity of at least a portion of the logical blocks.

5. A flash memory storage system, comprising:

a flash memory chip, having a plurality of physical blocks;

a connector, cinfigured to couple to a host system; and

a controller, coupled to the flash memory chip and the connector and configured to configure a plurality of logical blocks and map the logical blocks to a portion of the physical blocks,

wherein one of the physical blocks stores at least one management information of a rewritable disc device, and the flash memory storage system allows the host system to identify at least a portion of the logical blocks as a rewritable storage medium of the rewritable disc device when the flash memory storage system is coupled to the host system by configuring the controller and the at least one management information of the rewritable disc device.

6. The flash memory storage system according to claim 5, wherein the controller sends a single layer rewritable DVD message or a double layers rewritable DVD message back to the host system when the controller receives a storage medium feature inquiry command from the host system.

7. The flash memory storage system according to claim 5, wherein the controller sends a capacity value back to the host system when the controller receives a capacity inquiry command from the host system,

wherein the capacity value is smaller than or equal to a capacity of the flash memory chip.

8. The flash memory storage system according to claim 5, further comprising a non-volatile memory, wherein the at least one management information of the rewritable disc device is stored in the non-volatile memory.

9. The flash memory storage system according to claim 5, wherein the at

least one management information of the rewritable disc device comprises a storage status information, and the storage status information records a storage status of at least a portion of the logical blocks.

10. The flash memory storage system according to claim 9, wherein the controller marks the storage status information as a blank status when the controller receives a clear command from the host system.

11. The flash memory storage system according to claim 9, wherein the controller marks the storage status information as a non-blank status when the controller receives a format command or a close section command from the host system.

12. The flash memory storage system according to claim 9, wherein the controller determines whether the storage status information indicates a blank status when the controller receives a read command from the host system,

wherein the controller sends a null value back to the host system when the storage status information indicates the blank status.

13. The flash memory storage system according to claim 5, wherein the at least one management information of the rewritable disc device comprises a written capacity parameter, and the written capacity parameter records a written capacity of at least a portion of the logical blocks.

14. The flash memory storage system according to claim 13, wherein the controller initializes the written capacity parameter when the controller receives a format command or a clear command from the host system.

15. The flash memory storage system according to claim 13, wherein the controller updates the written capacity parameter after the controller receives a write command from the host system and writes the data into the physical blocks mapped to at

least a portion of the logical blocks.

16. The flash memory storage system according to claim 13, wherein the controller sends a next writable address back to the host system according to the written capacity parameter when the controller receives a next writable address inquiry command from the host system.

17. The flash memory storage system according to claim 5, wherein the controller uses a conversion parameter to calculate a logical address and a capacity of data indicated by a read command or a write command when the controller receives the read command or the write command from the host system.

18. The flash memory storage system according to claim 5, wherein the controller sends a non-support message back to the host system when the controller receives an error correction data inquiry command from the host system.

19. The flash memory storage system according to claim 5, wherein the controller calculates a logical address corresponding to a MSF read command through following expression when the controller receives the MSF read command from the host system,

$$LBA=((M\times60+S-2)\times75+F)\times4$$

wherein LBA represents the logical address corresponding to the MSF read command in the flash memory chip, M represents a minute parameter of the MSF read command, S represents a second parameter of the MSF read command, and F represents a frame parameter of the MSF read command.

20. The flash memory storage system according to claim 5, wherein the controller sends a received message or a ready message back to the host system when the controller receives a disc eject command from the host system.

21. The flash memory storage system according to claim 5, wherein the host system identifies another portion of the logical blocks as a rewritable storage medium of a disk device, a read only storage medium of a compact disc read only memory (CD-ROM) device, or a rewritable storage medium of a rewritable disc device.

22. The flash memory storage system according to claim 5, wherein the controller sets a write protect information according to a command and writes the write protect information into one of the physical blocks, wherein the write protect information records whether the portion of the logical blocks is in a write-protected status or a non-write-protected status.

23. The flash memory storage system according to claim 22, wherein the controller determines whether the write protect information indicates the write-protected status when the controller receives a clear command, a format command, or a close section command from the host system,

wherein the controller sends an error message back to the host system when the write protect information indicates the write-protected status.

24. The flash memory storage system according to claim 22, wherein the controller determines whether the write protect information indicates the write-protected status when the controller receives a storage medium feature inquiry command from the host system,

wherein the controller sends a single layer read only DVD message or a double layer read only DVD message back to the host system when the write protect information indicates the write-protected status.

25. A flash memory controller, for managing a plurality of physical blocks of a flash memory chip, the flash memory controller comprising:

a microprocessor unit;

a flash memory interface unit, coupled to the microprocessor unit and configured to couple to the flash memory chip;

a host interface unit, coupled to the microprocessor unit and configured to couple to a host system; and

a memory management unit, coupled to the microprocessor unit and configured to configure a plurality of logical blocks and maps the logical blocks to a portion of the physical blocks,

wherein the memory management unit identifies at least one rewritable disc command from the host system and writes data from the host system into the physical blocks mapped to at least a portion of the logical blocks according to the at least one rewritable disc command.

26. The flash memory controller according to claim 25, wherein the memory management unit establishes at least one management information of a rewritable disc device and stores the at least one management information of the rewritable disc device into one of the physical blocks.

27. The flash memory controller according to claim 26, wherein the at least one management information of the rewritable disc device comprises a storage status information, and the storage status information records a storage status of at least a portion of the logical blocks.

28. The flash memory controller according to claim 26, wherein the at least one management information of the rewritable disc device comprises a written capacity parameter, and the written capacity parameter records a written capacity of at least a portion of the logical blocks.

29. A computer system, comprising:

a central processing unit (CPU);

a main memory, coupled to the CPU; and

an embedded flash memory storage device, coupled to the CPU, wherein the embedded flash memory storage device comprises:

a flash memory chip, having a plurality of physical blocks; and

a controller, coupled to the flash memory chip and configured to configure a plurality of logical blocks and map the logical blocks to a portion of the physical blocks,

wherein the controller identifies at least one rewritable disc command from the CPU and writes data into the physical blocks mapped to at least a portion of the logical blocks according to the rewritable disc command.

30. The computer system according to claim 29, wherein the controller establishes at least one management information of a rewritable disc device and stores the at least one management information of the rewritable disc device into one of the physical blocks.

31. The computer system according to claim 30, wherein the at least one management information of the rewritable disc device comprises a storage status information, and the storage status information records a storage status of at least a portion of the logical blocks.

32. The computer system according to claim 30, wherein the at least one management information of the rewritable disc device comprises a written capacity parameter, and the written capacity parameter records a written capacity of at least a portion of the logical blocks.

33. A computer system, comprising:

a CPU;

a main memory, coupled to the CPU;

a card reader, coupled to the CPU, wherein the card reader comprises a card reader controller, and the card reader controller is configured to declare that the card reader is a rewritable disc device to the CPU; and

a removable memory card, plugged into the card reader, wherein the removable memory card comprises:

a flash memory chip, having a plurality of physical blocks; and

a memory card controller, coupled to the flash memory chip and configured to configure a plurality of logical blocks and maps the logical blocks to a portion of the physical blocks,

wherein the card reader controller identifies at least one rewritable disc command from the CPU and writes data into the physical blocks mapped to at least a portion of the logical blocks according to the rewritable disc command.

34. A flash memory storage system, comprising:

a card reader, comprising a card reader controller, wherein the card reader controller declares that the card reader is a rewritable disc device to a host system; and

a removable memory card, plugged into the card reader, wherein the removable memory card comprises:

a flash memory chip, having a plurality of physical blocks; and

a memory card controller, coupled to the flash memory chip and configured to configure a plurality of logical blocks and maps the logical blocks

to a portion of the physical blocks,

wherein the card reader controller identifies at least one rewritable disc command from the host system and writes data from the host system into the physical blocks mapped to at least a portion of the logical blocks according to the rewritable disc command.

35. A method for simulating a rewritable disc device by using a flash memory storage device, wherein the flash memory storage device has a flash memory chip, and the flash memory chip has a plurality of physical blocks, the method comprising;

configuring a plurality of logical blocks and mapping the logical blocks to a portion of the physical blocks, wherein the logical blocks have a plurality of logical addresses, the physical blocks have a plurality of physical addresses, and mapping relationships exist between the logical addresses and the physical addresses;

storing at least one management information of the rewritable disc device into one of the physical blocks;

responding to a host system to access the flash memory storage device according to the at least one management information of the rewritable disc device; and

using a conversion parameter to calculate a written logical address and a quantity of data corresponding to a write command of at least one rewritable disc command given by the host system, wherein the written logical address is one of the logical addresses.

36. The method according to claim 35, wherein when data is written into one of the physical blocks according to the rewritable disc command and a write error occurs, the data is written into another one of the physical blocks.

37. The method according to claim 35, wherein the at least one management information of the rewritable disc device comprises a storage status information, and the storage status information records a storage status of at least a portion of the logical blocks.

38. The method according to claim 35, wherein the at least one management information of the rewritable disc device comprises a written capacity parameter, and the written capacity parameter records a written capacity of at least a portion of the logical blocks.

Samsung Ex. 1002, Page 331 of 622

ABSTRACT OF THE DISCLOSURE

A flash memory storage system including a flash memory chip, a connector, and a controller is provided. The flash memory chip has a plurality of physical blocks. The connector is configured to couple to a host system. The controller is coupled to the flash memory chip and the connector. The controller configures a plurality of logical blocks and maps the logical blocks to a portion of the physical blocks. In addition, the controller identifies rewritable disc commands from the host system and writes data from the host system into the physical blocks mapped to the logical blocks according to the rewritable disc commands. Thereby, a rewritable disc device is simulated by using the flash memory storage system.

Samsung Ex. 1002, Page 333 of 622

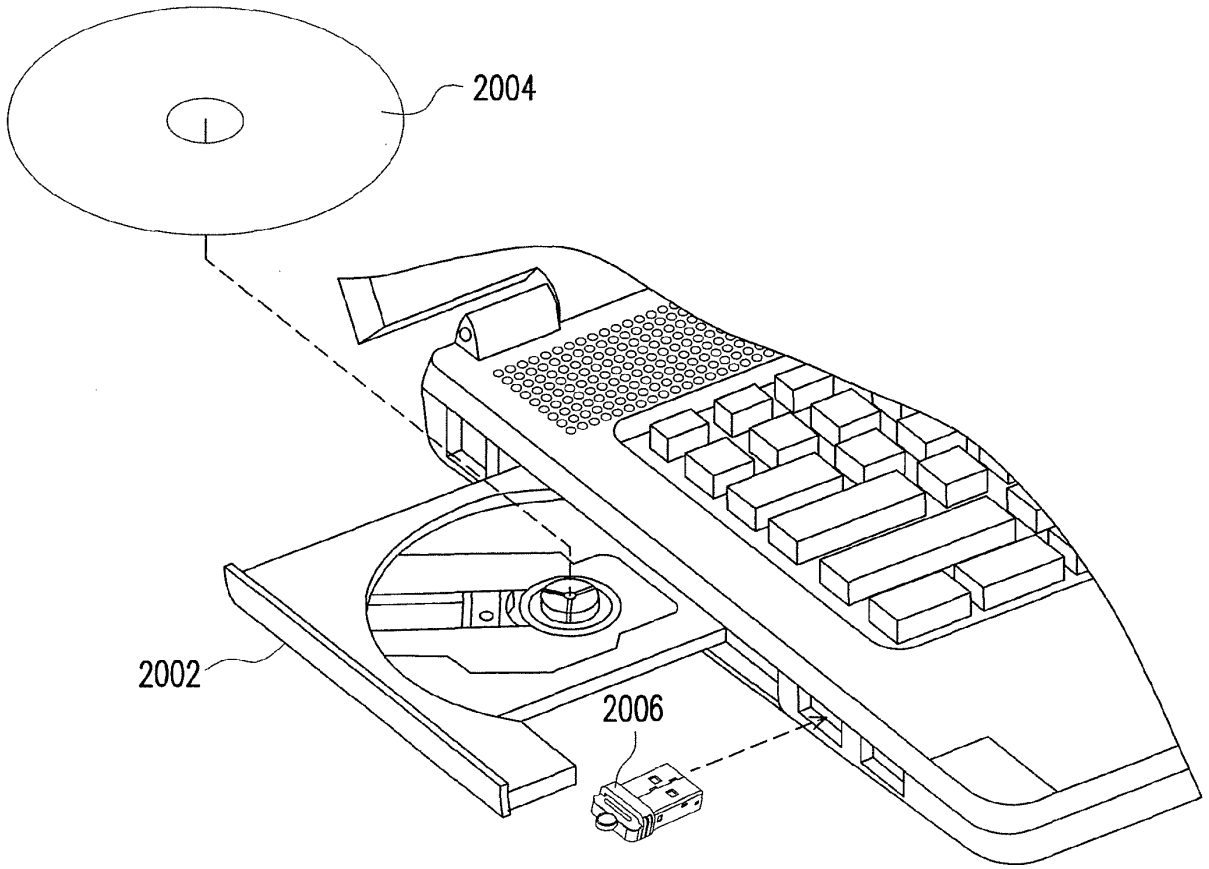


FIG. 1A (RELATED ART)

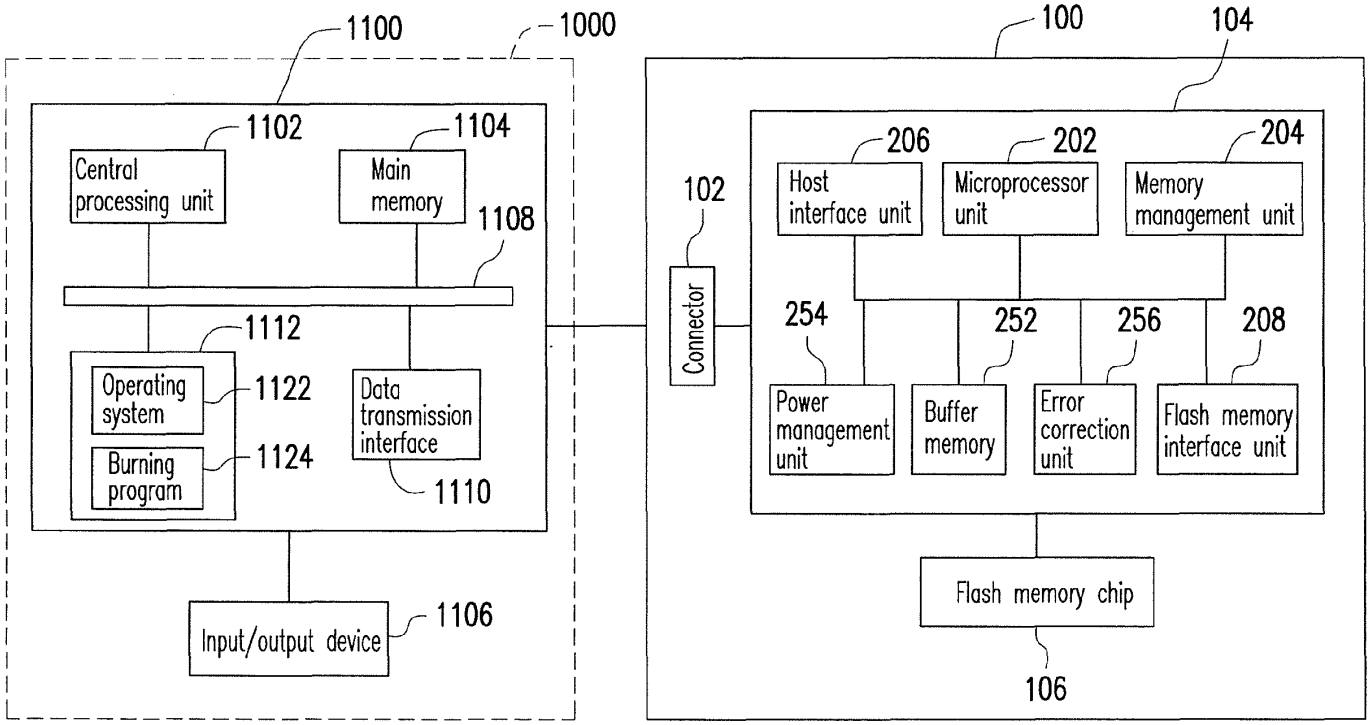

1100
1000
100
104
1102
1104
Central processing unit
Main memory
1108
1112
Operating system
1122
Burning program
1124
Data transmission interface
1110
Input/output device
1106
102
Connector
206
202
204
Host interface unit
Microprocessor unit
Memory management unit
254
252
256
208
Power management unit
Buffer memory
Error correction unit
Flash memory interface unit
Flash memory chip
106

FIG. 1B

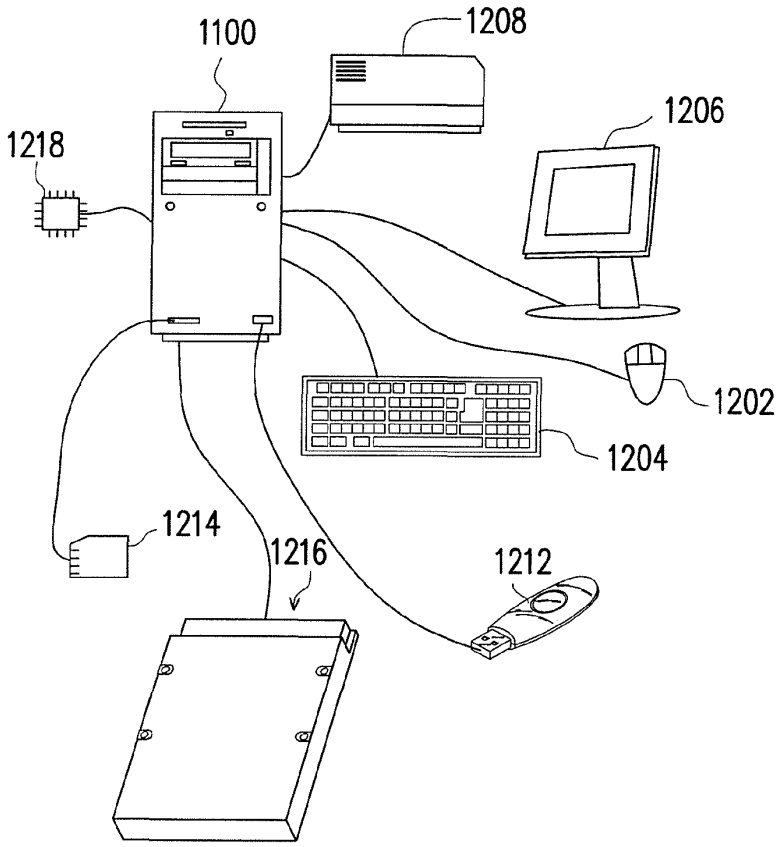


FIG. 1C

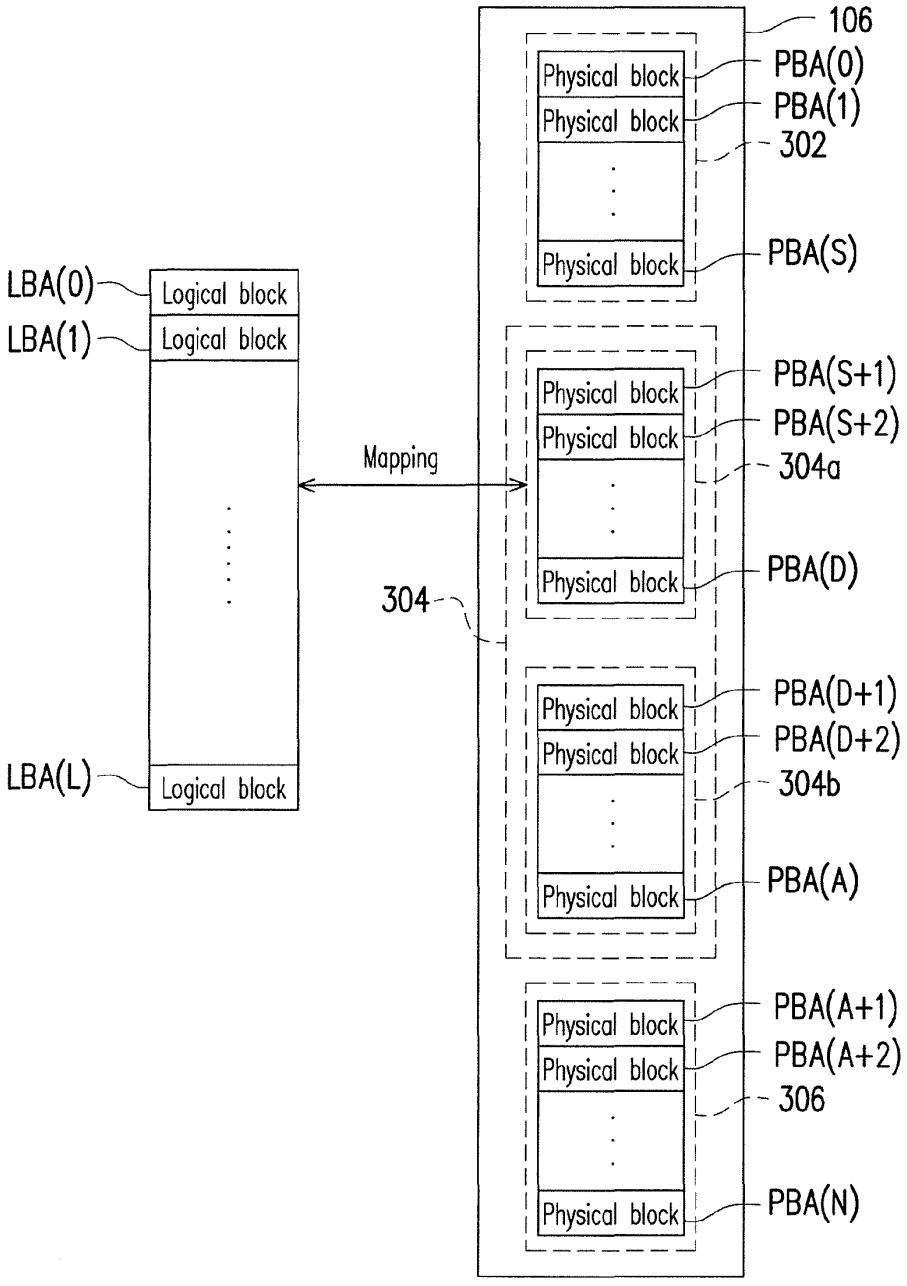

106
LBA(0)
LBA(1)
LBA(L)
Logical block
Logical block
Logical block
Mapping
304
Physical block
PBA(0)
PBA(1)
302
PBA(S)
PBA(S+1)
PBA(S+2)
304a
PBA(D)
PBA(D+1)
PBA(D+2)
304b
PBA(A)
PBA(A+1)
PBA(A+2)
306
PBA(N)

FIG. 2

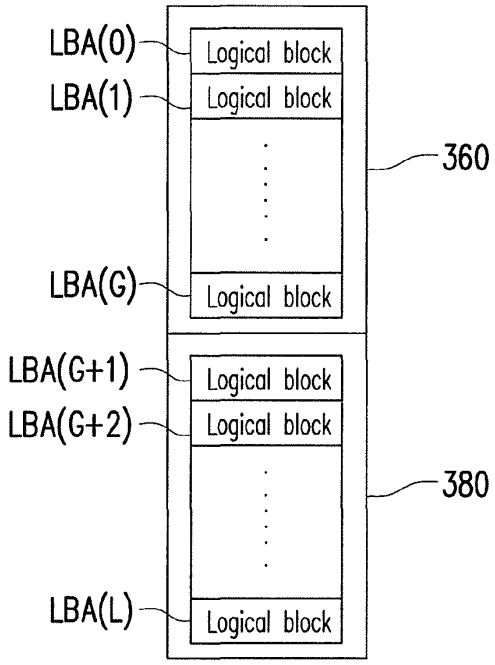

LBA(0)
Logical block
LBA(1)
Logical block
360
LBA(G)
Logical block
LBA(G+1)
Logical block
LBA(G+2)
Logical block
380
LBA(L)
Logical block

FIG. 3

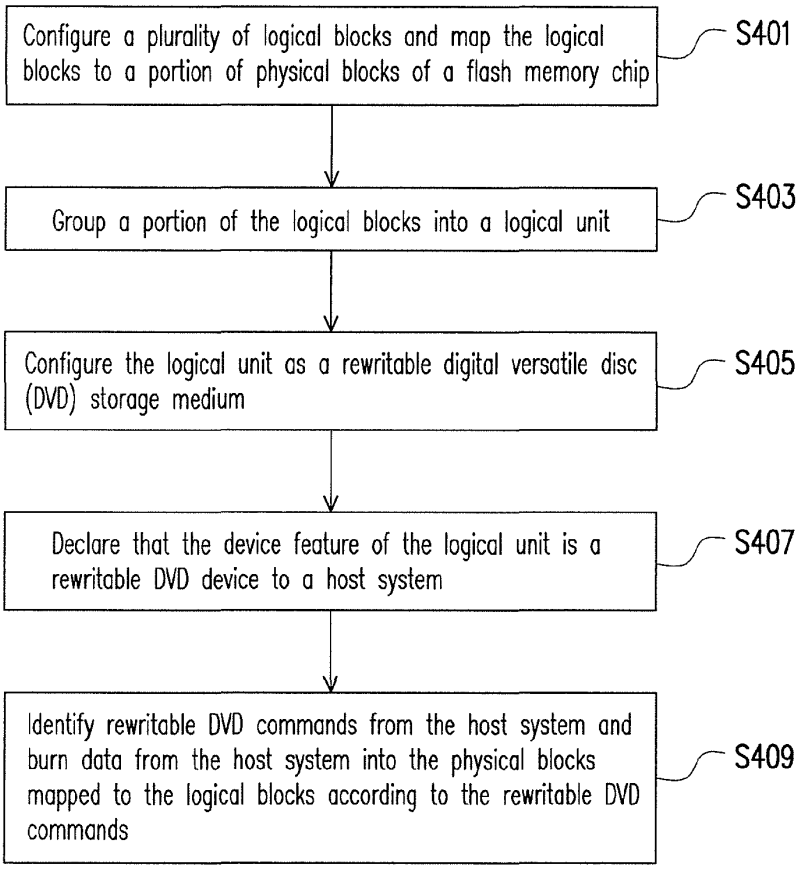
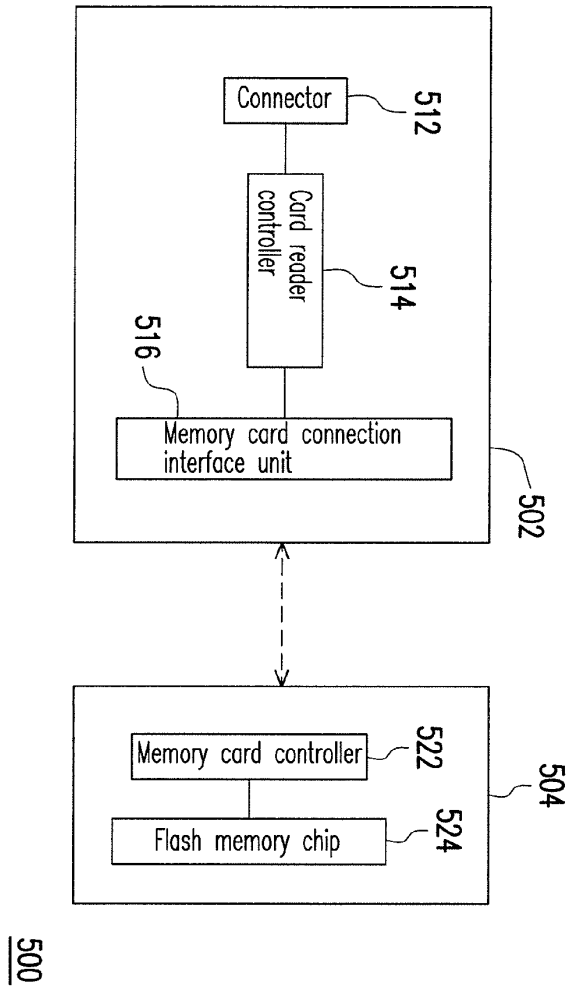

Configure a plurality of logical blocks and map the logical blocks to a portion of physical blocks of a flash memory chip
S401
Group a portion of the logical blocks into a logical unit
S403
Configure the logical unit as a rewritable digital versatile disc (DVD) storage medium
S405
Declare that the device feature of the logical unit is a rewritable DVD device to a host system
S407
Identify rewritable DVD commands from the host system and burn data from the host system into the physical blocks mapped to the logical blocks according to the rewritable DVD commands
S409

FIG. 4

FIG. 5


Connector
512
Card reader controller
514
516
Memory card connection interface unit
502
Memory card controller
522
504
Flash memory chip
524
500

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	16997471
Filing Date	2021-12-27
First Named Inventor	Andrey V. Kuzmin
Art Unit	2136
Examiner Name	Birkhimer, Christopher
Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	8626989	B2	2014-01-07	Van Aken	
	2	9383926	B2	2016-07-05	Law	
	3	9134918	B2	2015-09-15	Yurzola	
	4	8898410	B1	2014-11-25	Ehrenberg	
	5	11347639	B1	2022-05-31	Kuzmin	
	6	11347638	B1	2022-05-31	Kuzmin	
	7	11354234	B1	2022-06-07	Kuzmin	
	8	11354235	B1	2022-06-07	Kuzmin	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENT APPLICATION PUBLICATIONS						Remove
Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20130297880	A1	2013-11-07	Flynn	
	2	20130275682	A1	2013-10-17	Ramanujan	
	3	20120072645	A1	2012-03-22	Olbrich	
	4	20080034153	A1	2008-02-07	Lee	
	5	20170192901	A1	2017-07-06	Coffin	
	6	20150339187	A1	2015-11-26	Sharon	
	7	20170199703	A1	2017-07-13	Ravimohan	
	8	20170109078	A1	2017-04-20	Shaharabany	
	9	20050160227	A1	2005-07-21	Todd	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	10	20070136555	A1	2007-06-14	Sinclair	
	11	20140122781	A1	2014-05-01	Smith	
	12	20160070496	A1	2016-03-10	Cohen	
	13	20100287327	A1	2010-11-11	Li	
	14	20090216992	A1	2009-08-27	Greiner	
	15	20090144497	A1	2009-06-04	Withers	
	16	20110231623	A1	2011-09-22	Goss	
	17	20090172499	A1	2009-07-02	Olbrich	
	18	20170075620	A1	2017-03-16	Yamamoto	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	Y.. Hu et al., "Achieving page-mapping FTL performance at block-mapping FTL cost by hiding address translation," 2010 IEEE 26th Symposium On Mass Storage Systems And Technologies (MSST), 2010, pp1-12.	

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16997471
	Filing Date	2021-12-27
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	2136
	Examiner Name	Birkhimer, Christopher
	Attorney Docket Number	2020058 / RMS-02C1C1C14

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Marc P. Schuyler/	Date (YYYY-MM-DD)	2022-06-14
Name/Print	Marc P. Schuyler	Registration Number	35675

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Doc code: RCEX
Doc description: Request for Continued Examination (RCE)

PTO/SB/30EFS (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

REQUEST FOR CONTINUED EXAMINATION(RCE)TRANSMITTAL
(Submitted Only via EFS-Web)

Application Number	16997471	Filing Date	2020-08-19	Docket Number (if applicable)	2020058 / RMS-02C1C1C14	Art Unit	2136
First Named Inventor	Andrey V. Kuzmin			Examiner Name	Birkhimer, Christopher		

This is a Request for Continued Examination (RCE) under 37 CFR 1.114 of the above-identified application.
Request for Continued Examination (RCE) practice under 37 CFR 1.114 does not apply to any utility or plant application filed prior to June 8, 1995, or to any design application. The Instruction Sheet for this form is located at WWW.USPTO.GOV

SUBMISSION REQUIRED UNDER 37 CFR 1.114

Note: If the RCE is proper, any previously filed unentered amendments and amendments enclosed with the RCE will be entered in the order in which they were filed unless applicant instructs otherwise. If applicant does not wish to have any previously filed unentered amendment(s) entered, applicant must request non-entry of such amendment(s).

☐ Previously submitted. If a final Office action is outstanding, any amendments filed after the final Office action may be considered as a submission even if this box is not checked.

- ☐ Consider the arguments in the Appeal Brief or Reply Brief previously filed on _____
- ☐ Other _____

☒ Enclosed

- ☒ Amendment/Reply
- ☒ Information Disclosure Statement (IDS)
- ☐ Affidavit(s)/ Declaration(s)
- ☐ Other _____

MISCELLANEOUS

☐ Suspension of action on the above-identified application is requested under 37 CFR 1.103(c) for a period of months _____
(Period of suspension shall not exceed 3 months; Fee under 37 CFR 1.17(i) required)

☐ Other _____

FEES

☐ **The RCE fee under 37 CFR 1.17(e) is required by 37 CFR 1.114 when the RCE is filed.**
The Director is hereby authorized to charge any underpayment of fees, or credit any overpayments, to Deposit Account No _____

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

☒ Patent Practitioner Signature

Applicant Signature

PTO/SB/30EFS (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Signature of Registered U.S. Patent Practitioner			
Signature	/Marc P. Schuyler/	Date (YYYY-MM-DD)	2022-06-14
Name	Marc P. Schuyler	Registration Number	35675

This collection of information is required by 37 CFR 1.114. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Patent Application Fee Transmittal					
Application Number:	16997471				
Filing Date:	19-Aug-2020				
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS				
First Named Inventor/Applicant Name:	Andrey V. Kuzmin				
Filer:	Marc Philip Schuyler				
Attorney Docket Number:	2020058 / RMS-02C1C1C14				
Filed as Small Entity					
Filing Fees for	**Utility under 35 USC 111(a)**				
Description		**Fee Code**	**Quantity**	**Amount**	**Sub-Total in USD($)**
Basic Filing:					
Pages:					
Claims:					
Miscellaneous-Filing:					
Petition:					
Patent-Appeals-and-Interference:					
Post-Allowance-and-Post-Issuance:					
Extension-of-Time:					

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Extension - 1 month with $0 paid	2251	1	110	110
Miscellaneous:				
RCE- 1ST REQUEST	2801	1	680	680
		Total in USD ($)		**790**

Electronic Acknowledgement Receipt	
EFS ID:	45954444
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	14-JUN-2022
Filing Date:	19-AUG-2020
Time Stamp:	16:37:55
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	CARD
Payment was successfully received in RAM	$790
RAM confirmation Number	E20226DG38500623
Deposit Account	
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Form (SB08)	IDS-02-C14.pdf	613268 c042f5263e1db4a099f0b35fdc5c1f1831a39d57	no	6
Warnings:					
Information:					
2	Non Patent Literature	NPL1-17571394.pdf	2273612 afe16b4d297edcf074a859ab22bbad0b7cc8d9b1	no	12
Warnings:					
Information:					
3	Request for Continued Examination (RCE)	sb0030efs.pdf	1349938 7ae1a0d322e732a55513e3beeb03f094d3322e3b	no	3
Warnings:					
Information:					
4	Amendment Submitted/Entered with Filing of Continued Prosecution Application (CPA)/Request for Continued Examination(RCE)	2022-05-09-OAResponse-RMS-02C1C1C14.pdf	106931 742a0c41240da4c55204d2414f29a54581f926d4	no	11
Warnings:					
Information:					
5	Fee Worksheet (SB06)	fee-info.pdf	40936 036564c6496ea97815fd34e7e52e9ce49e3043a9	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	4384685		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application Serial Number:	16/997471
Confirmation Number:	1070
First Named Inventor:	Andrey V. Kuzmin
Title:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
Filing Date:	08/19/2020
Attorney Docket:	2020058 / RMS-02C1C1C14

APPLICANT RESPONSE TO 2/14/2022 OFFICE ACTION, FILED WITH REQUEST FOR CONTINUED EXAMINATION

Applicant hereby requests a one month extension of time to respond to the outstanding office action. A request for continued examination (RCE) is also submitted herewith. All fees are being concurrently paid via the EFS.

In The Claims.

Please amend the claims as follows. This listing replaces all prior listings of claims.

1. (Currently Amended) A memory controller to control ~~nonvolatile~~flash memory, the ~~nonvolatile~~flash memory having ~~nonvolatile memory dies~~subdivisions, each of the ~~nonvolatile memory dies~~subdivisions ~~having~~ corresponding to a respective group of erase blocks, each of the erase ~~block in the group~~ blocks such that it must be erased before ~~constituent~~ storage locations ~~associated with~~ in that erase block can be reprogrammed, the memory controller comprising:

a host interface;

a memory interface;

logic to track information indicating an extent to which each of the subdivisions is full;

logic to receive a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, wherein the logic to track is to, in association with execution of the write request, update the information indicating the extent to which the specific one of the subdivisions is full;

logic to send the host the information indicating the extent to which the specific one of the subdivisions is full and an address corresponding to the specific one of the subdivisions;

logic to receive an erasure ~~requests~~ request from ~~a~~ the host, via the host interface, the erasure ~~requests~~ request accompanied by ~~host-selected designations of~~the address corresponding to the specific one of the subdivisions~~space that is to be erased~~, ~~and~~ to responsively control erasure of each unerased ~~an integer set of one or more of the~~ erase ~~blocks~~ block in the group respective to the specific one of the subdivisions, and to update the information indicating the extent to which the specific one of the subdivisions is full~~corresponding to each of the host-selected designations of address space, by issuing one or more erasure commands to the nonvolatile memory dies via the memory interface~~; and

logic to, in connection with the erasure ~~of the integer set of one or more erase blocks~~, detect a defect~~defects~~ in one of the erase blocks in the group respective to the specific one of the subdivisions~~integer set and, responsive to detecting the defects~~, to responsively substitute one or more different erase blocks of the flash memory ~~remap host address information~~ for the one of the ~~at least one~~ erase ~~block~~ blocks for which the defect was detected~~in the integer set to one or more different erase blocks in the nonvolatile~~

~~memory; andlogic to receive data write requests from the host via the host interface and to responsively issue write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information;wherein the memory controller is to direct data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks~~;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

2. (Currently Amended) The memory controller of claim 1, wherein the host interface is also to receive a data copy ~~requests~~ request from the host, ~~and~~ wherein the memory controller is to responsively ~~read~~ retrieve data from ~~one of the constituent storage locations~~ a source location ~~associated with a first one of the erase blocks~~ in the flash memory, the source location associated with the specific one of the subdivisions, and write the retrieved data ~~which has been read~~ to a destination location in the flash memory,~~one of the constituent storage locations associated with a second one of the erase blocks~~ without sending the retrieved data ~~which has been read~~ to the host, and wherein the logic to track is to responsively update the tracked information for one of the subdivisions which corresponds to the destination location, in association with the write of the retrieved data into the destination location.

3. (Currently Amended) The memory controller of claim 2, wherein:

~~the host interface is also to receive data read requests;~~

~~the host interface is to receive the data write and the data read requests each accompanied by a respective host-supplied address;~~

the data copy requests are each accompanied by a source address and a destination address; and

each source ~~respective host-supplied~~ address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the ~~nonvolatile~~flash memory, the mapping such that

different values of the field compel the memory controller to access respective storage structures ~~within the nonvolatile memory~~at the specific tier, ~~the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data;~~

~~and each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the memory controller to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier~~

a value of the field for each source address is effective to cause the memory controller to select one of the respective storage structure corresponding to the source location, and

a value of the field for each destination address is effective to cause the memory controller to select one of the respective storage structures corresponding to the destination location.

4. (Original) The memory controller of claim 3, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

5-10. (Canceled).

11. (Previously Presented) A method of operation in a memory controller to control ~~nonvolatile~~flash memory, where the ~~nonvolatile~~flash memory has ~~nonvolatile memory dies~~subdivisions, each of the ~~nonvolatile memory dies having~~ subdivisions corresponding to a respective group of erase blocks, each of the erase ~~block in the group~~ blocks such that it must be erased before ~~constituent~~ storage locations ~~associated with~~ in that erase block can be reprogrammed, the memory controller having a host interface, a memory interface and at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that, when executed, are to control the operation of hardware circuitry, the method comprising:

tracking information indicating an extent to which each of the subdivisions is full;

receiving a write request from a host via the host interface, the write request accompanied by an address that designates a specific one of the subdivisions, and, in association with execution of the write request, updating the information indicating the extent to which the specific one of the subdivisions is full;

sending the host the information indicating the extent to which the specific one of the subdivisions is full and an address corresponding to the specific one of the subdivisions;

receiving an erasure requests from thea host, via the host interface, the erasure ~~requests~~ request accompanied by the ~~host-selected designations of~~ address corresponding to the specific one of the subdivisions, ~~space that is to be erased, and~~ responsively controlling erasure of each unerased erase block in the group respective to the specific one of the subdivisions, and updating the information indicating the extent to which the specific one of the subdivisions is full~~an integer set of one or more of the erase blocks corresponding to each of the host-selected designations of address space, by issuing erasure commands to the nonvolatile memory dies via the memory interface~~; and

~~detecting,~~ in connection with the erasure, detecting a defect in one of the ~~integer set of one or more~~ erase blocks in the group respective to the specific one of the subdivisions, responsively substituting ~~, defects in the integer set and, responsive to detecting the defects, remapping host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory; and~~

~~receiving data write requests from the host via the host interface and responsively issuing write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information; and~~

~~in connection with said receiving the data write requests and responsively issuing write commands, directing data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the~~ one or more different erase blocks of the flash memory for the one of the erase blocks for which the defect was detected.

12. (Currently Amended) The method of claim 11, further comprising also receiving via the host interface a data copy ~~requests~~ request from the host, ~~and~~ responsively retrieving ~~reading~~ data from a source location in the flash memory, the source location associated with the specific one of the subdivisions, ~~one of the constituent storage locations associated with a first one of the erase blocks~~ and writing the retrieved data which ~~has been read~~ to a destination location in the flash memory,~~one of the constituent storage locations associated with a second one of the erase blocks~~ without sending the data which has been read to the host, and responsively updating the tracked information for the one of the subdivisions which corresponds to the destination location, in association with the write of the retrieved data into the destination location.

13. (Currently Amended) The method of claim 12, wherein:

~~the method also comprises receiving data read requests;~~

the data copy requests are each accompanied by a source address and a destination address;

~~each of the data write and the data read requests is accompanied by a respective host-supplied address, and~~ each source ~~respective host-supplied~~ address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the ~~nonvolatile~~flash memory, the mapping such that

different values of the field compel the memory controller to access respective storage structures at the specific tier~~within the nonvolatile memory~~, ~~the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data; and~~

a value of the field for each source address is effective to cause the memory controller to select one of the respective storage structure corresponding to the source location, and

a value of the field for each destination address is effective to cause the memory controller to select one of the respective storage structures corresponding to the destination location.

~~each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the memory controller to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier.~~

14. (Original) The method of claim 13, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

15-20. (Canceled)

21. (New) The memory controller of claim 2, wherein:

the memory controller further comprises logic to track metadata for each of the subdivisions, the metadata indicating a time since data was programmed, to compare the tracked metadata to a threshold, and, responsive to a determination that the tracked metadata

for the specific one of the subdivisions satisfies the threshold, to transmit to the host the address corresponding to the specific one of the subdivisions; and

the data copy request is accompanied by the address corresponding to the specific one of the subdivisions.

22. (New) The memory controller of claim 21, wherein the address accompanying the data copy request is a source address for the retrieved data and wherein the data copy request is also accompanied by a destination address, wherein the specific one of the subdivisions is a first one of the subdivisions, and wherein the destination address is free to designate a second one of the subdivisions.

23. (New) The memory controller of claim 3, wherein the specific tier is a tier associated with the subdivisions, and wherein the respective storage structures comprise respective ones of the subdivisions.

24. (New) The memory controller of claim 1, wherein the memory controller further comprises logic to receive a query from the host, and wherein the logic to send the information to the host is to do so as a response to the query.

25. (New) The memory controller of claim 1, wherein:

the memory controller further comprises logic to assign a write destination in the flash memory, in which data accompanying the write request is to be stored, the write destination corresponding to the specific one of the subdivisions, and to transmit to the host a read address that will be used to retrieve the data specifically from the write destination; and

the memory controller further comprises logic to receive, from the host, and via the host interface, a read request for the data, accompanied by the read address, and to identify both of the specific one of the subdivisions and the write destination from the read address.

26. (New) The memory controller of claim 1, wherein the memory controller further comprises logic to store erase count information for each of the subdivisions, and to send to the host, via the host interface, the erase count information for the specific one of the subdivisions, together with the address corresponding to the specific one of the subdivisions, and wherein the logic to store the erase count information is to update the erase count information for the specific one of the subdivisions in association with the erasure.

27. (New) The method of claim 12, wherein:

the method further comprises tracking metadata for each of the subdivisions, the metadata indicating a time since data was programmed, comparing the tracked metadata to a threshold, and, responsive to a determination that the tracked metadata for the specific one of the subdivisions satisfies the threshold, transmitting to the host the address corresponding to the specific one of the subdivisions; and

the data copy request is accompanied by the address corresponding to the specific one of the subdivisions.

28. (New) The method of claim 27, wherein the address accompanying the data copy request is a source address for the retrieved data and wherein the data copy request is also accompanied by a destination address, wherein the specific one of the subdivisions is a first one of the subdivisions, and wherein the destination address is free to designate a second one of the subdivisions.

29. (New) The method of claim 13, wherein the specific tier is a tier associated with the subdivisions, and wherein the respective storage structures comprise respective ones of the subdivisions.

30. (New) The method of claim 11, further comprising receiving a query from the host, and sending the information to the host as a response to the query.

31. (New) The method of claim 11, wherein:

the method further comprises assigning a write destination in the flash memory, in which data accompanying the write request is to be stored, the write destination corresponding to the specific one of the subdivisions, and transmitting to the host a read address that will be used to retrieve the data specifically from the write destination; and

the method further comprises receiving, from the host, and via the host interface, a read request for the data, accompanied by the read address, and identifying both of the specific one of the subdivisions and the write destination from the read address.

32. (New) The method of claim 11, wherein the method further comprises:

storing erasing count information for each of the subdivisions;

sending to the host, via the host interface, the erase count information for the specific one of the subdivisions, together with the address corresponding to the specific one of the subdivisions; and

updating the erase count information for the specific one of the subdivisions in association with the erasure.

REMARKS

In the Office Action mailed 02/14/2022, the Examiner rejected claims 1-20 under 35 USC 102 as allegedly being anticipated by Selinger (USPB 20110161784). Applicant continues to disagree with the rejection, for reasons already of record, but to advance prosecution, as amended the claims to better define the recited subject matter. Reconsideration is requested for the additional reasons that are set forth below.

Applicant firstly notes that it has amended the claims to more specifically recite the claimed subject matter. Specifically addressing the amendments to the independent claims, Applicant has:

- changed 'nonvolatile memory' to specifically be "flash memory;"
- added the term 'subdivisions,' wherein each of the subdivisions is specifically associated with a respective 'group' of plural erase blocks;
- added a feature where the memory controller tracks the extent to which each subdivision is full (e.g., consistent with the "page utilization" features recited in connection with some of Applicant's embodiments);
- advanced within the claim those features relating to write request processing, i.e., instead of being discussed at the end of the claim (as was the case with pre-amendment text), these features have been rewritten earlier in the claim; the claimed memory controller updates the extent to which a specific (addressed) one of the subdivisions is full as part of the write request processing;
- added a feature where the memory controller sends to the host the updated information indicating extent to which the specific one of the subdivisions is full, together with an address for that subdivision;
- amended erase request processing claim features, to recite that the erase request is addressed to the address that was sent to the host for the specific one of the subdivisions, and that, in connection with erasure request processing, the memory controller detects a defect in one of the erase blocks that is part of the group of erase blocks corresponding to the 'specific one' of the subdivisions and substitutes in one or more a new (different) erase blocks to take the place of the defective block in the corresponding group, i.e., thereby effectively remapping physical resources associated with the subdivision so as to include the one or more new erase blocks.

It is respectfully submitted that these features are fully supported by the Specification and are made to enhance "chain of custody" associated with the respective claim elements, i.e., their interrelation. Notably, Applicant has also elected to simplify some of the dependent claims by canceling

certain claims relating to storage location release metadata; Applicant has added new claims to take the place of those which have been cancelled, and it is respectfully submitted that all subject matter is supported by the original specification, as filed.

Applicant respectfully requests reconsideration in connection with the request for continued examination. It is submitted that, whatever the Examiner may think of Selinger, Selinger does not disclose the subdivision-based features added the amended claims, and the associated addressing features. Because Selinger does not disclose these features (or other features of the claims, e.g., as outlined by Applicant's prior responses), a rejection under 35 USC 102 is not appropriate.

If it would be helpful for PTO Personnel to discuss any of the presented limitations, or request elaboration on what is meant, such personnel are respectfully requested to contact the undersigned using the contact information set forth below.

Respectfully submitted,
Attorney for Applicant

/Marc P. Schuyler/
Marc P. Schuyler, Reg. No. 35, 675
(408) 655-6375
marc@schuylerlawgroup.com

PTO/SB/84 (11-08)
Approved for use through 03/31/2021. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

AUTHORIZATION TO ACT IN A REPRESENTATIVE CAPACITY

In re Application of:	Andrey V. Kuzmin
Application No.	16/997471
Filed:	19 August 2020
Title:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
Attorney Docket No.	2020058 / RMS-02C1C1C14
Art Unit:	2136

The practitioner named below is authorized to conduct interviews and has the authority to bind the principal concerned. (Note: pursuant to 37 CFR 10.57(c), a practitioner cannot authorize other registered practitioners to conduct interviews without consent of the client after full disclosure.) Furthermore, the practitioner is authorized to file correspondence in the above-identified application pursuant to 37 CFR 1.34:

Name	Registration Number
Timothy B. Kang	46423

This is not a Power of Attorney to the above-named practitioner. Accordingly, the practitioner named above does **not** have authority to sign a request to change the correspondence address, a request for an express abandonment, a disclaimer, a power of attorney, or other document requiring the signature of the applicant, assignee of the entire interest or an attorney of record. If appropriate, a separate Power of Attorney to the above-named practitioner should be executed and filed in the United States Patent and Trademark Office.

SIGNATURE of Practitioner of Record

Signature	/Marc P. Schuyler/	Date	14 June 2022
Name	Marc P. Schuyler	Registration No. (if applicable)	35675
Telephone	408 655-6375		

This collection of information is required by 1.31, 1.32 and 1.34. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 15 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Acknowledgement Receipt

EFS ID:	45954716
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	14-JUN-2022
Filing Date:	19-AUG-2020
Time Stamp:	16:52:14
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Power of Attorney	2022-06-14-C14-sb0084.pdf	322837 465dae1eda8723802f67dd5667e988f889fe9ea5	no	2

Warnings:

Information:	
Total Files Size (in bytes):	322837

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 16/997,471	Filing Date 08/19/2020	☐ To be Mailed

ENTITY: ☐ LARGE ☑ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $230 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	06/14/2022	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $50 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $240 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	LIE
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/ZETA T ADAMS/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

73091 7590 02/14/2022

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	

ART UNIT	PAPER NUMBER
2136	

MAIL DATE	DELIVERY MODE
02/14/2022	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 16/997,471	**Applicant(s)** Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	**Art Unit** 2136	**AIA (FITF) Status** No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☑ Responsive to communication(s) filed on 01/03/2022.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☑ This action is **FINAL.** 2b) ☐ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 1-20 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 1-20 is/are rejected.

8) ☐ Claim(s) _____ is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☐ The specification is objected to by the Examiner.

11)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date _____.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

DETAILED ACTION

The current Office Action is in response to the papers submitted 01/03/2022. Claims 1 – 20 are pending.

Notice of Pre-AIA or AIA Status

1. The present application is being examined under the pre-AIA first to invent provisions.

Claim Rejections - 35 USC § 102

2. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of the appropriate paragraphs of pre-AIA 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claim(s) **1 - 20** is/are rejected under pre-AIA 35 U.S.C. 102(b) as being anticipated by Selinger et al. (Pub. No.: US 2011/0161784) referred to as Selinger.

Regarding **claim 1**, Selinger teaches a memory controller **[700, Fig 7]** to control nonvolatile memory **[730, Fig 7]**, the nonvolatile memory **[730, Fig 7]** having nonvolatile

memory dies **[Paragraph 0024; The memory is NAND flash memory comprised of dies]**, each of the nonvolatile memory dies **[Paragraph 0024; The memory is NAND flash memory comprised of dies]** having a group of erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, each erase block in the group **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** such that it must be erased before constituent storage locations **[Paragraph 0062; The pages in a block]** associated with that erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** can be reprogrammed **[Paragraph 0062]**, the memory controller comprising **[700, Fig 7]**:

a host interface **[725, Fig 7]**;

a memory interface **[735, Fig 7]**;

logic **[700, Fig 7]** to receive erasure requests from a host **[720, Fig 7]** via the host interface **[725, Fig 7]**, the erasure requests accompanied by host-selected designations of address space that is to be erased, and to responsively control erasure of an integer set of one or more of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** corresponding to each of the host-selected designations of address space, by issuing one or more erasure commands **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]** to the nonvolatile memory dies

[Paragraph 0024; The memory is NAND flash memory comprised of dies] via the memory interface **[735, Fig 7]**;

logic **[700, Fig 7]** to, in connection with the erasure of the integer set of one or more erase blocks **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]**, detect defects in the integer set and, responsive to detecting the defects, to remap host address information for at least one erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** in the integer set to one or more different erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** in the nonvolatile memory **[730, Fig 7; 908, Fig 9; Paragraphs 0040, 0080, 0091, and 0095; The controller detects an erase error, notifies the host, and performs the remapping operations of host logical address due to the erase error]**; and

logic **[700, Fig 7]** to receive data write requests **[510, Fig 5; 1440, Fig 14]** from the host **[720, Fig 7]** via the host interface **[725, Fig 7]** and to responsively issue write commands to, and effectuate related data transfer with, the nonvolatile memory **[Fig 6; 730, Fig 7; Figs 10 and 14; The controller sends corresponding commands to either read or write data dependent on the read or write command issues from the host]**, each of the data write requests being accompanied by the host address

information **[510, Fig 5; 1440, Fig 14; Each write request has a destination address]**;

wherein the memory controller **[700, Fig 7]** is to direct data write requests **[510, Fig 5; 1440, Fig 14]** from the host **[720, Fig 7]** for which the host address information has been remapped **[730, Fig 7; 908, Fig 9; Paragraphs 0040, 0080, 0091, and 0095; The controller detects an erase error, notifies the host, and performs the remapping operations of host logical address due to the erase error]** by said logic **[700, Fig 7]** to detect and remap to constituent storage locations **[Paragraph 0062; The pages in a block are written due to write requests]** associated with the one or more different erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**;

wherein each said logic **[700, Fig 7]** comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry **[725, 735, 740, 750, 760, and 770, Fig 7; Paragraph 0025]**.

Regarding **claim 2**, Selinger teaches the host interface **[725, Fig 7]** is also to receive data copy requests **[Fig 14]** from the host **[720, Fig 7]**, and wherein the memory controller **[700, Fig 7]** is to responsively read data from one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a first one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** and write the data which has been read to one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated

with a second one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** without sending the data which has been read to the host **[720, Fig 7; 1460 and 1470, Fig 14; Paragraph 0152; The data that is copied is not sent to the host]**.

Regarding **claim 3**, Selinger teaches the host interface **[Fig 1; 725, Fig 7]** is also to receive data read requests **[Paragraph 0025; The first interface receives all commands sent from the host to the memory devices which include read, write, erase, and other commands]**;

the host interface **[725, Fig 7]** is to receive the data write **[1440, Fig 14]** and the data read requests **[610, Fig 6; 1002, Fig 10; 1410, Fig 14; 1510, Fig 15]** each accompanied by a respective host-supplied address **[Figs 6, 10, 14, and 15; Each read or write request is accompanied with a host designated address]**;

each respective host-supplied address **[Figs 6, 10, 14, and 15; Each read or write request is accompanied with a host designated address]** includes a field that has been mapped in advance to a specific tier of a structural hierarchy **[Paragraphs 0065, 0108, and 0110; Each address provided from the host is mapped in advance in the mapping table to a physical structural tier address]** associated with storage in the nonvolatile memory **[730, Fig 7]**, the mapping such that different values of the field compel the memory controller **[700, Fig 7]** to access respective storage structures **[Paragraphs 0065, 0108, and 0110; Each address provided selects different tiers of the memory]** within the nonvolatile

memory **[730, Fig 7]**, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data **[Paragraphs 0062 and 0086; Each block structure is comprised of multiple pages]**; and

each of the data copy requests **[Fig 14]** is accompanied by a destination address **[1440, Fig 14]** also having the field, a value for the field of the destination address **[1440, Fig 14]** selected by the host **[720, Fig 2]** in a manner which compels the memory controller **[700, Fig 7]** to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier **[Fig 14; The destination address causes the controller to copy data read from a destination address to the destination address]**.

Regarding **claim 4**, Selinger teaches the specific tier is a tier associated with the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks which comprise pages on a lower tier of storage]**, and wherein the respective storage structures comprise respective erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]**.

Regarding **claim 5**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to maintain storage location release information **[900, Fig 9]** for at least one erase block in one of the groups **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]**, the storage location release information **[900, Fig 9]** to indicate a condition where at least one of the

constituent storage locations **[Paragraph 0062; The pages in a block]** for the at least one erase block **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** has been programmed but no longer stores valid data **[Paragraphs 0077, 0091, and 0165; The error reporting shows at least one page of the block has invalid data]**, the condition such that the at least one of the constituent storage locations **[Paragraph 0062; The pages in a block]** cannot be reprogrammed until the at least one erase block is erased **[Paragraph 0062; Pages in a block cannot be reprogrammed until the entire block is erased]**.

Regarding **claim 6**, Selinger teaches the memory controller **[700, Fig 7]** is to store the storage location release information **[Figs 8A – 8D; 900, Fig 9; Paragraphs 0074 – 0077; Status information is stored in registers and stored in the interface in the process of transmitting to the host]** for each of the constituent storage locations **[Paragraph 0062; The pages in a block]** for each erase block in at least one of the groups **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** and is to update the storage location release information **[900, Fig 9]** as an automated response **[Paragraphs 0074 – 0077; The status information is updated as a result of requests from the host]** to requests received from the host **[720, Fig 7]**, and wherein the memory controller **[700, Fig 7]** further comprises logic to receive and process a host-request to transmit, to the host **[720, Fig 7]**, the storage location release information **[900, Fig 9; Fig 10; Paragraph 0074; The requests from the host are requests to transmit associated status information relating to the read/write request from the host]** for at least one of the constituent storage locations

[Paragraph 0062; The pages in a block] associated with the at least one of the groups **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**.

Regarding **claim 7**, Selinger teaches the host **[720, Fig 7]** is to communicate an address space selection together with the host-request to transmit **[Figs 5 - 6, 10, and 14 – 15; Each request from the host includes an address selection to direct the given request to a location in memory]**, such that the memory controller **[700, Fig 7]** is to transmit, to the host **[720, Fig 7]**, storage release information **[900, Fig 9]** associated with a host-selected subset of storage structures **[Paragraph 0062; The pages in a block]**, the host-selected subset corresponding to the address space selection **[Paragraphs 0074 – 0077; The status information presented to the host is associated with pages in the selected block based on the address information in the request from the host]**.

Regarding **claim 8**, Selinger teaches the memory controller **[700, Fig 7]** is to receive from the host **[720, Fig 7]** a request to mark as released one or more of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with at least one of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, and wherein the logic **[700, Fig 7]** to maintain is to responsively update the storage location release information **[900, Fig 9]** to mark the one or more of the constituent storage locations **[Paragraph 0062; The pages in a block]** as released **[908, Fig 9; Paragraphs 0076 – 0077; The erase error indicates an associated block and pages are candidates for erase and thus released]**.

Regarding **claim 9**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to monitor the storage location release information **[900,**

Fig 9] to detect memory space that represents candidates for garbage collection **[Paragraphs 0076 – 0077 and 0159]**, and to transmit to the host **[720, Fig 7]** an identification of the memory space that represents candidates for garbage collection **[1012, Fig 10; Paragraphs 0074 – 0077 and 0159]**;

the host **[720, Fig 7]**, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests **[Figs 14 – 15; Paragraph 0159]**; and

the memory controller **[700, Fig 7]**, in response to each of the one or more data copy requests **[Figs 14 – 15; Paragraph 0159]**, is to read data **[1430, Fig 14; 1530, Fig 15]** from one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a first one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** and write the data **[1460 and 1470, Fig 14; 1570, Fig 15]** to one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a second one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062, 0086, and 0159; The blocks are erase blocks]** without sending the data to the host **[720, Fig 7; 1460 and 1470, Fig 14; Paragraph 0152; The data that is copied is not sent to the host]**.

Regarding **claim 10**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to monitor the storage location release information **[900, Fig 9]** to detect memory space that represents candidates for erasure **[908, Fig 9]**, and is to transmit to the host **[720, Fig 7]** an identification of the memory space that represents candidates for erasure **[908 and 910, Fig 9; Paragraphs 0074, 0080, 0091, and 0164 - 0165]**; and

the host **[720, Fig 7]** is to issue at least one of the erasure requests **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]** in response to said identification **[Paragraphs 0025, 0073, 0076, 0091, and 0164 – 0165]**.

Claims 11 – 20 are the method claims relating to the memory controller claims 1 – 10 and are thus rejected using the same prior art and reasoning.

Response to Arguments

4. Applicant's arguments filed 01/03/2022 have been fully considered but they are not fully persuasive.

The Applicant argues on pages 23 – 24, regarding claim 1, that Selinger fails to teach any structure that whereby a host knows that there is previously-written now free unerased space and the host can simply erase that space at a time of the host's convenience and has no structure where a host can simply physically erase a structure

at any time. After careful consideration of the Applicant's arguments the Examiner respectfully disagrees.

There is no mention in claim of a host knowing a location in memory that was previously written but is now free and is also considered unerased. The claim recites a controller receiving an erasure request from a host, controlling the erase request on erase blocks based on address information in the erase request, detecting defects in the erase blocks related to the erase operation, remapping the defective erase blocks, and performing writes to the remapped erase blocks. There is no mention of a host knowing that data stored in memory is no considered free and is also unerased. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., a host knows that there is previously-written now free unerased space and the host can simply erase that space at a time of the host's convenience) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Selinger discloses in paragraphs 0025, 0032, 0050, 0092, 0165, and multiple other locations that the host sends requests to the controller causing the controller to perform an erase operation in memory. This shows the request resulting in the erase operation can be considered at least an erase operation in part when part of a copy/move command or just a regular erase command alone. This shows the host sends multiple types of erase commands to the controller and the controller executes

the erase commands based on the address information associated with the erase commands.

The host sending write, read, erase, and other commands to the controller shows there was a trigger activated on the host that caused to the host to send the associated command. This shows the host is able to send specific commands when a specific trigger occurs. Nothing happens in a computer system totally random with no reasons or precondition. Everything that happens is based on instructions in a program that dictates when a system performs specific actions.

The Applicant argues on page 24 that Selinger fails to teach that during an erase operation the memory controller both detects a bad block and remaps the bad block which later results in a write from the host that will be directed to the remapped address. After careful consideration of the Applicant's arguments the Examiner respectfully disagrees.

There is no specific mention in the claims that the detection and remapping is occurring during an erase operation as argued. The only require that the detection and remapping of a bad block is performed in connection with an erase operation. That means the detection and remapping is linked or associated with the erase operations in some manner. The detection and remapping can occur before, during, or after the erase operations as claimed. A connection between two items means the two items are related to each other in some manner. This does not require that the two operations being performed at the same time as argued. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., during an erase operation the memory

controller both detects a bad block and remaps the bad block) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Selinger teaches status information is sent to the host which allows the host to know when there was an error in a block during an erase or program operation. That status information is also used to manage the bad, good, and spare blocks according to figure 11. When a bad block is detected the address is remapped to a spare block that is good so the next time a write is directed to the remapped address the good block is accessed. This is fully disclosed at figure 11 and associated paragraphs 0091 – 01000. The status information also includes a remap field that indicates the need to remap a block.

Conclusion

5. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER D BIRKHIMER whose telephone number is (571)270-1178. The examiner can normally be reached 8-5 Hoteling.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on 571-272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Christopher D Birkhimer/
Primary Examiner, Art Unit 2136

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	09/30/2021	02/10/2022							
	1	✓	✓							
	2	✓	✓							
	3	✓	✓							
	4	✓	✓							
	5	✓	✓							
	6	✓	✓							
	7	✓	✓							
	8	✓	✓							
	9	✓	✓							
	10	✓	✓							
	11	✓	✓							
	12	✓	✓							
	13	✓	✓							
	14	✓	✓							
	15	✓	✓							
	16	✓	✓							
	17	✓	✓							
	18	✓	✓							
	19	✓	✓							
	20	✓	✓							

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

CPC - Searched*		
Symbol	**Date**	**Examiner**
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	09/30/2021	CDB
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	02/10/2022	CDB

CPC Combination Sets - Searched*		
Symbol	**Date**	**Examiner**

US Classification - Searched*			
Class	**Subclass**	**Date**	**Examiner**

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	**Date**	**Examiner**
EAST search performed	09/30/2021	CDB
Assignee/Inventor search performed in EAST	09/30/2021	CDB
Prior Art/Assignee/Inventor/Google Internet database search	02/10/2022	CDB

Interference Search			
US Class/CPC Symbol	**US Subclass/CPC Group**	**Date**	**Examiner**

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L56	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"2	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM

		0150324264"|"20150347041"|"20150347296"|"20160018998"|"20160026564"|"20160342509"|"20160357462"|"20160364179"|"20170031699"|"5568423"|"5652857"|"5860082"|"6134631"|"7096378"|"7339823"|"7404031"|"7406563"|"7581078"|"7702846"|"7710777"|"7752381"|"7801561"|"7818489"|"7836244"|"7861122"|"7941692"|"7970983"|"7991944"|"8001318"|"8024545"|"8055833"|"8065471"|"8065473"|"8068365"|"8072463"|"8074022"|"8082389"|"8086790"|"8347042"|"8572331"|"8645634"|"9229854"|"9335939"|"9400749"|"9519578"|"9542118"|"9588904"|"9652376"|"9696917"|"9710377"|"9727454"|"9785572"|"9846541").PN.				
L57	38	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L58	114	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L59	19976	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L60	0	(L22 L23) and (constituent near2 storage near2 location$1) and cop$4 and (erase near2 request)	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L61	19976	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L62	62	L26 and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L63	38	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L64	114	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM

		Richard)).inv.					
L65	1	(L28 L29) and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L66	1282	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L67	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L68	15	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L69	60	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L70	31	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L71	305	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L72	101	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L73	43	((controller$1) near10 (transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM

		moved or moving))					
L74	22	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L75	2210	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving)))	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L76	1228	(host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L77	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L78	15	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L79	58	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L80	30	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L81	304	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L82	101	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L83	43	((controller$1) near10	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10

		(transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or moved or moving))					11:06 AM
L84	22	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L85	131	("10445229"|"10552058"|"10552085"|"20030028733"|"20030065866"|"20050144413"|"20060022171"|"20070046681"|"20070058431"|"20070233939"|"20070260811"|"20070283428"|"20080082596"|"20080147964"|"20080155204"|"20080189485"|"20090036163"|"20090046533"|"20090083478"|"20090089482"|"20090089490"|"20090172219"|"20090172250"|"20090172257"|"20090198946"|"20090254705"|"20090271562"|"20090300015"|"20090327602"|"20100042655"|"20100115172"|"20100161882"|"20100162012"|"20100182838"|"20100191779"|"20100241866"|"20100262761"|"20100281230"|"20100287217"|"20100329011"|"20110033548"|"20110055445"|"20110161784"|"20110197023"|"20110238943"|"20110276756"|"20110296133"|"20110314209"|"20120033519"|"20120059972"|"20120066441"|"20120131381"|"20120159039"|"20120204079"|"20130007343"|"20130013852"|"20130019062"|"20130073816"|"20130111295"|"20130124793"|"20130166825"|"20130242425"|"20130290619"|"20130297852"|"20140101371"|	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM

		"20140122781"|"20140189209"|"20140208004"|"20140208062"|"20140215129"|"20140297949"|"20150067297"|"20150113203"|"20150134930"|"20150149789"|"20150193148"|"20150212938"|"20150261456"|"20150324264"|"20150347041"|"20150347296"|"20160018998"|"20160026564"|"20160342509"|"20160357462"|"20160364179"|"20170031699"|"5568423"|"5652857"|"5860082"|"6134631"|"7096378"|"7339823"|"7404031"|"7406563"|"7581078"|"7702846"|"7710777"|"7752381"|"7801561"|"7818489"|"7836244"|"7861122"|"7941692"|"7970983"|"7991944"|"8001318"|"8024545"|"8055833"|"8065471"|"8065473"|"8068365"|"8072463"|"8074022"|"8082389"|"8086790"|"8347042"|"8572331"|"8645634"|"9229854"|"9335939"|"9400749"|"9519578"|"9542118"|"9588904"|"9652376"|"9696917"|"9710377"|"9727454"|"9785572"|"9846541").PN.				
L86	0	"store the storage location release information"	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L87	0	"store the storage location release information"	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM
L88	38	(Radian near3 Memory near3 Systems).as.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L89	114	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	(US-PGPUB; USPAT)	OR	ON	ON	2022/02/10 11:06 AM
L90	5	(L54 L53) and ((error$1 defect$1)).clm.	(US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/02/10 11:06 AM

PE2E SEARCH - Search History (Interference)

There are no Interference searches to show.

Web Search History

date, time	web site	search string
2/10/2022 1:28:07 PM	Google	"a relationship between two things or situations, especially where one thing affects the other"
2/10/2022 1:28:13 PM	Google	define link
2/10/2022 1:28:51 PM	Google	define connection

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 16/997,471	Filing Date 08/19/2020	☐ To be Mailed

ENTITY: ☐ LARGE ☑ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $230 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	01/03/2022	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $50 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $240 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	LIE
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/BRENDA V HARRISON/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Electronic Acknowledgement Receipt

EFS ID:	44652258
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	03-JAN-2022
Filing Date:	19-AUG-2020
Time Stamp:	12:36:15
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Amendment/Req. Reconsideration-After Non-Final Reject	2021-12-31-OAResponse-RMS-02C1C1C14.pdf	448948 9c2c38b2b3b087f3253831da71663bca12f4ed30	no	25

Warnings:

Information:	
Total Files Size (in bytes):	448948

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.
National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.
New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application Serial Number:	16/997471
Confirmation Number:	1070
First Named Inventor:	Andrey V. Kuzmin
Title:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
Filing Date:	08/19/2020
Attorney Docket:	2020058 / RMS-02C1C1C14

APPLICANT RESPONSE TO 10/01/2021 OFFICE ACTION

Applicant notes that the three-month response date for the 10/01/2021 Office Action is 1/3/2022 since 1/1/2022 falls on a Saturday, Sunday, or Federal Holiday.

In The Claims.

Please amend the claims as follows. This listing replaces all prior listings of claims.

1. (Currently Amended) A memory controller to control nonvolatile memory, the nonvolatile memory having nonvolatile memory dies, each of the nonvolatile memory dies having a group of erase blocks, each erase block in the group such that it must be erased before constituent storage locations associated with that erase block can be reprogrammed, the memory controller comprising:

a host interface;

a memory interface;

logic to receive erasure requests from a host via the host interface, the erasure requests accompanied by host-selected designations of address space that is to be erased, and to responsively control erasure of an integer set of one or more of the erase blocks corresponding to each of the host-selected ~~designation~~ designations of address space, by issuing one or more erasure commands to the nonvolatile memory dies via the memory interface;

logic to, in connection with the erasure of the integer set of one or more erase blocks, detect defects in the integer set and, responsive to detecting the defects, to remap host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory; and

logic to receive data write requests from the host via the host interface and to responsively issue write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information;

wherein the memory controller is to direct data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

2. (Original) The memory controller of claim 1, wherein the host interface is also to receive data copy requests from the host, and wherein the memory controller is to responsively read data from one of the constituent storage locations associated with a first one of the erase blocks and write the data which has been read to one of the constituent storage locations associated with a second one of the erase blocks without sending the data which has been read to the host.

3. (Currently Amended) The memory controller of claim 2, wherein:

the host interface is also to receive data read requests;

the host interface is to receive the data write and the data read requests each accompanied by a respective host-supplied address;

each respective host-supplied address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the nonvolatile memory, the mapping such that different values of the field compel the memory controller to access respective storage structures within the nonvolatile memory, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data; and

each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the ~~nonvolatile~~ memory ~~system~~ controller to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier.

4. (Original) The memory controller of claim 3, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

5. (Original) The memory controller of claim 1, wherein the memory controller further comprises logic to maintain storage location release information for at least one erase block in one of the groups, the storage location release information to indicate a condition where at least one of the constituent storage locations for the at least one erase block has been programmed but no longer stores valid data, the condition such that the at least one of the constituent storage locations cannot be reprogrammed until the at least one erase block is erased.

6. (Currently Amended) The memory controller of claim 5, wherein the memory controller is to store the storage location release information for each of the constituent storage locations for each erase block in at least one of the groups and is to update the storage location release information as an automated response to requests received from the host, and wherein the memory controller further comprises logic to receive and process a host-request to transmit, to the host, the storage location release information for at least one of the constituent storage locations associated with the at least one of the groups.

7. (Currently Amended) The memory controller of claim 6, wherein the host is to communicate an address space selection together with the ~~host request~~ host-request to transmit, such that the ~~nonvolatile~~ memory ~~system~~ controller is to ~~transmit~~ transmit, to the ~~host~~ host, storage release information associated with a host-selected subset of storage structures, the host-selected subset corresponding to the address space selection.

8. (Original) The memory controller of claim 5, wherein the memory controller is to receive from the host a request to mark as released one or more of the constituent storage locations associated with at least one of the erase blocks, and wherein the logic to maintain is to responsively update the storage location release information to mark the one or more of the constituent storage locations as released.

9. (Currently Amended) The memory controller of claim 8, wherein:

the memory controller further comprises logic to monitor the storage location release information to detect memory space that represents candidates for garbage collection, and to transmit to the host an identification of the memory space that represents candidates for garbage collection;

the host, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests; and

the memory controller, in response to each of the one or more data copy ~~request~~ requests, is to read data from one of the constituent storage locations associated with a first one of the erase blocks and write the data to one of the constituent storage locations associated with a second one of the erase blocks without sending the data to the host.

10. (Original) The memory controller of claim 8, wherein:

the memory controller further comprises logic to monitor the storage location release information to detect memory space that represents candidates for erasure, and is to transmit to the host an identification of the memory space that represents candidates for erasure; and

the host is to issue at least one of the erasure requests in response to said identification.

11. (Currently Amended) A method of operation in a memory controller to control nonvolatile memory, where the nonvolatile memory has nonvolatile memory dies, each of the nonvolatile memory dies having

a group of erase blocks, each erase block in the group such that it must be erased before constituent storage locations associated with that erase block can be reprogrammed, the memory controller having a host interface, a memory interface and at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry, the method comprising:

receiving erasure requests from a host via the host interface, the erasure requests accompanied by host-selected designations of address space that is to be erased, and responsively controlling erasure of an integer set of one or more of the erase blocks corresponding to each~~the~~ of the host-selected ~~designation~~ designations of address space, by issuing erasure commands to the nonvolatile memory dies via the memory interface;

detecting, in connection with the erasure of the integer set of one or more erase blocks, defects in the integer set and, responsive to detecting the defects, remapping host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory; and

receiving data write requests from the host via the host interface and responsively issuing write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information; and

in connection with said receiving the data write requests and responsively issuing write commands, directing data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks.

12. (Original) The method of claim 11, further comprising also receiving via the host interface data copy requests from the host, and responsively reading data from one of the constituent storage locations associated with a first one of the erase blocks and writing the data which has been read to one of the constituent storage locations associated with a second one of the erase blocks without sending the data which has been read to the host.

13. (Currently Amended) The method of claim 12, wherein:

the method also comprises receiving data read requests;

each of the data write and the data read requests is accompanied by a respective host-supplied address, and each respective host-supplied address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the

nonvolatile memory, the mapping such that different values of the field compel the memory controller to access respective storage structures within the nonvolatile memory, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data; and

each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the ~~nonvolatile~~ memory controller~~system~~ to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier.

14. (Original) The method of claim 13, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

15. (Original) The method of claim 11, further comprising maintaining storage location release information for at least one erase block in one of the groups, the storage location release information to indicate a condition where at least one of the constituent storage locations for the at least one erase block has been programmed but no longer stores valid data, the condition such that the at least one of the constituent storage locations cannot be reprogrammed until the at least one erase block is erased.

16. (Currently Amended) The method of claim 15, further comprising:

storing the storage location release information for each of the constituent storage locations for each erase block in at least one of the groups;

updating the storage location release information as an automated response to requests received from the host; and

receiving and processing a host-request to transmit, to the host, the storage location release information for at least one of the constituent storage locations associated with the at least one of the groups.

17. (Currently Amended) The method of claim 16, wherein the host is to communicate an address space selection together with the ~~host~~ host-request to transmit, and wherein the method further comprises ~~transmitting~~ transmitting, to the host, ~~host~~ storage release information associated with a host-selected subset of storage structures, the host-selected subset corresponding to the address space selection.

18. (Original) The method of claim 15, further comprising receiving from the host a request to mark as released one or more of the constituent storage locations associated with at least one of the erase blocks, and responsively updating the storage location release information to mark the one or more of the constituent storage locations as released.

19. (Currently Amended) The method of claim 18, wherein:

the method further comprises monitoring the storage location release information to detect memory space that represents candidates for garbage collection, and transmitting to the host an identification of the memory space that represents candidates for garbage collection;

the host, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests; and

the method further comprises, in response to each of the one or more data copy ~~request~~ requests, reading data from one of the constituent storage locations associated with a first one of the erase blocks and writing the data to one of the constituent storage locations associated with a second one of the erase blocks without sending the data to the host.

20. (Original) The method of claim 18, wherein:

the method further comprises monitoring the storage location release information to detect memory space that represents candidates for erasure, and transmitting to the host an identification of the memory space that represents candidates for erasure; and

the host is to issue at least one of the erasure requests in response to said identification.

REMARKS

In the Office Action mailed 10/01/2021, the Examiner: (1) objected to the wording of claims 7 and 17 and required correction; (2) issued numerous specification objections; (3) rejected claims 1-20 under 35 USC 112, first paragraph, as allegedly lacking written description; (4) rejected claims 1-20 under 35 USC 112, second paragraph, as allegedly being indefinite, on various grounds; and (5) rejected claims 1-20 under 35 USC 102 as allegedly being anticipated by Selinger (USPB 20110161784).

Applicant notes that it has amended the claims to address item (1) above and antecedent basis issues raised by the Examiner in connection with the rejection under 35 USC 112, second paragraph. Most of the rest the Examiner's objections and rejections lack merit.

As an initial and nonlimiting example of this, the Examiner objects to the term "constituent storage locations" and asserts that this term lacks antecedent basis in the Specification and also, therefore, that a corresponding written description rejection and indefiniteness rejection also necessarily flows from this objection. However, the law is quite clear that (a) in haec verba correspondence in the Specification to claim terms is not required, and (b) adequate support may be shown through drawings as well as in other manners (e.g., including without limit inherent disclosure), and (c) claim terms are not necessarily unclear because the Specification does not use the same words arranged in the same order. Pertinent to the term "constituent storage locations," for example, the present application relates to flash memory, and the Examiner's position is apparently that (1) Applicant clearly discloses memory having "storage locations," (2) such storage locations are a constituent part of the disclosed memory, but (3) the failure to spell out "constituent storage locations" as a formally defined term verbatim used in the specification, constitutes a per se failure of support, lack of antecedent basis, and claim language clarity. This is not the law. Applicant agrees that claim features must find support in the Specification, but where normal English usage makes that support clear, as it does here, it is unnecessary for an Applicant to do more. For example, it is unnecessary for Applicant to add an express definition which expressly narrates that "constituent storage locations" means "storage locations" that are a "constituent part" of a memory die or device, when this would already be understood from the Specification by those versant in English; this is to say, Applicant is entitled to assume that readers have a baseline understanding of the English language and that, with such an understanding, correspondence between disclosed structures and claim features is already clear. Tellingly, with this as well as most of the other outstanding objections and rejections, the Examiner points to no confusion in fact, and simply repackages his baseline position that there is no in haec verba support. To the contrary, as expressly cited by Applicant, in haec verba support is not required from any of a support, clarity or antecedent basis perspective; all that is required is that those having an ordinary level of skill in the art would be able to find associated structure. This is why

Applicant has quoted to the Examiner, and continues to quote to the Examiner, statements in the MPEP and caselaw stating that in haec verba support *is not required*.

To cite another example, the Examiner apparently acknowledges that Applicant's disclosure teaches that a host can send a memory access request in a manner accompanied by a logical address, but the Examiner apparently takes the position that a "host-supplied address" (i.e., supplied with a request) is unclear unless there is a formal definition of this exact wording combination found in the Specification; once again, and as with many of the claim features cited by the Examiner, this claim feature is used in accordance with its normal English meaning, i.e., in this case referring to an "*address*" which is "*supplied*" by a "*host*;" Applicant is not required to presume that a reader would be unable to understand this normal English language usage in absence of an express definition and, once again, the various objections and rejections seem to indicate that the Examiner is clearly misinterpreting the law (e.g., the MPEP) on this issue. So as to not push this application to yet another appeal to resolve what are essentially non-issues, the Examiner's cooperation is respectfully requested in reviewing these remarks and withdrawing these baseless rejections and objections and in not reasserting, repeatedly, exactly the same meritless positions case-by-case.

Below, Applicant addresses each enumerated ground of objection/rejection (other than the objection to claims 7 and 17, which should be rendered moot by the amendment above), under a separate heading.

A. THE OBJECTIONS TO THE SPECIFICATION ARE MERITLESS AND NEED TO BE WITHDRAWN.

As to the objections to the Specification, it is again noted that in haec verba correspondence between language of the Specification and the claims is *not required*; as long as those having ordinary skill in the art would understand that the Specification provides structure which operates in the manner of the claim language, this is all that is required. See MPEP 2163.02:

> "The subject matter of the claim need not be described literally (i.e., using the same terms or *in haec verba*) in order for the disclosure to satisfy the description requirement. If a claim is amended to include subject matter, limitations, or terminology not present in the application as filed, involving a departure from, addition to, or deletion from the disclosure of the application as filed, the examiner should conclude that the claimed subject matter is not described in that application. This conclusion will result in the rejection of the claims affected under 35 U.S.C.112, first paragraph - description requirement, or denial of the benefit of the filing date of a previously filed application, as appropriate." (Emphasis in original)

This is to say, the requirements of an applicant's specification in terms of supporting claim language is that it is enough that a specification provides support for a claimed feature that can be identified by those having an ordinary level of skill in the art. As stated explicitly by MPEP 608.01(o), in pertinent part:

> "While an applicant *is not limited to the nomenclature used in the application as filed*, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims *so as to have clear support **OR** antecedent basis* in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103. *Note that examiners should ensure that the terms* and phrases used in claims presented late in prosecution of the application (including claims amended via an examiner's amendment) *find clear support **OR*** antecedent basis in the description so that the meaning of the terms in the claims may be ascertainable by reference to the description." (emphasis added by Applicant).

The use of the disjunctive "OR" in this paragraph is significant, i.e., once again, Applicant does not have to use exactly the same language as long as corresponding structure is disclosed and one of ordinary skill in the art would be able to find correlation between that structure and claim feature. This is to say, adequate antecedent basis already exists in Applicant's Specification if one having ordinary skill in the art would be able to find support in the Specification for the associated claim language, even if precisely the same language of the claim isn't used.

In this case, each and every one of the Examiner's contentions as to alleged lack of specification antecedent basis is untenable, i.e., anyone having any skill in the art reviewing Applicant's Specification would clearly be able to identify support for recited claim terms.

1. The Examiner at page 3 of the Office Action firstly asserts that the Specification doesn't provide support for "constituent storage locations;" as discussed above, the Examiner's position is unreasonable, i.e., the Examiner is arguing effectively that a claim should be objected to even when it uses language having a clear ordinary meaning in the English language, unless it has a formal definition or is used in haec verba in the Specification. This position is unreasonable. This is to say, there is ample antecedent basis already in the Specification for a memory having "constituent storage locations" and Applicant need do nothing more. The term "constituent storage locations" refers, unsurprisingly, to "storage locations" that form a "constituent" part of the recited memory. The Examiner does not dispute that Applicant's specification clearly discloses memory, for example, dies of flash memory having storage for many erase units, pages and logical blocks. As a nonlimiting example of express support, paragraph 49 of Applicant's Specification, in describing addressing of storage location in various dies of flash memory, says that fields of an address arriving from a host "can identify for example the number of a communication channel within the flash memory interface 205, the number of a device 207 connected to that channel, the number of a die 209 within that device 207, the number of a plane 211 located on the die 209, the *location* of a block within the die 209, and the *location* of a page within the block." This is to say, the entirety of Applicant's disclosure is directed to nonvolatile memory (e.g., flash memory) having locations where data is stored, and one having ordinary skill in the art, in reviewing either of Applicant's

Specification or Drawings, would immediately recognize clear antecedence for "constituent storage locations" of nonvolatile memory.

Because clear antecedence is already provided by the Specification, this ground of objection is clearly erroneous. Withdrawal is therefore requested.

2. The Examiner at page 3 of the Office Action refers to "nonvolatile memory system" and argues that "there is no mention of a nonvolatile system in the specification as claimed." Applicant finds this assertion of the Examiner nothing short of *unbelievably frivolous*. This is to say, there are many, many references to "nonvolatile" memory in the Specification and to systems based on nonvolatile memory, and those having an ordinary level of skill in the art would clearly recognize explicit antecedence for a nonvolatile memory system.

Paragraph 7 of the Specification states, for example:

> "[0007] What is needed is a mechanism for improving control and data bandwidth for flash memory and other forms of *nonvolatile memory*. More particularly, a mechanism is needed that reduces control and data bandwidth encumbrances created by memory management functions and thereby decreases the issues referenced above. Still further, a need exists for a memory management scheme that does not create excessive write amplification and bandwidth competition. Finally, a need exists for a flash/*nonvolatile memory architecture* that has more consistent latency, is conductive to structured pipelining of commands, and permits ubiquitous management of SSDs and other forms of memory in direct-attached and network storage applications. Techniques provided by this disclosure satisfy these needs and provide further related advantages."

Paragraph 40 adds:

> "[0040] This disclosure therefore also provides *storage systems*, hosts, network-ready storage servers, methods, software and other implementations consistent with the principles introduced above. Generally speaking, the techniques disclosed herein can be implemented in any one or more of these components, in a manner where a component can be designed, manufactured and/or sold for optional inclusion in a complete storage system having a host, memory controller and memory. Consistent with these principles, *one implementation of the techniques referenced above is in a storage system having at least one memory controller and memory*. *Each memory controller and its associated memory can be bundled together as a storage drive* and, in one implementation, the storage system includes plural storage drives, at least one of which is a solid-state storage drive (SSD) *based in nonvolatile memory* such as flash memory."

Thus, for example, when the Examiner asserts that "there is no mention of a nonvolatile system in the specification as claimed," this assertion is clearly baseless. This is to say, anyone reading the Specification, or drawings, would immediately see support for a nonvolatile memory system. Because there is express antecedent basis in the Specification for a nonvolatile memory system, this ground of rejection is *unreasonable*.

Applicant notes for the record that, following amendment, to resolve antecedent basis issues, the word "system" is no longer used by the claims (i.e., this objection should be moot, in addition to being unreasonable in the first instance).

3. At page 3 of the Office Action, the Examiner objects to the term "respective host-supplied addresses" and argues that there is no mention of host-supplied addresses in the Specification. This objection is, likewise, completely baseless.

The present application relates to memory. The term "host-supplied address" refers, unsurprisingly, to an "*address*" which is "*supplied*" by "*a host*;" this is clearly disclosed by Applicant's Specification, which once again provides plenty of existing antecedence for this term. Once again, the Examiner's objection is rooted in an attempt to apply a strict in haec verba requirement.

The Examiner's position is inappropriate, i.e., Applicant should not need to formally define the meaning of an expression plainly already understood in the English language, and the Examiner's insistence on an in haec verba definition of ordinary language is in clear disregard of the MPEP and applicable rules pertaining to examination. One of ordinary skill in any field pertinent to memory or any form of data storage would clearly understand what is meant by a "host supplied" address, and clearly, Applicant's Specification discloses embodiments where a host supplies an address in connection with a read, write, or specifically-discussed other forms of requests. Applicant's Abstract, as one nonlimiting example, states "[b]y redefining host-controller responsibilities in this manner, much of the overhead associated with FTL functions can be substantially removed from the memory controller, with the host directly specifying physical addresses." The second paragraph of the detailed description states "[t]o provide an example of an action a host could take based on this exemplary information, a host armed with this information can issue an erase command directly addressed to a specific EU of memory." Paragraph 36 of the Specification extensively references host addressing in association with data access commands, and there are approximately 200 references to "addressing" in the Specification, e.g., addressing used by the host. One of ordinary skill would therefore unambiguously and clearly understand that the Specification clearly describes that a host provides, or supplies, an address with its various requests, that serve to identify operands for those requests within memory managed by a memory controller and, once again, in haec verba correspondence *is not required*. Applicant therefore, once again, requests that the Examiner withdraw this meritless objection.

4. At page 3 of the Office Action, the Examiner twice objects to the term "host-request" and argues that there is no mention of "host-request," and that this therefore constitutes a failure of adequate antecedent basis.[1] Once again, it is immediately apparent that the Examiner is attempting to apply a strict

[1]E.g., see claims 6-7, a 'host-request' to transmit storage location release information to the host.

in haec verba standard in disregard of the Rules. For example, Applicant's Specification uses the term "request" to refer to communications from the host to a memory controller approximately *sixty* times. See, e.g., paragraph 34 ("a host within this context can be any integrated circuit or other apparatus (e.g., such as virtual or actual machine) that is separate from the memory controller and that transmits a *request* to the memory controller for some type of processing or response"), and paragraph 49 ("the controller can serve detailed information to the host for each subdivision of memory, and the host can also issue query *requests* to the memory controller, designed to assist the host with management of the bandwidth-consuming functions referenced above"), emphasis added. See also paragraph 53:

> "Generally speaking, the host can *request* (a) return of raw information for the entire memory space managed by the memory controller, or for specific address ranges or EU ranges, or (b) processing or filtering of that data by the memory controller. For example, as referenced previously, the memory controller can be provided with logic operable to receive and interpret host commands for lists of blocks, e.g., the "top ten" candidates for garbage collection, ranked in order of page (under) utilization. Since the purpose of such an operation is to identify EUs for erasure, a memory controller receiving such a command interrogates the metadata storage to (a) identify EUs that are at least partially in use, (b) identify the extent of page utilization for each such EU, and (c) order a set of EUs that are candidates for erasure in the order of greatest number of released pages. The query/management logic block 317 uses internal registers 319 to manage functions like this and, when the requested processing is complete, the query/management logic block sends a reply message to the host with the requested information. Note that the host interface 305 includes buffers that receive this data, and queue the data for transmission to the host (e.g., as a reply packet that may be distinguished from read data). The mentioned example is only one type of query that can be processed by the host, i.e., there are many types of requests and queries that can be processed by a memory controller of the present disclosure. A skilled designer can implement any type of management processing desired. This *request* is an example of a synchronous query, because data is on-hand for the memory controller, and because the memory controller returns requested data as soon as its query processing is complete. In a simplified case, the host can *request* return of raw metadata."

These citations refer to the type of operation referenced by the claims in question, e.g., a host can issue a request to the memory controller to return information such as raw metadata (e.g., page release information) for a specified structure or address range. Once again, there is no requirement for Applicant to set forth a formal definition of "host-request" to explain that this refers to a "request" from a "host," and those having ordinary skill in the art would have no trouble understanding both what is being referred to and what structure articulated by the Specification can correspond to this claim language.[2]

---

[2]As should also be understood by PTO personnel, many applicants frequently need to use various language expressions to refer to disclosed features; for example, in the context of the present disclosure, there are many types of "requests" that can be issued by a host, e.g., write requests, read requests, erase requests, garbage collection requests and so forth. The use of various nomenclature to distinguish one request from another in the claims is no different from an applicant, for example, that might want to claim a "first" element and a "second" element, notwithstanding failure of a specification to formally define which elements the "first" and "second" labels correspond to. This is to say, once again, as stated by the MPEP, strict in haec verba correspondence is not required.

Because Applicant's specification clearly discloses that a host issues a request for information returns, including the information at issue in the claims, and because one of ordinary skill in the art would clearly identify support for a "host-request" for this information, this objection should also be withdrawn.

5. Finally, also at page 3 of the Office Action, the Examiner objects to the expression "host-selected subset" (e.g., of 'storage structures,' as referenced by claim 7), and argues that there is 'no mention' of such a subset. However, once again, this is entirely false. As just quoted in the last section (see paragraph 53 of Applicant's Specification, quoted above), there is express support for a host that issues a request accompanied by an address range that causes a memory controller to return raw metadata for a set of structures encompassed by the address range. See also paragraph 49 of Applicant's Specification, e.g., the addressing fields used by a host "can identify for example the number of a communication channel within the flash memory interface 205, the number of a device 207 connected to that channel, the number of a die 209 within that device 207, the number of a plane 211 located on the die 209, the location of a block within the die 209, and the location of a page within the block") and it is this addressing, and host knowledge that specific fields map to specific structures, that permits a host to query raw metadata return for a HOST-SELECTED SUBSET of structures. It is noted that these operations are referenced in the explicit context of a host-controlled erase operation, e.g., see paragraphs 88 et seq. of the Specification, i.e., the host requests a raw metadata return from the memory controller, and the memory controller returns information (e.g., a 'bitmap') that permits the host to make a decision about what it wants to erase by explicit request (i.e., at times of the host's choosing, which may not be at the same time as garbage collection is performed, or which may be performed completely independently of garbage collection/data moves). Once again, there is copious, extensive antecedent basis for the claim features referenced by the Examiner and the Rules require that Applicant do nothing more. Therefore, this objection also is unsupportable.

Thus, as demonstrated above, one having ordinary skill in the art would recognize that Applicant's Specification has ample antecedent basis for each of the cited claim terms, and withdrawal of the specification objection is requested.

B. <u>THE REJECTION UNDER 35 USC 112, FIRST PARAGRAPH, IS MERITLESS AND NEEDS TO BE WITHDRAWN.</u>

The Examiner argues at pages 4 et seq. of the Office Action that when there is a lack of in haec verba support, there must necessarily (i.e., per se) also be a written description violation.

For example, on page 4 of the Office Action, the Examiner argues that "the original specification discloses constituent pages, constituent space, constituent commands and constituent memory cell (sic)" "however, there is no mention" (which one of these) "is meant to be a constituent storage location."

This line of reasoning is improper and defies explanation; first of all, "commands" are not a "storage location," as would be understood by "anyone." Second, the term "storage location" can encompass multiple different structures, and there is nothing improper about this. For example, as stated by the MPEP 2173.02 in connection with indefiniteness:

> "A broad claim is not indefinite merely because it encompasses a wide scope of subject matter provided the scope is clearly defined. Instead, a claim is indefinite when the boundaries of the protected subject matter are not clearly delineated and the scope is unclear. For example, a genus claim that covers multiple species is broad, but is not indefinite because of its breadth, which is otherwise clear. "

As expressed, there is nothing improper about a claim feature covering multiple structures, as long as there is support, and one of ordinary skill in the art can determine what is or is not infringement. In this regard, a storage location can be identified by an address range, or a specific page address, or an erase unit address, etc.; this concept would be understood by one familiar with technology generally, i.e., a storage location in memory would be understood as a place where data can be stored. Applicant discloses these structures, and therefore supports constituent storage locations, and therefore satisfies the written description requirement.

As verbatim stated by the MPEP "[t]he subject matter of the claim ***need not be described literally (i.e., using the same terms or in haec verba)*** in order for the disclosure to satisfy the description requirement" and "[w]hile there is no *in haec verba* requirement, newly added claims or claim limitations must be supported in the specification through express, implicit, or inherent disclosure." [See generally MPEP 2163.] As stated by MPEP 2163.02:

> "Whenever the issue arises, the fundamental factual inquiry is whether the specification conveys with reasonable clarity to those skilled in the art that, as of the filing date sought, applicant was in possession of the invention as now claimed. See, e.g., *Vas-Cath, Inc. v. Mahurkar*, 935 F.2d 1555, 1563-64, 19 USPQ2d 1111, 1117 (Fed. Cir. 1991). An applicant shows possession of the claimed invention by describing the claimed invention with all of its limitations using such descriptive means as words, structures, figures, diagrams, and formulas that fully set forth the claimed invention. *Lockwood v. American Airlines, Inc.*, 107 F.3d 1565, 1572, 41 USPQ2d 1961, 1966 (Fed. Cir. 1997). Possession may be shown in a variety of ways including description of an actual reduction to practice, or by showing that the invention was "ready for patenting" such as by the disclosure of drawings or structural chemical formulas that show that the invention was complete, or by describing distinguishing identifying characteristics sufficient to show that the applicant was in possession of the claimed invention. See, e.g., *Pfaff v. Wells Elecs., Inc.*, 525 U.S. 55, 68, 119 S.Ct. 304, 312, 48 USPQ2d 1641, 1647 (1998); *Regents of the University of California v. Eli Lilly*, 119 F.3d 1559, 1568, 43 USPQ2d 1398, 1406 (Fed. Cir. 1997); *Amgen, Inc. v. Chugai*

Pharmaceutical, 927 F.2d 1200, 1206, 18 USPQ2d 1016, 1021 (Fed. Cir. 1991) (one must define a compound by "whatever characteristics sufficiently distinguish it").

Thus, as stated, if Applicant provides support for "storage locations" that are a constituent part of the claimed memory (e.g., such is shown in a figure, even if not stated in text), then Applicant satisfies written description as a matter of law. It is not required that Applicant use the same verbatim label in the Specification in order to satisfy the written description requirement.

The Examiner here expressly admits that Applicant supports these features, e.g., he states verbatim on page 4 of the Office Action that "[t]he original specification discloses … constituent pages, constituent space … and constituent memory cell (sic)." Since these are unambiguously 'storage locations,' Applicant necessarily discloses "*constituent storage locations*." Applicant again notes that the present Specification relates to nonvolatile memory, e.g., flash memory, where data is stored in many storage locations that are a constituent part of the claimed flash memory. Because this is clearly supported by the present Specification and Drawings, and because those having an ordinary level of skill in the art would clearly recognize Applicant's possession of this subject matter, this rejection is completely baseless and needs to be withdrawn.

At pages 4-5 of the Office Action, the Examiner argues that there is no support for "host-selected designations" of address space. Applicant refers the reader to the discussion above which was provided in connection with the discussion of the Specification objections; the present Specification discloses that a host can specify an address range, or an address that is known to the host to correspond to one or more structures (e.g., using discrete address fields) and can request a return of raw metadata (i.e., storage location release information, such as page release information) for these specified memory locations. See once again paragraph 53:

> "Generally speaking, the host can *request* (a) return of raw information for the entire memory space managed by the memory controller, or for specific address ranges or EU ranges, or (b) processing or filtering of that data by the memory controller. For example, as referenced previously, the memory controller can be provided with logic operable to receive and interpret host commands for lists of blocks, e.g., the "top ten" candidates for garbage collection, ranked in order of page (under) utilization. Since the purpose of such an operation is to identify EUs for erasure, a memory controller receiving such a command interrogates the metadata storage to (a) identify EUs that are at least partially in use, (b) identify the extent of page utilization for each such EU, and (c) order a set of EUs that are candidates for erasure in the order of greatest number of released pages. The query/management logic block 317 uses internal registers 319 to manage functions like this and, when the requested processing is complete, the query/management logic block sends a reply message to the host with the requested information. Note that the host interface 305 includes buffers that receive this data, and queue the data for transmission to the host (e.g., as a reply packet that may be distinguished from read data). The mentioned example is only one type of query that can be processed by the host, i.e., there are many types of requests and queries that can be processed by a memory controller of the present disclosure. A skilled

> designer can implement any type of management processing desired. This request is an example of a synchronous query, because data is on-hand for the memory controller, and because the memory controller returns requested data as soon as its query processing is complete. In a simplified case, the host can request return of raw metadata."

The host-specified designation of address (or an address range) is a *host designation* of a portion of memory space for which metadata is to be returned to the host, and would clearly be understood as such by those having ordinary skill in the art. Therefore, once again, adequate support clearly exists in the Specification, and the rejection is unreasonable, and must be withdrawn. That is, once again, this rejection is solely predicated on this Examiner's improper insistence for in haec verba correspondence between claims and Specification; such is not required, as already discussed, and there is consequently no basis for this rejection.

At page 5 of the Office Action, the Examiner (*completely inappropriately, and unbelievably*) cites "nonvolatile memory system" and reasons "[t]he specification discloses the use of a nonvolatile memory but there is no mention as to what is considered a nonvolatile memory system as claimed." It is firstly once again noted that the claims no longer use the expression "nonvolatile memory system," and therefore this rejection should in all events be moot.

Secondly however, and for purposes of the record, Applicant identifies this rejection as in all events *completely unreasonable*; as expressly discussed in the section above (relating to objections to the specification) Applicant expressly identifies a nonvolatile memory system as a device having a memory controller and memory. As a further example of clear and notorious support, paragraph 134 of Applicant's Specification begins "FIGS. 14-19 are used to illustrate functions introduced above in an exemplary storage *system*." Paragraph 139 continues "the storage server can collect wear metrics for all flash memory managed as well as for *other types of nonvolatile memory*, if present."[3] FIG. 14 is reproduced below for the reader's convenience:

[3]Flash memory is introduced by the Specification as a form of nonvolatile memory, and would be understood by those having any experience with memory or computer-related technology to be a form of nonvolatile memory.

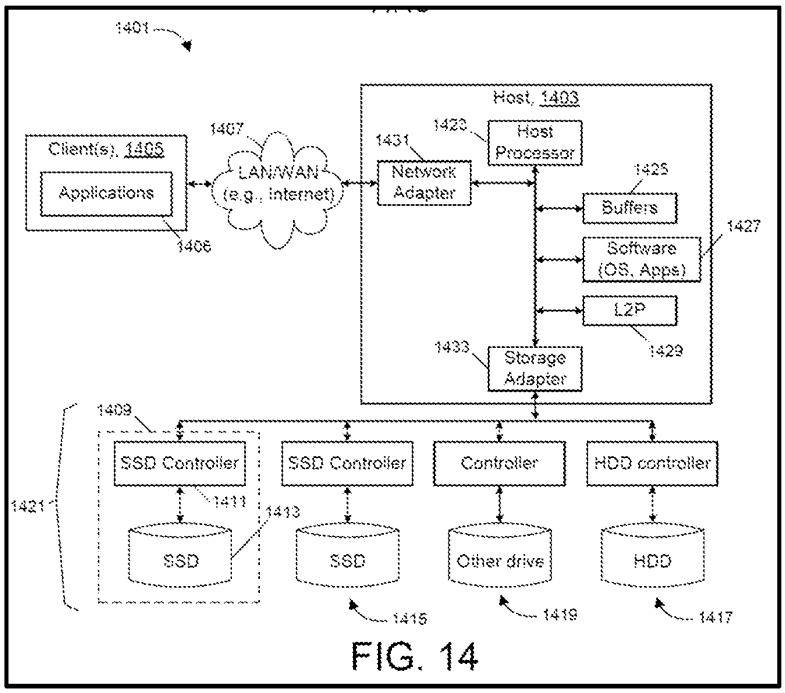


FIG. 14

In this regard, the term "system" would be well-understood by those having ordinary skill in the art, for example, as a set of interconnected parts that cooperate to perform some type of function. As non-limiting examples, a storage drive (such as 1409) would be a nonvolatile memory system, as would drives 1415, 1417, and potentially aggregate 1421, to the extent rooted in nonvolatile memory. As quoted earlier, the Specification expressly indicates that a system typically has a memory controller and memory managed by that memory controller. Therefore, possession of these features would clearly be identified, in the original disclosure as filed, notwithstanding any argument of the Examiner, and this rejection is meritless.

Because this rejection is moot, not to mention clearly erroneous, and because copious support clearly exists for a *nonvolatile memory system*, withdrawal of this rejection is requested.

Also at page 5 of the Office Action, the Examiner cites lack of in haec verba support for (a) host-supplied address, (b) host-request, and (c) host-selected subset of storage structures; the Examiner reasons that "[t]he specification fails to disclose anything as being host-supplied as claimed let alone a host-supplied address," "[t]he specification does disclose the host sending multiple different types of request to the memory controller to be performed on memory" but "[t]here is no mention of any of the requests being considered a host-request as claimed," and "[t]he specification fails to disclose the storage structures that are host-selected as claimed." Even a cursory review of the Examiner's position and the Specification demonstrates that the Examiner is once again arguing that a strict in haec verba standard

should be applied.[4] To the contrary, copious support exists for these claim features. First, as already quoted herein repeatedly (and in dozens of other Office Action responses to this Examiner in sister applications), ***in haec verba correspondence IS NOT REQUIRED***. This is to say, the claims reference a host-issued request for metadata, where an address is supplied by the host (e.g., to designate a structure or address range) and where the memory controller uses that address supplied by the host to identify and return specific metadata, or derivative information. These features clearly disclose a "host-request" (to the memory controller), a "host-supplied address," and "wherein the host is to communicate an address space selection together with the host-request to transmit, such that the memory controller is to transmit, to the host, storage release information associated with a host-selected subset of storage structures, the host-selected subset corresponding to the address space selection." As once again recited by paragraph 53 of Applicant's Specification, as non-exhaustive support for the claim features:

> "Generally speaking, the host can *request* (a) return of raw information for the entire memory space managed by the memory controller, or for specific address ranges or EU ranges, or (b) processing or filtering of that data by the memory controller. For example, as referenced previously, the memory controller can be provided with logic operable to receive and interpret host commands for lists of blocks, e.g., the "top ten" candidates for garbage collection, ranked in order of page (under) utilization. Since the purpose of such an operation is to identify EUs for erasure, a memory controller receiving such a command interrogates the metadata storage to (a) identify EUs that are at least partially in use, (b) identify the extent of page utilization for each such EU, and (c) order a set of EUs that are candidates for erasure in the order of greatest number of released pages. The query/management logic block 317 uses internal registers 319 to manage functions like this and, when the requested processing is complete, the query/management logic block sends a reply message to the host with the requested information. Note that the host interface 305 includes buffers that receive this data, and queue the data for transmission to the host (e.g., as a reply packet that may be distinguished from read data). The mentioned example is only one type of query that can be processed by the host, i.e., there are many types of requests and queries that can be processed by a memory controller of the present disclosure. A skilled designer can implement any type of management processing desired. This request is an example of a synchronous query, because data is on-hand for the memory controller, and because the memory controller returns requested data as soon as its query processing is complete. In a simplified case, the host can request return of raw metadata."

Thus, for each feature cited by the Examiner, support clearly exists, and there is no basis for a written description rejection. This is to say, contrary to the Examiner's arguments, each claim feature cited by the Examiner is very clearly supported, and the rejection here also is unreasonable.

---

[4]E.g., the Examiner states verbatim "[t]he specification *does disclose the host sending multiple different types of request to the memory controller to be performed on memory*" but "[t]here is no mention of any of the requests being considered a host-request as claimed;" he clearly acknowledges support for requests, and is complaining that the *exactly identical* wording is not used in the Specification.

C. THE REJECTION UNDER 35 USC 112, SECOND PARAGRAPH, IS MERITLESS AND NEEDS TO BE WITHDRAWN.

Applicant initially notes that there are two general classes of rejections for which the Examiner cites 35 USC 112, second paragraph: (1) rejections founded on assertions of lack of antecedent basis within the claims; and (2) rejections that bootstrap the Examiner's lack of in haec verba support allegations, i.e., the Examiner argues that one of ordinary skill in the art would (for unexplained reasons) be unable to interpret the claims, based solely on the fact that the Examiner asserts there is no in haec verba support.

As to the first class of rejections, Applicant notes that it has, by the present response, amended the claims so as to (1) remove reference to 'nonvolatile memory system,' (2) revise reference to 'designation' and (3) address antecedent basis issues identified by the Examiner, e.g., at pages 12-14 of the Office Action; it is submitted that the amendments set forth above should be fully self-explanatory and should moot the associated grounds of rejection. Therefore, withdrawal of the Examiner's positions on the corresponding issue is requested.

As to the second class of rejections, and relative to the same positions advanced by the Examiner for "host-selected," "host-request," "constituent storage locations," "host-supplied address," and other items discussed above in connection with the objection to the Specification, these grounds of rejection for alleged lack of clarity are completely unreasonable - the Examiner here is arguing that any time there is no in haec verba support, claim terms are per se confusing and, that as a consequence, the Examiner thinks incorrectly that he no longer needs to demonstrate any credible argument that claim wording itself would be unclear. This is clearly not the law, and it once again ignores the many admonitions in case law and the MPEP that *in haec verba correspondence is not required*. Applicant also notes that Applicant has argued this issue to this Examiner ad nauseum, and the Examiner continues to flatly ignore the law on this issue and promulgate, repeatedly, the same completely baseless and unreasonable rejections (see, e.g., any of the copending sister applications).

For example, at pages 6 et seq. of the Office Action, the Examiner asserts (preposterously) that one of ordinary skill in the art would be unable to understand the terms "constituent storage locations," "nonvolatile memory system," "host-selected designation of address space," "host-supplied address" and "host-request" SOLELY because the Specification does not use these terms verbatim. Each of these grounds of rejection is unreasonable.

For example, relative to "host-supplied address," as discussed above, one of ordinary skill in the art would clearly understand that this refers to an "*address*" which is "*supplied*" by a "*host*." There is nothing at all vague or confusing in the claim terms at issue, and there is no plausible reason cited by the Examiner in the Office Action as to how or why one of ordinary skill in the art would find this language

vague or in any way confusing. This is to say, these terms are clearly understood in the English language, they correspond to clear description in the Specification and drawings (as discussed above) where a host provides an address in association with requests (e.g., data access, garbage collection and/or erase requests). Consequently, this ground of rejection is baseless and needs to be withdrawn.

Similarly, relative to "constituent storage locations," the Specification clearly discloses for example, that dies of flash memory can have one or more planes, which can each have erase units, which can each have pages, and so forth. The Examiner also clearly acknowledges this disclosure, e.g., he states verbatim on page 4 of the Office Action that "[t]he original specification discloses … constituent pages, constituent space … and constituent memory cell (sic)." Since these are unambiguously 'storage locations' and would be understood as such by those having ordinary skill in the art, there is no indefiniteness issue; the Examiner has not articulated any grounds of confusion in fact, i.e., a 'storage location' in memory is a 'location' at which data can be 'stored,' and it would be interpreted as such by those having an ordinary level of skill in the art. This is not a confusing term, and there is no specific ground of vagueness even alleged by the Examiner, that is, apart from his assertion that the Specification does not use *exactly the same wording*. The same is also true for each of the other claim features cited by the Examiner.

As variously stated by MPEP 2173.02 (emphasis added):

> "A decision on whether a claim is indefinite under 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph ***requires*** a determination of whether those skilled in the art would understand what is claimed when the claim is read in light of the specification. Power-One, Inc. v. Artesyn Techs., Inc., 599 F.3d 1343, 1350, 94 USPQ2d 1241, 1245 (Fed. Cir. 2010); Orthokinetics, Inc. v. Safety Travel Chairs, Inc., 806 F.2d 1565, 1 USPQ2d 1081 (Fed. Cir. 1986)…."
>
> "The ***essential inquiry*** pertaining to this requirement is whether the claims set out and circumscribe a particular subject matter with a reasonable degree of clarity and particularity.... Definiteness of claim language must be analyzed, not in a vacuum, but in light of:
>
> (A) The content of the particular application disclosure;
>
> (B) The teachings of the prior art; and
>
> (C) The claim interpretation that would be given by one possessing the ordinary level of skill in the pertinent art at the time the invention was made.
>
> In reviewing a claim for compliance with 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph, the examiner must consider the claim as a whole to ***determine whether the claim apprises one of ordinary skill in the art*** of its scope and, therefore, serves the notice function required by 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph, ***by providing clear warning to others as to what constitutes infringement of the patent***."

As these recitations make clear, the Examiner's inquiry is to determine whether the claim is worded in a manner that provides metes and bounds for determining what does or does not constitute an infringement of the claims; if using broadest reasonable interpretation, the language is found to be reasonably clear, *then the Examiner **may not** reject the claims (under 35 USC 112, Second Paragraph)*. As

indicated above, the Examiner has not demonstrated any justification for asserting that the claim language in question is confusing - the Examiner's arguments, that failure to use exactly the same language in the Specification as is found in the claims always necessitates an indefiniteness rejection, flies in the face of applicable law, and is improper.

Rather, as stated above, the test for indefiniteness is simply whether the language of the claims is intelligible, in English, such that one of ordinary skill would understand where the boundaries of infringement lie. As noted above by Applicant, each term cited by the Examiner (e.g., 'host-supplied,' 'nonvolatile memory system,' 'constituent storage location,' 'host-request,' 'host-selected,' etc.) would be readily understood and clearly understood by those having skill in the art, notwithstanding any of the Examiner's arguments. Therefore, as argued by Applicant, ***again***, each of these rejections is meritless and needs to be withdrawn.

D. THE REJECTION UNDER 35 USC 102 IS ERRONEOUS AND MUST BE WITHDRAWN.

The Examiner rejected all pending claims over Selinger under 35 USC 102 as allegedly anticipated (USPB 20110161784). As is known, a rejection under 35 USC 102 is ONLY APPRORIATE if a cited reference shows all of the limitations of a rejected claim, arranged as in the rejected claim. Selinger CLEARLY fails to satisfy this requirement. This is to say, Applicant specifically asserts that Selinger clearly fails to disclose or teach at least those limitations of the independent claims which are underlined below (with exemplary reference to rejected independent claim 1). Because the underlined claim features are found in each independent claim and thus, each pending claim of record, Selinger does not anticipate any claim of record, and the rejection is clearly unsupportable.

> 1. A memory controller to control nonvolatile memory, the nonvolatile memory having nonvolatile memory dies, each of the nonvolatile memory dies having a group of erase blocks, each erase block in the group such that it must be erased before constituent storage locations associated with that erase block can be reprogrammed, the memory controller comprising:
>
> a host interface;
>
> a memory interface;
>
> *logic to receive erasure requests from a host via the host interface, the erasure requests accompanied by host-selected designations of address space that is to be erased*, and to responsively control erasure of an integer set of one or more of the erase blocks corresponding to each of the host-selected designations of address space, by issuing one or more erasure commands to the nonvolatile memory dies via the memory interface;
>
> *logic to, in connection with the erasure of the integer set of one or more erase blocks, detect defects in the integer set and, responsive to detecting the defects, to remap host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory;* and
>
> logic to receive data write requests from the host via the host interface and to responsively issue write commands to, and effectuate related data transfer with,

the nonvolatile memory, each of the data write requests being accompanied by the host address information;

wherein the memory controller is to direct data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

As should be apparent, the pending claims relate to a structure where a memory controller detects a bad block during an erasure and remaps host address information, such that, when a write request is later received which might designate that bad block, the memory controller redirects that write request to the (substitute) erase block.[5] Notwithstanding anything cited by the Examiner in the Office Action, Applicant fails to see how or where Selinger shows any of these features.

The Examiner argues that Selinger teaches erasure requests from a host, and variously cites to FIG. 7, FIG. 12A-12B, FIG. 9, and paragraphs 62, 86, 24, 73-74, and 164-165; however, Applicant does not see how Selinger supports the Examiner's allegations. For example, the Examiner references paragraphs 164-165 of Selinger, at which Selinger indicates that a copy back operation can feature a bit which is set to, in connection with a request to copy, cause the memory controller to erase a source location after data is moved. This operation in turn is apparently triggered by Selinger's detection of a read error or cold data. However, this is not the same thing as an erase request as claimed, i.e., Applicant sees no structure in Selinger where Selinger's host can simply instruct a memory controller to erase host-selected address space. In this regard, the Examiner appears to be arguing that Selinger's copy back operation is effectively 'the same thing,' but it is not, i.e., Selinger does not, that Applicant can identify, disclose any structure whereby a host "knows" that there is previously-written, now free, unerased space in flash memory, and where a host can issue requests to simply erase that space at a time of the host's convenience. As Applicant reads Selinger, Selinger appears to relate a memory controller that can identify that data should be moved (e.g., cold data, or data with a read error), such ostensibly being the trigger for Selinger's copy

[5]Applicant points out that (1) for systems that rely on conventional physical (wire) addressing, such that there is no address translation, a host simply always addresses the block it wants, and there is no "remapping" of host address information for an erased block; and (2) for systems that rely on conventional FTL address translation, there is no remapping of host address information for an erased block, i.e., the host continues to use its logical addressing (without regard to any erasure or defective bad block), and the memory controller simply assigns available space "anywhere in memory" as needed to logical addresses, on an at large basis. In other words, it is unconventional for a memory controller to detect a bad block during an erasure of that block, and to nonetheless REMAP host address information, and the associated claim features are not shown by Selinger.

back command, but Applicant sees no structure whereby a host can simply at any time, and for whatever reason, simply physically erase a structure.

This is not the only difference between the claims and Selinger, as noted. The claims also recite that, during an erase operation, a bad block can be detected, and that the memory controller then remaps host address information such that when a host designates an address space (having a defective block), this address space has already been remapped to one or more different erase blocks in the nonvolatile memory. Applicant sees nothing in Selinger that corresponds to or necessarily performs these features.

The Examiner cites again to 62, 86, 24, 73-74, and 164-165, and to FIGS. 7 and 12, but Applicant sees no correspondence with Applicant's claim features in question. Paragraph 86 of Selinger references bad block management but says little about this. Paragraph 77 of Selinger (uncited by the Examiner) is more relevant, indicating that errors can be detected in various memory controller operations (including 'erase failures') and reported to the host, and that a memory controller can mark associated erase blocks as "bad;" however, all Selinger says is that valid data is moved and that this happens transparently to the host (i.e., there is no remapping that occurs for host address information *upon erasure* of an erase block, as claimed by Applicant); with Selinger, because this operation is transparent, this indicates that the host continues to use logical addressing (i.e., consistent with conventional FTL operations, as referenced in a footnote above). That is, here also, Selinger does not correspond to, and does not teach, the claim features in question.

Thus, Applicant sees nothing at all in Selinger that corresponds to at least those features of the independent claims which were underlined above.

As is known, a rejection for anticipation is only proper if a cited reference shows exactly the same thing as claimed, "arranged as in the rejected claim." If a cited reference is unclear as to its manner of operation, the Examiner has the sole burden of demonstrating that the cited reference relied on for anticipation *necessarily* operates in the manner of the rejected claim, and it is insufficient if the cited reference "might" operate in the manner claimed. See, e.g., Par Pharmaceuticals, Inc. v. TWI Pharmaceuticals, Inc., CAFC Appeal No. 2014-1391, Dec. 3, 2014: "A party must ... meet a high standard in order to rely on inherency to establish the existence of a claim limitation in the prior art in an obviousness analysis— the limitation at issue necessarily must be present, or the natural result of the combination of elements explicitly disclosed by the prior art" (see p. 16, emphasis added by Appellant). The MPEP (§2112) also underscores this, i.e., "[t]he fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic.... 'In relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily

flows from the teachings of the applied prior art'" (emphasis once again added by Appellant, text citing Ex parte Levy, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990)).

Selinger does not show those limitations underlined above, notwithstanding the Examiner's arguments, and the rejection therefore fails to set forth a prima facie position for unpatentability. Because the claim features at issue are recited by each independent claim of record, they are also effectively present in each claim of record, i.e., there is no justifiable position of unpatentability for any pending claim of record. Therefore, the rejection under 35 USC 102 cannot be supported on the present record and needs to be withdrawn.

Applicant notes that it is unnecessary to discuss *additional* reasons why the various claims are patentable and why the rejection is additionally erroneous (e.g., as to the other independent claims or dependent claims). It is believed that the errors in the rejection extend to the Examiner's positions on these claims as well, and Applicant does not acquiesce to any ground or position advocated by the Office Action, or regarding claim features not expressly discussed in this response relative to the cited art.

CONCLUSION

For the various reasons set forth above, Applicant submits that (1) the amendment set forth above should fully address the claim antecedence arguments advanced by the Examiner, and that (2) there otherwise is no justifiable rejection or objection remaining of record, i.e., the claims should, following the amendment set forth above, be allowed in their present form. Issuance of a Notice of Allowance is therefore requested.

If it would be helpful for PTO Personnel to discuss any of the presented limitations, or request elaboration on what is meant, such personnel are respectfully requested to contact the undersigned using the contact information set forth below.

Respectfully submitted,
Attorney for Applicant

/Marc P. Schuyler/
Marc P. Schuyler, Reg. No. 35, 675
(408) 655-6375
marc@schuylerlawgroup.com

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/997,471	08/19/2020	Andrey V. Kuzmin	2020058 / RMS-02C1C1C14	1070

73091 7590 10/01/2021

Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

EXAMINER	
BIRKHIMER, CHRISTOPHER D	

ART UNIT	PAPER NUMBER
2136	

MAIL DATE	DELIVERY MODE
10/01/2021	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 16/997,471	**Applicant(s)** Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	**Art Unit** 2136	**AIA (FITF) Status** No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☑ Responsive to communication(s) filed on 08/19/2020.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☐ This action is **FINAL.** 2b) ☑ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 1-20 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 1-20 is/are rejected.

8) ☐ Claim(s) _____ is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☑ The specification is objected to by the Examiner.

11)☑ The drawing(s) filed on 08/19/2020 is/are: a)☑ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)

2) ☑ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date 08/19/2020.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

DETAILED ACTION

The current Office Action is in response to the papers submitted 08/19/2020. Claims 1 – 20 are pending.

Notice of Pre-AIA or AIA Status

1. The present application is being examined under the pre-AIA first to invent provisions.

Claim Objections

2. Claims **7 and 17** are objected to because of the following informalities:

Claim 7 is worded oddly to the examiner. It is suggested to add a comma to separate the host from the storage release information in line 3 of the claim.

Claim 17 is worded oddly to the examiner. It is suggested to add a comma to separate the host from the storage release information in line 3 of the claim.

Appropriate correction is required.

Specification

3. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

4. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required:

- The claims disclose constituent storage locations. There is no mention of constituent storage locations in the specification as claimed.
- The claims disclose a nonvolatile memory system. There is no mention of a nonvolatile system in the specification as claimed.
- The claims disclose a respective host-supplied address. There is no mention of a host-supplied address as claimed.
- The claims disclose a host-request. There is no mention of a host-request as claimed.
- The claims disclose a host-selected subset. There is no mention of a host-selected subset as claimed.
- The claims disclose a host-request. There is no mention of a host-request as claimed.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112(a):

> (a) IN GENERAL.—The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor or joint inventor of carrying out the invention.

The following is a quotation of the first paragraph of pre-AIA 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to

> enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.
>
> The following is a quotation of 35 U.S.C. 112(b):
> (b) CONCLUSION.—The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the inventor or a joint inventor regards as the invention.
>
> The following is a quotation of 35 U.S.C. 112 (pre-AIA), second paragraph:
> The specifications shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims **1 - 20** are rejected under 35 U.S.C. 112(a) or 35 U.S.C. 112 (pre-AIA), first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor or a joint inventor, or for applications subject to pre-AIA 35 U.S.C. 112, the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claims 1 – 2, 5 – 6, 8 – 9, 11 – 12, 15 – 16, and 18 – 19 disclose constituent storage locations. The original description fails to disclose any mention of constituent storage locations as claimed. The original specification discloses constituent pages, constituent space, constituent commands, and constituent memory cell. However, there is no mention of which one of the constituent pages, space, commands, or memory cell is meant to be a constituent storage location or if there is another item that is supposed to be considered a constituent storage location. There is a lack of support for the claimed constituent storage locations in the original specification.

Claims 1 and 11 disclose host-selected designations of address space and host-selected designation of address space. The only mention of anything being host-

selected is a threshold in paragraph 0053. There is a lack of any designations of an address space or any other designations being host-selected as claimed.

Claims 3, 7, and 13 disclose a nonvolatile memory system. The specification discloses the use of a nonvolatile memory but there is no mention as to what is considered a nonvolatile memory system as claimed. There is a lack of any designation of a nonvolatile memory system as claimed.

Claims 3 and 13 disclose a respective host-supplied address. The specification fails to disclose anything being host-supplied as claimed let alone a host-supplied address. There is a lack of any designation of a host-supplied address as claimed. The specification does not mention the host-supplied limitation.

Claims 6 and 16 disclose a host-request. The specification fails to disclose the term "host-request" as claimed. The specification does disclose the host sending multiple different types of requests to the controller to be performed on memory. There is no mention of any of the requests being considered a host-request as claimed.

Claims 7 and 17 disclose a host-selected subset of storage structures. The specification fails to disclose the storage structures that are host-selected as claimed. The specification does disclose the host-selected limitation but it is related to a threshold in paragraph 0053. There is no mention of any subset of storage structures being host-selected as claimed.

All remaining claims are rejected for being dependent on a rejected base claim.

7. Claims **1 - 20** are rejected under 35 U.S.C. 112(b) or 35 U.S.C. 112 (pre-AIA), second paragraph, as being indefinite for failing to particularly point out and distinctly

claim the subject matter which the inventor or a joint inventor (or for applications subject to pre-AIA 35 U.S.C. 112, the applicant), regards as the invention.

Claims 1 – 2, 5 – 6, 8 – 9, 11 – 12, 15 – 16, and 18 - 19 recite "constituent storage locations" in multiple locations. There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does disclose constituent pages of an erase unit but not constituent storage locations or that the locations are associated with an erase block as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies…Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification…The use of a confusing variety of terms for the same thing should not be permitted…While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is

suggested to amend the specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claims 1 and 11 recite "host-selected designations of address space" and "host-selected designation of address space". There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does disclose a host-selected threshold but not mention of host-selected designations or designation as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies…Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification…The use of a confusing variety of terms for the same thing should not be permitted…While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is suggested to amend the

specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claims 3, 7, and 13 recite "the nonvolatile memory system". There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does disclose the use of nonvolatile memory in connection with a controller and a host. There is no mention though of anything being referred to as a nonvolatile memory system as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies…Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification…The use of a confusing variety of terms for the same thing should not be permitted…While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is suggested to amend the

specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claims 3 and 13 recite "host-supplied address". There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does not mention anything being host-supplied as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies…Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification…The use of a confusing variety of terms for the same thing should not be permitted…While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is suggested to amend the specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claims 6 and 16 recite "host-request". There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does mention the host sending multiple types of requests to the controller including read, write, and erase requests. There is no mention of any request being referred to as a host-request as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies…Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification…The use of a confusing variety of terms for the same thing should not be permitted…While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is suggested to amend the specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claims 7 and 17 recite "host-selected subset of storage structures". There is no recitation of such a term in the specification. The specification fails to properly disclose such a limitation. The specification does mention the host-selected limitation in paragraph 0053. However, the item that is host-selected is a threshold and not a subset of storage structures as claimed. There is no mention of any subset of any structure being host-selected as claimed. There is insufficient support for this term and its use in the specification making the use of such a limitation indefinite. According to MPEP 608.01(o) "The meaning of every term used in any of the claims should be apparent from the descriptive portion of the specification with clear disclosure as to its import; and in mechanical cases, it should be identified in the descriptive portion of the specification by reference to the drawing, designating the part or parts therein to which the term applies...Usually the terminology of the claims present on the filing date of the application follows the nomenclature of the specification...The use of a confusing variety of terms for the same thing should not be permitted...While an applicant is not limited to the nomenclature used in the application as filed, he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims. This is necessary in order to insure certainty in construing the claims in the light of the specification See 37 CFR 1.75, MPEP § 608.01(i) and § 1302.01 and § 2103." The terms used in the claims are different from what is in the specification making it unclear what and where support for certain claim limitations are in the specification. It is suggested to amend the

specification or claims to make the language consistent with each other to show clear support for the claimed limitations.

Claim 1 recites the limitation "host-selected designation of address space" in line 10. The claim previously mentions host-selected designations of address space. It is unclear if the single host-selected designation is meant to refer to the multiple host-selected designations of address space previously mentioned in the claim or is another separate host-selected designation. There is insufficient antecedent basis for this limitation in the claim. It is being assumed for examination that the single host-selected designation refers back to one of the host-selected designations of address space.

Claim 1 recites the limitation "the integer set of one or more blocks" in line 12. There is no previous mention of an integer set of one or more blocks in the claim. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the integer set of one or more blocks is meant to refer to the integer set of one or more of the erase blocks in line 9 of the claim.

Claim 1 recites the limitation "the integer set" in lines 13 - 14. The claim previously discloses an integer set of one or more of the erase blocks and an integer set of one or more blocks. It is unclear which integer set is referred to in lines 13 – 14. There is insufficient antecedent basis for this limitation in the claim. For examination is assumed that the integer set disclosed in lines 13 – 14 refers to the integer set of one or more of the erase blocks previously disclosed in the claim.

Claim 3 recites the limitation "the data read requests" in line 2. There is no previous mention of data read requests in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 3 recites the limitation "the nonvolatile memory system" in line 12. There is no previous mention of a nonvolatile memory system in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 3 recites the limitation "host-supplied address" in lines 3 - 4. As indicated above, the specification fails to mention the host-supplied limitation let alone a host-supplied address. The lack of support for the host-supplied address in the specification makes the scope of a host-supplied address indefinite.

Claim 6 recites the limitation "host-request" in line 5. As indicated above, the specification fails to mention the host-request limitation. The lack of support for the host-request in the specification makes the scope of a host-request indefinite.

Claim 6 recites the limitation "the storage release information" in line 5. There is no previous mention of storage release information in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the storage release information refers to the storage location release information in line 3 of claim 6.

Claim 7 recites the limitation "the host request to transmit" in line 2. There is no previous mention of a host request to transmit. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the host request to transmit is meant to refer to the host-request to transmit from claim 6.

Claim 7 recites the limitation "the nonvolatile memory system" in line 2. There is no previous mention of a nonvolatile memory system in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 7 recites the limitation "host-selected subset" in lines 3 - 4. As indicated above, the specification fails to mention the host-selected subset limitations. The lack of support for the host-selected subset in the specification makes the scope of a host-selected subset indefinite.

Claim 9 recites the limitation "the one or more data copy request" in line 8. The claim previously discloses one or more data copy requests not one or more data copy request. It is unclear if the one or more data copy request is meant to be one of the one or more data copy requests or another request. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the one or more data copy request refers back to the one or more data copy requests in the claim.

Claim 11 recites the limitation "host-selected designation of address space" in line 11. The claim previously mentions host-selected designations of address space. It is unclear if the single host-selected designation is meant to refer to the multiple host-selected designations of address space previously mentioned in the claim or is another separate host-selected designation. There is insufficient antecedent basis for this limitation in the claim. It is being assumed for examination that the single host-selected designation refers back to one of the host-selected designations of address space.

Claims 11 recites the limitation "the integer set of one or more blocks" in line 13. There is no previous mention of an integer set of one or more blocks in the claim. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the integer set of one or more blocks is meant to refer to the integer set of one or more of the erase blocks in line 10 of the claim.

Claim 11 recites the limitation "the integer set" in lines 13 - 15. The claim previously discloses an integer set of one or more of the erase blocks and an integer set of one or more blocks. It is unclear which integer set is referred to in lines 13 – 15. There is insufficient antecedent basis for this limitation in the claim. For examination is assumed that the integer set disclosed in lines 13 – 15 refers to the integer set of one or more of the erase blocks previously disclosed in the claim.

Claim 13 recites the limitation "the data read requests" in line 2. There is no previous mention of data read requests in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 13 recites the limitation "the nonvolatile memory system" in line 12. There is no previous mention of a nonvolatile memory system in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 13 recites the limitation "host-supplied address" in lines 2 - 3. As indicated above, the specification fails to mention the host-supplied limitation let alone a host-supplied address. The lack of support for the host-supplied address in the specification makes the scope of a host-supplied address indefinite.

Claim 16 recites the limitation "host-request" in line 6. As indicated above, the specification fails to mention the host-request limitation. The lack of support for the host-request in the specification makes the scope of a host-request indefinite.

Claim 16 recites the limitation "the storage release information" in line 6. There is no previous mention of storage release information in the claim or any base claim. There is insufficient antecedent basis for this limitation in the claim. It is assumed for

examination that the storage release information refers to the storage location release information in line 2 of claim 6.

Claim 17 recites the limitation "the host request to transmit" in line 2. There is no previous mention of a host request to transmit. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the host request to transmit is meant to refer to the host-request to transmit from claim 6.

Claim 17 recites the limitation "host-selected subset" in lines 3 - 4. As indicated above, the specification fails to mention the host-selected subset limitations. The lack of support for the host-selected subset in the specification makes the scope of a host-selected subset indefinite.

Claim 19 recites the limitation "the one or more data copy request" in line 8. The claim previously discloses one or more data copy requests not one or more data copy request. It is unclear if the one or more data copy request is meant to be one of the one or more data copy requests or another request. There is insufficient antecedent basis for this limitation in the claim. It is assumed for examination that the one or more data copy request refers back to the one or more data copy requests in the claim.

All remaining claims are rejected for being dependent on a rejected base claim.

Claim Rejections - 35 USC § 102

8. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of

rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of the appropriate paragraphs of pre-AIA 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

9. Claim(s) **1 - 20** is/are rejected under pre-AIA 35 U.S.C. 102(b) as being anticipated by Selinger et al. (Pub. No.: US 2011/0161784) referred to as Selinger.

Regarding **claim 1**, Selinger teaches a memory controller **[700, Fig 7]** to control nonvolatile memory **[730, Fig 7]**, the nonvolatile memory **[730, Fig 7]** having nonvolatile memory dies **[Paragraph 0024; The memory is NAND flash memory comprised of dies]**, each of the nonvolatile memory dies **[Paragraph 0024; The memory is NAND flash memory comprised of dies]** having a group of erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, each erase block in the group **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** such that it must be erased before constituent storage locations **[Paragraph 0062; The pages in a block]** associated with that erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** can be reprogrammed **[Paragraph 0062]**, the memory controller comprising **[700, Fig 7]**:

a host interface **[725, Fig 7]**;

a memory interface **[735, Fig 7]**;

logic **[700, Fig 7]** to receive erasure requests from a host **[720, Fig 7]** via the host interface **[725, Fig 7]**, the erasure requests accompanied by host-selected designations of address space that is to be erased, and to responsively control erasure of an integer set of one or more of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** corresponding to the host-selected designation of address space, by issuing erasure commands **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]** to the nonvolatile memory dies **[Paragraph 0024; The memory is NAND flash memory comprised of dies]** via the memory interface **[735, Fig 7]**;

logic **[700, Fig 7]** to, in connection with the erasure of the integer set of one or more blocks **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]**, detect defects in the integer set and, responsive to detecting the defects, to remap host address information for at least one erase block **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** in the integer set to one or more different erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** in the nonvolatile memory **[730, Fig 7; 908, Fig 9; Paragraphs 0040, 0080,**

0091, and 0095; The controller detects an erase error, notifies the host, and performs the remapping operations of host logical address due to the erase error]; and

logic **[700, Fig 7]** to receive data write requests **[510, Fig 5; 1440, Fig 14]** from the host **[720, Fig 7]** via the host interface **[725, Fig 7]** and to responsively issue write commands to, and effectuate related data transfer with, the nonvolatile memory **[Fig 6; 730, Fig 7; Figs 10 and 14; The controller sends corresponding commands to either read or write data dependent on the read or write command issues from the host]**, each of the data write requests being accompanied by the host address information **[510, Fig 5; 1440, Fig 14; Each write request has a destination address]**;

wherein the memory controller **[700, Fig 7]** is to direct data write requests **[510, Fig 5; 1440, Fig 14]** from the host **[720, Fig 7]** for which the host address information has been remapped **[730, Fig 7; 908, Fig 9; Paragraphs 0040, 0080, 0091, and 0095; The controller detects an erase error, notifies the host, and performs the remapping operations of host logical address due to the erase error]** by said logic **[700, Fig 7]** to detect and remap to constituent storage locations **[Paragraph 0062; The pages in a block are written due to write requests]** associated with the one or more different erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**;

wherein each said logic **[700, Fig 7]** comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry **[725, 735, 740, 750, 760, and 770, Fig 7; Paragraph 0025]**.

Regarding **claim 2**, Selinger teaches the host interface **[725, Fig 7]** is also to receive data copy requests **[Fig 14]** from the host **[720, Fig 7]**, and wherein the memory controller **[700, Fig 7]** is to responsively read data from one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a first one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** and write the data which has been read to one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a second one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** without sending the data which has been read to the host **[720, Fig 7; 1460 and 1470, Fig 14; Paragraph 0152; The data that is copied is not sent to the host]**.

Regarding **claim 3**, Selinger teaches the host interface **[725, Fig 7]** is to receive the data write **[1440, Fig 14]** and the data read requests **[610, Fig 6; 1002, Fig 10; 1410, Fig 14; 1510, Fig 15]** each accompanied by a respective host-supplied address **[Figs 6, 10, 14, and 15; Each read or write request is accompanied with a host designated address]**;

each respective host-supplied address **[Figs 6, 10, 14, and 15; Each read or write request is accompanied with a host designated address]** includes a field that has been mapped in advance to a specific tier of a

structural hierarchy **[Paragraphs 0065, 0108, and 0110; Each address provided from the host is mapped in advance in the mapping table to a physical structural tier address]** associated with storage in the nonvolatile memory **[730, Fig 7]**, the mapping such that different values of the field compel the memory controller **[700, Fig 7]** to access respective storage structures **[Paragraphs 0065, 0108, and 0110; Each address provided selects different tiers of the memory]** within the nonvolatile memory **[730, Fig 7]**, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data **[Paragraphs 0062 and 0086; Each block structure is comprised of multiple pages]**; and

each of the data copy requests **[Fig 14]** is accompanied by a destination address **[1440, Fig 14]** also having the field, a value for the field of the destination address **[1440, Fig 14]** selected by the host **[720, Fig 2]** in a manner which compels the nonvolatile memory system **[Fig 7]** to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier **[Fig 14; The destination address causes the controller to copy data read from a destination address to the destination address]**.

Regarding **claim 4**, Selinger teaches the specific tier is a tier associated with the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks which comprise pages on a lower tier of storage]**, and wherein the respective storage structures comprise respective erase

blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]**.

Regarding **claim 5**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to maintain storage location release information **[900, Fig 9]** for at least one erase block in one of the groups **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]**, the storage location release information **[900, Fig 9]** to indicate a condition where at least one of the constituent storage locations **[Paragraph 0062; The pages in a block]** for the at least one erase block **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** has been programmed but no longer stores valid data **[Paragraphs 0077, 0091, and 0165; The error reporting shows at least one page of the block has invalid data]**, the condition such that the at least one of the constituent storage locations **[Paragraph 0062; The pages in a block]** cannot be reprogrammed until the at least one erase block is erased **[Paragraph 0062; Pages in a block cannot be reprogrammed until the entire block is erased]**.

Regarding **claim 6**, Selinger teaches the memory controller **[700, Fig 7]** is to store the storage location release information **[Figs 8A – 8D; 900, Fig 9; Paragraphs 0074 – 0077; Status information is stored in registers and stored in the interface in the process of transmitting to the host]** for each of the constituent storage locations **[Paragraph 0062; The pages in a block]** for each erase block in at least one of the groups **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]** and is to update the storage location release information **[900, Fig 9]** as an automated response **[Paragraphs 0074 – 0077; The status information is**

updated as a result of requests from the host] to requests received from the host **[720, Fig 7]**, and wherein the memory controller **[700, Fig 7]** further comprises logic to receive and process a host-request to transmit, to the host **[720, Fig 7]**, the storage release information **[900, Fig 9; Fig 10; Paragraph 0074; The requests from the host are requests to transmit associated status information relating to the read/write request from the host]** for at least one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with the at least one of the groups **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**.

Regarding **claim 7**, Selinger teaches the host **[720, Fig 7]** is to communicate an address space selection together with the host request to transmit **[Figs 5 - 6, 10, and 14 – 15; Each request from the host includes an address selection to direct the given request to a location in memory]**, such that the nonvolatile memory system **[Fig 7]** is to transmit to the host **[720, Fig 7]** storage release information **[900, Fig 9]** associated with a host-selected subset of storage structures **[Paragraph 0062; The pages in a block]**, the host-selected subset corresponding to the address space selection **[Paragraphs 0074 – 0077; The status information presented to the host is associated with pages in the selected block based on the address information in the request from the host]**.

Regarding **claim 8**, Selinger teaches the memory controller **[700, Fig 7]** is to receive from the host **[720, Fig 7]** a request to mark as released one or more of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with at least one of the erase blocks **[1210, Figs 12A – 12B; Paragraphs 0062 and 0086; The blocks are erase blocks]**, and wherein the logic **[700, Fig 7]** to maintain is to

responsively update the storage location release information **[900, Fig 9]** to mark the one or more of the constituent storage locations **[Paragraph 0062; The pages in a block]** as released **[908, Fig 9; Paragraphs 0076 – 0077; The erase error indicates an associated block and pages are candidates for erase and thus released]**.

Regarding **claim 9**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to monitor the storage location release information **[900, Fig 9]** to detect memory space that represents candidates for garbage collection **[Paragraphs 0076 – 0077 and 0159]**, and to transmit to the host **[720, Fig 7]** an identification of the memory space that represents candidates for garbage collection **[1012, Fig 10; Paragraphs 0074 – 0077 and 0159]**;

the host **[720, Fig 7]**, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests **[Figs 14 – 15; Paragraph 0159]**; and

the memory controller **[700, Fig 7]**, in response to each of the one or more data copy request **[Figs 14 – 15; Paragraph 0159]**, is to read data **[1430, Fig 14; 1530, Fig 15]** from one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated with a first one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062 and 0086; The blocks are erase blocks]** and write the data **[1460 and 1470, Fig 14; 1570, Fig 15]** to one of the constituent storage locations **[Paragraph 0062; The pages in a block]** associated

with a second one of the erase blocks **[1210, Figs 12A – 12B; 1420 and 1430, Fig 14; Paragraphs 0062, 0086, and 0159; The blocks are erase blocks]** without sending the data to the host **[720, Fig 7; 1460 and 1470, Fig 14; Paragraph 0152; The data that is copied is not sent to the host]**.

Regarding **claim 10**, Selinger teaches the memory controller **[700, Fig 7]** further comprises logic to monitor the storage location release information **[900, Fig 9]** to detect memory space that represents candidates for erasure **[908, Fig 9]**, and is to transmit to the host **[720, Fig 7]** an identification of the memory space that represents candidates for erasure **[908 and 910, Fig 9; Paragraphs 0074, 0080, 0091, and 0164 - 0165]**; and

the host **[720, Fig 7]** is to issue at least one of the erasure requests **[Paragraphs 0025, 0073 – 0074, and 0164 – 0165; The controller performs erase operations based on commands and addresses received from the host in response to status information sent to the host from the controller]** in response to said identification **[Paragraphs 0025, 0073, 0076, 0091, and 0164 – 0165]**.

Claims 11 – 20 are the method claims relating to the memory controller claims 1 – 10 and are thus rejected using the same prior art and reasoning.

Conclusion

10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Oh (Pat 8,069,284) discloses a semiconductor memory device includes a nonvolatile memory device having a plurality of physical sectors, and a memory controller configured to translate a logical address received from a host to a physical address, with reference to mapping data that defines a correspondence between the logical address and the physical address. The nonvolatile memory device is configured to access a first physical sector corresponding to the physical address, and, when a data delete command is provided from the host to the memory controller to delete first data that is stored in the first physical sector, the memory controller delays an erase and/or merge operation for the first physical sector in which the first data is stored.

Oh (Pub. No.: US 2009/0292839) discloses a semiconductor memory device includes a nonvolatile memory device having a plurality of physical sectors, and a memory controller configured to translate a logical address received from a host to a physical address, with reference to mapping data that defines a correspondence between the logical address and the physical address. The nonvolatile memory device is configured to access a first physical sector corresponding to the physical address, and, when a data delete command is provided from the host to the memory controller to delete first data that is stored in the first physical sector, the memory controller delays an erase and/or merge operation for the first physical sector in which the first data is stored.

Afriat (Pub. No.: US 2009/0172246) discloses a controller may be further operative to disregard an access command, which does not include a logical address satisfying the predefined condition, if the construction of the mapping table has not yet completed. Alternatively, the controller may be further operative to delay servicing an

access command, which does not include a logical address satisfying the predefined condition, until after completing the construction of the mapping table.

Calder et al. (Pub. No.: US 2010/0106695) discloses at least one extent is determined to be collected as part of the garbage collection method. In an exemplary embodiment, the extents to be collected are determined based on a utilization ratio. The utilization ratio is a comparison of active portions, portions being utilized by properties that have not expired, relative to inactive (dead) portions that are associate with expired properties. Therefore, once the utilization ration passes a predefined threshold that is dependent on the particular schema or heuristics of the structured storage system, the extent is identified as one to be collected. Additionally, extents, in an embodiment, are identified based on last I/O activity or percent of utilization of the extent.

Conley et al. (Pub. No.: US 2005/0195635) discloses garbage collection operations may be scheduled for a time when they will have reduced impact on host data write operations. Operations may be scheduled so that they are carried out as background operations and thus cause little or no disruption to other operations.

Conley et al. (Pub. No.: US 2008/0250202) discloses garbage collection operations may be scheduled for a time when they will have reduced impact on host data write operations. Operations may be scheduled so that they are carried out as background operations and thus cause little or no disruption to other operations.

Danilak (Pub. No.: US 2011/0167199) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2011/0125956) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pat 7,904,619) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pat 7,809,900) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pat 7,747,813) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2009/0138671) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2008/0126724) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2008/0126720) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2008/0126719) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2008/0126685) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Danilak (Pub. No.: US 2012/0060060) discloses a throttling module coupled to a controller that sends information to the controller which instructs the controller to delay execution of commands.

Flynn et al. (Pub. No.: US 2010/0211737) discloses the storage controller may delay or defer performing the erase operation of the non-volatile storage media comprising the unused blocks until later in time or until the storage media for the unused blocks is needed.

Gonzalez et al. (Pat 7,012,835) discloses a comprehensive scrub operation is included in a flash memory system to compensate for disturbs of storage levels in one group of memory cells that are caused by programming, reading or erasing operations performed on another group of memory cells on the same integrated circuit chip. The potential for such data disturbs increases as the memory cell arrays are shrunk in size. There is a tendency, in order to save space, for various signal lines to be shared among groups of memory cells such that one group experiences potentially repetitive exposure to voltages and/or currents during programming, reading or erasing the memory cells of another group. By scrubbing the data stored in all groups of memory cells in an organized and continuous manner, corruption of stored data over time is significantly

reduced. Also, the storage of an increased number of bits per memory cell with an increased number of smaller voltage threshold state ranges is made practical when the disturbed charge levels are corrected by methodically restoring them to their intended levels.

Gorobets (Pub. No.: US 2008/0082596) discloses performing garbage collection in phases interleaved with accesses from a host.

Harari et al. (Pub. No.: US 2011/0041039) discloses a host scheduling read scrubbing in flash memory based on status information of the flash memory received from a controller of the flash memory.

Menon et al. (Pat 5,933,840) discloses different methodologies exist for determining when garbage collection should be performed, such as the percentage of empty segments in the LSA. These methodologies do not form a part of this invention. Because the LSA controller considers a segment for garbage collection only if its age is greater than the age threshold value, the next processing step is to check segment age, which is represented by the flow diagram box numbered. Each segment that is determined to be a candidate for garbage collection is preferably designated in some way, such as by setting a flag in the LSA directory entry for that segment. Next, the utilization of each segment that is older than the age threshold is determined.

O'Connor et al. (Pat 8,099,570) discloses scrubbing routines read unused sections of memory to attempt to detect and deal with memory faults before critical data is stored in these locations. Reading this unimportant data allows the error to be detected and dealt with before it holds critical information.

Oh (Pat 8,069,284) discloses when a data delete command is provided from the host to the memory controller to delete first data that is stored in the first physical sector, the memory controller delays an erase and/or merge operation for the first physical sector in which the first data is stored.

Oh (Pub. No.: US 2009/0292839) discloses when a data delete command is provided from the host to the memory controller to delete first data that is stored in the first physical sector, the memory controller delays an erase and/or merge operation for the first physical sector in which the first data is stored.

Seol et al. (Pub. No.: US 2011/0296089) discloses if the flash chip storing the original page is performing a writing operation, the controller is required to delay reading of the original page until the writing operation is terminated.

Sinclair (Pat 7,984,233) discloses if no host command is pending, garbage collection is carried out, including non-priority garbage collection, if it is determined that the host has been inactive or idle for a predetermined length of time, or that the most recently received host command was the Idle command, or that the pending host command is the Idle command. Under these circumstances, the garbage collection is then most likely performed entirely in the background.

Sinclair et al. (Pat 7,610,437) discloses garbage collection operations may be scheduled in two ways. Background operations may be initiated by the host when it is not making read or write access to the device, and are executed continuously by the device until the host makes another access. Foreground operations may be scheduled by the device whilst it is being accessed by the host, and are executed in bursts interleaved with bursts of program operations for file data received from the host. The

lengths of the interleaved bursts may be adaptively controlled to maintain the garbage collection rate to the minimum required at all times.

Sinclair (Pub. No.: US 2007/0033325) discloses immediately after a file is indicated to be deleted or erased by a host, the blocks containing the file are scheduled for garbage collection.

Tsirkin (Pub. No.: US 2012/0030669) discloses storing commands in a delay command queue and eventually executing the delayed commands from the delay command queue.

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER D BIRKHIMER whose telephone number is (571)270-1178. The examiner can normally be reached on 8-5 Hoteling.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on 571-272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see https://ppair-my.uspto.gov/pair/PrivatePair. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Christopher D Birkhimer/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 1 of 3

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-8069284-B2	11-2011	Oh; Sang-Jin	G06F13/1668	710/52
*	B	US-20090292839-A1	11-2009	OH; Sang-Jin	G06F13/1668	710/55
*	C	US-20110296089-A1	12-2011	Seol; Jin-Ho	G06F12/121	711/103
*	D	US-20110167199-A1	07-2011	Danilak; Radoslav	G06F12/0246	711/103
*	E	US-20110125956-A1	05-2011	Danilak; Radoslav	G06F12/0246	711/103
*	F	US-7904619-B2	03-2011	Danilak; Radoslav	G06F3/0616	710/56
*	G	US-7809900-B2	10-2010	Danilak; Radoslav	G06F12/0246	711/154
*	H	US-20100211737-A1	08-2010	Flynn; David	G06F12/0246	711/114
*	I	US-7747813-B2	06-2010	Danilak; Radoslav	G06F12/0246	711/103
*	J	US-20090172246-A1	07-2009	Afriat; Itshak	G06F12/0246	711/103
*	K	US-20090138671-A1	05-2009	Danilak; Radoslav	G06F3/0659	711/170
*	L	US-20080126724-A1	05-2008	Danilak; Radoslav	G06F12/0246	711/162
*	M	US-20080126720-A1	05-2008	Danilak; Radoslav	G06F12/0246	711/158

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 2 of 3

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20080126719-A1	05-2008	Danilak; Radoslav	G06F12/0246	711/154
*	B	US-20080126685-A1	05-2008	Danilak; Radoslav	G06F3/0616	711/103
*	C	US-20070033325-A1	02-2007	Sinclair; Alan Welsh	G06F3/0679	711/103
*	D	US-20080082596-A1	04-2008	Gorobets; Sergey Anatolievich	G06F12/0253	1/1
*	E	US-20120030669-A1	02-2012	Tsirkin; Michael	G06F9/46	718/1
*	F	US-20110161784-A1	06-2011	Selinger; Robert D.	G06F11/1016	714/768
*	G	US-20100106734-A1	04-2010	CALDER; BRADLEY GENE	G06F16/93	707/758
*	H	US-20100106695-A1	04-2010	CALDER; BRADLEY GENE	G06F16/2255	707/696
*	I	US-5933840-A	08-1999	Menon; Jaishankar Moothedath	G06F3/0607	1/1
*	J	US-20120060060-A1	03-2012	Danilak; Radoslav	G11C16/14	714/37
*	K	US-8099570-B2	01-2012	O'Connor; James A.	G06F11/1666	711/162
*	L	US-20110041039-A1	02-2011	Harari; Eliyahou	G11C29/76	714/773
*	M	US-7012835-B2	03-2006	Gonzalez; Carlos J.	G06F11/1068	365/185.11

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Notice of References Cited	Application/Control No. 16/997,471	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner CHRISTOPHER D BIRKHIMER	Art Unit 2136	Page 3 of 3

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-7984233-B2	07-2011	Sinclair; Alan W.	G06F12/0246	711/103
*	B	US-20080250202-A1	10-2008	CONLEY; Kevin M.	G06F12/0862	711/115
*	C	US-20050195635-A1	09-2005	Conley, Kevin M.	G06F12/0862	365/149
*	D	US-7610437-B2	10-2009	Sinclair; Alan W.	G06F3/0679	711/103
	E					
	F					
	G					
	H					
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	Art Unit
	CHRISTOPHER D BIRKHIMER	2136

CPC - Searched*		
Symbol	Date	Examiner
G06F12/0246 or G06F2212/7208 or G06F2212/2022 with and without text	09/30/2021	CDB

CPC Combination Sets - Searched*		
Symbol	Date	Examiner

US Classification - Searched*			
Class	Subclass	Date	Examiner

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	Date	Examiner
EAST search performed	09/30/2021	CDB
Assignee/Inventor search performed in EAST	09/30/2021	CDB

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner

/Christopher D Birkhimer/ Primary Examiner, Art Unit 2136	

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/997,471	Kuzmin et al.
	Examiner	**Art Unit**
	CHRISTOPHER D BIRKHIMER	2136

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS										
☐ Claims renumbered in the same order as presented by applicant						☐ CPA		☐ T.D.		☐ R.1.47
CLAIM		**DATE**								
Final	Original	09/30/2021								
	1	✓								
	2	✓								
	3	✓								
	4	✓								
	5	✓								
	6	✓								
	7	✓								
	8	✓								
	9	✓								
	10	✓								
	11	✓								
	12	✓								
	13	✓								
	14	✓								
	15	✓								
	16	✓								
	17	✓								
	18	✓								
	19	✓								
	20	✓								

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S21	131	("10445229" | "10552058" | "10552085" | "20030028733" | "20030065866" | "20050144413" | "20060022171" | "20070046681" | "20070058431" | "20070233939" | "20070260811" | "20070283428" | "20080082596" | "20080147964" | "20080155204" | "20080189485" | "20090036163" | "20090046533" | "20090083478" | "20090089482" | "20090089490" | "20090172219" | "20090172250" | "20090172257" | "20090198946" | "20090254705" | "20090271562" | "20090300015" | "20090327602" | "20100042655" | "20100115172" | "20100161882" | "20100162012" | "20100182838" | "20100191779" | "20100241866" | "20100262761" | "20100281230" | "20100287217" | "20100329011" | "20110033548" | "20110055445" | "20110161784" | "20110197023" | "20110238943" | "20110276756" | "20110296133" | "20110314209" | "20120033519" | "20120059972" | "20120066441" | "20120131381" | "20120159039" | "20120204079" | "20130007343" | "20130013852" | "20130019062" | "20130073816" | "20130111295" | "20130124793" | "20130166825" | "20130242425" | "20130290619" | "20130297852" | "20140101371" | "20140122781" | "20140189209" | "20140208004" | "20140208062" | "20140215129" | "20140297949" | "20150067297" | "20150113203" | "20150134930" | "20150149789" | "20150193148" | "20150212938" | "20150261456" | "20150324264" | "20150347041" | "20150347296" | "20160018998" | "20160026564" | "20160342509" | "20160357462" | "20160364179" | "20170031699" | "5568423" | "5652857" | "5860082" | "6134631" | "7096378" | "7339823" | "7404031" |	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33

		"7406563" | "7581078" | "7702846" | "7710777" | "7752381" | "7801561" | "7818489" | "7836244" | "7861122" | "7941692" | "7970983" | "7991944" | "8001318" | "8024545" | "8055833" | "8065471" | "8065473" | "8068365" | "8072463" | "8074022" | "8082389" | "8086790" | "8347042" | "8572331" | "8645634" | "9229854" | "9335939" | "9400749" | "9519578" | "9542118" | "9588904" | "9652376" | "9696917" | "9710377" | "9727454" | "9785572" | "9846541").PN.				
S22	30	(Radian near3 Memory near3 Systems).as.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S23	127	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S24	19,163	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S25	0	(S22 S23) and (constituent near2 storage near2 location$1) and cop$4 and (erase near2 request)	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S26	19,163	(G06F12/0246 or G06F2212/7208 or G06F2212/2022).cpc.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S27	58	S26 and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S28	30	(Radian near3 Memory near3 Systems).as.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S29	127	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S30	1	(S28 S29) and (controller same ((defer$4 or delay$4) near5 (copy erase erasure)) same host) and flash	US-PGPUB; USPAT	OR	ON	2021/09/27 14:33
S31	1,196	(host) near5 (schedule$1) near5 operation	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S32	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33

S33	13	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S34	62	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S35	47	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S36	302	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S37	99	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S38	38	((controller$1) near10 (transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or moved or moving))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S39	22	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33

S40	2,146	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/27 14:33
S41	1,147	(host) near5 (schedule$1) near5 operation	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S42	4	(status near2 information) near10 (host) near5 (schedule$1) near5 operation	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S43	13	(controller near5 (notifies or identifies or indicates) near5 (need or desire or threshold) near5 (erase or erasure))	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S44	61	(controller near5 (notifies or identifies or indicates) near5 host near5 (erase or erasure))	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S45	46	(controller near5 (defer$4 or delay$4 or wait$4) near5 (operation$1 or execution$1 or command$1)) same ((notifies or notification or notified) near5 host near5 controller)	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S46	302	(eras$4 or move$4) and ((controller near10 host) same (eras$4 and (move or moved or moving))) and (interleav$4)	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S47	98	((host) near10 (receiv$4) near10 (command$1 or notif$4 or notificat$4) near10 (controller)) same (eras$4 same (move or moved or moving))	US-PGPUB; USPAT	OR	ON	2021/09/28 09:18
S48	38	((controller$1) near10 (transmit$4 or send$1) near10 (command$1 or notif$4 or notificat$4) near10 (host)) same (eras$4 same (move or moved or moving))	US-PGPUB; USPAT	OR	ON	2021/09/28 09:19
S49	22	((controller near5 (transmit$4 or send$1) near5 (command$1 or notif$4 or notificat$4) near10 host) same (eras$4 same (move or moved or moving)))	US-PGPUB; USPAT	OR	ON	2021/09/28 09:19
S50	131	("10445229" | "10552058" | "10552085" | "20030028733" | "20030065866" | "20050144413" | "20060022171" | "20070046681" | "20070058431" | "20070233939" | "20070260811" | "20070283428" | "20080082596" | "20080147964" |	US-PGPUB; USPAT	OR	ON	2021/09/28 09:30

		"20080155204" | "20080189485" | "20090036163" | "20090046533" | "20090083478" | "20090089482" | "20090089490" | "20090172219" | "20090172250" | "20090172257" | "20090198946" | "20090254705" | "20090271562" | "20090300015" | "20090327602" | "20100042655" | "20100115172" | "20100161882" | "20100162012" | "20100182838" | "20100191779" | "20100241866" | "20100262761" | "20100281230" | "20100287217" | "20100329011" | "20110033548" | "20110055445" | "20110161784" | "20110197023" | "20110238943" | "20110276756" | "20110296133" | "20110314209" | "20120033519" | "20120059972" | "20120066441" | "20120131381" | "20120159039" | "20120204079" | "20130007343" | "20130013852" | "20130019062" | "20130073816" | "20130111295" | "20130124793" | "20130166825" | "20130242425" | "20130290619" | "20130297852" | "20140101371" | "20140122781" | "20140189209" | "20140208004" | "20140208062" | "20140215129" | "20140297949" | "20150067297" | "20150113203" | "20150134930" | "20150149789" | "20150193148" | "20150212938" | "20150261456" | "20150324264" | "20150347041" | "20150347296" | "20160018998" | "20160026564" | "20160342509" | "20160357462" | "20160364179" | "20170031699" | "5568423" | "5652857" | "5860082" | "6134631" | "7096378" | "7339823" | "7404031" | "7406563" | "7581078" | "7702846" | "7710777" | "7752381" | "7801561" | "7818489" | "7836244" | "7861122" | "7941692" | "7970983" | "7991944" | "8001318" | "8024545" | "8055833" | "8065471" | "8065473" | "8068365" | "8072463" | "8074022" | "8082389" | "8086790" | "8347042" | "8572331" | "8645634" | "9229854" | "9335939" | "9400749" | "9519578" | "9542118" | "9588904" | "9652376" | "9696917" | "9710377" | "9727454" | "9785572" | "9846541").PN.			

S51	0	"store the storage location release information"	US-PGPUB; USPAT	OR	ON	2021/09/28 11:07
S52	0	"store the storage location release information"	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/28 11:11
S53	30	(Radian near3 Memory near3 Systems).as.	US-PGPUB; USPAT	OR	ON	2021/09/29 12:05
S54	127	((Kuzmin near3 Andrey) (Jadon near3 Mike) (Mathews near3 Richard)).inv.	US-PGPUB; USPAT	OR	ON	2021/09/29 12:05
S55	3	(S54 S53) and ((error$1 defect$1)).clm.	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2021/09/29 12:05

EAST Search History (Interference)

<This search history is empty>

9/30/2021 9:09:17 AM
C:\Users\cbirkhimer\Documents\EAST\Workspaces\16997471.wsp

Electronically filed

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

INFORMATION DISCLOSURE STATEMENT
McKesson Statement

The present application contains subject matter and/or claims similar to subject matter and/or claims present in one of more of the US patent applications listed below, and/or a continuation of one of the US patent applications listed below. The Examiner is encouraged to review the file wrappers of cases identified below, parents and continuations of the cases identified below, and to consider arguments advanced by US patent examiners and Applicant's remarks. To the extent the Examiner has any questions, he/she is respectfully requested to contact the undersigned using the information set forth below.

Applications identified:

Application No.	***First Inventor***	***Status***
13/767723	Kuzmin	ISSUED - USPT 9652376
14/466167	Kuzmin	ISSUED - USPT 9727454
15/009275	Kuzmin	ISSUED - USPT 9400749
15/625931	Kuzmin	on appeal, appeal docket no. 2019-005230
15/625946	Kuzmin	pending
14/047193	Kuzmin	ISSUED - USPT 9229854
14/951708	Kuzmin	ISSUED - USPT 9519578
15/346641	Kuzmin	ISSUED - USPT 9710377
15/621888	Kuzmin	ISSUED - USPT 10445229
16/570922	Kuzmin	pending
16/751925	Kuzmin	pending
16/783083	Kuzmin	pending
16/794805	Kuzmin	pending
16/833547	Kuzmin	pending
14/848273	Kuzmin	ISSUED - USPT 10642505
14/880529	Lercari	USPT 9542118
15/053372	Lercari	USPT 9588904
15/074778	Lercari	USPT 9785572
15/690006	Lercari	ISSUED - USPT 10642748
15/211927	Jadon	USPT 10552058
16/591735	Jadon	pending
15/211939	Chen	USPT 10552085
16/702736	Chen	pending

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

16/707934	Lercari	pending
16/779918	Lercari	pending

Attorney For Applicant
Respectfully submitted,
/Marc P. Schuyler/
(408) 655-6375
marc@schuylerlawgroup.com

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

Bibliographic Data

Application No: 16/997,471

Foreign Priority claimed: ○ Yes ◉ No

35 USC 119 (a-d) conditions met: ☐ Yes ☑ No ☐ Met After Allowance

Verified and Acknowledged: /Christopher D Birkhimer/ (Examiner's Signature) | (Initials)

Title: NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
08/19/2020	711	2136	2020058 / RMS-02C1C1C14
RULE			

APPLICANTS

Radian Memory Systems, Inc., Calabasas, CA,

INVENTORS

Andrey V. Kuzmin, Moscow, RUSSIAN FEDERATION

Mike Jadon, Manhattan Beach, CA, UNITED STATES

Richard M. Mathews, Porter Ranch, CA, UNITED STATES

CONTINUING DATA

This application is a CON of 15625931 06/16/2017 PAT 10838853

15625931 is a CON of 14466167 08/22/2014 PAT 9727454

14466167 is a CON of 13767723 02/14/2013 PAT 9652376

13767723 has PRO of 61757464 01/28/2013

FOREIGN APPLICATIONS

IF REQUIRED, FOREIGN LICENSE GRANTED**

08/25/2020

**** SMALL ENTITY ****

STATE OR COUNTRY

RUSSIAN FEDERATION

ADDRESS

Marc P. Schuyler

P.O. Box 2535

Saratoga, CA 95070

UNITED STATES

FILING FEE RECEIVED

$785

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Andrey V. Kuzmin
Art Unit	
Examiner Name	unknown
Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	8065473	A1	2011-11-22	Ito et al.	
	2	6134631	A	2000-10-17	Jennings III	
	3	7861122	B2	2010-12-28	Cornwell et al.	
	4	5652857	A	1997-07-29	Shimoi et al.	
	5	7096378	B2	2006-08-22	Stence et al.	
	6	7404031	B2	2008-07-22	Oshima	
	7	7702846	B2	2010-04-20	Nakanishi et al.	
	8	7752381	B2	2010-07-06	Wong	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	9	7801561	B2	2010-09-21	Parikh et al.	
	10	7818489	B2	2010-10-19	Karamcheti et al.	
	11	7941692	B2	2011-05-10	Royer et al.	
	12	7970983	B2	2011-06-28	Nochimowski	
	13	7991944	B2	2011-08-02	Lee et al.	
	14	8024545	B2	2011-09-20	Kim et al.	
	15	8074022	B2	2011-12-06	Okin et al.	
	16	8082389	B2	2011-12-20	Fujibayashi	
	17	8086790	B2	2011-12-27	Roohparvar	
	18	5568423	A	1996-10-22	Jou et al.	
	19	7836244	B2	2010-11-16	Kim et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	20	8065471	B2	2011-11-22	Yano et al.	
	21	9229854	B2	2016-01-05	Kuzmin et al.	
	22	5860082	A	1999-01-12	Smith et al.	
	23	7339823	B2	2008-03-04	Nakayama et al.	
	24	7581078	B2	2009-08-25	Ware	
	25	7710777	B1	2010-05-04	Mintierth	
	26	8055833	B2	2011-11-08	Danilak et al.	
	27	8068365	B2	2011-11-29	Kim	
	28	9542118	B1	2017-01-10	Lercari et al.	
	29	9400749	B1	2016-07-26	Kuzmin et al.	
	30	9519578	B1	0216-12-13	Kuzmin et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	31	8001318	B1	2011-08-16	Iyer	
	32	8572331	B1	2013-10-29	Shalvi	
	33	8072463	B1	2011-12-06	Van Dyke	
	34	9335939	B2	2016-05-12	Bennett et al.	
	35	8347042	B2	2013-01-01	You	
	36	9727454	B1	2017-08-08	Kuzmin et al.	
	37	9652376	B1	2017-05-16	Kuzmin et al.	
	38	9710377	B1	2017-07-18	Kuzmin et al.	
	39	10445229	B1	2019-10-15	Kuzmin et al.	
	40	9588904	B1	2017-03-07	Lercari et al.	
	41	9785572	B1	2017-10-10	Lercari et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	42	10552058	B1	2020-02-04	Jadon et al.	
	43	10552085	B1	2020-02-04	Chen et al.	
	44	7406563	A1	2008-07-29	Nagshain	
	45	8645634	B1	2014-02-04	Cox et al.	
	46	9696917	B1	2017-07-04	Sareena et al.	
	47	9846541	B2	2017-12-19	Miyamoto et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20080189485	A1	2008-08-07	Jung et al.	
	2	20110033548	A1	2011-02-10	Kimmel et al.	
	3	20100281230	A1	2010-11-04	Rabii et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	4	20090327602	A1	2009-12-31	Moore et al.	
	5	20110296133	A1	2011-12-01	Flynn et al.	
	6	20120033519	A1	2012-02-09	Confalonieri et al.	
	7	20090172219	A1	2009-07-02	Mardiks	
	8	20090254705	A1	2009-10-08	Abali et al.	
	9	20100042655	A1	2010-02-18	Tse et al.	
	10	20100162012	A1	2010-06-24	Cornwell et al.	
	11	20100329011	A1	2010-12-30	Lee et al.	
	12	20110055445	A1	2011-03-03	Gee et al.	
	13	20110296133	A1	2011-10-20	Soran et al.	
	14	20100115172	A1	2010-05-06	Gillingham et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	15	20080147964	A1	2008-06-19	Chow et al.	
	16	20060022171	A1	2006-10-05	Maeda et al.	
	17	20090036163	A1	2009-02-05	Kimbrell	
	18	20090300015	A1	2009-12-03	Kazan et al.	
	19	20140215129	A1	2014-07-31	Kuzmin et al.	
	20	20070233939	A1	2007-10-04	Kim	
	21	20130166825	A1	2013-06-27	Kim et al.	
	22	20110238943	A1	2011-09-29	Devendran et al.	
	23	20090089490	A1	2009-04-02	Ozawa et al.	
	24	20030065866	A1	2003-04-03	Spencer	
	25	20090172250	A1	2009-07-02	Allen et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	26	20130007343	A1	2013-01-03	Rub	
	27	20140208004	A1	2014-07-24	Cohen	
	28	20120131381	A1	2012-05-24	Eleftheriou	
	29	20090083478	A1	2009-03-26	Kunimatsu	
	30	20120066441	A1	2012-03-15	Weingarten	
	31	20090271562	A1	2009-10-29	Sinclair	
	32	20110197023	A1	2011-08-11	Iwamitsu et al.	
	33	20130290619	A1	2013-10-31	Knight	
	34	20080155204	A1	2008-06-26	Qawami et al.	
	35	20090089482	A1	2009-04-02	Traister	
	36	20150324264	A1	2015-11-12	Vidypoornachy et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	37	20050144413	A1	2005-06-30	Kuo et al.	
	38	20070283428	A1	2007-12-06	Ma et al.	
	39	20090172257	A1	2009-07-02	Prins et al.	
	40	20100287217	A1	2010-11-11	Borchers et al.	
	41	20120059972	A1	2013-01-10	Chen	
	42	20130013852	A1	2013-01-10	Hou et al.	
	43	20130297852	A1	2013-11-07	Fai et al.	
	44	20140101371	A1	2014-04-10	Nguyen et al.	
	45	20150113203	A1	2015-04-23	Dancho et al.	
	46	20150261456	A1	2015-09-17	Alcantara et al.	
	47	20140122781	A1	2014-05-01	Smith et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	48	20130124793	A1	2013-05-16	Gyl et al.	
	49	20130019062	A1	2013-01-17	Bennett et al.	
	50	20100262761	A1	2010-10-14	Borchers et al.	
	51	20070058431	A1	2007-03-15	Chung et al.	
	52	20090198946	A1	2009-08-06	Ebata	
	53	20140189209	A1	2014-07-03	Sinclair et al.	
	54	20120204079	A1	2012-08-09	Takefman et al.	
	55	20070260811	A1	2007-11-08	Merry JR et al.	
	56	20080082596	A1	2008-04-03	Gorobets	
	57	20110161784	A1	2011-06-30	Sellinger et al.	
	58	20100161882	A1	2010-06-24	Stern et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	59	20150212938	A1	2015-06-30	Chen et al.	
	60	20130111295	A1	2013-05-02	Li et al.	
	61	20160018998	A1	2016-01-21	Mohan et al.	
	62	20160357462	A1	2016-12-15	Nam et al.	
	63	20130073816	A1	2013-03-21	Seo et al.	
	64	20150149789	A1	2015-05-28	Seo et al.	
	65	20160364179	A1	2016-12-15	Tsai et al.	
	66	20030028733	A1	2003-02-06	Tsunoda et al.	
	67	20100191779	A1	2010-01-29	Hinrichs	
	68	20090046533	A1	2009-02-19	Jo	
	69	20150347296	A1	2015-12-03	Kotte et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	70	20100241866	A1	2010-09-23	Rodorff	
	71	20130242425	A1	2013-09-19	Zayas et al.	
	72	20110314209	A1	2011-12-22	Eckstein	
	73	20100182838	A1	2010-07-22	Kim et al.	
	74	20170031699	A1	2017-02-02	Banerjee et al.	
	75	20150193148	A1	2015-07-09	Miwa et al.	
	76	20150134930	A1	2015-05-14	Huang et al.	
	77	20130124793	A1	2013-05-16	Gyl et al.	
	78	20140297949	A1	2014-10-02	Nagawaka	
	79	20150347041	A1	2015-12-03	Kotte et al.	
	80	20150067297	A1	2015-03-05	Arroyo et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	81	20160342509	A1	2016-11-24	Kotte et al.	
	82	20140189209	A1	2014-07-03	Sinclair et al.	
	83	20070046681	A1	2007-03-01	Nagashima	
	84	20140208062	A1	2014-07-24	Cohen	
	85	20120159039	A1	2012-06-21	Kegel et al.	
	86	20110276756	A1	2011-11-10	Bish et al.	
	87	20080147964	A1	2008-06-19	Chow et al.	
	88	20150324264	A1	2015-11-12	Chinnakkonda et al.	
	89	20160026564	A1	2016-01-28	Manning	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

EFS Web 2.1.17

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	1	2009100149	WO	A1	2009-08-13	Shaeffer et al.	entire doc.	

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T5
	1	NVM Express, Version 1.0b, July 12, 2011, pages 1-126, published at http://www.nvmexpress.org/resources/ by the NVM Express Work Group.	
	2	John D. Strunk, "Hybrid Aggregates: Combining SSDs and HDDs in a single storage pool," December 15, 2012, ACM SIGOPS Operating Systems Review archive, Volume 46 Issue 3, December 2012, Pages 50-56.	
	3	Yiying Zhang, Leo Prasath Arulraj, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Computer Sciences Department, University of Wisconsin-Madison, "De-indirection for Flash-based SSDs with NamelessWrites," published at https://www.usenix.org/conference/fast12/de-indirection-flash-based-ssds-nameless-writes, 2-7-2012, pages 1-16.	
	4	Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, and Vijayan Prabhakaran, "ResearchRemoving The Costs Of Indirection in Flash-based SSDs with NamelessWrites," 6-22-2010, pp. 1-5, published at www.cs.wisc.edu/wind/Publications/hotstorage10-nameless.pdf by Computer Sciences Department, University of Wisconsin-Madison and Microsoft Research.	
	5	Stan Park and Kai Shen, Department of Computer Science, University of Rochester, "FIOS: A Fair, Efficient Flash I/O Scheduler," 2-23-2012, pages 1-15, published at www.usenix.org/event/fast12/tech/full_papers/Park.pdf by the Advanced Computing Systems Association, Fast'12, 10th Usenix Conference On File And Storage Technologies, San Jose.	
	6	Eric Seppanen, Matthew T. O'Keefe, David J. Lilja, Department of Electrical and Computer Engineering, University of Minnesota, "High Performance Solid State Storage Under Linux," 4-10-2010, MSST '10 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), Pages 1-12	
	7	Xiangyong Ouyangyz, David Nellansy, Robert Wipfely, David Flynny, Dhabaleswar K. Pandaz, "Beyond Block I/O: Rethinking Traditional Storage Primitives," 8-20-2011, published at http://www.sciweavers.org/read/beyond-block-i-o-rethinking-traditional-storage-primitives-327868, by Fusion IO and the Ohio State University	
	8	Intel Corp, "PCI-SIG SR-IOV Primer - An Introduction to SR-IOV Technology,: 321211-002, Revision 2.5, January 2011, 28 pages	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	9	Open NAND Flash Interface (ONFI), specification, version 2.0, 174 pages, 02-27-2008	
	10	Open NAND Flash Interface (ONFI), specification, version 3.1, 296 pages 09-19-2012	
	11	NVM Express, V. 1.2.1, 217 pages, 06/03/2016	
	12	Garth Gibson, Greg Ganger, "Principles of Operation for Shingled Disk Devices," Canregie Mellon Parallel Data Laboratory, CMU-PDL-11-107, April 2011, 9 pages	
	13	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, ASPDAC 2008, 26 pages	
	14	Hua Wangx, Ping Huangxz, Shuang Hex, Ke Zhoux, Chunhua Lix, and Xubin He, "A Novel I/O Scheduler for SSD with Improved Performance and Lifetime," Mass Storage Systems and Technologies (MSST), 2013 IEEE 29th Symposium on, 6-10 May 2013, 5 pages.	
	15	Altera Corp. et al., "Hybrid Memory Cube" specification, 2012, 122 pages.	
	16	JEDEC Standard, JESD229, Wide IO, December 2011, 74 pages.	
	17	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, 978-1-4244-1922-7/08, 2008 IEEE, 6 pages	
	18	Optimizing NAND Flash Performance, Flash Memory Summit, Santa Clara, CA USA August 2008, Ryan Fisher, pp. 1-23.	
	19	High-Speed NAND Flash: Design Considerations to Maximize Performance, Flash Memory Summit, Santa Clara, CA USA August 11,2009, , Robert Pierce, pp. 1-19.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	20	NAND 201: An Update on the Continued Evolution of NAND Flash, Jim Cooke, Micron White Paper, September 6, 2011, pp. 1-10.	
	21	Spansion SLC NAND Flash Memory for Embedded, data sheet, S34ML01G1, S34ML02G1, S34ML04G1, Sept. 6, 2012, pp. 1-73.	
	22	Wang et al., "An Efficient Design and Implementation of LSM-Tree based Key-Value Store on Open Channel SSD,"EuroSys '14 Proceedings of the Ninth European Conference on Computer Systems, Article No. 16, 2014-04-14, 14 pages.	
	23	Ouyang et al., "SDF: Software-defined flash for web-scale internet storage systems," Computer Architecture News - ASPLOS '14, Volume 42 Issue 1, March 2014, 14 pages.	
	24	Macko et al., "Tracking Back References in a Write-Anywhere File System,"FAST'10 Proceedings of the 8th USENIX conference on File and storage technologies, 14 pages, 2-23-2010	
	25	Ohad, Rodeh, "IBM Research Report Defragmentation Mechanisms for Copy-on-Write File-systems," IBM white paper, April 26, 2010, 10 pages, available at domino.watson.ibm.com/library/CyberDig.nsf/papers/298A0EF3C2CDB17B852577070056B41F/$File/rj10465.pdf	

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature	/Christopher D Birkhimer/	Date Considered	09/28/2021

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Marc P. Schuyler/	Date (YYYY-MM-DD)	2020-08-18
Name/Print	Marc P. Schuyler	Registration Number	35675

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

/Christopher D Birkhimer/

09/28/2021

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

INFORMATION DISCLOSURE STATEMENT

Dear Sir:

In accordance with 37 C.F.R. §§1.97 and 1.98, and in compliance with the duty of disclosure set forth in 37 C.F.R. §1.56, applicant submits attached Form PTO/SB/08a (EFS-Web) for consideration and request the references cited therein be made of record by the U.S. Patent and Trademark Office in the above-captioned application.

Applicant respectfully points out that the submission of the listed references in this Information Disclosure Statement is not an admission that they are prior art or that they are material to patentability of any claims of the application. Also, the submission of this Information Disclosure Statement is not an indication that a search has been made by Applicant.

It is respectfully submitted that copies of all cited materials (and translations, as appropriate) have already been cited in one or more parent applications relied upon for priority, in a manner compliant with 37 C.F.R. §1.98(a-c); these one or more applications are identified below -

USSN 13/767723, USSN 14/466167, USSN 15/625931

Accordingly, per the provisions of 37 C.F.R. §1.98(d), copies of such materials which were previously provided and are not again being provided herewith. To the extent not already submitted in connection with the referenced parent application, NPLs and Foreign References (and translations thereof, as appropriate) are being furnished herewith. Consideration of the foregoing plus the prompt return of a copy of the enclosed Form SB/08A with the Examiner's initials in the left column in accordance with MPEP §609 are respectfully requested.

Respectfully submitted,

August 18, 2020

/Marc P. Schuyler/
Marc P. Schuyler, Reg. No. 35,675
Attorney for Applicant

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /CDB/

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
16/997,471	08/19/2020	2827	785	2020058 / RMS-02C1C1C14	20	2

CONFIRMATION NO. 1070

73091
Marc P. Schuyler
P.O. Box 2535
Saratoga, CA 95070

FILING RECEIPT

OC000000119480862

Date Mailed: 08/26/2020

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Andrey V. Kuzmin, Moscow, RUSSIAN FEDERATION;
Mike Jadon, Manhattan Beach, CA;
Richard M. Mathews, Porter Ranch, CA;

Applicant(s)

Radian Memory Systems, Inc., Calabasas, CA;

Power of Attorney: The patent practitioners associated with Customer Number 73091

Domestic Priority data as claimed by applicant

This application is a CON of 15/625,931 06/16/2017
which is a CON of 14/466,167 08/22/2014 PAT 9727454
which is a CON of 13/767,723 02/14/2013 PAT 9652376
which claims benefit of 61/757,464 01/28/2013

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

Permission to Access Application via Priority Document Exchange: No

Permission to Access Search Results: No

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 08/25/2020
The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 16/997,471**
Projected Publication Date: Request for Non-Publication Acknowledged
Non-Publication Request: Yes
Early Publication Request: No
**** SMALL ENTITY ****
Title

NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

Preliminary Class

365

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific

countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop

technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

PATENT APPLICATION FEE DETERMINATION RECORD

Substitute for Form PTO-875

Application or Docket Number: 16/997,471

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE($)	SMALL ENTITY FEE($)	OR	OTHER THAN SMALL ENTITY RATE($)	OTHER THAN SMALL ENTITY FEE($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A	75		N/A	
SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A	330		N/A	
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A	380		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	20 minus 20 =	*	x 50 =	0.00	OR		
INDEPENDENT CLAIMS (37 CFR 1.16(h))	2 minus 3 =	*	x 230 =	0.00			
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).			0.00			
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))				0.00			
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL	785		TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE($)	SMALL ENTITY ADDITIONAL FEE($)	OR	OTHER THAN SMALL ENTITY RATE($)	OTHER THAN SMALL ENTITY ADDITIONAL FEE($)
AMENDMENT A	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	RATE($)	ADDITIONAL FEE($)	OR	RATE($)	ADDITIONAL FEE($)
AMENDMENT B	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest found in the appropriate box in column 1.

PTO/AIA/80 (07-12)
Approved for use through 11/30/2014. OMB 0651-0035
U.S. Patent and Trademark Office; U.S DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

POWER OF ATTORNEY TO PROSECUTE APPLICATIONS BEFORE THE USPTO

I hereby revoke all previous powers of attorney given in the application identified in the attached statement under 37 CFR 3.73(c).

I hereby appoint:

■ Practitioners associated with Customer Number: 73091

OR

☐ Practitioner(s) named below (if more than ten patent practitioners are to be named, then a customer number must be used):

Name	Registration Number	Name	Registration Number

As attorney(s) or agent(s) to represent the undersigned before the United States Patent and Trademark Office (USPTO) in connection with any and all patent applications assigned only to the undersigned according to the USPTO assignment records or assignments documents attached to this form in accordance with 37 CFR 3.73(c).

Please change the correspondence address for the application identified in the attached statement under 37 CFR 3.73(c) to:

■ The address associated with Customer Number: 73091

OR

☐

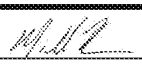
Firm or Individual Name			
Address			
City		State	Zip
Country			
Telephone		Email	

Assignee Name and Address: Radian Memory Systems, Inc., 5010 N. Parkway Calabasas Unit 205, Calabasas, CA 91302

A copy of this form, together with a statement under 37 CFR 3.73(c) (Form PTO/AIA/96 or equivalent) is required to be Filed in each application in which this form is used. The statement under 37 CFR 3.73(c) may be completed by one of The practitioners appointed in this form, and must identify the application in which this Power of Attorney is to be filed.

SIGNATURE of Assignee of Record

The individual whose signature and title is supplied below is authorized to act on behalf of the assignee

Signature		Date	
Name	Mike Jadon	Telephone	
Title	Officer, Radian Memory Systems, Inc.		

This collection of information is required by 37 CFR 1.31, 1.32 and 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

ABSTRACT

This disclosure provides for host-controller cooperation in managing NAND flash memory. The controller maintains information for each erase unit which tracks memory usage. This information assists the host in making decisions about specific operations, for example, initiating garbage collection, space reclamation, wear leveling or other operations. For example, metadata can be provided to the host identifying whether each page of an erase unit has been released, and the host can specifically then command each of consolidation and erase using direct addressing. By redefining host-controller responsibilities in this manner, much of the overhead association with FTL functions can be substantially removed from the memory controller, with the host directly specifying physical addresses. This reduces performance unpredictability and overhead, thereby facilitating integration of solid state drives (SSDs) with other forms of storage. The disclosed techniques are especially useful for direct-attached and/or network-attached storage.

We Claim:

1. (Original) A memory controller to control nonvolatile memory, the nonvolatile memory having nonvolatile memory dies, each of the nonvolatile memory dies having a group of erase blocks, each erase block in the group such that it must be erased before constituent storage locations associated with that erase block can be reprogrammed, the memory controller comprising:

a host interface;

a memory interface;

logic to receive erasure requests from a host via the host interface, the erasure requests accompanied by host-selected designations of address space that is to be erased, and to responsively control erasure of an integer set of one or more of the erase blocks corresponding to the host-selected designation of address space, by issuing erasure commands to the nonvolatile memory dies via the memory interface;

logic to, in connection with the erasure of the integer set of one or more blocks, detect defects in the integer set and, responsive to detecting the defects, to remap host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory; and

logic to receive data write requests from the host via the host interface and to responsively issue write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information;

wherein the memory controller is to direct data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks;

wherein each said logic comprises at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry.

2. (Original) The memory controller of claim 1, wherein the host interface is also to receive data copy requests from the host, and wherein the memory controller is to responsively read data from one of the constituent storage locations associated with a first one of the erase blocks and write the data which has

been read to one of the constituent storage locations associated with a second one of the erase blocks without sending the data which has been read to the host.

3. (Original) The memory controller of claim 2, wherein:

the host interface is to receive the data write and the data read requests each accompanied by a respective host-supplied address;

each respective host-supplied address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the nonvolatile memory, the mapping such that different values of the field compel the memory controller to access respective storage structures within the nonvolatile memory, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data; and

each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the nonvolatile memory system to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier.

4. (Original) The memory controller of claim 3, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

5. (Original) The memory controller of claim 1, wherein the memory controller further comprises logic to maintain storage location release information for at least one erase block in one of the groups, the storage location release information to indicate a condition where at least one of the constituent storage locations for the at least one erase block has been programmed but no longer stores valid data, the condition such that the at least one of the constituent storage locations cannot be reprogrammed until the at least one erase block is erased.

6. (Original) The memory controller of claim 5, wherein the memory controller is to store the storage location release information for each of the constituent storage locations for each erase block in at least one of the groups and is to update the storage location release information as an automated response to requests received from the host, and wherein the memory controller further comprises logic to

receive and process a host-request to transmit, to the host, the storage release information for at least one of the constituent storage locations associated with the at least one of the groups.

7. (Original) The memory controller of claim 6, wherein the host is to communicate an address space selection together with the host request to transmit, such that the nonvolatile memory system is to transmit to the host storage release information associated with a host-selected subset of storage structures, the host-selected subset corresponding to the address space selection.

8. (Original) The memory controller of claim 5, wherein the memory controller is to receive from the host a request to mark as released one or more of the constituent storage locations associated with at least one of the erase blocks, and wherein the logic to maintain is to responsively update the storage location release information to mark the one or more of the constituent storage locations as released.

9. (Original) The memory controller of claim 8, wherein:

the memory controller further comprises logic to monitor the storage location release information to detect memory space that represents candidates for garbage collection, and to transmit to the host an identification of the memory space that represents candidates for garbage collection;

the host, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests; and

the memory controller, in response to each of the one or more data copy request, is to read data from one of the constituent storage locations associated with a first one of the erase blocks and write the data to one of the constituent storage locations associated with a second one of the erase blocks without sending the data to the host.

10. (Original) The memory controller of claim 8, wherein:

the memory controller further comprises logic to monitor the storage location release information to detect memory space that represents candidates for erasure, and is to transmit to the host an identification of the memory space that represents candidates for erasure; and

the host is to issue at least one of the erasure requests in response to said identification.

11. (Original) A method of operation in a memory controller to control nonvolatile memory, where the nonvolatile memory has nonvolatile memory dies, each of the nonvolatile memory dies having a group of erase blocks, each erase block in the group such that it must be erased before constituent storage locations associated with that erase block can be reprogrammed, the memory controller having a host interface, a memory interface and at least one of hardware circuitry or instructions stored on non-transitory, machine-readable media that when executed are to control the operation of hardware circuitry, the method comprising:

receiving erasure requests from a host via the host interface, the erasure requests accompanied by host-selected designations of address space that is to be erased, and responsively controlling erasure of an integer set of one or more of the erase blocks corresponding to the host-selected designation of address space, by issuing erasure commands to the nonvolatile memory dies via the memory interface;

detecting, in connection with the erasure of the integer set of one or more blocks, defects in the integer set and, responsive to detecting the defects, remapping host address information for at least one erase block in the integer set to one or more different erase blocks in the nonvolatile memory; and

receiving data write requests from the host via the host interface and responsively issuing write commands to, and effectuate related data transfer with, the nonvolatile memory, each of the data write requests being accompanied by the host address information; and

in connection with said receiving the data write requests and responsively issuing write commands, directing data write requests from the host for which the host address information has been remapped by said logic to detect and remap to constituent storage locations associated with the one or more different erase blocks.

12. (Original) The method of claim 11, further comprising also receiving via the host interface data copy requests from the host, and responsively reading data from one of the constituent storage locations associated with a first one of the erase blocks and writing the data which has been read to one of the constituent storage locations associated with a second one of the erase blocks without sending the data which has been read to the host.

13. (Original) The method of claim 12, wherein:

each of the data write and the data read requests is accompanied by a respective host-supplied address, and each respective host-supplied address includes a field that has been mapped in advance to a specific tier of a structural hierarchy associated with storage in the nonvolatile memory, the mapping such that different values of the field compel the memory controller to access respective storage structures within the nonvolatile memory, the respective storage structures corresponding to the specific tier, each of the respective storage structures in the specific tier capable of storing multiple pages of data; and

each of the data copy requests is accompanied by a destination address also having the field, a value for the field of the destination address selected by the host in a manner which compels the nonvolatile memory system to copy the valid data from a first one of the respective storage structures at the specific tier to a second one of the specific storage structures at the specific tier.

14. (Original) The method of claim 13, wherein the specific tier is a tier associated with the erase blocks, and wherein the respective storage structures comprise respective erase blocks.

15. (Original) The method of claim 11, further comprising maintaining storage location release information for at least one erase block in one of the groups, the storage location release information to indicate a condition where at least one of the constituent storage locations for the at least one erase block has been programmed but no longer stores valid data, the condition such that the at least one of the constituent storage locations cannot be reprogrammed until the at least one erase block is erased.

16. (Original) The method of claim 15, further comprising:

storing the storage location release information for each of the constituent storage locations for each erase block in at least one of the groups;

updating the storage location release information as an automated response to requests received from the host; and

receiving and processing a host-request to transmit, to the host, the storage release information for at least one of the constituent storage locations associated with the at least one of the groups.

17. (Original) The method of claim 16, wherein the host is to communicate an address space selection together with the host request to transmit, and wherein the method further comprises transmitting to the host storage release information associated with a host-selected subset of storage structures, the host-selected subset corresponding to the address space selection.

18. (Original) The method of claim 15, further comprising receiving from the host a request to mark as released one or more of the constituent storage locations associated with at least one of the erase blocks, and responsively updating the storage location release information to mark the one or more of the constituent storage locations as released.

19. (Original) The method of claim 18, wherein:

the method further comprises monitoring the storage location release information to detect memory space that represents candidates for garbage collection, and transmitting to the host an identification of the memory space that represents candidates for garbage collection;

the host, in dependence on the identification, is to select address space corresponding to the candidates, and transmit to the memory controller one or more data copy requests; and

the method further comprises, in response to each of the one or more data copy request, reading data from one of the constituent storage locations associated with a first one of the erase blocks and writing the data to one of the constituent storage locations associated with a second one of the erase blocks without sending the data to the host.

20. (Original) The method of claim 18, wherein:

the method further comprises monitoring the storage location release information to detect memory space that represents candidates for erasure, and transmitting to the host an identification of the memory space that represents candidates for erasure; and

the host is to issue at least one of the erasure requests in response to said identification.

NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

[0001] This application is a continuation of US Patent Application No. 15/625931, filed on behalf of first-named inventor Andrey V. Kuzmin on June 16, 2017, which in turn is a continuation of US Patent Application No. 14/466167 (issued on August 8, 2017 as US Patent No. 9727454), filed on behalf of first-named inventor Andrey V. Kuzmin on August 22, 2014, which in turn is a continuation of US Patent Application No. 13/767723 (issued on May 16, 2017 as US Patent No. 9652376), filed on behalf of first-named inventor Andrey V. Kuzmin on February 14, 2013, which in turn claims priority to US Provisional Patent Application No. 61/757464, filed on January 28, 2013 on behalf of first-named inventor Andrey V. Kuzmin; each of the aforementioned patent applications are hereby incorporated by reference.

[0002] This disclosure relates to storage systems and, more specifically, to the architecture of storage systems that utilize flash memory. Still more particularly, the present disclosure relates to flash management techniques that facilitate more efficient integration and utilization of flash-based solid-state drives in network- and direct-attached storage systems.

Background

[0003] Flash memory is a type of non-volatile storage medium. It is characterized by being partitioned into storage areas called erase units (EU). Different portions of an EU can be written (programmed) at different times, but each portion can only be written once without erasing the particular EU in its entirety. Once an EU is erased, all portions of that EU are again available for writing. Erase units can be large relative to many file system operations. For example, NAND flash memory features programming units of "pages" with each erase unit consisting of a large number of pages. This asymmetry in programming and erasing data is referred to as program-erase (P/E) asymmetry. Flash memory can be embodied in different forms, for example, solid-state drives (SSDs) that utilize NAND flash memory devices. "Flash memory" as used herein includes other technologies that share these asymmetries or the other control functions mentioned below, e.g., the term can include other forms of nonvolatile memory.

[0004] A flash memory controller is typically used to manage operations within flash memory. Other types of memory, for example random access memory (RAM) and hard disk drives (HDDs), also utilize controllers dedicated to managing operations within those types of memory. The use of a memory controller independent from a host is often desirable particularly for flash memory because without such a scheme the host would be encumbered with a number of management functions unique to, or characteristic of, flash memory. Such functions typically include caching of write data to reduce frequency of programming operations, wear leveling, bad block management and space reclamation. These tasks are typically managed by a flash memory controller using a flash translation layer (FTL), which keeps records of logical-to-physical translations, wear count, bad blocks and so forth using RAM that is built-in to the flash memory controller. For example, owing to P/E asymmetry and typically large erase block size, a flash memory controller can use this RAM as a cache to help reduce the frequency of P/E operations. That is, to minimize the frequency of flash programming operations, pages of data from flash memory can be temporarily stored in the RAM and only occasionally programmed into flash memory. This helps reduce write counts and thus dilutes wear, effectively extending useful flash life. Some flash memory controllers also use wear leveling to help mitigate wear caused by accumulated writes to particular locations of flash memory. That is, to avoid disproportionate wear associated with frequently-written logical addresses, wear leveling is employed to shuffle memory contents to new physical locations within flash memory, to distribute (and thereby level) wear for each specific logical address across different physical locations. A flash memory controller typically handles this function in a manner transparent to the host and, to this effect, tracks new physical locations for each logical address using the FTL; a memory operation from the host that specifies a logical address is translated to substitute in a physical address where the desired data can be found. The RAM mentioned above can be used to store logical-to-physical (L2P) translation tables used for this purpose. Note, however, that as memory capacity grows, the typical table sizes often exceed RAM capacity, which leads to further complications. The use of L2P translation built-in to the memory controller helps present flash memory to the host as ubiquitous memory, such that a host operating system does not have to concern itself with the P/E asymmetry or other special issues for flash memory. Also, as memory cells in flash memory lose their ability to retain data reliably, portions of physical storage can be marked as "bad;" some flash memory controllers therefore also track "bad blocks" and use the FTL to remap valid memory so as to avoid these bad blocks. The FTL can also be used to detect when a host attempts to write data to an already-programmed location (i.e., without an intervening erase operation); when such an operation is detected, the FTL remaps the respective logical address to a free EU and marks “overwritten” space at

the original location as stale. Due to this and other manifestations of the P/E asymmetry, it is possible to have valuable data stored in one individually programmable unit of memory, while stale data is stored in other individually programmable units of memory within the same erase block. That is, many of the pages (e.g., a hundred pages or more) of an EU can remain unutilized while a small subset of the EU still contains data in active use, a problem which increases with time. To better utilize available storage space, some flash memory controllers therefore possess logic that periodically consolidates active data and, in so doing, frees up (reclaims) stale space, which can then be erased; this function is sometimes combined with wear leveling.

[0005] Each of these functions contributes substantial overhead and write amplification in flash memory. That is to say, substantial data and control bandwidth is consumed in implementing these functions, which can both increase the number of writes to memory (i.e., increase wear) as well as compete with new writes initiated by a host. The use of search trees or other L2P translation functions can also substantially encumber control bandwidth and input/output (IO) latency. For example, on host read commands, L2P translation is performed with the logical address provided in a command to obtain a physical address from which data should be read. At a 4KB logical block size, L2P translation is organized into a B+-tree that requires extensive space, for example, 2GB of space for 1TB flash storage device. Inability to fit a L2P tree into local (fast) RAM can result in tree blocks being swapped out to flash memory media, penalizing the address look-up for the need to load the tree blocks, with high latency. The possibility of power loss further requires L2P mapping updates to be persistent, necessitating update logging or other techniques to provide fault-tolerance. Other techniques in some flash memory implementations, for example, error protection schemes such as RAID techniques, can also substantially encumber control and data bandwidth.

[0006] These encumbrances create unpredictable response latency in flash memory. In turn, these encumbrances inhibit the use of flash memory in many applications, particularly in non-homogenous storage systems (e.g., that include dissimilar types of memories, such as both flash and magnetic memory), direct-attached storage systems and storage systems directed to network-based applications. That is, the ability of a storage system to operate effectively typically depends on structured pipelining of memory commands. Unpredictable latencies often associated with flash memory can inhibit this pipelining and, further, inhibits the use of flash memory in multiple drive storage systems, particularly

non-homogenous systems, as the unpredictable latency renders it difficult to pipeline commands for a flash drive with other memory types or drives.

[0007] What is needed is a mechanism for improving control and data bandwidth for flash memory and other forms of nonvolatile memory. More particularly, a mechanism is needed that reduces control and data bandwidth encumbrances created by memory management functions and thereby decreases the issues referenced above. Still further, a need exists for a memory management scheme that does not create excessive write amplification and bandwidth competition. Finally, a need exists for a flash/nonvolatile memory architecture that has more consistent latency, is conductive to structured pipelining of commands, and permits ubiquitous management of SSDs and other forms of memory in direct-attached and network storage applications. Techniques provided by this disclosure satisfy these needs and provide further related advantages.

Brief Description Of The Drawings

[0008] FIG. 1 is an illustrative diagram of a system having a memory controller 103, a host 105 and a memory 107.

[0009] FIG. 2 is a block diagram of a solid-state drive (SSD) having a memory controller and NAND flash memory 207.

[0010] FIG. 3A is a block diagram of a memory controller.

[0011] FIG. 3B is a block diagram of memory controller logic used in supporting cooperative functions.

[0012] FIG. 4 is a flow diagram that illustrates host tasks in a cooperative memory management scheme.

[0013] FIG. 5 is a flow diagram that illustrates host tasks in a cooperative memory management scheme.

[0014] FIG. 6 a diagram showing examples of information that can be kept by a memory controller for each of plural physical subdivisions of memory.

[0015] FIG. 7 shows a process associated with wear-aware writes.

[0016] FIG. 8 shows a process associated with a delegated copy operation.

[0017] FIG. 9 shows a process associated with an explicit erase operation (or relocation of data and recycling of previously used memory space).

[0018] FIG. 10A shows a process associated with host-owned defect management.

[0019] FIG. 10B shows a process associated with shared defect management.

[0020] FIG. 10C shows a process associated with memory-controller owned defect management.

[0021] FIG. 11A shows a process associated with host-owned and shared garbage collection.

[0022] FIG. 11B shows a process associated with memory controller-managed garbage collection.

[0023] FIG. 12A shows a process associated with host-owned and shared wear leveling.

[0024] FIG. 12B shows a process associated with memory controller-managed wear leveling.

[0025] FIG 13 shows a process associated with assignment of available space at memory controller, and ensuing reverse lookup and communication to a host of assigned physical address(es).

[0026] FIG. 14 illustrates a block diagram of a storage server having multiple solid state drives (SSDs) and hard disk drives (HDDs).

[0027] FIG. 15 illustrates a block diagram of functions performed a storage server operating system.

[0028] FIG. 16 illustrates a block diagram of storage management tasks used by a storage server.

[0029] FIG. 17 illustrates how space reclamation is applied to mitigate P/E asymmetry.

[0030] FIG. 18 illustrates a garbage collection process.

[0031] FIG. 19 illustrates a process associated with wear leveling.

[0032] The subject matter defined by the enumerated claims may be better understood by referring to the following detailed description, which should be read in conjunction with the accompanying drawings. This description of one or more particular embodiments, set out below to enable one to build and use various implementations of the technology set forth by the claims, is not intended to limit the enumerated claims, but to exemplify their application to certain methods and devices. The description set out below exemplifies methods supporting cooperative memory management between a host and a memory controller, and improved designs for a memory controller, host, and memory system. While the specific examples are presented, particularly in the context of flash memory, the principles described herein may also be applied to other methods, devices and systems as well.

Detailed Description

I. Introduction.

[0033] This disclosure provides techniques for cooperative interaction between a memory controller and host. The techniques call for the memory controller to store information specific to each of plural subdivisions of memory, and to make data based on that stored information accessible to the host to assist with management of the memory. For example, the memory controller can store a table with information for each logical unit or physical unit spanning the entire memory managed by the controller. The data provided to the host can be the stored information itself, or the result of processing or filtering performed by the memory controller based on that information. In detailed embodiments, the memory controller includes logic that processes host queries and that provides data back to the host responsive to those queries. Further, the information stored by the memory controller can include status information (that is, information regarding the state or history of either a unit of memory cells, or

regarding the state or history of logic contents of those cells); some non-limiting examples of such information are depicted in FIG. 6. The data provided back to the host can either be generated synchronously (in response to host trigger) or asynchronously (i.e., only upon satisfaction of a processing condition). Once in receipt of the data, the host can then electively take action in dependence on that data.

[0034] In one embodiment, the stored information can include one or more fields of metadata representing each erase unit (EU) or physical page of a flash memory. Examples of data that a memory controller could provide to the host responsive to this data include (i) extent to which a set of constituent pages of a particular EU have been released, (ii) a list of unerased EUs in order of amount of stale space (e.g., candidates for space consolidation), and (iii) data representing frequency of individual page or EU usage; naturally, these examples are non-limiting. Note that a host within this context can be any integrated circuit or other apparatus (e.g., such as virtual or actual machine) that is separate from the memory controller and that transmits a request to the memory controller for some type of processing or response. In one embodiment, the memory controller has interface logic that permits a host to request any of these pieces of information by issuing respective commands. To provide an example of an action a host could take based on this exemplary information, a host armed with this information can issue an erase command directly addressed to a specific EU of memory. Similar examples exist for garbage collection, space reclamation, wear leveling and other memory management functions.

[0035] Note that in one embodiment, this infrastructure can be employed to substantially eliminate the need for a flash memory controller to implement a flash translation layer (FTL). That is, a flash memory controller can maintain per-subdivision data, which is accessible to the host (e.g., retrievable by, or against which the host can issue function calls or queried). The host uses this information to issue new write commands that are targeted to specific physical locations in flash memory, thus substantially avoiding the need for translation at a memory controller, and reducing the likelihood of uneven wear. In addition, the host receives alerts, and issues queries as necessary, to ensure that it is generally apprised of events that signify the presence of wear, cold data, excessive "released" space and low available memory space and so forth. This information is then advantageously used in host-dictated scheduling of maintenance operations in flash memory, such that those operations do not interfere with writes and reads needed by the host. To provide one optional implementation illustration, in a storage

aggregate having multiple storage drives, maintenance operations can be scheduled by the host for a first drive (and delegated to a memory controller for that drive) while the host is occupied with transactions directed to a second drive, i.e., the maintenance for one drive is "stacked" behind operations in another.

[0036] In embodiments below, a memory controller facilitates these actions by maintaining a stored repository of information, for each physical subdivision of memory, and updating this information as memory transactions are processed, e.g., as pages are released, and other events such as wear and defects transpire. If supported by the specific implementation, the host at any time can issue "synchronous" commands to the memory controller for an immediate return of information. Also if supported by the implementation, the host can issue asynchronous commands, which for example, cause the memory controller to "automatically respond" if and when a specified threshold is reached. The host has software which periodically issues these queries, maintains data necessary for L2P translation, and responsively schedules maintenance operations such as wear leveling and garbage collection. That is, host commands advantageously equate physical and logical space from the vantage point of the flash memory controller, because those commands directly address physical pages, EUs or other unit that are the target of intended operations, with at most insubstantial address translation in the memory controller. For example, in an overwrite (copy-on-write scenario), the host is made aware of a new memory location available for writes and directly writes modified data to the new memory location; while the flash memory controller can keeps track of metadata such as page release status, the host tracks the updated page address including any logical to physical translation and therefore can directly address it in the future (e.g., using the physical address). Note that the flash memory controller can still perform some mapping and address translation, even in this scenario – as one example, if a write error occurs, the memory controller can itself remap logical space on a limited basis to available memory space (e.g., with or without notifying the host). Even if the host is not immediately informed of the remapping, the discrepancy between physical and logical space will eventually be worked out through garbage collection and bad block management, and the host ultimately consolidates remapped data in physical address space that is directly addressed by host commands.

[0037] Employed in the context of flash memory, these techniques can help mitigate the concerns mentioned above, and thereby facilitate greater usage of nonvolatile memory in direct direct-attached and/or network-attached storage environments. That is, the techniques presented above can help

reduce flash control bandwidth competition with host-initiated reads and writes and can help minimize write amplification. These techniques, in turn, can help substantially eliminate the need for an FTL as conventionally used, which leads to further efficiencies. By redefining host and/or controller responsibilities, host-controller management features duplication and associated disadvantages can also be avoided, leading to a simpler and less expensive memory controller design.

[0038] For example these techniques facilitate better pipelining of commands in flash memory systems. In an implementation where there is no FTL table that must be loaded into a flash memory controller, and no associated search tree, flash memory is capable of servicing host read requests more quickly. The reduction of write amplification and controller-initiated erase, wear leveling and garbage collection operations reduces unintended competition with host-initiated commands, i.e., the host is far less likely to find flash memory "busy" when it seeks to have a command filled, because the host is vested with scheduling of the potentially competing functions. In turn, the better pipelining permits a host to more easily interact with a storage aggregate having one or more discrete flash drives, optionally including other types of storage drives (i.e., mixed or non-homogenous memory).

[0039] In another embodiment, this disclosure provides for cooperative interaction between a host and memory controller where the host manages logical to physical (L2P) address translation. The host stores a L2P translation table; when the host has a need to issue a command to memory, it performs translation and directly addresses specific storage units in memory. In this manner, a memory controller (e.g., a flash memory controller) does not need to perform local address translation that might add latency in responding to host commands. Optionally, the memory controller of this embodiment uses the stored information for each of multiple subdivisions of memory, as referenced above.

[0040] This disclosure therefore also provides storage systems, hosts, network-ready storage servers, methods, software and other implementations consistent with the principles introduced above. Generally speaking, the techniques disclosed herein can be implemented in any one or more of these components, in a manner where a component can be designed, manufactured and/or sold for optional inclusion in a complete storage system having a host, memory controller and memory. Consistent with these principles, one implementation of the techniques referenced above is in a storage system having at least one memory controller and memory. Each memory controller and its associated memory can be bundled together as a storage drive and, in one implementation, the storage system includes plural

storage drives, at least one of which is a solid-state storage drive (SSD) based in nonvolatile memory such as flash memory. Each memory controller maintains the information referenced above for each associated physical subdivision of memory, in one embodiment, for each unit of memory representing a smallest group of memory cells that can be erased in a single operation. The information maintained for each subdivision by the memory controller can optionally include one or more of:

whether the respective subdivision has been marked as bad;
whether a memory operation is currently in progress in the respective subdivision;
number of erase operations performed on the respective subdivision;
a period since data was last programmed within the respective subdivision;
number of reads to memory within the respective subdivision since last erase;
a logical address for data stored in the respective subdivision;
a number of pages released within the respective subdivision;
a number of pages used within the respective subdivision;
information representing page usage for all pages within the respective subdivision;
whether the respective subdivision has been programmed out of order relative to others of the plural subdivisions; or
whether the respective subdivision has been assigned to defect management.

[0041] Other types of information can also be stored by the memory controller.

[0042] In one contemplated implementation, each subdivision is an EU or page of NAND flash memory, and a NAND flash memory controller can provide a map to the host indicating relative information (e.g., page utilization) for all subdivisions of the specific drive or a storage volume spanning multiple drives. Alternatively, the flash memory controller can provide filtered or derived information based on such a map to the host, for example, a listing of EUs best suited for space consolidation based on some threshold applied by the flash memory controller. Such a threshold can be defined as a default and/or can be dynamically programmed by the host (e.g., by asynchronous command).

[0043] FIG. 1 illustrates a first embodiment of a storage system 101 and associated memory controller 103, host 105 and memory 107. In the illustrated embodiment, the memory controller is structured to cooperate with the host 105 in the control of the memory 107. The memory controller

103 has at least one first interface 109 to exchange commands and data with the host. Although two such interfaces and corresponding transmission paths are seen in FIG. 1, these interfaces may be combined (e.g., with communications occurring via a packet-based transmission scheme). The commands generally relate to operations in memory such as read and write operations, although commands can also be directed to the memory controller 103 to assist in memory functions. In one embodiment, the commands and signaling protocol are compatible with one or more standards, for example, with Non-Volatile Memory Express (NVMe) or the Small Computer System Interface (SCSI) (in the case of commands) and Peripheral Component Interconnect Express (PCIe) or Serial-Attached SCSI/Serial ATA (SAS/SATA) (in the case of signaling formats). The memory 107 generally has an array of memory cells and array control circuitry that may support one or more planes or banks depending on design. The memory core in turn has one or more subdivisions of memory cells for which subdivision-specific usage data will be tracked by the memory controller 103. In embodiments where the memory is flash memory and the memory controller a flash memory controller, each subdivision can include one or more erase blocks or units (EUs), with each EU having a minimum number of memory cells that must be erased at once.

[0044] The memory controller tracks subdivision-specific-usage data using internal storage 111. In one embodiment, this storage can be volatile memory such as synchronous random access memory (SRAM); in another embodiment, this storage can be non-volatile memory, for example an internal flash array. As denoted by reference numeral 113, the storage retains information for each subdivision of the memory governed by the memory controller, in this case, for a physical subdivision of the memory 107. In embodiments where the memory 107 is a NAND flash memory, the storage retains information for each EU or physical page of the flash memory (e.g., EUs *1-n* as indicated by reference numeral 113). Note that for flash memory, each EU can also correspond to multiple pages, as indicated by numeral 115 (e.g., pages *1-j*). For example, depending on manufacturer and design, there can be 128-256 pages per EU, with each EU corresponding to a substrate well, and each page corresponding to an independently controlled wordline for memory cells tied to that substrate well. The memory controller also has logic 117 that is operable to send to a host either some or all of the "raw" information retained in the storage 111, or derived or processed information based that storage 111. This logic for example can include circuitry within the memory controller that is adapted to respond to host commands seeking specific data; alternatively, this logic can also include circuitry that applies pertinent filters or comparisons and that notifies the host when a tracked metric meets an assigned threshold. This information or an alert

representing a particular condition can be transmitted to the host via the at least one first interface 109, via a dedicated connection or via a backplane connection.

[0045] Several configurations are also represented by the embodiment of FIG. 1. First, as represented by numeral 121, the memory controller 103 can be designed as a standalone integrated circuit with the host 105 and the memory implemented as one or more discrete integrated circuits (e.g., the host in the form of a host processor). Second, as represented by dashed-line box 123, the memory controller 103 can instead be co-packaged or otherwise combined with the memory 107 as a storage subsystem. For example, dashed-line box 123 can represent a discrete solid-state drive (SSD) where the memory controller is implemented as an IC and where the memory is embodied as one or multiple NAND flash devices. It is also possible (as represented by dotted-line box 125) to combined the memory controller 103, the host 105 and the memory 107 as a single system, for example, a network-attached storage system or a storage system adapted for connection with another digital system (e.g., via a USB, PCIe, SATA, Ethernet or other standard signaling protocol). Finally, as represented by reference numeral 127, cooperative management functions can be embodied as instructions stored on non-transitory machine readable media, for example, for use in controlling a host processor, memory controller or other circuit. That is, software or firmware can be used to control actions of a host, memory controller or other circuits.

[0046] With principle parts of a cooperative memory system thus introduced, this disclosure will now proceed to a more detailed discussion of examples of memories, memory controllers, hosts and systems that implement principles of this disclosure. Generally speaking, FIG. 2 is used to provide additional detail on a NAND-flash implementation (e.g., a solid-state drive having flash memory and a memory controller). FIGS. 3A-13 are used to provide additional detail on a cooperative memory controller that implements principles of this disclosure, including various types of usage information that can be tracked in various embodiments and ways in which that information can be used to facilitate cooperative host-memory controller management of memory. Finally, FIGS. 14-19 will be used to describe possible system implementations, including a storage server and a heterogeneous memory system.

II. Architecture Of An Exemplary, Cooperative Flash Controller.

[0047] FIG. 2 shows a solid-state drive (SSD) having a memory controller 200 and NAND flash memory comprising one or more NAND flash memory devices 207. The flash memory controller 200 includes storage to store information for each subdivision of memory as referenced above, as well as logic that services host commands relating to that information. The logic is partially in the form of an embedded processor 202, which receives commands from the host and fulfills those commands under the auspices of firmware. This logic and firmware will be further exemplified below in reference to FIGS. 3A and 3B, but for purposes of FIG. 2, it is noted that this logic relies on internal memory 203 including volatile storage (e.g., DRAM, or another very low-latency storage technology, for example, using a double-data rate or "DDR" signaling scheme) and nonvolatile storage (e.g., internal flash memory for the firmware). The memory, and associated firmware and data, are accessed via a dedicated interface 204. Once again, in one embodiment, the flash memory controller interacts with a host using exchanges of commands and data that are compatible with one or more well-known communication standards, such as NVMe or SCSI. Each of these standards provide for commands to be sent from an initiator (such as the host) to a target (such as a storage device). Signaling formats used by these commands structures can follow any desired signaling standard, for example, a version Peripheral Computer Interconnect Express (PCIE), serial ATA (SATA) or another signaling standard. The interaction can take the form of commands for memory transactions (e.g., read and write transactions), configuration commands to the memory controller (e.g., asynchronous commands), query commands (e.g., commands for synchronous or asynchronous returns of information based on memory controller processing requested by the host), and alerts and returns of various operating codes and other information from the memory controller. Generally speaking, a "synchronous command" as used herein will refer to a command to the memory controller which initiates a function in the memory controller that returns a result as soon as processing is completed. A synchronous command is analogous to a query. An "asynchronous command" as used herein will refer to a command that requests a return only once another condition is satisfied. Such a return of information can be triggered immediately (e.g., if the condition is already satisfied) or in the future (e.g., the memory controller alerts the host immediately and without solicitation when a condition specified by the host is later satisfied). An asynchronous command can be thought of as the host setting an operating mode or condition in the memory controller, e.g., setting a mode calling to trigger an immediate alert if previously released space exceeds a host-specified level.

[0048] To perform input/output (IO) operations, controller firmware interacts with a low-level flash memory interface 205 to translate high-level IO commands into flash memory operations. In this

embodiment, the flash memory consists of one or more NAND storage devices (e.g., integrated circuits) 207, each coupled to the flash memory interface 205 via a multidrop channel. Each device 207 includes a multiplex interface 208 that connects to one or more co-packaged dies 209. Each die can have one or more planes 210, each with independent control and data registers 211, so that each die is capable of performing multiple IO commands simultaneously (e.g., an operation for each plane). These registers and respective planes can be delegated commands (e.g., programming commands) by the memory controller, or alternatively, the memory controller can use a fractional programming scheme. Following SCSI protocol tradition, a logical unit (LUN) is used to refer to the smallest device unit addressable by IO operations.

[0049] As mentioned, the controller can serve detailed information to the host for each subdivision of memory, and the host can also issue query requests to the memory controller, designed to assist the host with management of the bandwidth-consuming functions referenced above. Advantageously, to avoid delays associated with memory-controller-resident address translation, the host sends IO requests to the memory controller that directly specify physical address. Note that for a NVMe compatible embodiment, the predicate of direct physical addressing can be supported simply by configuring the memory controller to manage the host-specified address as a physical address, or otherwise with limited memory-controller-side address translation. The performance of commands (e.g., management of multiple program-verify cycles, or "P/V" cycles, of one write command) is then managed by the memory controller which alerts the host upon command completion. The meaning of the physical address in this context depends on flash memory geometry but, in this embodiment, includes multiple fields. These fields can identify for example the number of a communication channel within the flash memory interface 205, the number of a device 207 connected to that channel, the number of a die 209 within that device 207, the number of a plane 211 located on the die 209, the location of a block within the die 209, and the location of a page within the block. Thus, physical address in this embodiment includes a quadruple of channel, device, die and logic unit number (LUN).

[0050] FIG. 3A helps illustrate layout of one possible flash memory controller, with an emphasis on command processing. In particular, the flash memory controller can be implemented as a single integrated circuit 301. As before, a host interface 305 is used to exchange commands and data with a host, and a flash interface 307 is used to issue commands to and exchange data with one or more flash memory devices (not shown in FIG. 3A). Note that in this embodiment, a single path is illustrated as

coupling the memory controller with the host, for packetized exchange of both commands and data; that is, the host interface can include logic that extracts commands and data from request packets, and can also packetize read data, alerts, metadata and other communications to the host. Other implementations are also possible, for example, using separated command and data busses. In the scheme depicted in FIG. 3A, it should be assumed that the host interface has a high speed serdes interface, such that communications with the host occur over one or more differential signaling paths, for example, compliant with a PCIe, SATA or other signaling scheme. Note that one or more flash devices can be copackaged with the memory controller, and thus the flash interface 307 does not necessarily have to use the same signaling scheme as the host interface, e.g., communications can be via wide-bus single-ended communications, using command and data busses.

[0051] The host interface 305 separates controller commands from any received packets (as necessary), and routes these commands over a command bus 308 internal to the flash memory controller. Generally speaking, commands will relate to memory operations (e.g., reading or writing of data) or to queries for data and memory management functions. To this effect, separate logic blocks within the memory controller are dedicated to each of these functions.

[0052] A memory operations logic block 313 manages operations involving the memory device. For example, as is well-known, NAND flash memory typically utilizes incremental programming – that is, array control circuitry for the flash memory device uses a minimum programming voltage, and results of programming are checked against contents of a write data register to determine whether those results are correct. This is performed using a "program-verify" (P/V) cycle having separate "program" and "verify" phases. During validation, a programmed page is typically sensed and an exclusive-or function is performed with contents of the write data register; for any bits that remain set, programming has not been effective, so the programming voltage is raised and the process repeated in a second P/V cycle. This process typically continues until proper programming has been achieved or some limit has been reached, the latter case resulting in a write error. The memory operations logic block 313 performs control over these various phases of programming using buffers 315. Since a memory device can include multiple planes (as discussed above), the memory command processing logic block 315 optionally uses multiple buffers, for example, with one dedicated to each plane or with buffers assigned on a rotating basis for individual commands received from the host. The memory command processing logic block also manages any functions of reporting write error and consequent remapping of data, as

well as updating L2P mapping information in metadata storage 311 (for embodiments that perform such mapping). Note that this information can be part of a much larger class of metadata indexed by EU as discussed above (see, e.g., FIG. 6 and the accompanying discussion below for examples on types of information that can be tracked using this storage).

[0053] Commands relating to the return of query information (e.g., synchronous commands) or the setting of an operating mode (e.g., asynchronous commands) are sent to query/management logic block 317. Generally speaking, the host can request (a) return of raw information for the entire memory space managed by the memory controller, or for specific address ranges or EU ranges, or (b) processing or filtering of that data by the memory controller. For example, as referenced previously, the memory controller can be provided with logic operable to receive and interpret host commands for lists of blocks, e.g., the "top ten" candidates for garbage collection, ranked in order of page (under) utilization. Since the purpose of such an operation is to identify EUs for erasure, a memory controller receiving such a command interrogates the metadata storage to (a) identify EUs that are at least partially in use, (b) identify the extent of page utilization for each such EU, and (c) order a set of EUs that are candidates for erasure in the order of greatest number of released pages. The query/management logic block 317 uses internal registers 319 to manage functions like this and, when the requested processing is complete, the query/management logic block sends a reply message to the host with the requested information. Note that the host interface 305 includes buffers that receive this data, and queue the data for transmission to the host (e.g., as a reply packet that may be distinguished from read data). The mentioned example is only one type of query that can be processed by the host, i.e., there are many types of requests and queries that can be processed by a memory controller of the present disclosure. A skilled designer can implement any type of management processing desired. This request is an example of a synchronous query, because data is on-hand for the memory controller, and because the memory controller returns requested data as soon as its query processing is complete. In a simplified case, the host can request return of raw metadata. For asynchronous queries or functions, the host typically provides a parameter (such as a mode setting and/or a host-selected threshold of some type) and the memory controller then operates a continuous or intermittent process to check for the specified condition; when the condition occurs, the memory controller immediately alerts the host, typically providing information that triggered the alert (such as EU identity and metadata for the EU pertinent to the function being tracked). That is, if the condition does not occur, there is no responsive alert sent to the host, and the condition is monitored for occurrence at any time in the indeterminate future. As should also be apparent,

sophisticated queries can be run that involve multiple metadata parameters. For example, a host might request an alert any time a partially written EU simultaneously reaches a specific page utilization threshold (e.g., less than 50% utilization) and has a time since last data access greater than a second threshold. Many examples of asynchronous processes are possible and will no doubt occur to the skilled memory architect. Once again, any suitable thresholds or modes are stored in registers 319.

[0054] A media icon 321 is depicted to indicate the use of software or firmware by the memory controller. The memory operations and query/management logic blocks 313 and 317 and registers 319 are all depicted within a dashed-line box 323 denoting that, in one implementation, these elements can optionally reside on a single die (e.g., a single processor or coprocessor); in such an embodiment, the media icon 321 typically denotes the use of firmware, for example, stored in memory within the single die in question. In this regard, such firmware can be designed to respond to vendor-specific NVMe extensions to implement specific query/management functions. For example, any desired asynchronous query can be implemented as a function call supported by firmware; when the asynchronous command in question is triggered, it is run as an open process in the die (323) or a coprocessor dedicated to the query/management functions. Alternatively, many processors can be provided, each assigned queries/asynchronous processes as they are invoked. As implied earlier, a specific asynchronous process can be associated with a mode set by the host; for example, in one mode defined by the host, the memory controller can automatically alert the host any time it identifies a space reclamation (garbage collection) candidate, based on default or host-identified parameters – in a second mode, this function is turned "off." Note that in the future, newer versions of standards such as NVMe can be structured so as to inherently support generic commands calls consistent with the operations discussed above.

[0055] The metadata storage 311 is indicated as separate from dashed-line box 323, reflecting that the metadata storage optionally can exist independent of the processing logic, e.g., on a separate die. That is, in one embodiment, the metadata storage consists of nonvolatile memory, such that it is persistent in through power cycling. In another embodiment, the metadata storage can exist in SRAM (e.g., internal to optional die 323), with data switched to nonvolatile memory and loaded back again in response to power cycling. In still another embodiment, as denoted by optional connection block 325, the metadata storage can be read directly by the host, i.e., via a form of commands for direct memory

access. In such an embodiment, the host simply reads a special memory location which it knows will contain metadata for a specific EU or EU range, or for the flash memory as a whole.

[0056] FIG. 3B shows a block diagram of a cooperative memory controller 351; more particularly, FIG. 3B is used to show how logic functions can be implemented using hardware and firmware logic 353. This logic 353 supports a number of optional processing features. In one embodiment, this memory controller uses this logic to perform many traditional flash controller functions (e.g., management of an FTL and associated search trees). This is to say, while one advantage of the memory controller 351 is the substantial reduction or omission of an FTL layer, this is not required for all embodiments. In another embodiment, the memory controller implements comprehensive cooperative functions that help reduce the need for complex FTL processes, as referenced above. In each embodiment, the memory controller 351 serves to the host unit-specific information for all units of memory governed by that memory controller; with this information, the host can directly address specified physical locations, thereby substantially reducing or obviating need for complex FTL processes resident at the memory controller.

[0057] For the embodiment of FIG. 3B, logic 353 is seen to have a number of basic function blocks, including interface logic 355 to interact with the host using packetized commands and responses, logic 357 used for local metadata management, command processing logic 359 used for query processing and other management functions, and IO scheduling logic 361 used to manage memory transactions (e.g., program and erase operations). As noted, even in an embodiment where it is desired to substantially reduce the FTL layer, a memory controller can still optionally implement some address translation, for example, for defect remapping as well as other functions. This will be described below. The metadata management logic 357 maintains locally-stored information in the form of metadata 363, as mentioned, for each unit of memory of the memory device. Non-limiting examples of information that can be tracked are once again discussed below in connection with FIG. 6. This information can include L2P or P2L mappings for each memory unit; that is, as discussed below, even for an embodiment where the host uses direct physical addressing, a scheme of shared address assignment responsibility can be implemented, with the memory controller assigning physical addresses based on an initial logical address specified by the host, with the memory controller reporting back assigned physical address(es) to the host, for the host to build a L2P table for future reads to the data. This shared addressing scheme can also optionally be employed for certain other delegated functions, e.g., valid data relocation, unit erase, garbage collection, defect management and other functions. Note that this is not required for all

embodiments, i.e., as will be detailed below, a host can also support these functions and dictate physical addresses for new writes. In many embodiments described below, an adaptable memory controller architecture will be described, providing implementation flexibility in assigning specific tasks to either host or memory controller. To these ends, the metadata management logic can include structure to assign physical address and generate updates to stored metadata 363 and to report back to the host an index of physical locations assigned the newly-written file for situations where address assignment is to be performed by the memory controller.

[0058] To provide another example of use of a limited address translation scheme notwithstanding address directed by the host, the memory controller can be configured to identify write error and to transparently remap the subject data over to reserved memory space. Because such reassignment might affect only a very small portion of data written to memory, the memory controller can advantageously keep track of this reassignment using the metadata 363. Future reads specifying the remapped EU are intercepted by the memory controller logic using locally-stored metadata 363 and redirected to the proper physical location in memory for defective blocks. In this manner, the memory controller is freed from having to implement extensive search trees to find physical locations based on supplied logical addresses, i.e., the memory controller need only track defective memory reassignments, which ultimately become stale as the memory controller progresses through erase operations, garbage collection and updates of data (the latter being directly written to new pages or EUs). Note that such addressing can be made even simpler if memory controller simply allocates remapped space to a reserved EU using a like page assignment.

[0059] The command processing logic 359 receives commands from the host directed to general configuration of memory operations and for queries. Generally speaking, this logic manages and updates the metadata 363 and runs queries on the metadata, returning information to the host as appropriate via the host interface logic 355. The returns to the host can be immediate returns to synchronous commands and immediate or later responses (or alerts) to asynchronous commands. Exemplifying some command options, the command logic can (a) serve information up to the host drawn from metadata 363 for use in wear aware writes, and (b) assist with wear leveling (WL), garbage collection (GC), defect management (DM) and integrity management (IM) functions in memory. Note that in network storage applications with multiple drives, this further enables certain novel host capabilities, as will be described in the section dealing with an exemplary storage system further below.

Note also that in some embodiments the host can also directly access raw metadata, e.g., as a form of direct memory access.

[0060] An exemplary memory controller can assume varying levels of host support in a manner that can be customized to any specific memory system design. That is, memory controller 351 possesses dedicated logic infrastructure to perform WL, GC, DM and IM specific functions (369, 371, 373 and 375, respectively), each of which can be tailored to a specific level of interaction with the host pertinent to the specific implementation. Depending on the desired level of interaction, the memory controller 351 helps avoid the need for remote storage and retrieval of large address translation tables and the use of complex search trees, e.g., address translation can be performed using a greatly simplified address translation table or omitted in the memory controller entirely. In addition, the configured level of cooperation can advantageously permit a host to directly assume scheduling of many flash management functions that might interfere with (i.e., compete with) host-directed writes, such as garbage collection, data relocation, wear leveling and so forth. That is to say, an architecture will be described below that permits a memory controller to serve sophisticated information to the host to assist with this scheduling. This, combined with less FTL overhead, provides for faster, more consistent flash response, and facilitates multiple drive storage aggregates based on solid state (flash) drives (SSDs) as well as mixed or heterogeneous systems that combine SSDs with other memory types.

[0061] Note that this is an example only, e.g., the architecture described herein can optionally also support a traditional FTL design, or memory controller management of complex functions.

[0062] To assist with host scheduling of flash management tasks, the memory controller can have firmware or hardware logic (or both) dedicated to specific types of host commands and host queries. In the embodiment of FIG. 3B, this structure is illustrated as optionally including structure to assist with wear leveling (WL), garbage collection (GC), defect management (DM) and integrity management (IM) functions, but other functions or logic can also or instead be used. To support these functions, the memory controller uses command processing logic 359 to manage space allocation and space reclamation, and otherwise to service host calls for the return of management data. For example, this command processing logic can facilitate direct physical addressing by identifying for the host available (i.e., available, erased) address space, candidates for erase (released space), candidates for data

relocation and garbage collection (e.g., based on low page utilization), assistance with cold data location (e.g., wear leveling), or more general functions.

[0063] For both embodiments that use wear-aware writes as well as those that do not, the memory controller can include wear leveling logic 369. That is, to account for a limited number of flash memory P/E cycles (typically on the order of tens to hundreds of thousands of cycles for NAND flash), the logic on board the memory controller can be designed to track wear as part of metadata 363 and to provide this information to the host. If over time, certain units of memory are determined to represent disproportionately high or low wear relative to overall memory, wear leveling can then be performed. Note that for embodiments where wear-aware writes are used, wear leveling can be highly localized, i.e., performed as a data relocation option simply to redistribute cold data. The memory controller 351 can generate alerts when predetermined wear thresholds are reached, and can otherwise perform low level queries relating to wear leveling. In support of the techniques presented by this disclosure, the wear accounting logic 377 can keep a changing-list of EUs, ranked in order of coldest data, least wear, greatest wear or in another manner. In one embodiment, this logic can be prompted via an explicit host command to synchronously compile such a list or to asynchronously notify the host of EU identity any time a wear metric (e.g., EU erase count) exceeds a programmably-defined value. Then, when and as wear leveling is scheduled by the host, the host issues a command to the memory controller to relocate cold data and erase the old space (e.g., using relocation logic 379), thereby redistributing that space into a pool of available space used for active writes (and potentially more frequently-cycled data). Note that in an embodiment where the host directly addresses physical space and performs wear-aware address assignment, distribution of wear can be inherently minimized as part of the write process. However, disproportionate wear can still occur for data that is held for a long time and which is therefore deemed "cold;" that is, cold data can keep EUs out of circulation while other EUs are more frequently recycled. The memory controller architecture presented by this disclosure supports memory controller cooperation with wear management through the use of "limited" data relocation and wear leveling processes (e.g., directed only to specific address ranges within flash) as well as (if pertinent to the implementation), the scheduling and management of more extensive wear leveling, e.g., for entire flash devices or across multiple flash devices or drives.

[0064] As discussed earlier, a copy-on-write process can result in retaining old pages in flash memory that are stale. This is because a given EU can have other pages that are still in use, and the old

page location typically cannot be reused until the entire associated EU is recycled. Over time, substantial portions of flash memory can be locked up simply because a small fraction of space in many respective EUs is still in use. This situation can occur whether the host or the memory controller performs address translation. To address this, the memory controller of FIG. 3B therefore uses garbage collection logic 371 to assist with functions of periodically consolidating data. That is, the garbage collection logic can track information pertinent to whether an EU is mostly stale or is still efficiently being used and can process host queries relating to this information. One form of this tracked information is page utilization information, e.g., where the memory controller stores information indicating whether each page in a given EU is available (erased but not yet written), has valid data (cannot be written to without an erase), or is released (a page with stale data that cannot be rewritten until it is first erased). Garbage collection involves accounting for released pages, for example, using release accounting logic 381 to track the mentioned page utilization information for each page (or other logical unit) for each EU; EUs with relatively few used pages can have those pages consolidated with pages from other EUs having similarly low page utilization, with the then-concentrated valid date being rewritten to a new destination. In an embodiment where the memory controller tracks this information, the memory controller can compute an extent of page utilization (e.g., 10% valid data) and can provide this information to a host with EU identity to permit the host to decide which EUs should have data consolidated and moved. The host can then schedule any resultant operation in a manner that does not compete for needed read data elsewhere in the subject memory. Note that "page utilization" as used herein generally refers to the extent to which pages of a given EU are valid (in use) or are stale or not in use. For example, if most pages in a given EU were stale or unused and only a few pages of the EU had valid data, the extent of page utilization in the given EU would be low. Conversely, if most pages in an EU were in active use, the extent of page utilization for the given EU would be high.

[0065] In an embodiment where the host cooperates with the garbage collection task, the host can query the memory controller using a command, with processing of the command performed in cooperation with the release accounting logic 381. In more detailed embodiments, the release accounting logic can be designed to perform low level inquiries, for example, to return a list of EUs where page utilization falls below a specific threshold (e.g., 50%). This type of function can also be managed as an asynchronous task, e.g., the host can request that the memory controller alert the host if at any time an EU that has been written-to (or that has just had a page released) experiences less than a threshold level of page utilization; in this regard, the release accounting logic 381 tracks explicit page

release with each command information update, and can perform any processing necessary to alert the host in response to any asynchronous queries. The release accounting logic 381 also has circuitry and/or firmware that performs other forms of processing, for example, optionally providing a list of "the 10 best" candidates for garbage collection in order of page (under)utilization. In another embodiment, some or all of the data relocation functions can be managed by the memory controller, for example, with relocation logic 383 being delegated specific tasks by the host (such as the identification of EUs to the host for relocation of data, or relocation of data in response to a host-specified target memory address). Once relocation has been performed, with respective L2P mappings updated and associated physical pages are released, the full EU is reclaimable. In one embodiment, this is performed by the host, which issues an explicit EraseBlock command for an address-specified EU – logic 355 processes this command and, once the command is completed, returns the freed EU to a pool of available EUs for future data allocation.

[0066] Write and erase errors discovered by the flash memory controller are handled by defect management flash management logic 373. Pages found defective due to write error are remapped by defect remapping logic 385, with write operation retried transparent to the host. The original page in error is marked defective or "bad" and added to a bad block list or map to avoid further use of associated physical pages. Unit relocation logic 387 performs background relocation of data that experienced write error to minimize possibility of further write errors. Unit erase errors are handled by the defect management logic as well, with defective EUs also reflected on a bad block list kept by the flash memory controller. As indicated, in such a scheme, the entire EU can optionally be moved, preserving relative page layout and simplifying translation issues.

[0067] While flash memory typically provides strong error detection and correction (EDC), the memory controller may also provide onboard data integrity management logic 375. Data scrubbing logic 389 is responsible for background data integrity checks based on EDC or additional data integrity metadata. Suspect blocks with transient errors identified are relocated by suspect relocation logic 391 using mechanisms similar to wear leveling and garbage-collection relocation processes described above.

[0068] As operations are performed in memory, whether as part of a management operation (such as data relocation) or in servicing a write or read command, IO scheduling logic 361 detects completion of the command. Pertinent information is added by metadata management logic 357 to the stored

metadata 363 for the pertinent EU, and the host is then signaled with any completion codes as necessary. For example, if a data relocation operation has been performed, the metadata 363 can be updated with new information for both source and target blocks of memory (e.g., new page release information, L2P and P2L mapping, wear information and so forth), and the host can be signaled with new physical addresses for relocated valid data.

[0069] Note that, depending on embodiment, the flash memory controller can support one or more of the functions or units of logic described above, i.e., a memory controller can include subsets of the described logic to assist in the performance of specific tasks. For example, one hypothetical memory controller could omit the data relocation logic 383 if the host was to perform this function. Also, other logic functions can also be supported or provided for beyond those discussed. As mentioned, the embodiment of FIG. 3B presents a single memory controller architecture adaptable to multiple, different implementations, but this is not required for all embodiments.

[0070] Clearly, many examples exist for the layout of a cooperative memory controller. In various embodiments, these layouts can support different cooperative functions. FIGS. 4-13B are used to provide non-limiting examples of different types of functions that can be supported.

A. General Flow.

[0071] As discussed above, a cooperative flash memory controller can participate in and support periodic management tasks, such as relates to defect management, wear leveling, and garbage collection. A given task (e.g., an asynchronous task) will typically have an associated management condition that is triggered when a specific, tracked parameter exceeds a predefined threshold. Once the management condition occurs, data which triggered the condition is evaluated by the memory controller, the host or both based on pertinent metadata. Depending on implementation, individual management tasks and the need for action can be addressed solely by the memory controller, solely by the host (e.g., based on reported information), or cooperatively by both the memory controller and the host. That is to say, in any given embodiment, provision can be made for the handling of any one or more of the management tasks described above on a cooperative basis. In addition, these functions can also be made multimodal, i.e., invoked by components that invoke controller-owned processing, host-owned processing or cooperative processing, dependent on system design or implementation. That is, a

typical implementation can feature a single flash memory controller design which is customizable to a number of different memory systems or system configurations; depending on features supported by the specific system (e.g., the specific customer), the memory controller can support adaptation or configuration at design time (or programmatically at initial system configuration) to support the desired architecture. As an example, a particular cooperative memory controller might support cooperative garbage collection, but a particular design many not implement this type of cooperation between host and memory system – in such an implementation, the memory controller can be programmed (e.g., using a command register or mode register) to turn this feature (and associated functions) off. Typically, once initially configured for a specific implementation, a memory controller will maintain its configuration throughout system life or until the system is reformatted, although it is also possible to support dynamic modes for some purposes.

[0072] When configured to execute in the cooperative flash management mode for any one or more of the areas mentioned above, a cooperative memory controller provides methods and mechanisms for a host (e.g., host software) to define and detect the management condition, to query metadata in connection with that condition, and to electively take responsive action.

[0073] FIG. 4 illustrates a process of the host-controller interaction in management condition detection. The process starts at step 401. At step 402, either of the host or the memory controller determines a need to assess the management condition, for example, as a function of running a routine or based on some type of trigger. For example, in one embodiment, the host can initiate a process such as periodic wear leveling (either via explicit command or as a result of setting an asynchronous process in the memory controller). The memory controller can also be set for a default operation, e.g., automatically alerting the host or providing data unsolicitedly to the host any time all pages have been released for a given EU; such a process can be triggered by the memory controller, which notifies the host of a need for intervention or simply (unsolicitedly) provides pertinent information to the host. Such a notification can also be the result of an asynchronous process or mode selectively set by the host, as mentioned above. Many other examples also exist. In one such example, a particular implementation may call for specifically alerting the host "automatically" any time a page release is determined to be the last page release for an EU (e.g., the EU is completely released); in different implementation, this information might always be tracked by the host (or not tracked at all). Irrespective of how the process is initiated, at step 403, the memory controller provides data to the host based on tracked, subdivision-

specific (e.g., EU-specific) information associated with a particular parameter or metric. In one embodiment, each exchange between memory controller and host can be performed in a manner compliant with NVMe, for example, using a vendor-specific extension or a vendor specific NVMe administrative command. At step 404, host software checks the value returned at step 403 against a specified threshold to determine existence of a condition requiring intervention. On a negative determination, the process ends or otherwise loops to step 403 after a suitable delay. On a positive determination, host software proceeds to take management action (step 405). Once management action completes, the process either concludes or is looped back to step 402.

[0074] As an example of management action, a host can initiate an explicit erase operation that specifies an EU physical address. Because such an operation is host-initiated, it does not collide with a host-issued read or write request, thereby improving system performance. In a multiple drive system, the host can hide an explicit erase (and garbage collection and other functions) behind a read or write request to another system drive. For example, in a configuration where multiple drives are coupled to a host through (e.g., through a multidrop connection, a bridge or other routing scheme), commands from the host can be interleaved such that as read and write operations are performed in one drive, garbage collection and unit erase tasks are performed in another. In a system which has multiple SSDs, this parallel processing ability can lead to further performance benefits. Note that this type of parallel processing can also be performed where drives are connected point-to-point with a host.

[0075] FIG. 5 shows a method for host software to query metadata pertaining to pending management tasks; the host software can then determine the existence of a condition and optionally execute the decision it deems appropriate. The process starts at step 501. At step 502, host software – either in response to the management condition being detected with one of the methods described above or at will – requests the controller for metadata. For example, the requested information can be in the form of a map that verbatim passes tracked subdivision-specific usage data to the host, for example, by providing a bad block, EU or page utilization map to the host. At step 503, for each metadata range returned, host software performs any processing appropriate on returned metadata. At step 504, if management action is to be performed in response to the management condition being detected, host software either evaluates a current metric based on actions performed or queries the controller for a value. Based on the value obtained, at step 505 the host software checks if the condition still holds and if more metadata needs to be requested. On a positive determination, the

process loops to step 502 where additional metadata is requested. If the determination is negative, the process concludes (step 506).

[0076] As mentioned, the cooperative memory controller of this disclosure can track many different types of information representing subdivision-specific-usage. This tracking information can be in the form of metadata 600 that represents one or more parameters, illustrated in FIG. 6. A first tracked parameter represents unit state 605; in the case of an EU, this field can be used to store data indicated whether the EU is bad. Alternatively, this field can also be used to denote whether the EU contains valid host data, or is otherwise available for allocation to the host or to the controller. Status 610 can be used to track any active controller operation. Erase count 615 keeps track of the number of program/erase cycles to date for the particular unit (e.g. the particular EU). An age field 620 keeps track of how long it has been since data was written to the erase unit. Time in this context can be represented in many different equivalent manners, for example, based on a time stamp or any other desired numerical value or other metric. Read count 625 identifies how many reads have been performed for data stored in the unit (e.g. EU) since the last erasure. In one embodiment, time since last read can be computed from the erase or another metric, e.g., with those memory controller or host calculating time relative to other EUs or metrics, tracked either by the memory controller or the host. Address 630 keeps track of a logical address associated with the erase unit. Release Count 635 identifies number of pages within the particular unit (e.g., EU) that have been released. The bitmap field 640 lists or otherwise identifies pages within the erase unit that are free (i.e., available for writes), valid (with actively-used data), or released. Depending on embodiment, this type of data can be maintained in a variety of ways; for example, more than one field can be used to track each of these parameters, or release status can be inferred from valid data status, or vice versa. Other alternatives are also possible. The bitmap field can also be used to indicate whether the pages in the particular EU were written out-of-order. The remapped flag field 645 indicates whether the EU has been allocated to defect management. In another embodiment, the above metadata could instead be tracked on the basis of logical erase unit with a mapping to the physical erase unit.

[0077] FIG. 6 should be understood as providing non-limiting examples only, that is, not all of the listed data must be maintained by a memory controller, and each identified field should be viewed as optional. Similarly, there are also many other types of data, beyond those specifically illustrated in FIG. 6, which can be used.

B. Specific Flash Management Tasks.

[0078] Flash management tasks discussed below can be grouped according to three primary categories: Defect Management, Wear Leveling Management, and Garbage Collection Management. The novel memory controller-host interaction described in this disclosure provides a unique ways of handling these management functions in a way that can be customized to the specific implementation. That is, a single memory controller design can support a number of configurable options (or modes) for allocating responsibility for these tasks as described below. Prior to discussing these options, however, it would first be helpful to revisit some of the capabilities provided by the structure described above. Once again, in one embodiment, information can be tracked by the memory controller and made accessible to the host in a manner that helps reduce reliance on a memory controller to maintain detailed address translation tables. That is, for purposes of the discussion below, it should be assumed as a default that (1) the host maintains a reasonably-accurate understanding of physical location of data in flash memory and is thus capable of directly addressing data to the specific physical locations in memory at which the data will be stored, and (2) many of the operations which might traditionally compete with host read/write requests will instead be managed by the host, i.e., such that tasks are schedule so that there is no competition.

1. Capabilities.

a. Wear-Aware Writes.

[0079] As discussed above, the host can issue commands to the memory controller requesting a listing of space that is available for writes, that is, either EUs that are erased, or pages in EUs that are not yet written to following an erase operation; this information can be returned with wear data used by the host to selectively direct new writes as a priority to space that is relatively more virgin. That is, generally speaking, the host need not track available space and in one embodiment can simply at any time request an update of available space by explicit command to the memory controller. The host can rely on this information for a period of time (e.g., until its list of available space grows low) and the host can then schedule garbage collection and erase operations at a time convenient to the host.

[0080] FIG. 7 shows flow for a method 701 by which a host targets writes of new data to specific physical addresses in flash memory. The method begins at 703 in FIG. 7. Note that invocation of the

method can be triggered by the need for an application or an operating system to write data to memory, per numeral 705. The host is responsible for having a list on-hand with available free space; this list can be periodically updated by the host by query to the memory controller, e.g., after an erase operation is performed. Note that a steps associated with such a query are illustrated in phantom-line boxes in FIG. 7, i.e., are designated by function blocks 707, 709 and 713. That is, optionally in connection with an erase operation, the host requests the memory controller to identify all free space, sorted or prioritized in order of least wear; this listing is determined by reference to the memory controller's metadata repository 711. In a system having multiple SSDs, the host can maintain a dedicated table for each SSD or, alternatively, it can instead build a table spanning memory space for multiple SSDs using sorted information from each SSD. "Available space" or "free space" in this context refers to space that has previously been erased in flash memory but has not yet been written to, meaning it is available for immediate programming (i.e., writes). Per numeral 715, the host then chooses a write address for data based on the list. Note that other priority schemes besides those listed above can also be used; as a non-limiting example, space can also be assigned for writes based on data type (e.g., specific file types) and other criteria, tracked or otherwise. After selecting a suitable destination address, the host then issues a write command to the memory controller specifying a desired physical address within flash memory, per numeral 717. As indicated by function block 719, the memory controller then manages the write process and, once successful, returns a code to the host confirming a successful write. The memory controller also updates the metadata repository (711) stored for each pertinent EU (e.g., to indicate that the assigned space is now taken, and to update any other tracked parameters regarding the data or use of the particular memory space). As reinforced by function block 721, the host then updates its own translation tables (723) as appropriate, i.e., with little to no L2P translation performed by the memory controller. In one embodiment to be discussed below, in which a memory controller is to perform its own defect management, the memory controller can remap data and mark the intended physical address as "bad" without immediately reporting this information to the host – advantageously, L2P translation is restricted to few, notable situations such as this. For a NVMe compatible implementation, these processes can once again be implemented with a vendor specific command. For example, the host can issue a command requesting LUN ranges and the amount of space that the host software is considering to allocate. Upon successful completion, the command returns a list of erase unit size logical block address ranges, prioritized by wear, and described by (LUN, offset) pair or single offset value (if the LUN was specified in the host request). A returned command status code can further specify whether the requested amount of space was available,

whether the request was satisfied or whether further calls are required. The host then directs its writes as appropriate, directly specifying addresses for pertinent memory operations.

[0081] Note that writes targeted in the manner indicated above will inherently tend to level wear in flash memory, thereby reducing the need for wear leveling relative to some conventional schemes. As further noted below, even with such a scheme, asymmetric wear is possible and, therefore, cold data detection and consequent data relocation is advantageously performed even where wear-aware programming is performed, i.e., to perform "limited" wear leveling as described above. Also note that because the host has or is provided with direct physical addresses of free space, there is no need (or greatly reduced need) for L2P address translation at the memory controller.

b. Page Underutilization And Stale Space Reclamation.

[0082] To address page utilization issues, including the presence of stale (unused) data, the host can advantageously enquire at any time as to the extent of page utilization and/or aged data present in a flash device or SSD. First, note that updates of memory content in flash memory is typically processed as a copy-on-write operation, which writes updated data to a new memory location, leaving the old memory location in an unusable state (at least until the subject EU is the target of an erase operation). Note that in the context of the embodiments proposed herein, this operation can be performed directly by the host, i.e., with the host reading data from a first physical location, modifying that data, and writing the modified data directly to a different physical address based on a listing of available space. Over time, EUs can acquire greater and greater numbers of unused (released) pages, and so, experience low page utilization. Also, as certain times, it can be beneficial to inquire as to infrequently written data, so as to proactively assess whether any data should remain at its current location. That is, while wear-aware writes does effectively distribute wear, data that is only infrequently changed (e.g., a media library) can still be effectively tied up as infrequently written memory space while other memory space is more frequently cycled. "Cold data" detection processes (as well as "hot data" detection processes) can therefore be run to identify this data and determine whether this data should be moved, and optionally consolidated with other data, or potentially moved to other storage drives. This is to say, "cold data" can be moved to better distribute wear in a given memory space, and if that "cold data" has read/write characteristics similar to other "cold data," the respective data can be consolidated in one or more specific EUs or moved to a different storage drive or to another tier of a multi-tier storage system. This also enables the host to better match data access patterns to media characteristics, such as access

latency or available capacity. Note that shuffling various data based on type can help improve the efficiency with which memory is used and also help better distribute wear.

[0083] As with the various schemes presented above, either the host or both the host and the memory controller can participate in this type of evaluation. For example, the host can issue a command (e.g., a vendor-specific NVMe command) specifying an EU or a LUN or a range of memory space. Upon successful completion of the command, the memory controller returns information representing the amount of capacity used by, e.g., cold data, either on a global basis (e.g., by returning a map) or can identify "cold" or "hot" status for a specific LUN identified by the host. The host can also (or instead) instruct the memory controller to asynchronously generate a notification any time data meeting a specific "hot" or "cold" threshold meets a predetermined threshold. For example, in one implementation, a cooperative memory controller receiving an erase command automatically queries EUs present in the pertinent memory space to determine whether a deviation exists from a target wear level (e.g., based on erase count deviation). The memory controller can also provide other types of asynchronous updates to the host, for example, in response to a periodic cold data check, with either raw metadata being returned to the host, or with the memory controller performing a query and returning pertinent data (e.g., an indication that data at a specified location does not meet thresholds such that it would be considered "cold"). Other operations are also possible. As with the schemes identified above, logic on board the memory controller receives the pertinent command, and sets registers and initiates routines as necessary to synchronously or asynchronously provide the requested data.

[0084] In some embodiments, as mentioned, the host is provided with the ability to request stale space identification either for a flash memory as a whole, or for a specific LUN range. A function call to this effect can be implemented with a vendor-specific command that returns the list of EUs described by one or more offset values (e.g., based on an offset to a LUN if a LUN was specified in host request). Host software can proactively use this method to assess units with stale data or in the context of a request for relocation candidates. Such a synchronous process could be advantageously applied in response to capacity pressure or other situations requiring immediate action. Asynchronous processes can also be used, for example, pursuant to a host command that the memory controller is to notify the host any time global page utilization, EU-specific utilization, or available memory space match host-specified thresholds. Note that in a storage system having multiple drives, a host can choose the most suitable

destination for data relocation, potentially including another drive. For example, a host can elect to relocate data from one SSD onto another SSD, or even onto another memory type (e.g., a HDD). If host chooses to relocate valid or "cold data" within the boundaries of the flash memory device where this data resides, these techniques provide a method to relocate data without first reading it into host memory, thereby saving IO bandwidth and other system resources. That is, the host can delegate a data relocation operation to the memory controller; with such an operation, the host first queries the memory controller for a list of free address space (optionally meeting some criteria), schedules a relocation and then (at the appropriate time) provides a relocation command to the memory controller, with both source address and target address specified.

i. Delegated Copy/Data Relocation.

[0085] A delegated copy operation can be scheduled by the host in a manner (a) so as to not collide with ongoing data operations, and (b) without requiring movement of the affected data either to or from the host. Such a process is illustrated in FIG. 8 by reference numeral 801. More specifically, the host first determines that it needs to relocate or copy data, per reference numeral 803. As a non-limiting example, such an operation could be performed for an EU determined to have one or more defective memory cells. Alternatively, such an operation could be performed if page utilization for two or more EUs is low, and the host wants to consolidate "valid" data so that one or more EUs can be erased, thereby freeing all pages or other logical blocks associated with that EU. Still further, such an operation can be performed for cold data, to move it to a new physical memory address. Depending on the amount of data that must be copied, the host then identifies free space needed to hold the data to be copied, per function block 805. This query can be satisfied, for example, by retrieving information (e.g. a bitmap) representing all space managed by the memory controller; alternatively, the host can also query specific EUs or can indicate the amount of space needed, with the memory controller returning a list of available space. These options are generally indicated by reference numerals 807, 809 and 811. With addresses for both existing and recipient data locations identified, the host then issues a command (813) containing these addresses for the memory controller to perform a delegated copy operation. As an optional, implicit part of this command, the memory controller can be configured to release pages of the original source addresses following such a copy operation; note that with pertinent pages "released," the prior memory locations are in a condition where they cannot be written to until the entire EU associated with those pages is first erased. As another optional command feature, the

memory controller can be configured to automatically check whether such a release completely releases the pertinent EU (i.e., as a release of the "last page") and to initiate an erase operation for the EU if this is the case. The memory controller then (815) returns a code to the host confirming successful write and updates metadata for the pertinent EUs. Finally, per numerals 817 and 819, the host updates its translation tables such that the host can directly address physical locations of memory, thereby freeing the memory controller from the bulk of translation duties.

[0086] In one embodiment, a vendor-specific NVMe copy command can be used to specify a source logical or physical block address and destination logical block or physical block address. Multiple pairs of source and destination addresses can be specified by a single command; if such pairs are specified, any such copy operations are executed transparently by the memory controller until the argument list is exhausted or until the operation fails.

[0087] Note that a delegated copy operation as just described can provide substantial performance benefits, i.e., the memory controller is relieved from the bulk of address translation duties, with the host being primarily responsible for issuing commands that directly specify physical address. Furthermore, the use of the delegate copy operation charges the host with scheduling of copy operations, with the memory controller being responsible for completing a delegated copy operation once issued; since the host is in charge of scheduling such a command, it can once again pipeline command issuance so as to no unduly interfere with read and write operations, and it can hide a delegated copy operation behind operations in other memory (e.g., other planes or SSDs). Delegating the copy operation to the memory controller frees up host-controller interface bandwidth that might otherwise be consumed by the need to send data to be copied from the controller to the host and then back from the host from the controller.

ii. Explicit Erase.

[0088] Memory bandwidth competition between host and memory controller can be further reduced through the use of an explicit erase command. That is, one optional design takes scheduling of erasing of flash memory out of the hands of the memory controller and vests it directly with the host. The host therefore pipelines issuance of explicit erased commands at times when flash memory (e.g., a particular SSD) has bandwidth to process such a command; as before, in an implementation featuring multiple drives, a host can hide (stack) an explicit erase of one SSD behind data operations to another

SSD. Note that in a multidrop or other shared command path system, such an embodiment can interleave commands across multiple SSDs (each having their own memory controller), such that explicit erase operations are performed in one or more SSDs while data is exchanged in the performance of a write or read command in a different SSD (e.g., with data being exchanged over a shared data path).

[0089] FIG. 9 illustrates a method 901 for explicit data relocation and/or erase operations. The host schedules an explicit erase operation to occur either immediately, or when it has sufficient spare bandwidth. As indicated by numeral 903, typically this will be triggered by a condition when the host needs space (e.g., of amount X). The host then issues a command (905) to the memory controller querying erase candidates based on low degree of page utilization. Note that with such a command, available space is not counted, i.e., such space is already free and available – rather, in responding to such a query, the host seeks to identify space that is released (not in active use) and for which an erase operation must be performed for entire EUs before constituent space can once again be made available for programming. An explicit erase command or data relocation command (e.g., delegated copy, referenced above) can be performed as part of a garbage collection operation.

[0090] As with commands discussed previously, synchronous or asynchronous commands can be issued by the host, depending on desired implementation. Where the host immediately needs free space, it can issue a synchronous command to the memory controller, for example, requiring a listing of units where page utilization falls below a specific threshold (e.g., any EU where released page space is greater than a threshold, e.g., 50% of an EU's capacity). Many choices of metric are possible, and in some embodiments, complex conditions can be evaluated (e.g., EUs where more than 50% of pages are released, but where less than 10% of space is currently available). In response to such a command, the memory controller returns a listing of EUs (or logical units), sorted by any desired priority scheme (e.g., by lowest amount of wear). Asynchronous commands can also be issued, as represented by numerals 907, 909 and 911, seen at the right of FIG. 9. For example, some designers may choose to set a condition where the memory controller unsolicitedly sends an alert to the host any time "available space" tracked by the memory controller falls below a specific threshold, e.g., less than 10% of total capacity managed by that memory controller. Alternatively, the memory controller can be configured in some implementations to alert the host in connection with any page release if that page release causes the subject-EU to have more than a specific released page threshold. As yet another example, a memory controller can be configured to notify the host any time overall memory (under) utilization

(e.g., released page count to valid page count) exceeds a specific threshold. Commands that invoke these functions are examples of asynchronous commands, because the memory controller might respond immediately (e.g., if the specified condition were already satisfied) or at a later point in time (e.g., at a time of later occurrence of the condition specified by the command). As noted by numeral 913, a delegated copy command can then be executed by the host to relocate any valid data in an EU selected for an erase, with all constituent pages of that EU then being released. Then, per numeral 915, the host issues an explicit erase commands; the memory controller manages the erase operation (e.g., as a sequence of P/V cycles applied to a substrate well of affected memory) and returns a code to the host confirming successful erase (917). In a NVMe-compatible embodiment, an explicit erase command can be implemented using a vendor-specific extension of the "Deallocate" administrative command. The memory controller at this time also updates its local metadata repository for the memory in question, for example, identifying all logical units of the pertinent EU as "available," adding the EU to a list of available space, and so forth. Finally, upon receipt of the confirmation code from the memory controller, the host then updates its translation tables as appropriate (919/921).

[0091] Note that once again, explicit erase provides a benefit in that this operation is scheduled by the host (i.e., to avoid conflicts), with the host maintaining primarily responsibility for L2P address translation.

c. Determination Of Geometry and Physical Addressing.

[0092] To facilitate cooperative flash management and otherwise permit the use of discrete storage drives, a cooperative host and/or cooperative memory controller can provide a mechanism to permit the host to have insight into flash array geometry. To this effect, the host can request (or the memory controller can transmit to the host) information indicating the type of memory present, capabilities of the memory controller and other parameters. Such a task can be implemented with a vendor-specific extension to the NVMe Identify Controller command, where the memory controller responds by returning information such as the number of channels, targets per channel, LUNs per target, and number of EUs per LUN, and whether the memory controller supports specific management schemes for defect management, wear leveling or garbage collection, discussed below. Geometry information can also additionally include physical EU and page size, as well as many other types of information. This information can be used by a host to configure operation with the particular memory

device or drive, and to configure wear leveling, garbage collection and other schemes as appropriate. For example, the host and/or the memory controller can upon power-up request device configuration, and can then set the pertinent scheme(s) by responsively configuring a mode register. The pertinent scheme can be selected according to policy (e.g., based on the configuration of memory available or other system requirements).

2. Management Tasks: Configurable Allocation Of Responsibilities.

[0093] A cooperative memory controller and cooperative host can generally support a host-owned, memory controller-owned, or shared schemes for managing various tasks associated with flash memory. The particular choice of configuration is an implementation choice, and can depend in any given system of the type of data being managed, desired performance parameters, file management or network policies, or other factors. For example, a music server or video server application (e.g., where data has read-mostly characteristics and where data is read intermittently) might be more amenable to a host-owned scheme than to a memory controller-owned or shared scheme.

a. Defect Management.

[0094] Flash memory, as with any storage media, is potentially subject to data errors. While read errors are proactively handled using EDC techniques, write and erase errors may occur at random over the device lifetime, with some units initially marked unusable (i.e., bad) and others being initially good, but later being found to be bad and marked as bad over time. These units, whether initially bad or later marked as bad, are tracked by either the flash memory controller or the host so that writes of data are not directed to these units. Thus, a cooperative memory controller and host will typically support a scheme for managing defects which are detected in flash media. As with other management functions, whether a given implementation uses host-owned, memory controller-owned or shared defect management can be configured according to preference, for example, adapted at initial power-up (for example, in dependence on detected device or system geometry, policy and other considerations), via the programming of a mode register. For example, the registers illustrated in FIG. 3A can be used for this purpose. Note that in each mode or scheme discussed, the memory controller will still typically provide some form of cooperative input to the host, for example, by serving stored metadata, providing functional returns to host-initiated queries, using on-board storage for metadata and associated processing logic. That is, a cooperative or shared management task should not be confused with a

cooperative memory controller that provides certain functional services (e.g., including the serving of subdivision-specific data) that can be electively used by the host.

i. Host-Owned Defect Management.

[0095] For host-owned defect management, units marked unusable by the chip manufacturer are identified during first time device initialization. Host software is then expected to query the flash memory controller or the flash memory for this defect list using the method provided by the controller.

[0096] In an embodiment where host-controller communications are NVMe compatible, the host software and flash memory controller can interact using a vendor-specific GetLogPage command. When issued by host software, such a custom command is structured so as to return a log page describing logical address ranges marked as unusable, with each range identified by starting logical block address and the length in logical blocks (e.g., in pages where a page is the logical block size identified by the host). Recording defect areas so as to proactively manage further read and write access to the correct address is then the responsibility of the host. Note that the effect of such host-owned defect management is that once again, there is no need for L2P translation at the memory controller - the host registers and tracks defects and consequent changes in physical locations of data, and then it directly accesses that data as appropriate.

[0097] Subsequent write errors that occur can be incrementally reported back to the host software without further memory controller involvement. Such general interaction is generally designated by reference numeral 1001 in FIG. 10A. Generally speaking, the host first initiates a write command, per numeral 1003, and the memory controller detects the occurrence of a write error, for example, a timeout after reaching a programming limit (e.g., maximum write time out or number of program-verify cycles); this write error is then reported back to the host (1005). The error condition is then recorded by the host (1007) so as to avoid further writes to the area in error, with writes directed by the host to another location. The host marks the unit (EU or other unit as bad). In addition, any memory operation retry in this mode is advantageously controlled and performed solely by the host software, with the memory controller confirming a successful write to the host and also updating local metadata for correctly written data (and optionally, for the bad blocks as well). Note that, once again, the fact that the host updates translation tables so as to avoid future writes to the location avoids the need to

maintain L2P translation infrastructure at the memory controller (1011, 1013). When configured to execute in this host-owned defect management mode, the controller does not need to reserve any spare for defect area remapping and thus can advertise full device capacity to the host, such that the memory controller is not required to implement even a minimal L2P capability for this mode. That is to say, the memory controller need not reserve spare capacity since all address assignment and defect tracking is owned by the host, i.e., such that there is no form or remapping performed by the memory controller and all physical address assignment to new writes is exclusively owned by the host.

[0098] When the host owns defect management, the memory controller is typically inhibited from attempting a unit erase operation due to the inability to handle erase errors that could occur. To overcome this issue, for host-owned defect management, the host explicitly initiates all erase operations, with a status code returned to the host software that specifies whether the erase operation completed successfully or failed. Recording the unit that experienced an erase error so as to avoid further writes to a "bad" EU is then once again the responsibility of the host.

[0099] Note that it was referenced above that the memory controller, the host, or both can track lists of free erase units. In connection with defect management, a reserve of available space is kept on hand in order to provide a pool for immediate reassignment of writes when a defect is detected. Such a list can be implemented in a manner that supports midstream insertion of an EU after erasure. In one embodiment for example, when spare capacity is low, the host can initiate an operation that identifies candidate blocks for erasure in a given memory device or drive. A memory controller sharing responsibility for defect management might already be charged with managing a list of erased units that it holds for spare capacity. A cooperative host can identify pages with low page utilization, move any active data, instruct the erasure of associated EUs, and instruct the memory controller to add newly erased units to its list of free pages. If desired, this list can be organized according to a desired priority criteria, e.g., as a FIFO list, according to wear, or using another priority scheme. In one embodiment, the list can be organized as a B-tree, or alternatively, as a hierarchical doubly linked list.

ii. <u>Shared Defect Management.</u>

[0100] While providing host software with complete control over media defects, host-owned defect management can exert an extra burden of defect list management on the host software; this may

be undesirable for some applications. Shared responsibility for defect management using principles of this disclosure can instead be used. That is, the memory controller can be used to monitor and track errors and other parameters, leaving error detection and/or defect list management to the host.

[0101] In one embodiment, write errors as they occur are reported back to the host software. Host software can then instruct the memory controller to mark the particular physical block of memory cells associated with the error as bad. In a NVMe context, such a method can be performed using a WriteUncorrectable command that, when issued, instructs the controller to add a specified logical block address range to the bad block list, maintained in storage of the memory controller.

[0102] Units marked unusable by a chip manufacturer can be identified during first time device initialization and saved in the defect list in the controller's internal memory. The memory controller preferably further guarantees persistence of the defect list across power-cycles, so that defect information is always up-to-date. Host software is permitted to inquire at any time whether a given address range or block has been previously marked as bad, or to query the memory controller for the complete defect list. In a NVMe embodiment, such a query can once again be implemented with a vendor-specific NVMe administrative command.

[0103] As seen in FIG. 10B, as with host-owned defect management, a cooperative memory controller can once again provide for an write operation to be initiated by host software (1023), with an alert code being returned in the event that the write operation failed (1025). Host software may then request the memory controller to identify the bad unit in the memory controller's defect list and the host then attempts a rewrite (1027). Once again, once the retried write is successful, both the host and the memory controller update translation tables and metadata respectively (1029, 1031, 1033). As before, because the host is informed of a failed write and locally maintains precise physical address identifiers for each page in flash memory, the memory controller is substantially freed from having to provide for an L2P mechanism. When configured for cooperative defect management, a memory controller once again does not need to reserve any spare capacity for defect area remapping, and can instead advertise full device capacity to the host software.

[0104] In some embodiments, host software can be permitted to classify an EU or a smaller address range as unreliable due to read error detected using the additional data integrity features mentioned

earlier or other host-specific features. Host software can therefore be provided with an option to request that this range to be added to the defect list using the method described above. This option provides for greater flexibility in error detection and correction at a system level.

iii. Memory Controller-Owned (Transparent) Defect Management.

[0105] In situations where the memory controller manages defects, when write or erase errors occur, they are not reported to the host software unless the controller is out of spare capacity (e.g., to remap data). That is, if sufficient spare capacity exists, defective areas are automatically remapped by the memory controller, transparent to host, and added to the defect list maintained by the memory controller. This configuration is represented by FIG. 10C, and is generally denoted using reference numeral 1041.

[0106] As before, a defect is detected in the event of a failed write (1041, 1043). The memory controller detects this error and updates local metadata (1045). However, in this mode, the memory controller does not immediately notify the host. Rather, the locally maintained metadata is updated in a manner that flags a remapping situation, and the memory controller reserves some spare capacity for redirected writes. The memory controller also updates a locally-maintained bad block list (1047); note that as indicated in FIG. 10C, each of these pieces of information can be once again maintained on a per-EU or per-logical unit (LU) basis. When the data is correctly written to substitute EU(s), the memory controller returns a code indicating completion of a successful write (1049). Per function blocks 1049 and 1051, ensuing reads to the remapped data are detected by the memory controller, which detects correspondence of a read address with the physically-defective "bad" block, transparently obtains the remapped address from the metadata associated with the bad block, and services read requests directly from the remapped space. Note that when configured in this mode, the memory controller continues to store information tracking usage for each subdivision of memory, and continues to serve this information to the host. The memory controller can also provide query capability as referenced above. As use of the memory progresses over time, and as wear leveling and garbage collection occur, the host will be informed of new available memory that does not include marked "bad" blocks; as a consequence, physical address assignment for new writes ultimately moves remapped data to a different memory location, thereby avoiding the "bad" blocks. In turn, this results in the host once again having a direct physical address for the data in question (1053, 1055). In other words, irrespective of

the fact that the memory controller owns defect tracking in this scheme, the extent of L2P remapping is advantageously both minimal and self-limiting. The host manages address translation for data reads, but because defect space in practice will be limited, the memory controller should not require large search trees or other complex mechanisms to perform address translation.

[0107] When spare area reserved for defect remapping is exhausted, the controller issues an alert to the host or otherwise flags an error condition to draw the attention of the host. In a NVMe-compatible embodiment, such an alert can be raised with an asynchronous event notification. The host software can then take the appropriate actions such as switching to another supported defect management mode, initiating garbage collection or data relocation, or using some other process.

[0108] Note that if spare capacity is exhausted, this even can threaten and potentially halt normal device operations. To overcome this potential issue, a cooperative host can be permitted to poll the spare capacity for defects in any flash device or drive, or set a state that requests asynchronous notification if spare capacity for defects falls below a host-defined threshold. Methods for host software to detect this management condition are described above.

[0109] Also, as alluded-to above, the host can also take management action to increase the amount of defect spare when a defect-spare management condition is detected. That is, host software may pro-actively address potential spare capacity run-out by improving spare capacity by, for example, initiating space reclamation, garbage collection or data relocation operations or using another process that can improve usable device capacity.

b. Garbage-Collection Management.

[0110] As discussed earlier, a page in NAND flash memory cannot be rewritten until an entire EU including that page has been erased; this is the P/E asymmetry referred to earlier. Some flash memory controllers operate by detecting an attempt to write to a previously-written page (i.e., a page that has already been programmed but not yet erased) and responsively implementing a copy-on-write operation. With such an operation, the memory controller remaps the page or other logical quantum of data by remapping it to erased location. Over time, the number of unused pages grows without any free space being reclaimed due to the P/E asymmetry. To reclaim released space, memory controllers are periodically forced to run a garbage collection task, where they locate and consolidate partially used

erase blocks to reclaim space. Conventional garbage collection practices can require substantial reserved buffer space, up to 20% of available storage capacity, to compensate for non-reclaimable released space, and can compete with write coincidentally initiated by the host. Performance penalties associated with these conventional practices management can be especially pronounced when page utilization exceeds 50-60%.

[0111] Host-owned and shared garbage collection techniques permit significant reduction of these performance penalties and associated overhead. As before, in one embodiment, a cooperative memory controller and/or cooperative host can be made configurable so as to support host-owned, memory controller-owned, or shared garbage collection management responsibilities. Host-owned and shared garbage collection techniques are discussed with reference to FIG. 11A, while memory controller-owned techniques are discussed in reference to FIG. 11B.

i. Host-Owned And Shared Garbage Collection.

[0112] In a host-owned garbage collection process, generally designated 1101 in FIG. 11A, the host can assume full control and responsibility for garbage collection, including released space accounting, candidate unit selection, and relocation of valid (active) data. The operation is initiated when a host process detects a threshold condition related to garbage collection, as referenced by numeral 1106. Unit erase operations and actions to reclaim free space are thereafter initiated by host software with an explicit erase command, for example, as described in connection with defect management above. The host is further expected to appreciate P/E asymmetry, to track released pages for each unit, and to apply any garbage collection candidate identification logic to ensure the desired amount of free units or available capacity. All of these functions can be facilitated via the information stored and made available by the memory controller presented by this disclosure, and the queries that can be run to such a memory controller. That is, the memory controller can provide page utilization information to the host, which can determine scheduling, pull data, issue erase commands and rewrite data as necessary. Based on this information, the host schedules garbage collection and selects both source locations and destination locations for any data that is to be relocated (1115). As indicated by dashed-line block 1117, if supported by the particular implementation, the host can delegate a copy operation, for example, as was discussed earlier. Such an implementation has the advantage that a data relocation operation does not require moving data back and forth to and from the host, and thus, does not encumber a data

communication path between the host and the controller. Alternatively, if it is desired to copy the data to the host (e.g., to move data to another SSD), the copy/relocation operation can be directly performed by the host. When data is properly written as part of such an operation, the memory controller returns with a confirmation to the host and successfully updates its metadata as appropriate (1119). As denoted by numeral 1121 and as previously discussed, the memory controller can be configured as an option to automatically release old pages that were the source of relocated data, and to automatically erase any EU for which the last page has been released. Alternatively, if this function is not automatically performed, the host then issues an explicit erase command 1123, and the memory controller then returns a code indicating successful erase. Per numerals 1125 and 1127, as the host schedules the operations and is informed of associated physical addresses, the host can once again directly update its own translation tables, without need for a complex translation mechanism at the memory controller.

[0113] Dashed-lines in FIG. 11A are also used to discuss shared responsibility for garbage collection management. In addition to host detection of a threshold condition relating to garbage collection, this can also be performed at the memory controller (1105), as a default function or programmatically configured as a result of an asynchronous command from the host. The first task associated with garbage collection scheme is the identification of released, but not erased, memory space present. Accordingly, the host is advantageously provided with the ability to query each flash memory controller present in a memory system as to whether such space is present. Again, this information can be returned in the form of a map or in a manner specific to ranges of addressable memory (1111). In addition, a cooperative memory controller can also be configured to return general metrics such as the extent of page utilization or the number of released pages for a memory device in general. This latter information could be used by a host, for example, in selecting a specific one of multiple SSDs as a garbage collection operand. Further, these techniques can also be performed unsolicitedly, for example, they can be triggered automatically by a different type of command or in connection with an asynchronous process; otherwise stated, a different type of command can trigger an alert of the sending of data to the host as a result of a metadata update that causes specific metadata to satisfy a threshold.

[0114] In one embodiment, the ability to query a flash memory controller is implemented with vendor-specific NVMe command specifying a particular LUN, or requesting a return of information for an entire flash memory device or drive. Upon successful completion of the command, the memory

controller returns information to the host representing the amount of released space that may be reclaimed and the amount of valid data that needs to be relocated for reclamation to happen. In yet another embodiment, the memory controller can asynchronously notify the host when released non-reclaimed space reaches a predetermined threshold. For example, a host can issue vendor-specific NVMe asynchronous event request to a particular memory controller that specifies a threshold reclaimable space needed to trigger alert of the host. The flash memory controller then responds asynchronously when it detects the specified condition, notifying host software of the condition and delivering condition specific information.

[0115] As mentioned, as an option, the host can query (1109) the memory controller for a suggestion of suitable garbage collection candidates. Logic on board the memory controller receives this requires, processes stored metadata (1111), and responds as appropriate (1113). For example, depending on implementation, a response can identify a predetermined number of EUs in order of page (under) utilization. Alternatively, the response could rank all EUs in the flash memory being managed in order of suitability for garbage collection. As a further option, if the host command specified an amount of space to free up, the memory controller could return an identification of EUs which, when consolidated, would provide the specified amount of free space. Other options are also possible. As with other functions described above, the memory controller services this query by processing on locally stored information (e.g., metadata, 1111).

iii. Memory Controller-Owned Garbage Collection.

[0116] Memory controller-owned garbage collection typically requests some type of L2P tracking local to the memory controller and it therefore entails different considerations than the embodiments discussed above. Nevertheless, a skilled designer could implement such a scheme in a manner consistent with a memory controller that tracks and serves information (e.g., metadata) for host access. FIG. 11B depicts one possible method (1141) of managing this process. More specifically, a memory controller detecting a need for garbage collection (1143) performs local space consolidation, data relocation and erase operations (1145). In association with these actions, the memory controller updates its local metadata (1147) and transmits a consequent logical to physical mapping back to the host, together with metadata that enables the host to perform a reverse lookup and consequently build

its own address translation table for files (1149, 1151, 1153). Thereafter, the host performs memory operations (such as read commands) based on the physical address of data.

[0117] Other schemes consistent with memory controller-managed garbage collection are also possible; a designer, however, wishing to minimize control overhead associated with managing flash memory, not to mention avoiding L2P overhead on a flash memory device, will typically elect the clear advantages presented by the host-owned and shared configurations discussed above.

c. Wear leveling Management.

[0118] As noted earlier, flash memory typically can sustain a limited number of P/E cycles, e.g., tens to hundreds of thousands. As the number of times the unit was erased approaches the manufacturer defined limit, the unit wears out and finally becomes unusable such that data cannot be retained.

[0119] The cooperative management techniques of this disclosure permit host software to take full or partial responsibility of wear leveling. As before, a cooperative memory controller and/or cooperative host can support configuration of a host-owned, memory controller-owned, or shared wear leveling scheme. Also as discussed before, the pertinent scheme can be selected in dependence on a mode register or system configuration process or at system design time.

i. Host-Owned And Shared Wear leveling.

[0120] Host-owned and shared wear leveling schemes are generally represented by FIG. 12A. Generally speaking, whichever scheme is selected, the host can generally improve wear leveling by performing wear-aware writes, as generally denoted by reference numeral 1203. As mentioned, in some cases, wear leveling might still sometimes have to be performed, for example, due to differences in garbage collection differently affecting "hot" and "cold" data. Generally speaking, the need for wear leveling can be detected either by the memory controller or the host, as represented by numerals 1205 and 1206; a number of measures can be used for this purpose, such as a detection of a difference in EU wear for a given memory range exceeding a predetermined threshold. Alternatively, a statistical measure can also be used (e.g., statistical variance) to inherently apply a weighted measure across the entire memory range being managed by a memory controller.

[0121] For host-owned wear leveling, once again, the host can poll a memory controller to request a bitmap from the memory controller or can maintain its own tracked parameters representing uneven wear. Note that for many embodiments, tracked wear parameters will be maintained at a location consistent with where processes are run to detect wear leveling, but it is also possible for example to host-monitoring of wear based on memory controller-maintained metadata. Irrespective of the source of the wear data, for host-owned wear leveling, the host identifies candidate address ranges that are to be redistributed or recycled. For example, if wear-aware writes fail to evenly distribute wear, it could be that cold data is being recycled differently than hot data; note that this situation is slightly different than the issue of low page utilization mentioned earlier, i.e., in this case, an EU might be highly utilized for reads, but might have pages that are released only infrequently. If this data is periodically moved, then this underwritten memory can be used to dilute wear attributed to frequently written data. The host can also continuously track such information and, irrespective of the mechanism used to acquire data, the host monitors data to detect occurrence of one or more threshold conditions relative to wear (1206). With host-owned wear leveling, host software assumes full control and responsibility for scheduling and initiating wear leveling, and for selecting source data addresses and destination addresses associated with copy operations (1215). Once again, the host can perform copy operations itself, pulling data back to the host (e.g., for transferal to another drive), or it can issue a delegated copy command as discussed above (1217). Successful operations are confirmed by the memory controller and also form the basis of updates to metadata (1219, 1211). Release of source pages for copy operations and unit erase operations are again initiated by the host software (1221), for example using an explicit erase command, as described above. When old space is erased and available to receive new data, the memory controller provides a confirmation code to the host (1223), which can then serve as a recipient address for another data copy/relocation operation. Note that many types of tracked data can be used to assist in wear leveling management, for example, tracked metadata representing data type, read frequency, write frequency, and other measures can be considered in determining whether data is "hot" or "cold," and thus whether data should be moved. Also note that with the mentioned confirmation codes, the host once again advantageously maintains a current list of physical addresses for all data stored in flash memory, updated as appropriate following transaction completion (1225, 1227). The use of these physical addresses in association with file structure by the host will be discussed further below in connection with description of an exemplary storage server.

[0122] FIG. 12A also represents a scheme for shared responsibility over wear leveling. In such a scheme, the host can be permitted to query the memory controller as to what units are most suitable for allocation based on wear considerations (1207, 1209, 1213). A synchronous command can once again be used (1209) to cause the memory controller to run a query based on stored metadata (1211) and to return a result to the host (1213); as indicated in FIG. 12A, this result can be expressed in the form of a list that identifies a "suggestion" of candidate address ranges that are to be redistributed or recycled. Per numeral 1213, a list can be provided to the host based on time since last write, low wear, and so forth. The host can then explicitly direct new writes to specific EUs or other physical units based on this information. In addition, the memory controller can also be programmed using an asynchronous command to alert the host when a predetermined wear threshold or set of thresholds is achieved (1205). Note that, as discussed elsewhere herein, some limited L2P mapping can still be performed by the memory device, but with the use of direct addressing, it is expected that translation issues can be greatly minimized, thereby greatly reducing the possibility of memory controller task competition with host requests.

ii. Controller-Owned Wear Leveling.

[0123] A storage system can also implement a scheme where wear leveling is managed by the flash memory controller; the memory controller manages wear detection transparently to host by itself detecting the need to remap units to ensure uniform wear distribution. In one embodiment, the host can then schedule wear leveling (e.g., data relocation and unit erase), and in a second embodiment, the memory controller can transparently schedule and one or both of these functions. Such schemes still have context in a cooperative host-controller management scheme. For example, a memory controller transparently scheduling and performing data relocation can periodically report new L2P mappings to the host, which then implements direct physical addressing as indicated earlier. Further, a memory controller can still make metadata available to the host for most other management functions, for example, cold data relocation via both asynchronous and synchronous mechanisms described above. It should be noted though that if a memory controller transparently manages these functions, the functions can potentially collide with other host requests unless measures are taken to avoid collision; for this reason, in many embodiments, scheduling of wear leveling and other operations involving data relocation, unit erase and so forth can advantageously be reserved to the host.

[0124] Memory controller-owned wear leveling typically utilizes some type of L2P assignment and tracking local to the memory controller and it therefore entails different considerations than the embodiments presented earlier. That is, such an embodiment features additional overhead associated with an L2P translation mechanism in the memory controller; even in a situation where such reassignment is limited (e.g., entire EUs are simply remapped to spare EUs, such that for example, internal page ordering is preserved), this can potentially compete with host operations and create unpredictable latencies. Nevertheless, a skilled designer could implement such a scheme in a manner consistent with a memory controller that tracks and serves information (e.g., metadata) for host access. FIG. 12B depicts one possible method (1241) of managing this process. More specifically, a memory controller detecting disparate wear (1243) performs local space consolidation, data relocation and erase operations (1245); as mentioned, these can be advantageously scheduled by the host, with the memory controller then commanded to oversee these operations at the scheduled time. In association with these actions, the memory controller updates its local metadata (1247) and transmits a consequent logical to physical mapping back to the host, together with metadata that enables the host to perform a reverse lookup and consequently build its own address translation table for files (1249, 1251, 1253). Thereafter, the host performs memory operations (such as read commands) based on the physical address of data.

[0125] Other schemes consistent with memory controller-managed wear leveling are also possible; a designer, however, wishing to minimize control overhead associated with managing flash memory, not to mention avoiding L2P overhead on a flash memory device, will typically elect the clear advantages presented by the host-owned and shared configurations discussed above.

C. Shared Responsibility For Physical Address Management.

[0126] In some implementations, it might be desired for the memory controller to assign physical addresses for new data writes and then report these addresses back to the host; this is to say, in such an implementation, the host might not have architectural features required to assign physical addresses on its end so as to account for in-flight IO operations and to ensure maximum attainable device bandwidth utilization. Under such circumstances, this task can be beneficially delegated to the memory controller. To minimize latency and otherwise benefit from direct addressing in such circumstance, it is advantageous to provide physical addressing assigned by the memory controller back to the host, in

order that the host can associate physical addressing with file-level and other logical addressing structures and appropriately update its L2P translation tables. This is to say, as part of the write operation, the memory controller extracts logical address, assigns physical address, updates its tracked metadata to track these addresses, and then reports this addressing information back to the host; the host, based on the logical address it specified and the physical address(es) returned by the memory controller, updates its tracked information so as to maintain L2P mapping. Such a process can also be used by the memory controller to provide address information back to the host in associated with a delegated copy or relocation operation, and with forms of controller-managed wear leveling and garbage collection (if implemented).

[0127] In another embodiment, the memory controller further cooperates with the host by keeping track of backward address translations, known as back references. This enables the host to derive logical addresses of the data block given their physical addresses as returned by the controller in response to queries such as used for managing garbage collection (GC) and "cold" data relocation (e.g., WL) candidates (e.g., such that the host is able to map new physical addressing to the pertinent logical files and addresses). Where the host software requires logical addresses to perform relocation while at the same time missing back references, it can request the controller to return, for each block range suggested in the query response, the metadata provided by host software when the block was written. That metadata, in particular, can contain a physical-to-logical mapping allowing for the physical block address as returned by the controller to be translated to the logical block address used by the host storage software. Once again, this permits the host to thereafter use physical addressing, avoiding costly extraneous address translation in the memory controller for IO operations. In one embodiment, this scheme is supported by native memory controller hardware and, once again, can be selected at initial power-up by programming pertinent registers on board the memory controller, such as the registers seen in FIG. 3B.

[0128] In a NVMe-compatible embodiment, both types of addressing-related exchanges between the host and the memory controller can be implemented using a NVMe logical block metadata exchange mechanism. To inform the host of memory controller assigned physical addressing (i.e., L2P correspondence), the memory controller specifies as part of the logical block metadata vector the physical addresses assigned to each logical block associated with the host write command. To store the host-provided logical addresses for subsequent retrieval and use (e.g., for GC or WL processes), logical

addresses are sent from the host to the memory controller as logical block metadata accompanying a write command, and then are stored as part of the per-subdivision (e.g. per EU) metadata storage maintained by the memory controller for all managed memory.

[0129] Note that by using the NVMe logical block metadata structure, both types of exchanges retain the standard IO addressing interface and are compatible with the NVMe structure while at the same time providing the host with the ability to specify appropriate (i.e., logical or physical) addresses as needed.

[0130] Such a process is generally designated by reference numeral 1301 in FIG. 13. In such a process, the host initiates a write command to the memory controller, as generally indicated by function block 1303. This write command specifies a logical address. The memory controller, upon receipt of this command, queries available space and assigns physical addressing as appropriate to the new write data. It also updates its local metadata storage with the pertinent information, indexed by physical unit (1305, 1307). This is to say, the memory controller maintains this information per erase unit (or other unit) information for the entire managed memory space. In the case where physical units such as EUs or physical pages are tracked, the memory controller stores each corresponding logical address as metadata for the pertinent physical unit. As mentioned, in another embodiment, the memory controller can also track this information in a manner indexed by assigned logical unit (e.g. block or page). The assigned physical address is then transmitted back to the host (e.g., as a confirmation code), per numeral 1309, and the host builds a L2P translation table by processing this data. Finally, the host updates its translation tables as appropriate (1313, 1315); as before, because the host maintains L2P information for all files stored in flash memory, it uses direct addressing (i.e., addressing of physical units) notwithstanding the presence of the memory controller (1317). That is, the memory controller is advantageously not encumbered by time consuming translation or address lookup mechanisms. Thus, for delegated data assignment (or the other mentioned processes), the memory controller can support residence space assignment, while at the same time providing the benefits of a physical access to specific addresses dictated by the host.

[0131] FIG. 13 also represents a process usable by the memory controller in connection with garbage collection (GC), wear leveling (WL) and certain other processes. That is, in a scheme where the host manages physical addressing (and specifies destination addresses for GC and WL processes) but

where the memory controller is charged with suggesting garbage collection and/or wear leveling candidates, the memory controller advantageously informs the host of logical data correspondence associated with those candidates at the time of making suggestions; in this manner, the host missing back references can obtain logical addresses and peruse its L2P mapping tables as part of the GC or WL process. This functionality is various represented by dashed line blocks 1310 and 1311 in FIG. 13, and by solid-line blocks 1313, 1315 and 1317.

[0132] Those skilled in the art will recognize that other combinations of sole/shared management schemes other than those shown above are also possible.

[0133] With basic implementation of a cooperative flash memory controller and/or cooperative host thus described, this disclosure will now proceed to a description of an exemplary storage system configuration, with reference to FIGS. 14-19.

III. Architecture Of An Exemplary Storage System Or Subsystem.

[0134] FIGS. 14-19 are used to illustrate functions introduced above in an exemplary storage system. It should be assumed that a storage system or subsystem includes plural storage modules or drives, at least one of which is based on NAND flash memory solid-state devices (i.e., integrated circuits). A "drive" as used in this section denotes any grouping of memory that has a common memory controller, meaning a common interface that interprets commands and that directs memory access operations and associated physical addressing to dies or other forms of memory as appropriate. A memory controller can be in the form of an integrated circuit, and a drive can be in the form of a memory module, board, or other type of subsystem. Each drive can have one or more memory integrated circuits, disks or other storage media that are selected and accessed by the respective memory controller. Other configurations are also possible. The storage system can optionally be a hybrid or heterogeneous storage system; that is, the storage system can have plural drives, at least one of these is NAND flash-based solid-state drive (SSD), and one or more of these can optionally be a magnetic drive, such as a hard disk drive (HDD). In the context of a network-based storage appliance, the storage system or subsystem can be configured as a storage server having plural drives, each of which is a separate memory module.

[0135] As has been introduced previously, each SSD (and optionally, each other drive) in such an embodiment supports functional calls from the storage system to identify drive geometry. In one embodiment, drives can therefore be made interchangeable, with storage system dynamically adjusting its practices for any particular drive and the system as a whole according to whether the particular drive is a NAND-flash based and according to drive geometry. This architecture provides for host-controller cooperation in managing NAND flash memory-based storage devices in direct-attached and/or network-attached storage environments based on existing host storage software features, with minimal modifications of the latter. To that end, host storage software is modified and configured (relative to conventional designs) to perform management tasks as required by unique NAND flash memory requirements, with NAND-based storage device controller providing supportive information and features needed. By redefining host and controller responsibilities, with management task control regained by the host storage software, and each controller providing locally maintained metadata for cooperative management task execution, host-controller management feature duplication and associated disadvantages are avoided.

[0136] FIG. 14 illustrates an exemplary system 1401. In particular, a storage server 1403 receives requests for data or to store data, and optionally, requests for processing that data from one or more clients 1405. These clients each have respective applications 1406 which generate the respective data-related needs. The clients can each be desktop or laptop computers, smart phones, pads or other devices, or other types of digital systems. Each client 1405 sends requests to the storage server 1401 and receives responses via a wide area network, e.g., a local area network (LAN) or a wide area network (WAN) such as the Internet. The storage server, in turn, manages plural drives, each of which can be an SSD (e.g., a flash based drive), a HDD or another type of drive. As seen in FIG. 14, each drive also has its own dedicated memory controller, either bundled together as a single assembly 1409, or as separate devices; for example, assembly 1409 is seen to include a SSD controller 1411 and a SSD 1413. To simplify FIG. 14, a dashed-line box is shown only for assembly 1409. The SSD controller 1411 can be a NAND flash memory controller and the SSD 1413 can have one or more NAND flash memory devices. FIG. 14 also shows optional presence of other dedicate assemblies, 1415, 1417 and 1419, in this case seen to include a second SSD, an HDD and another unspecified type of memory, respectively. The collection of memory is seamlessly managed as a storage aggregate 1421, with the storage server managing scheduling for all drives so as to avoid collisions with storage-server-directed reads and writes, as described earlier. In this regard, the storage aggregate and the host are bundled together to form the

storage server, but this is not required for all embodiments. The storage server has a storage operating system that implements a file system to organize the stored data as a hierarchical structure of logical data containers (e.g., volumes, logical units, directories and/or files) on electronic and/or magnetic storage media. It will be understood by those skilled in the art that this description may apply to any type of special-purpose computer (e.g., file server or storage serving appliance) or general-purpose computer embodied as, or having, a storage server or portion thereof. Moreover, the teachings of this description can be adapted to a variety of storage server architectures including, but not limited to, a network-attached storage (NAS), storage area network (SAN), or a disk assembly directly-attached to a client or host computer. The term "storage server" should therefore be taken broadly to include such arrangements.

[0137] The storage server 1403 includes a host processor 1423, which uses various buffers 1425 under the auspices of the host software 1427 to schedule and manage memory operations, including both memory transactions and memory maintenance operations. The host maintains a local L2P translation table so as to access files which can be fragmented in multiple memory locations and potentially multiple drives. The storage server also optionally employs policy-based space allocation, supporting data- and media-characteristic-aware data placement across the entire storage aggregate 1421. The storage server communicates with the clients 1405 via a network adapter 1431 and with the storage aggregate 1421 via a storage adapter 1433 (although it is also possible to have all communications performed via network adapter or otherwise remotely over a network such as a WAN or LAN).

[0138] In some implementations, the client(s) can issue packets including file-based access protocols such as the Common Internet File System (CIFS) protocol or Network File System (NFS) protocol over TCP/IP when accessing information in the form of files. In other implementations, the client(s) can issue packets including block-based access protocols such as the Small Computer Systems Interface (SCSI) protocol encapsulated over TCP (iSCSI) and SCSI encapsulated over FC (FCP) when accessing information in the form of LUNs or blocks. Also in some implementations, the storage adapter includes input/output (IO) interface circuitry that couples to SSDs (1409 and 1415) and HDD (1417) over an IO interconnect arrangement, such as a conventional high-performance Fibre Channel serial link topology.

[0139] The storage server manages flash memory using a log-structured copy-on-write file system, with physical block addressing and wear-aware writes of new data (as has been described earlier, e.g., in a manner that obviates need for an extensive FTL layer on the side of the SSD and that better distributes wear). The storage server is configured to access NAND flash memory, such as in drives 1409 and 1415. using physical block addressing, and to account for NAND flash memory-based device wear-out, and to execute garbage collection with associated valid data migration to manage NAND flash P/E asymmetry and other NAND flash management functions, all as referenced earlier. This is to say, each flash-based drive such as assemblies 1409 and 1415 advantageously have a cooperative memory controller that implements the principles described earlier. Depending on system architecture, host software manages interaction with each such controller. This architecture provides for host-memory-controller cooperation in managing NAND flash memory-based storage devices in direct-attached and/or network-attached storage environments. For example, each flash-based drive has a respective controller that serves information to the host regarding each subdivision of the associated memory. The storage server (in this case, the host) can manage wear distribution across multiple drives to help lessen wear to any one area of memory; for example, in the context of the wear-aware writes, the storage server can collect wear metrics for all flash memory managed as well as for other types of nonvolatile memory, if present. The storage server can combine bitmaps from multiple drives and can allocate if desired new writes to a single drive only if needed to better distribute wear. As mentioned, the host can also electively direct writes of data based on policy, for example, in a manner that avoids fragmentation or that groups certain types of data together based on read-write characteristics. In one embodiment, each SSD memory controller (such as controller 1411) is rooted in a configurable architecture that depends on host capabilities, policy, or other considerations. For example, in such an architecture, if the host processor does not support host-assignment of physical addresses for new writes, the host configures each SSD memory controller to perform this function and to report back assigned physical addresses as described earlier.

[0140] FIG. 15 provides a flow diagram 1501 used to discuss some considerations used by the host processor of FIG. 14 in performing wear-aware writes and in managing addressing. First, the host processor receives a trigger to write new data to memory or the host otherwise determines that a need for management action is warranted, for example, to relocate data (e.g., hot or cold data) and/or to reclaim space (1503). The host processor proceeds to determine target addresses (1505) that will serve as the write destination for the new/moved data, and it does this based on a list of available free space

(1507). This list can if desired represent multiple drives as well as mixed memory types. For locations in flash, the data advantageously also is prioritized in order of least wear, i.e., such that the first location represents the most virgin space, and so forth. If the host processor/operating system does not have such a list, it proceeds to request one or otherwise initiates garbage collection and erase operations as necessary to generate free space; system designers if desired can implement a policy that ensures the list references at least a first threshold level of available free space, and that calls garbage collection and explicit erase to raise available free space to a second level (e.g., employing hysteresis) any time this list represents free space less than the first threshold. With the target physical addresses thereby identified, the storage server then proceeds to schedule memory operations (1509). In a multiple drive system, this can be achieved using one or more transaction queues (one for each control path or bus) which interleave commands as appropriate to multiple drives. Once the host processor receives confirmations of completion of the various commands (1511), the host processor then updates its L2P translation tables and file references (1513). Numeral 1515 references an exemplary file naming structure, i.e., where each file is an object having a name, and where the file is stored across multiple, not-necessarily-contiguous blocks. The file has one or more indirect references (for example, to memory locations that will store lists of physical addresses for blocks having ordered portions of the file), and it has physical address and logical position within the file for each such file reference. When a transaction is completed, for example, changing physical address in association with a data copy operation, the physical address of data move from a first block to a second is changed (updated), so that the host software can thereafter use direct physical addressing to identify locations of the subject-data. Note that FIG. 15 shows two options in dashed lines, including a delegated copy operation 1517, and a copy-via-host operation 1519. That is, in situations where a data copy operation is dictated, the delegated copy operation can be employed to perform data transfers within a single SSD as described earlier, which does not encumber the host processor-memory controller connection with the requirement of data transfer. As indicated by function block 1519, transfers in between different drives will typically be processed by first transferring data back to the host processor, and then onto the second drive.

[0141] FIG. 16 shows a block diagram for functions of host software 1601 in managing SSDs. Generally speaking, the host software 1601 first identifies a condition associated with a SSD, for example, a new data access request from a client (1603) or an alert condition (or asynchronous return from memory), represented by numeral 1605. The host software then invokes a storage manager 1607

to determine actions that need to be taken with respect to flash memory. The storage manager, in turn, sends actions to a scheduler 1609. Actions can include requests that are to be immediately sent to memory as well as conditional requests, for example, for background processing. As an illustrative example of the latter operation, it might be desired to perform space reclamation and release of additional space as a background operation (for example, to provide buffering of available space not critically needed by the host); such a request can be queued by the scheduler and invoked at a time when background traffic (e.g., frequency of other IO operations) are below a threshold level of traffic. As operations are scheduled, constituent commands associated with those operations are sent to a transaction queue 1611 for pipelined issuance to the pertinent SSD(s). In this regard, the storage manager 1607 relies on interaction with a host-maintained L2P address translation table 1613 to select pertinent "direct" physical addresses associated with each command. Note that the scheduler can interleave commands for one SSD with commands for a second SSD (or an HDD or other drive type). As another illustrative example, the storage server might be called upon to retrieve a large media file (e.g., for a streaming application) from one drive while freeing space in a second drive. The transaction queue can interleave periodic requests to the first drive for the read data (e.g., to maintain sufficient streaming buffer occupancy) with periodic commands to the second drive to relocate data and erase EUs of that drive until the desired amount of space has been released. Additional detail will be presented below relating to performance of these types of operations. As depicted by FIG. 16, the storage manager 1607 can include a number of components, including a segment cleaner module 1615, and a write allocator module 1617. The storage manager can interact with policy module 1619 and a policy repository 1621 in configuring and managing each of two modules. That is, the host software in this exemplary system configures the segment cleaner module 1615 to operate (e.g., to free additional space, perform garbage collection, weal leveling, defragmentation and relocation of existing data) within homogeneous storage media, between heterogeneous storage media, or both, in accordance with any specified policies. The write allocator module 1617 is used to allocate physical subdivisions that may then be written to, as previously described above. Note that, as was alluded to previously, the host software beneficially is provided with the ability to query each controller and/or drive of the storage aggregate to determine drive type, manufacturer, capabilities and other information pertinent to the management of memory and system configuration. This information is stored for use by the storage manager 1607 in accessing and managing the drives, as generally indicated by reference numeral 1623.

A. Reclamation In NAND-Flash Memory (SSDs).

[0142] Host-assisted space reclamation uses metadata from a flash controller to reclaim and consolidate space owing to P/E asymmetry in flash, using principles described earlier. If it were not for the P/E asymmetry, the storage server could equate physical memory no longer used for valid data with free space. Such is generally the case for HDDs and other forms of random access memory, to the extent managed in common by the storage server. However, for flash drives (or other memory types having P/E asymmetry), released space cannot typically be reused until (a) all independently-writable units that form an EU have been released, and (b) an explicit erase procedure has been performed, in a manner that returns the floating gates of each constituent memory cell back to a default state. FIG. 17 illustrates a method 1701 that can be used applied by the host software. Note that it can be beneficial to postpone or batch reclamation operations. In particular, it may be advantageous to schedule erase operations for a later point in time, scheduled in a manner appropriate to load and other system-level considerations; as an example, if free space shortage is not critical, it can be advantageous to batch erase operations to times of low network traffic or when there is low demand for a particular storage drive. The EU units and its pages are only marked as free once the erasure has successfully completed and confirmed by return of completion code by the particular memory controller.

[0143] FIG. 17 generally shows two methods, including a method of automatic erasure (if appropriate) each time a page is released, represented by solid lines only, and a method of batched space reclamation appropriate for address ranges, specified "candidate" EUs, or for a drive as a whole. The latter method will be described first. The method 1701 generally starts at step 1703, for which at least one EU is identified. For each EU identified, at 1704, the method proceeds to check each page of that EU to see if it is released; that is, as indicated by steps 1705, 1707 and 1711, the method tests whether the pertinent page is no longer referenced by any file system structure. If a page is still referenced, the method ends for that EU at step 1712, and the method then continues to the next EU (1703). If it is not referenced, at step 1713 the page is marked as released rather than free. If a page is determined to be free, the bitmap is checked (1715) to determine whether it is the last page of that EU; for EUs that are determined to be fully released, the method continues at 1717, with an explicit erase operation being called (1719) and the drive summary information being updated (1721); otherwise, the method ends for that EU at step 1716, and the method then proceeds to the next EU (1703). The method then proceeds to the next EU (1723). For example, if the method is called for a specific range or list of EUs, the method 1701 proceeds in this manner for each EU until it has processed the last EU.

Note that, as indicated by reference numeral 1709, this method can optionally be performed for a storage aggregate or for each drive on an independent basis.

[0144] As noted, FIG. 17 also shows steps associated with a page release operation (i.e., solid lines only). That is, in some implementations, it might be desired to have the storage server immediately schedule an erase operation each time "the last page" in any EU is released. Such a method begins at 1713, where the host processor identifies a page release – the host processor checks the pertinent bitmap to see whether the latest release has resulted in the EU being fully released (1715); if it is not fully released, the method ends. If the EU is fully released, the method continues at the right side of FIG. 17, where an explicit erase operation is automatically queued (1719); when this operation is confirmed, the bitmap for the drive or aggregate as pertinent is updated (1721), and the method ends.

B. Garbage Collection.

[0145] In one embodiment, garbage collection is performed to reclaim space of EUs where the amount of released space relative to unreleased space exceeds a predetermined threshold. Garbage collection is achieved by copying valid data to another tier or another location in the same tier to release source locations. The valid data can be combined with other data, for example, data from the same file (e.g., for purposes of defragmentation), or data belonging to another file. Once all valid data for an EU has been relocated, all portions of a partially used EU are thereby released, and the EU can be marked as fully released and erasable. The method just discussed can then (or at a later time associated with batch processing) be called to trigger the erase operation. Note that any desired garbage collection policy or trigger can be used, and that such a policy or trigger can be dynamically adjusted. For example, garbage collection can be performed using a batch process, in a manner analogous to the batched processing just discussed. Alternatively, garbage collection could be initiated (or a host alerted) any time released space to valid (unreleased, written space) exceeds a 1:1 ratio (e.g., less than 50% page utilization). Such a metric could be dynamically changed, e.g., this metric could be dynamically varied such that (depending on policy or other system considerations) garbage collection would be initiated when this ratio exceeds 1:5 (e.g., at least twenty percent of valid space in an EU has been released), or implemented by policy module (1619 from FIG. 16) as a policy applicable to supportive flash-memory controllers and/or drives. Many such examples are possible.

[0146] FIG. 18 illustrates a method followed by the host processor/software (e.g., by the segment cleaner of FIG. 16) in response to a garbage collection (GC) alert raised by a flash memory controller. The method starts at step 1803. While such an alert remains in effect (1805), the host software identifies a specific EU under consideration (e.g., from a prioritized list of EUs) and then loops from one EU to another (1808) through a range of EUs. When the list is exhausted, the method queries whether the GC condition alert still holds (1805) - as long as the condition that triggered the alert remains valid (e.g., an available space shortage, or EUs not satisfying a threshold level of utilization), the method retrieves a successor list and processes each EU in that list. Once the condition is cleared, the process stops at step 1807.

[0147] At step 1809, the host software queries the memory controller for candidate segments that are suitable for GC. Step 1811 will loop for each returned GC candidate. If storage manager (1607 from FIG. 16) does not employ back references, controller-provided backward address translation as described earlier may be applied at this step, to derive logical addresses of the returned candidate ranges identified by their physical flash addresses. At step 1813, host software screens whether each suggested candidate is an appropriate candidate for relocation; for example, due to policy or other considerations, it might be desired to not relocate certain types of data (e.g., a host policy is in effect that groups certain types of data together at one location based on read-write characteristics). If the candidate is not appropriate, the host software then proceeds to examine the next candidate in the list, per numeral 1815. If the candidate is appropriate, the host identifies pages needing relocation (1817), typically from information provided by the memory controller together with suggested candidates; this can optionally be provided in the form of a bitmask of valid data that needs relocation. The host software (e.g., the segment cleaner) at step 1819 then schedules the relocation operation (1821), identifying source and target physical addresses as appropriate. As referenced earlier, such an operation can be performed using a delegated copy process for relocation within a source SSD, or alternatively can be performed to relocate data to a different drive (and potentially a different memory type). As indicated by dashed-line numerals 1823 and 1825, such an operation typically calls for reading data to the host processor and attendant memory, and marking the data as pseudo-dirty (i.e., as modified even though the data is not in this example). The pseudo-dirty modification will trigger scheduling of a copy-on-write operation that then rewrites this data to available memory space. The process of FIG. 17 can then be called to fully erase the pertinent EU (and to add that EU to the pertinent list of available space). Note that, if desired, the destination of such a copy-on-write operation can be

selected according to policy. Finally, per numeral 1827, the next EU in the candidate list is processed. If the segment cleaner chooses not to relocate data immediately, it will save the candidate for later relocation.

C. Shared and Host-Owned Wear Leveling.

[0148] With shared and host-owned wear leveling, the storage server becomes aware of physical blocks (e.g., EUs) that represent "cold" (or optionally, "hot") data and, thus, that potentially signify uneven wear. In one implementation of a "shared" scheme, the host can periodically ask for a drive- or system-wide wear metric and an updated list of wear leveling candidates in advance of any wear leveling process to ensure quick allocation is possible when they are needed. The storage server then can then schedule processing of this list as a periodic maintenance operation, or as a background process. If and when the list starts being exhausted, a new list can be requested. While in a shared scheme the list request is sent to the memory controller, in a host-owned scheme, this query can be sent to another host process.

[0149] In one implementation, the write allocator of FIG. 16 interacts with the memory controller to replenish a pool of wear-ordered EUs when either the pool becomes small or when the memory controller (or host software) detects a wear leveling related condition that requires host's attention. The latter can occur for example, when due to recalculation of device wear distribution, the memory controller or the host detects the existence of other EUs that are now considered less worn out relative to mean wear; alternatively, this condition can also exist when one or more EUs are detected to have significantly-less-than-normal wear. The pool in this context can include EUs having "cold data" that otherwise represent low wear relative to the mean, and that are to be recycled, such that future writes will be directed to these "low wear" EUs instead of further contributing to wear variance in a particular SSD.

[0150] A wear leveling process from the host processor's perspective is seen in FIG. 19, and is generally designated by numeral 1901. The illustrated method is substantively similar to the method of FIG. 18, used for garbage collection.

[0151] The method starts at step 1903. While the need for wear leveling remains in effect (1905), the host software identifies specific EUs under consideration (e.g., from a prioritized list of EUs) and

then loops from one EU to another (1908) through the EUs specified in any provided list or range. When the list is exhausted, the method queries whether the WL condition alert still holds (1905) - as long as the condition that triggered the alert remains valid (e.g., an available space shortage, or EUs not satisfying a threshold wear metric utilization), the method retrieves a successor list and processes each EU in that list. Once the condition is cleared, the process ends (1907).

[0152] At step 1909, the host software queries the memory controller for candidate segments that are suitable for WL. Step 1911 will loop for each WL candidate. The host software screens whether each suggested candidate is an appropriate candidate for relocation (1913), for example, not redistributing space which according to policy might be used for areas reserved for important system parameters. If the candidate is not appropriate, the host software then proceeds to examine the next candidate in the list, per numeral 1915. If the candidate is appropriate, the flash memory controller identifies pages needing relocation (1917); this can optionally be permed by calling on the memory controller to provide the starting physical block number (e.g., page number) and bitmask of valid data that needs relocation. The host software (e.g., the segment cleaner) at step 1919 then schedules the relocation operation (1921), identifying source and target physical addresses as appropriate. As referenced earlier, such an operation can be performed using a delegated copy process for relocation within a source SSD, or alternatively can be performed to relocate data to a different drive (and potentially a different memory type). As indicated once again by optional process blocks 1923 and 1925, the data can be read back to the host operating system, and the data can be marked as dirty so as to trigger a copy-on-write operation that then rewrites this data to available memory space. The process of FIG. 17 can then once again be called to fully erase the pertinent (and to add that EU to the pertinent list of available space). The destination of such a copy-on-write operation can once again be selected according to policy. Finally, per numeral 1927, the next EU in the candidate WL list is processed.

IV. Multi-Modal Operation.

[0153] As should be apparent from the description above, a cooperative storage architecture facilitates performance of management and access functions across multiple drives, regardless of type of memory. Reserving some SSD (i.e., flash) functions to the host in this architecture helps ameliorate latency concerns and drive expense issues that have traditionally hindered use of SSD drives in direct-

attached and network storage applications. That is, the techniques and functions that contribute to this architecture can be embodied in memory devices themselves, controllers for memory devices or drives, the host, or in any combination of these elements. Note however that in practice, it might be desired to fabricate a memory, memory controller or host architecture (e.g. host software) in a manner that supports both implementations with other cooperative storage elements and implementations with legacy devices. In addition, it is possible to fabricate elements and systems that, depending on design or network management considerations, selectively use certain cooperative techniques but not others.

[0154] Thus, in one embodiment, a memory controller supports any one or more of the host-owned, cooperative, or memory-controller owned modes or features described above. A register (for example, part of the memory controller storage 111 from FIG. 1) can be programmed with a value, for example, by hardwiring a signal connection or setting a fuse, or by configuration at initial power-up. Providing a number of programmable modes in this manner permits a cooperative memory controller and/or cooperative host to be used both in the cooperative management of memory (e.g., where the host processes data provided by the controller and where the memory controller tracks subdivision-specific usage information) and also with legacy components that do not support these cooperative features. In addition, providing these capabilities also provides flexibility to a host or system designer in the selection of mode based on performance criteria, intended application, or other factors.

V. Conclusion.

[0155] The description presented above provides examples of a cooperative memory controller as well as a host that interacts with such a memory controller, a storage system architecture, and various memory systems. As indicated, these techniques can optionally be employed with any one or more of these components. Further, while various embodiments shown above have described structure and functions in the context of a host, memory controller and/or memory, it should be understood that various ones of these components can be combined with one another or packaged, mounted or sold together. For example, depending on implementation, a host processor can be copackaged with a memory controller or mounted to a common board. Conversely, it is possible to copackage a memory controller with memory, such as mounted them on a single die, multichip module or system on chip (SoC). In the system context, solid state drives (SSDs) and hard disk drives (HDDs) have been described as generally having their own, dedicated controllers, but it is possible to have multiple types of memory

managed by a single controller, i.e., with a drive or memory itself constituting a hybrid. Further to the system context, embodiments have been described having two tiers of memory, but it is possible to have a single tier of memory (e.g., all NAND-flash), or three or more tiers of memory. As an example, one contemplated implementation features a host with multiple tiers of fast-access memory (e.g., local SRAM and DRAM), one or more SSDs and one or more HDDs, configured as different tiers. Some embodiments may omit one or more of these in favor of other memory types, for example, the HDDs can be omitted, or a different type of storage media can be used.

[0156] To provide a few further examples, instead of using flash memory, the techniques applied above can be applied to other forms of both volatile and nonvolatile memory. For example, it is possible to apply some of the management and network configuration schemes discussed above to DRAM and other forms of volatile memory. In addition, some or all of the garbage collection, space reclamation, geometry reporting, wear leveling, defragmentation and other schemes discussed above can be applied to other forms of nonvolatile memory. Although known by a variety of names, these memories typically have bistable elements have properties where resistance change is a manifestation of some type of change in physical materials. For example, RRAM, magnetic random access memory (MRAM), phase change random access memory (PCRAM), nanowire RAM, and similar designs all operate based on similar state change principles where the current that flows through the material (or equivalently, voltage drop seen across the material) changes depending on state of the underlying materials. Therefore, in many of the detailed examples provided above, flash memory is used, but as mentioned, other designs are possible. Also, the examples discussed above were focused on NAND-flash memory, but it should be equally possible to apply the teachings herein to NOR-flash memory designs.

[0157] Also, the techniques discussed above, when implemented on a comprehensive basis, permit substantial reductions in memory controller functions which would be conventionally needed to manage an FTL. However, as should be apparent from the examples provided above, many of the cooperative techniques provided by this disclosure and supporting circuitry can be implemented in embodiments that utilize a traditional FTL. For example, it is possible to combine cooperative garbage collection and space reclamation processes with a conventional FTL. Advantageously, by providing ready-host visibility into EU-specific, tracked parameters (e.g., metadata) and making liberal use of cooperative capabilities, a given design can eliminate much of the overhead that would be traditionally needed to manage an FTL.

[0158] It should be noted that the subject matter disclosed herein can be expressed (or represented), as data and/or instructions embodied in various computer-readable storage media, meaning physical non-transitory media such as computer memory, storage disks and the like. In this context, the described subject matter can take the form of instructions or data for fabricating an integrated circuit (or integrated circuits) that, when built and operated, possesses the circuits and structures, and performs the techniques described herein. An output of a process for designing an integrated circuit, or a portion of an integrated circuit, comprising one or more of the circuits described herein may be such a computer-readable medium such as, for example, a magnetic tape or an optical or magnetic disk, or other non-volatile physical storage. The computer-readable medium may be encoded with data structures or other information describing circuitry that may be physically instantiated as an integrated circuit or portion of an integrated circuit. Although various formats may be used for such encoding, these data structures are commonly written in Caltech Intermediate Format (*CIF*), Calma *GDS* II Stream Format (*GDS*II), or Electronic Design Interchange Format (*EDIF*). Those of skill in the art of integrated circuit design can develop such data structures from schematic diagrams of the type detailed above and the corresponding descriptions and encode the data structures on computer readable medium. Those of skill in the art of integrated circuit fabrication can use such encoded data to fabricate integrated circuits comprising one or more of the circuits described herein. Some or all of the functions described above can also be embodied as instructions stored on machine-readable media, including software code or firmware that, when executed, cause a machine (e.g. a microprocessor or device having a microprocessor) to perform the described functions. Generally speaking, any such instructions can be alternatively implemented as hardware logic, or a combination of hardware and software logic, depending on implementation.

[0159] The foregoing description and in the accompanying drawings, specific terminology and drawing symbols have been set forth to provide a thorough understanding of the disclosed embodiments. In some instances, the terminology and symbols may imply specific details that are not required to practice those embodiments. The terms “exemplary” and "embodiment" are used to express an example, not a preference or requirement.

[0160] Various modifications and changes may be made to the embodiments presented herein without departing from the broader spirit and scope of the disclosure. For example, features or aspects of any of the embodiments may be applied, at least where practicable, in combination with any other of

the embodiments or in place of counterpart features or aspects thereof. Accordingly, the specification and drawings are to be regarded in an illustrative rather than a restrictive sense.

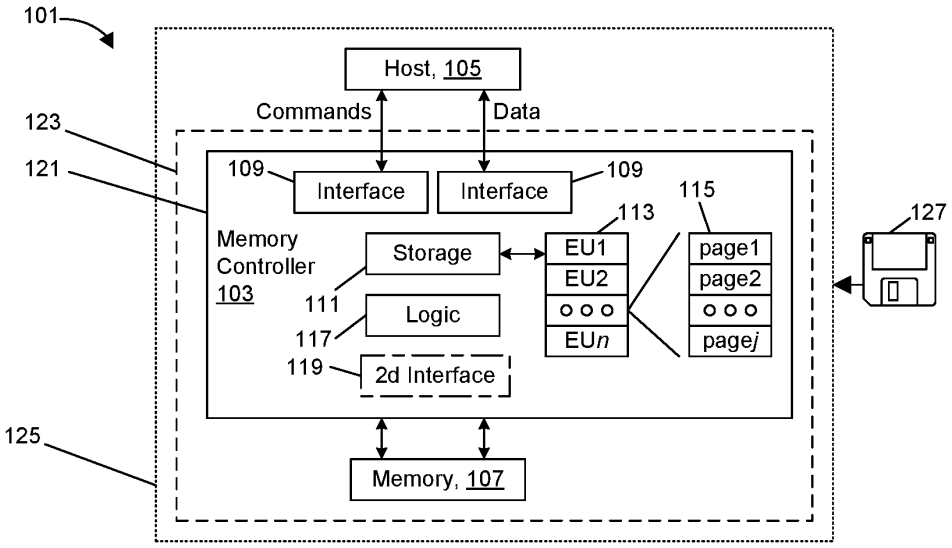

101
Host, 105
Commands
Data
123
121
109
Interface
Interface
109
115
113
127
Memory Controller 103
Storage
EU1
EU2
EUn
page1
page2
pagej
111
Logic
117
119
2d Interface
125
Memory, 107

FIG. 1

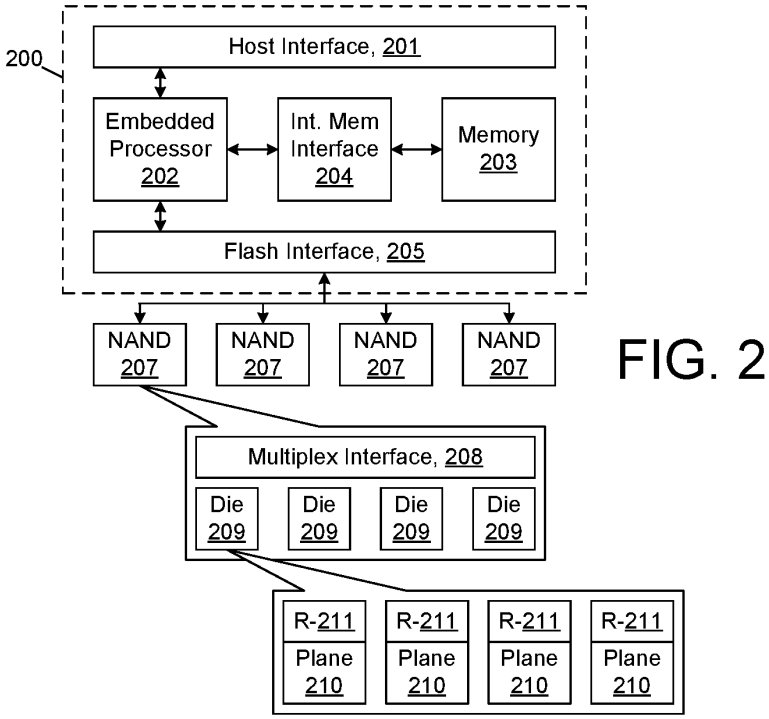

200
Host Interface, 201
Embedded Processor 202
Int. Mem Interface 204
Memory 203
Flash Interface, 205
NAND 207
NAND 207
NAND 207
NAND 207
Multiplex Interface, 208
Die 209
Die 209
Die 209
Die 209
R-211 Plane 210
R-211 Plane 210
R-211 Plane 210
R-211 Plane 210

FIG. 2

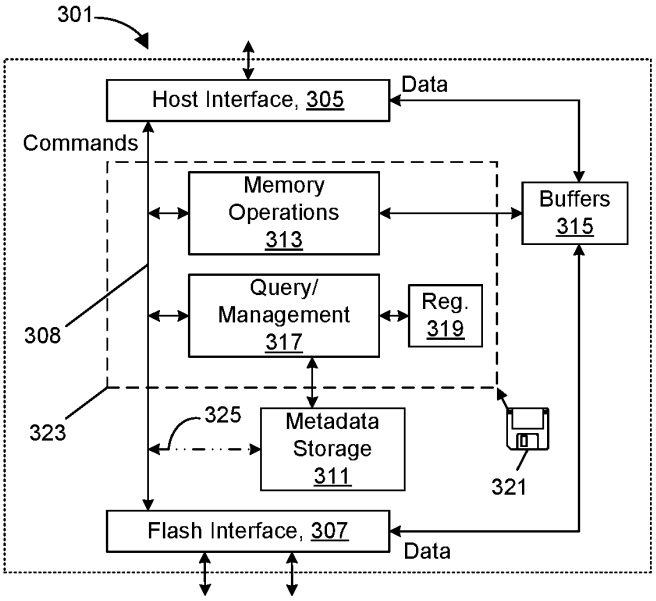

301
Host Interface, 305
Data
Commands
Memory Operations 313
Buffers 315
Query/ Management 317
Reg. 319
308
323
325
Metadata Storage 311
321
Flash Interface, 307
Data

FIG. 3A

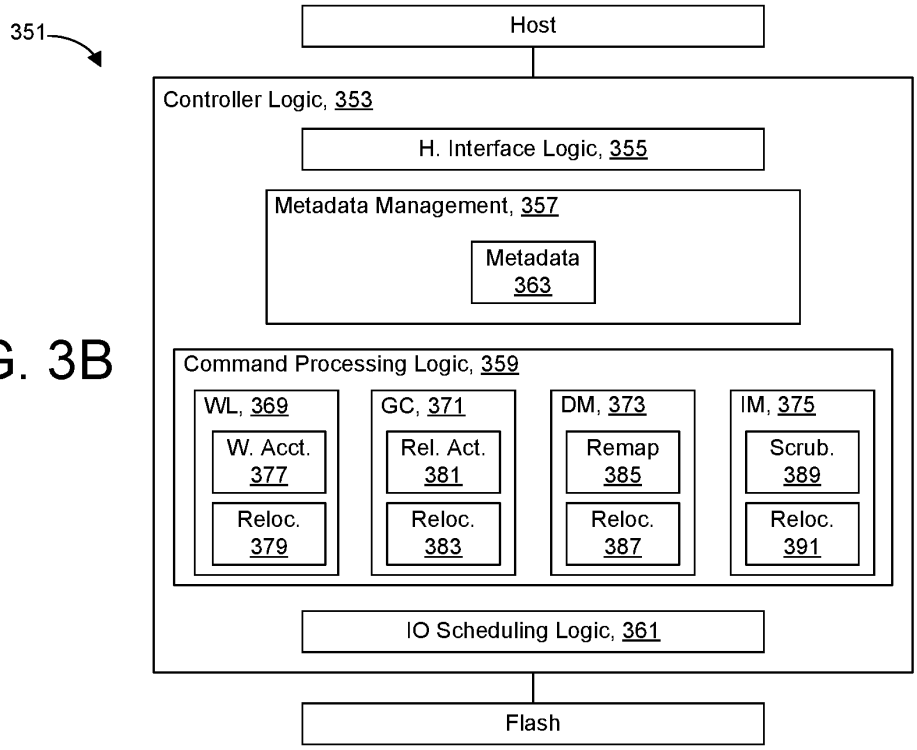

351
Host
Controller Logic, 353
H. Interface Logic, 355
Metadata Management, 357
Metadata 363
Command Processing Logic, 359
WL, 369
W. Acct. 377
Reloc. 379
GC, 371
Rel. Act. 381
Reloc. 383
DM, 373
Remap 385
Reloc. 387
IM, 375
Scrub. 389
Reloc. 391
IO Scheduling Logic, 361
Flash

FIG. 3B

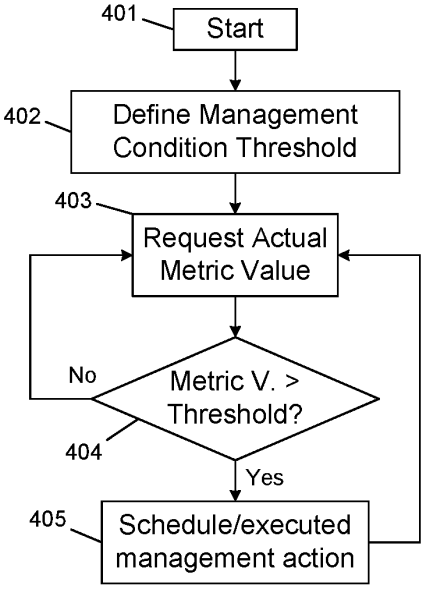

401
Start
402
Define Management Condition Threshold
403
Request Actual Metric Value
No
Metric V. > Threshold?
404
Yes
405
Schedule/executed management action

FIG. 4

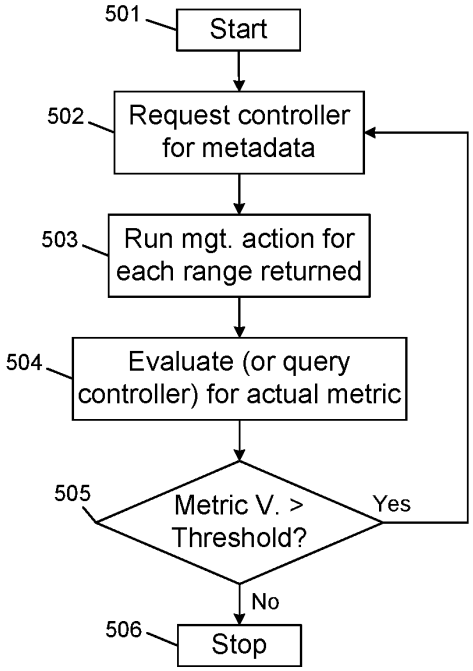

501
Start
502
Request controller for metadata
503
Run mgt. action for each range returned
504
Evaluate (or query controller) for actual metric
505
Metric V. > Threshold?
Yes
No
506
Stop

FIG. 5

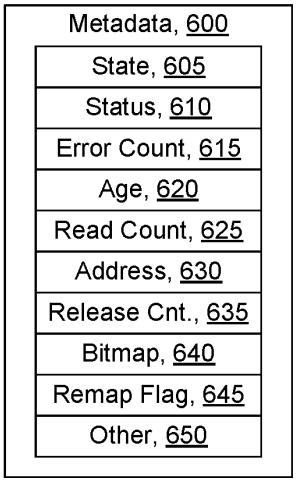

Metadata, 600
State, 605
Status, 610
Error Count, 615
Age, 620
Read Count, 625
Address, 630
Release Cnt., 635
Bitmap, 640
Remap Flag, 645
Other, 650

FIG. 6

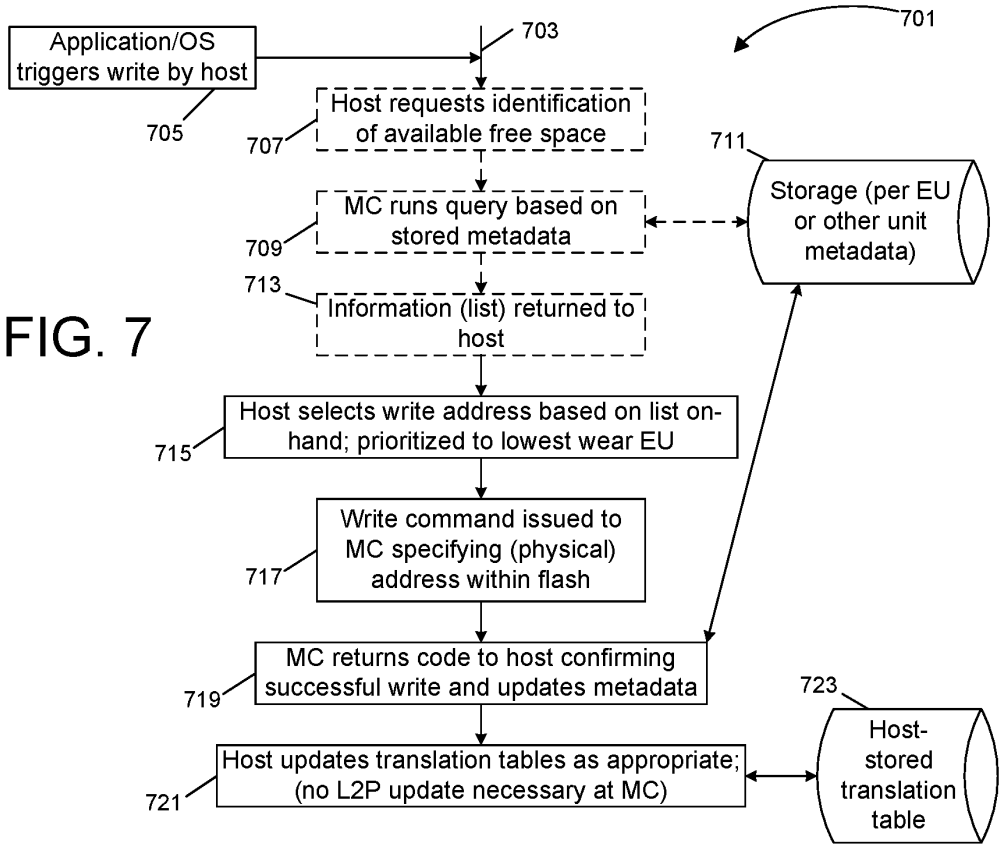

Application/OS triggers write by host
705
703
701
Host requests identification of available free space
707
711
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
709
713
Information (list) returned to host
Host selects write address based on list on-hand; prioritized to lowest wear EU
715
Write command issued to MC specifying (physical) address within flash
717
MC returns code to host confirming successful write and updates metadata
719
723
Host-stored translation table
Host updates translation tables as appropriate; (no L2P update necessary at MC)
721

FIG. 7

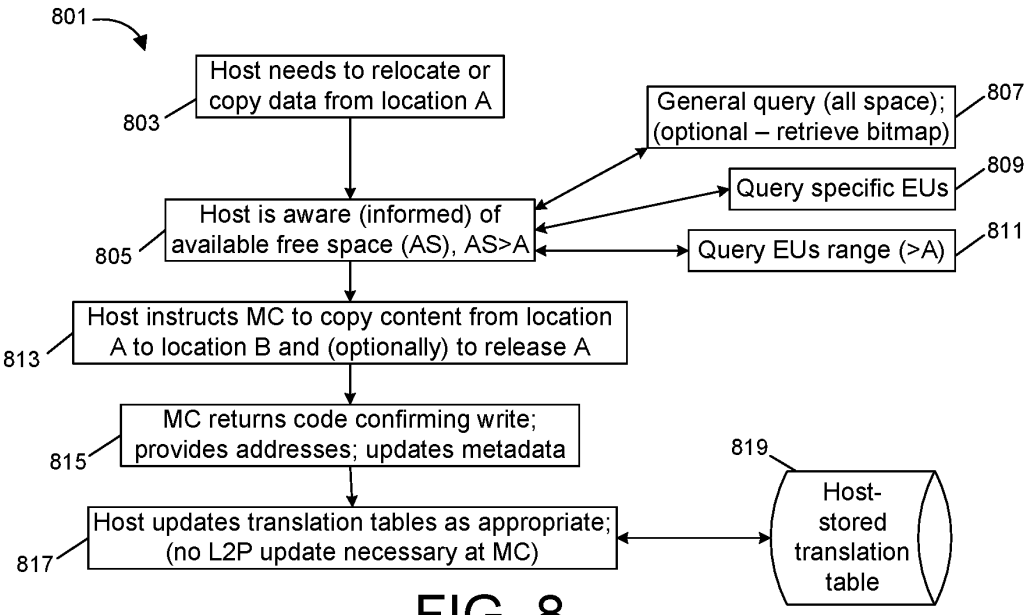

801
Host needs to relocate or copy data from location A
803
General query (all space); (optional – retrieve bitmap)
807
Query specific EUs
809
Host is aware (informed) of available free space (AS), AS>A
805
Query EUs range (>A)
811
Host instructs MC to copy content from location A to location B and (optionally) to release A
813
MC returns code confirming write; provides addresses; updates metadata
815
819
Host-stored translation table
Host updates translation tables as appropriate; (no L2P update necessary at MC)
817

FIG. 8

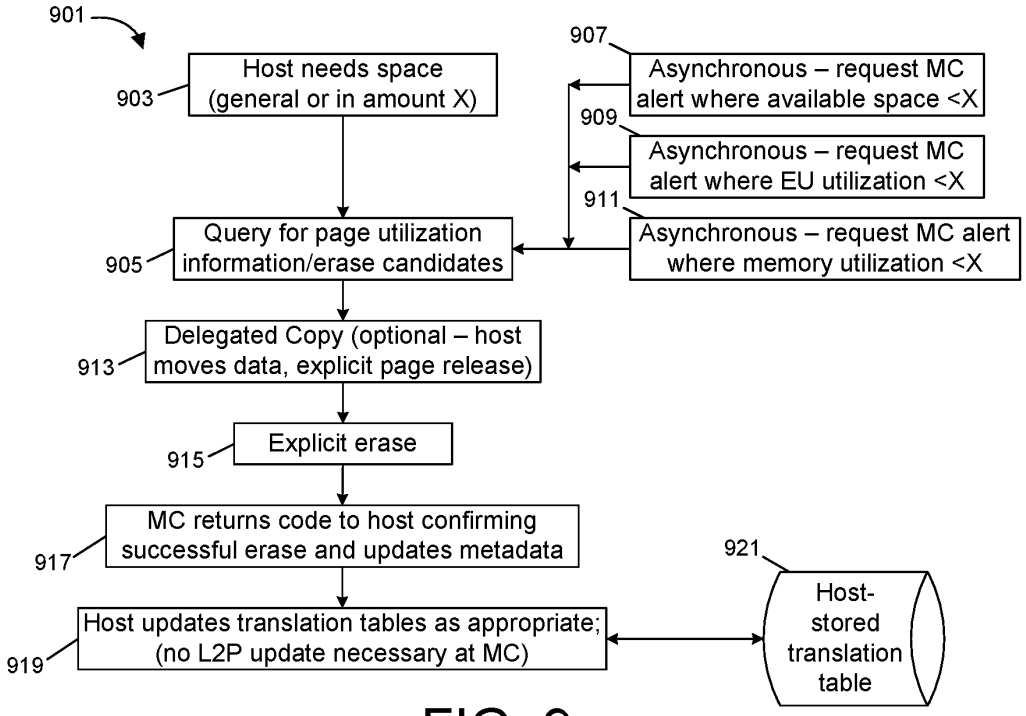

901
Host needs space (general or in amount X)
903
907
Asynchronous – request MC alert where available space <X
909
Asynchronous – request MC alert where EU utilization <X
911
Asynchronous – request MC alert where memory utilization <X
Query for page utilization information/erase candidates
905
Delegated Copy (optional – host moves data, explicit page release)
913
Explicit erase
915
MC returns code to host confirming successful erase and updates metadata
917
921
Host-stored translation table
Host updates translation tables as appropriate; (no L2P update necessary at MC)
919

FIG. 9

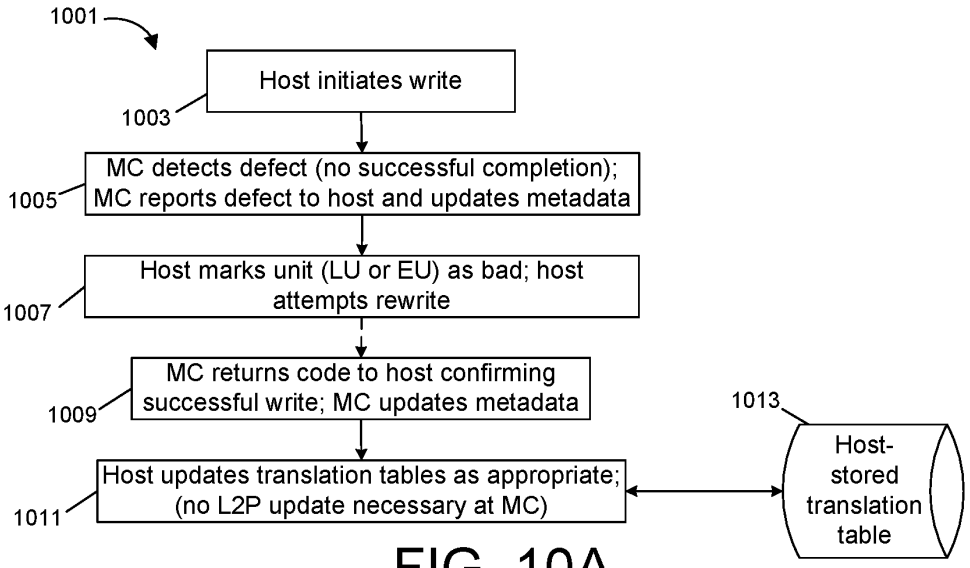


FIG. 10A

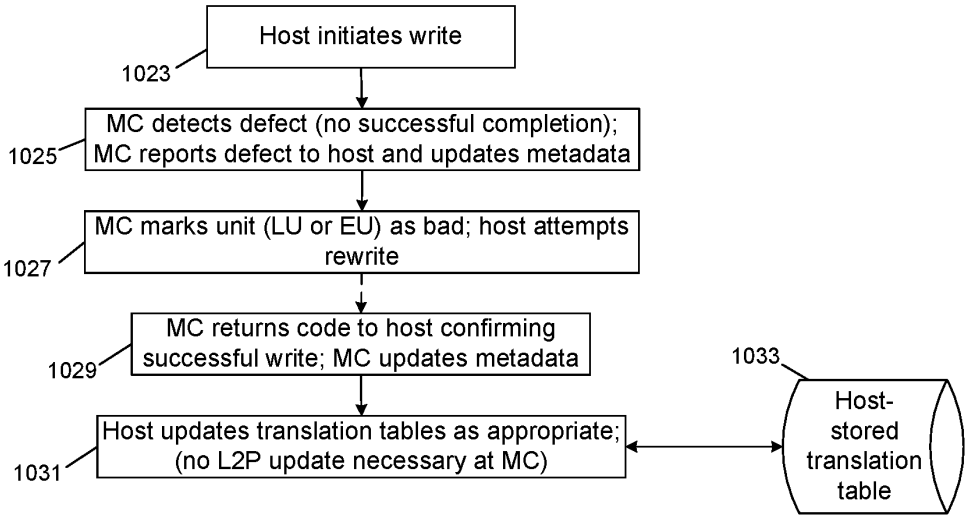


FIG. 10B

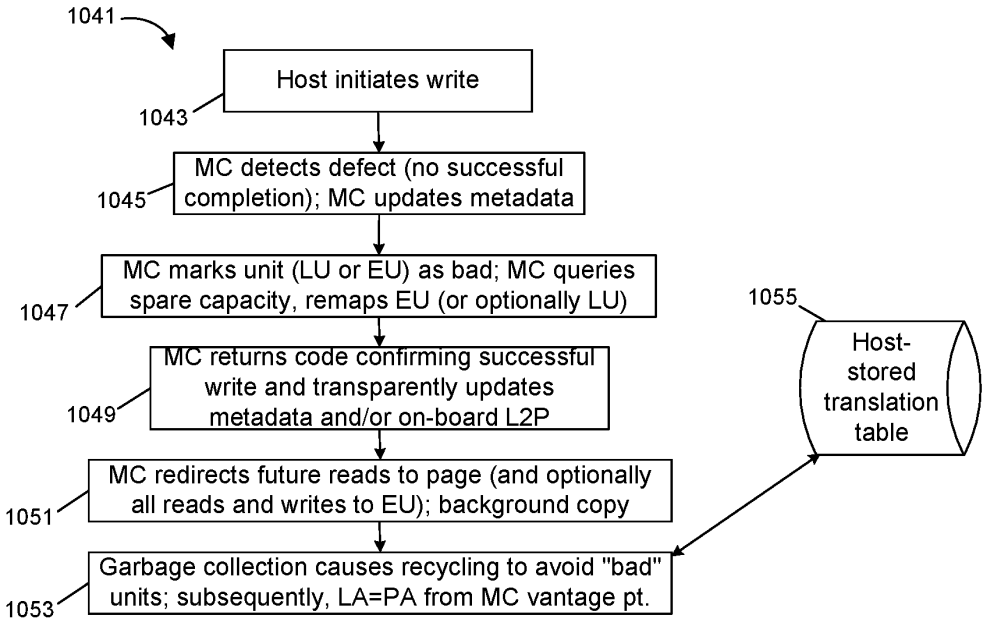


FIG. 10C

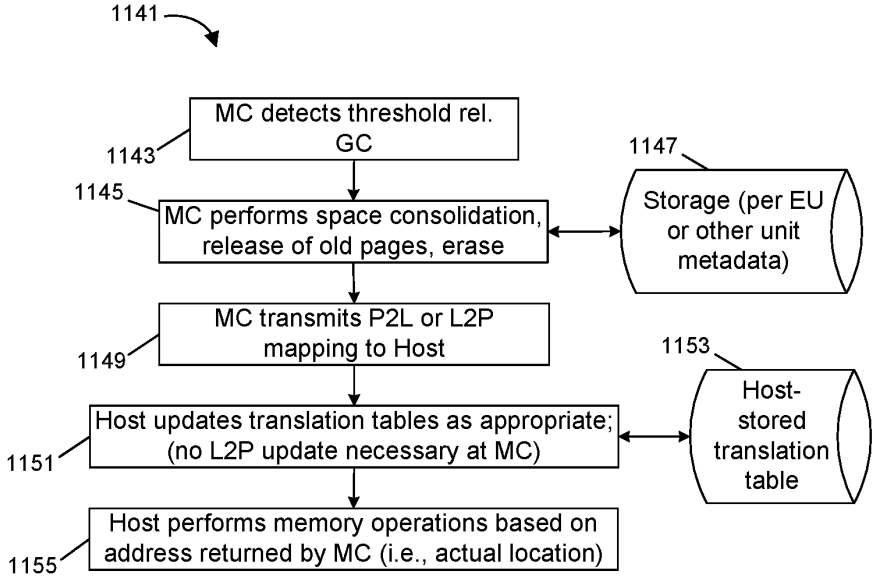


FIG. 11B

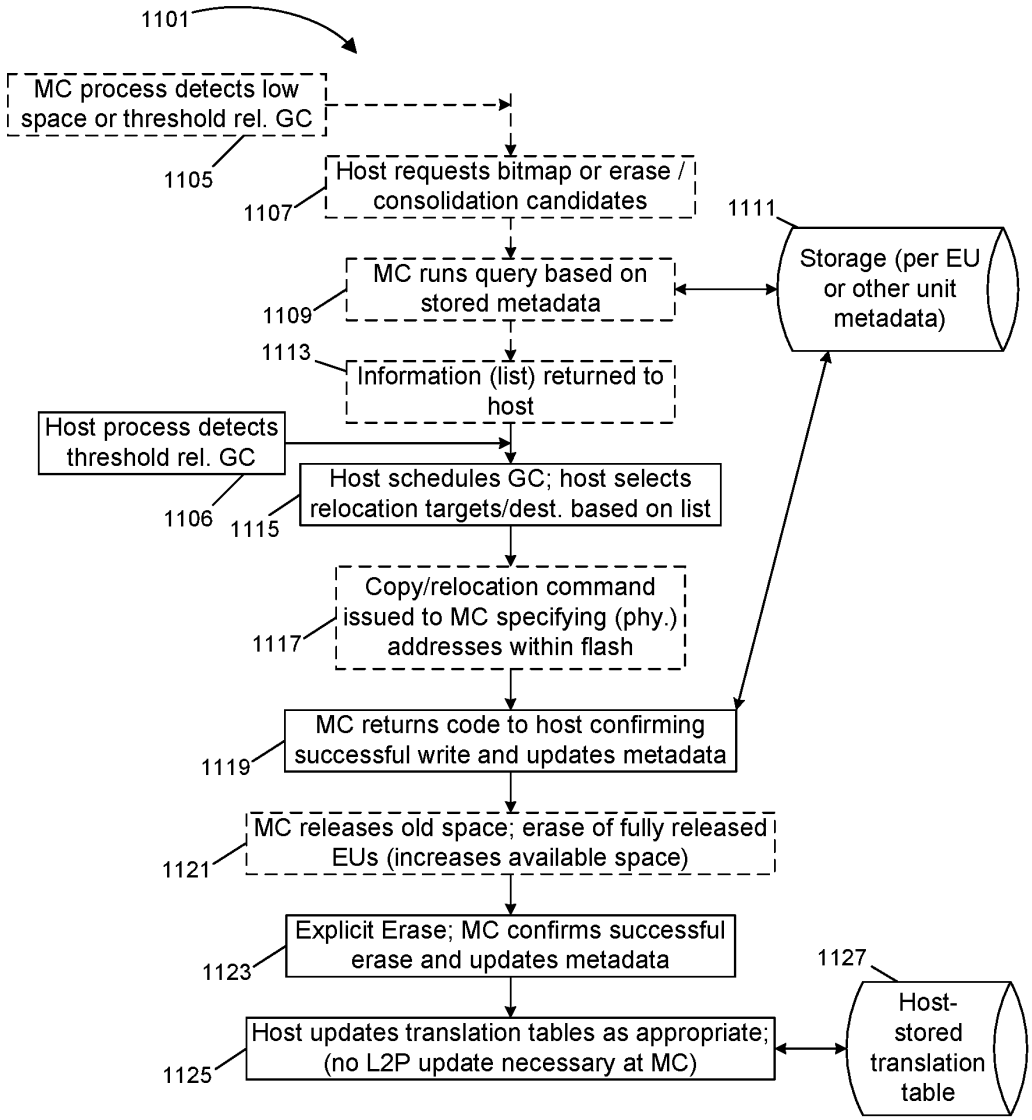

1101
MC process detects low space or threshold rel. GC
1105
Host requests bitmap or erase / consolidation candidates
1107
1111
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
1109
1113
Information (list) returned to host
Host process detects threshold rel. GC
1106
Host schedules GC; host selects relocation targets/dest. based on list
1115
Copy/relocation command issued to MC specifying (phy.) addresses within flash
1117
MC returns code to host confirming successful write and updates metadata
1119
MC releases old space; erase of fully released EUs (increases available space)
1121
Explicit Erase; MC confirms successful erase and updates metadata
1123
1127
Host-stored translation table
Host updates translation tables as appropriate; (no L2P update necessary at MC)
1125

FIG. 11A

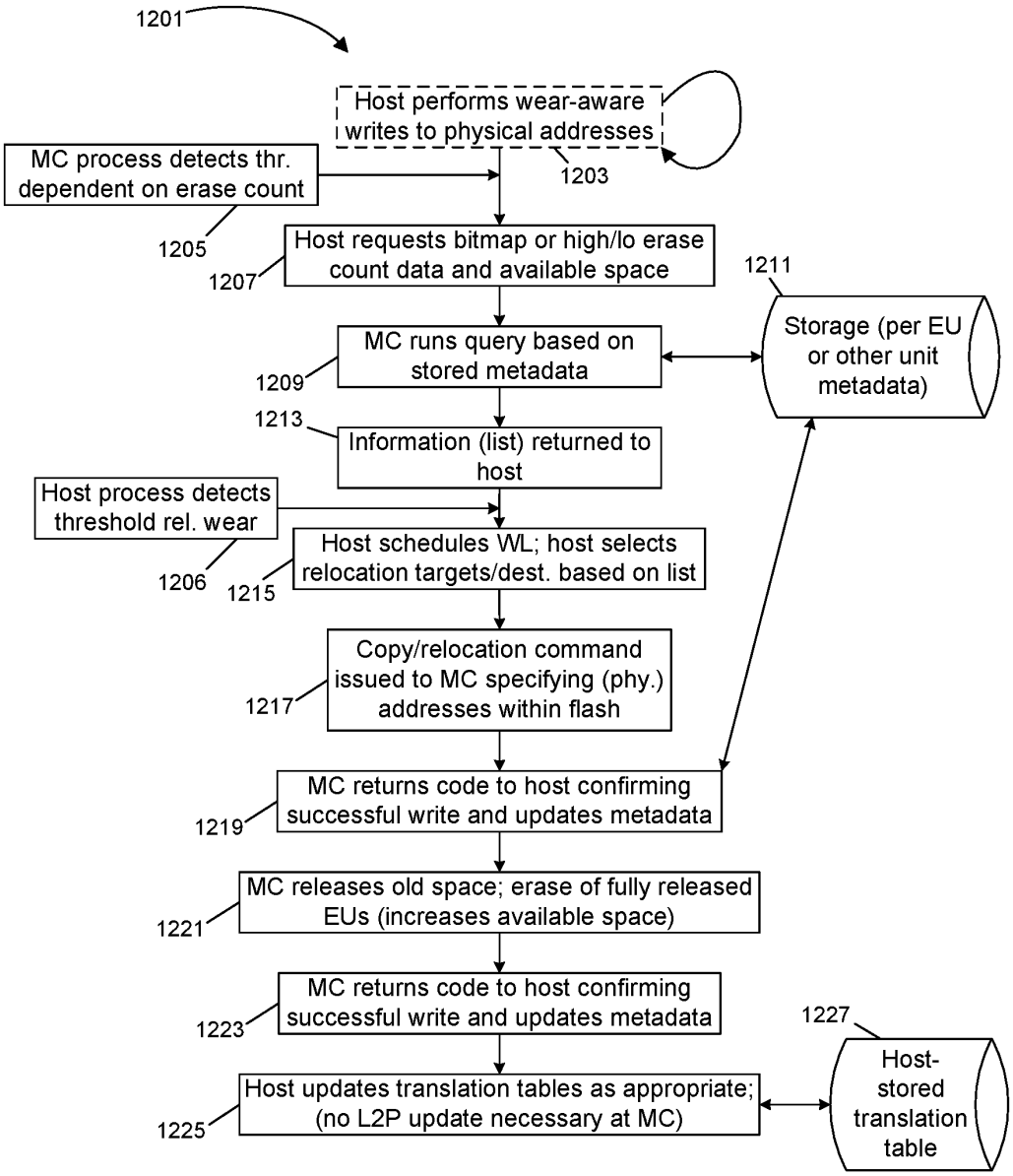

1201
Host performs wear-aware writes to physical addresses
1203
MC process detects thr. dependent on erase count
1205
Host requests bitmap or high/lo erase count data and available space
1207
1211
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
1209
1213
Information (list) returned to host
Host process detects threshold rel. wear
1206
Host schedules WL; host selects relocation targets/dest. based on list
1215
Copy/relocation command issued to MC specifying (phy.) addresses within flash
1217
MC returns code to host confirming successful write and updates metadata
1219
MC releases old space; erase of fully released EUs (increases available space)
1221
MC returns code to host confirming successful write and updates metadata
1223
1227
Host-stored translation table
Host updates translation tables as appropriate; (no L2P update necessary at MC)
1225

FIG. 12A

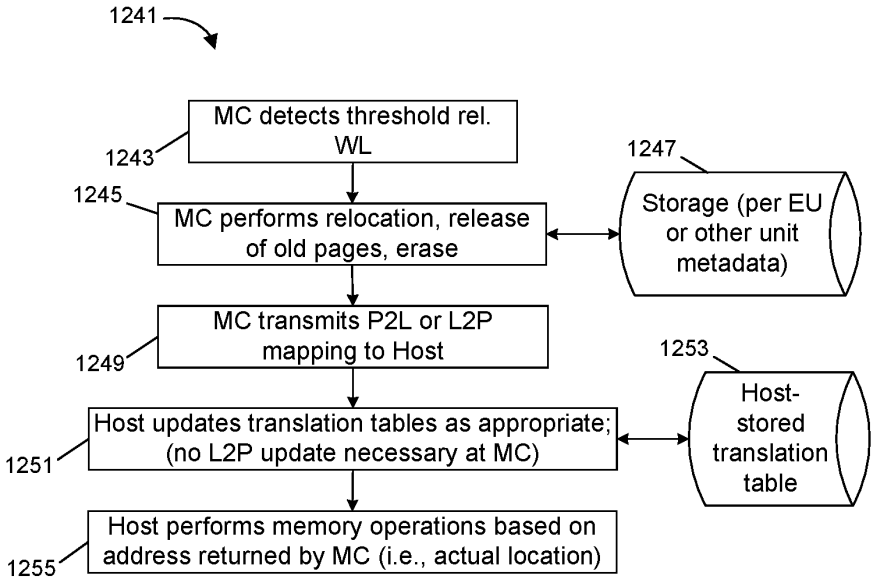


FIG. 12B

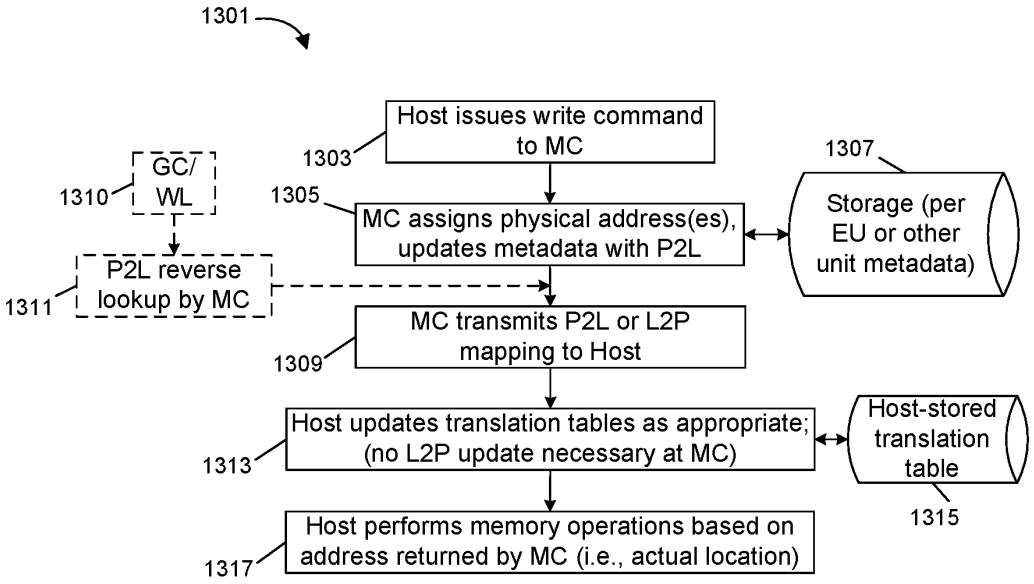


FIG. 13

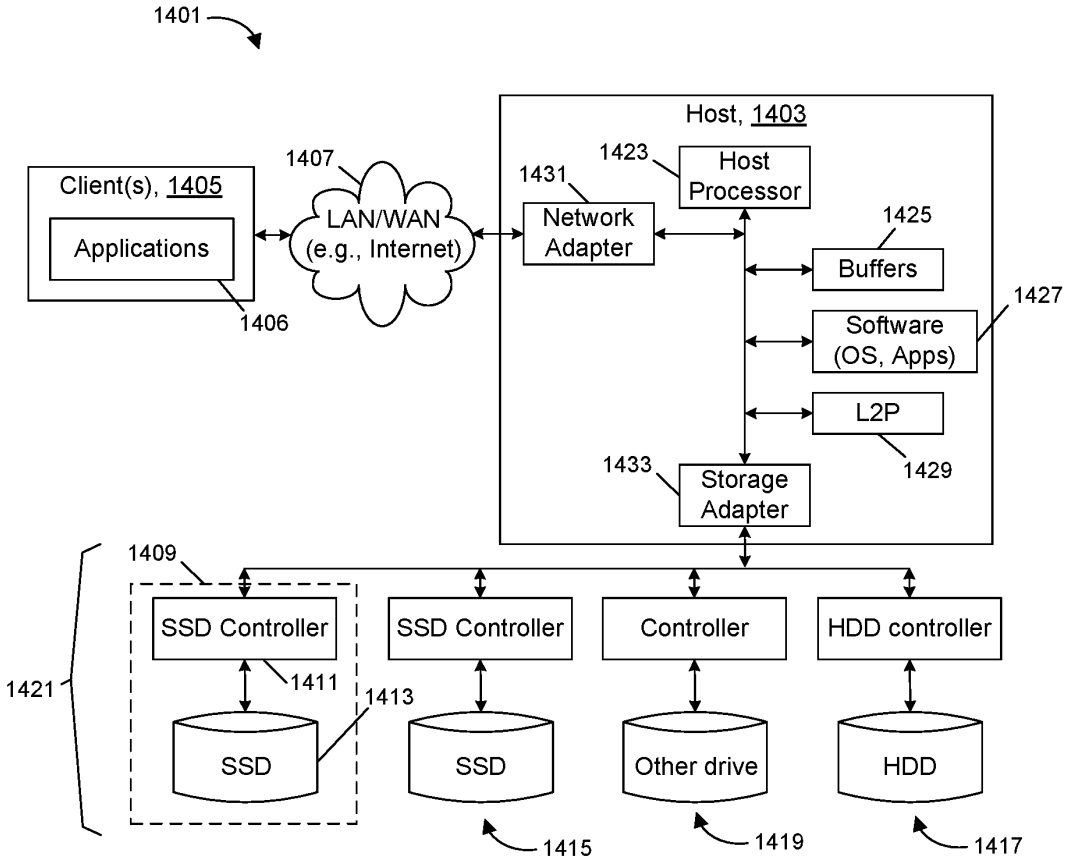

1401
Host, 1403
1407
1423
Host Processor
1431
Client(s), 1405
LAN/WAN (e.g., Internet)
Network Adapter
1425
Applications
Buffers
1406
1427
Software (OS, Apps)
L2P
1429
1433
Storage Adapter
1409
SSD Controller
SSD Controller
Controller
HDD controller
1421
1411
1413
SSD
SSD
Other drive
HDD
1415
1419
1417

FIG. 14

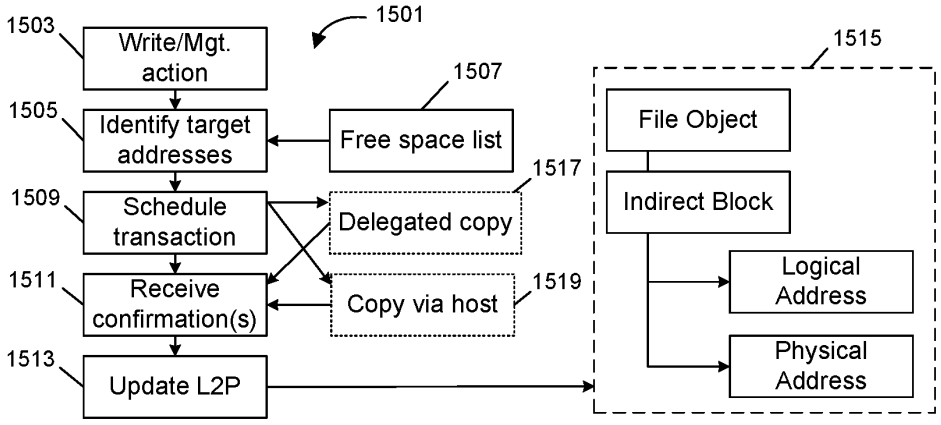

1501
1503
Write/Mgt. action
1507
1515
1505
Identify target addresses
Free space list
File Object
1517
1509
Schedule transaction
Delegated copy
Indirect Block
1511
Receive confirmation(s)
Copy via host
1519
Logical Address
1513
Update L2P
Physical Address

FIG. 15

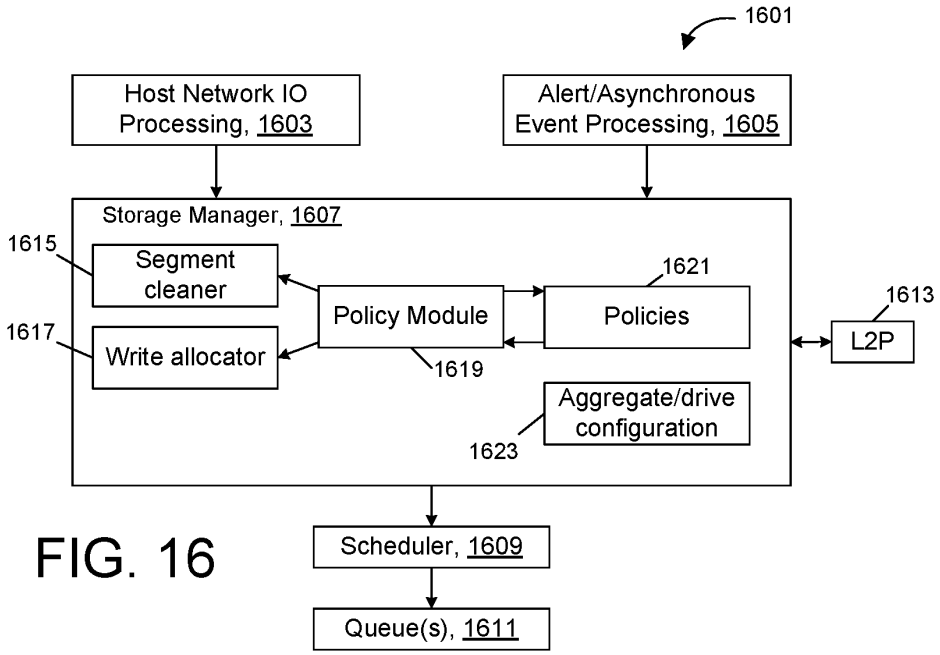

1601
Host Network IO Processing, 1603
Alert/Asynchronous Event Processing, 1605
Storage Manager, 1607
1615
Segment cleaner
1617
Write allocator
Policy Module
1619
1621
Policies
1613
L2P
Aggregate/drive configuration
1623
Scheduler, 1609
Queue(s), 1611

FIG. 16

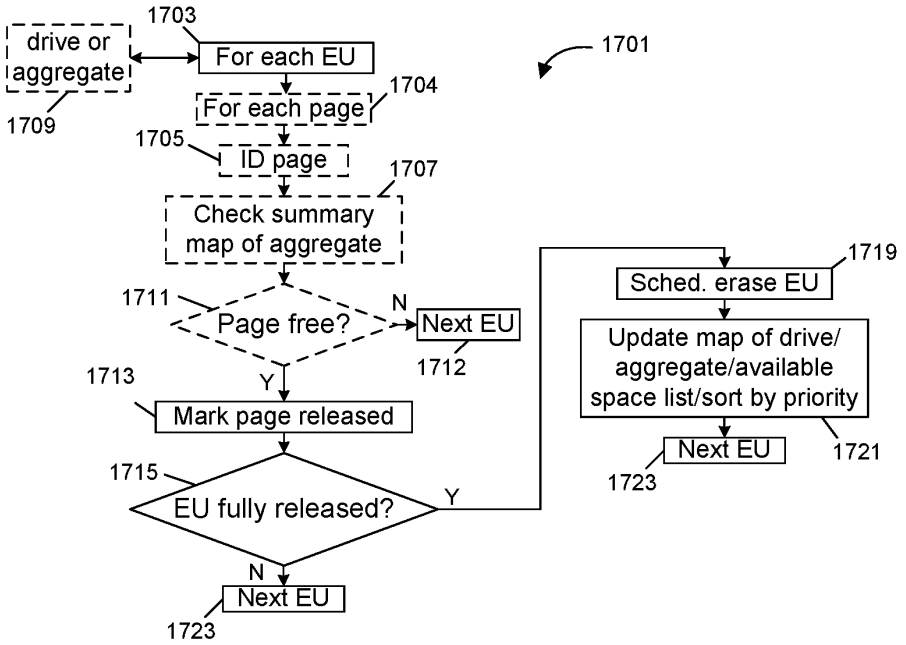

1703
drive or aggregate
1709
For each EU
1704
For each page
1705
ID page
1707
Check summary map of aggregate
1701
1711
Page free?
N
Next EU
1712
Y
1713
Mark page released
1715
EU fully released?
Y
N
Next EU
1723
1719
Sched. erase EU
Update map of drive/ aggregate/available space list/sort by priority
Next EU
1721
1723

FIG. 17

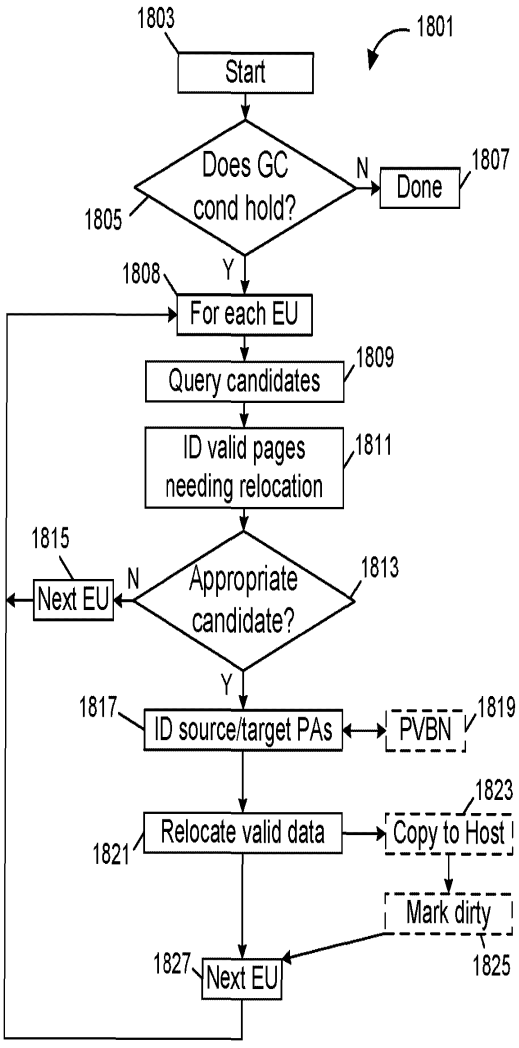

1803
1801
Start
Does GC cond hold?
1805
N
Done
1807
Y
1808
For each EU
Query candidates
1809
ID valid pages needing relocation
1811
1815
Next EU
N
Appropriate candidate?
1813
Y
1817
ID source/target PAs
PVBN
1819
1823
Relocate valid data
1821
Copy to Host
Mark dirty
1827
Next EU
1825

FIG. 18

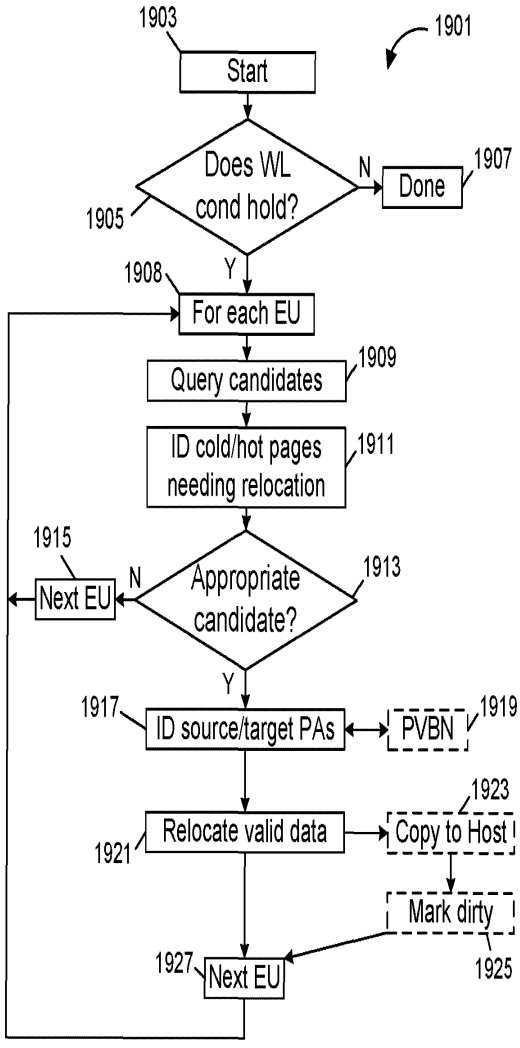

1903
1901
Start
Does WL cond hold?
1905
N
Done
1907
Y
1908
For each EU
Query candidates
1909
ID cold/hot pages needing relocation
1911
1915
Next EU
N
Appropriate candidate?
1913
Y
1917
ID source/target PAs
PVBN
1919
1923
Relocate valid data
1921
Copy to Host
Mark dirty
1927
Next EU
1925

FIG. 19

Electronic Patent Application Fee Transmittal

Application Number:	
Filing Date:	
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Filer:	Marc Philip Schuyler
Attorney Docket Number:	2020058 / RMS-02C1C1C14

Filed as Small Entity

Filing Fees for Utility under 35 USC 111(a)

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
UTILITY FILING FEE (ELECTRONIC FILING)	4011	1	75	75
UTILITY SEARCH FEE	2111	1	330	330
UTILITY EXAMINATION FEE	2311	1	380	380
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**785**

Electronic Acknowledgement Receipt	
EFS ID:	40330149
Application Number:	16997471
International Application Number:	
Confirmation Number:	1070
Title of Invention:	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	73091
Filer:	Marc Philip Schuyler
Filer Authorized By:	
Attorney Docket Number:	2020058 / RMS-02C1C1C14
Receipt Date:	19-AUG-2020
Filing Date:	
Time Stamp:	16:21:11
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	CARD
Payment was successfully received in RAM	$785
RAM confirmation Number	E20208IG22122551
Deposit Account	
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Application Data Sheet	0-aia0014-ADS-RMS-02C1C1C14.pdf	1397023 a455d6a119ddc557d3f4402d02c68e4025b6ba42	no	6
Warnings:					
Information:					
2	Assignee showing of ownership per 37 CFR 3.73	1-aia0095a-AssnStatement-RMS-02C1C1C14-2020-08-19.pdf	121670 423b8d86c40d69ca4c7874d9dcb06e4871f0b737	no	3
Warnings:					
Information:					
3	Oath or Declaration filed	2-RMS02-ExecutedDecs-All.pdf	2873101 4e3a3454a72cb0d99cb05459a59d152e1693e971	no	3
Warnings:					
Information:					
4	Information Disclosure Statement (IDS) Form (SB08)	3-IDS-cover-RMS-02.pdf	37805 012c75adb1d2fe11670c26e7d8a3d964a2dc3961	no	1
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
5	Information Disclosure Statement (IDS) Form (SB08)	4-IDS-2020-08-18.pdf	618595 368a0c7b962a032e1311674d9021667022c16c39	no	18
Warnings:					
Information:					
6	Information Disclosure Statement (IDS) Form (SB08)	5-RMS-McKesson-Statement.pdf	51950 cda64822b24c719ca9ba35b23ffa9cdce72f148d	no	2
Warnings:					

Information:					
This is not an USPTO supplied IDS fillable form					
7	Power of Attorney	6-aia0080-POA-2016-11-08-s.pdf	44477 2ad25a7b24c7336c2ae3391199615 4b407397931	no	1
Warnings:					
Information:					
8		7-2020-08-18-Radian-Utility-02C1C1C14-CLEAN.pdf	511844 83c172aefac137c5e239159fc6ebfed0ed7c92f4	yes	72

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Abstract	72	72
	Claims	66	71
	Specification	1	65

Warnings:					
Information:					
9	Drawings-only black and white line drawings	8-Visio-2020-03-27-RMS-02-Generic-Figures-FIN.pdf	137067 c79902c34692dd781769150f7216fd0e4d10cfe3	no	13
Warnings:					
Information:					
10	Fee Worksheet (SB06)	fee-info.pdf	35146 b8e7f8f55c42836dd9936a92b7d881db43d91d78	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	5828678		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	2020058 / RMS-02C1C1C14
	Application Number	
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Inventor Information:

Inventor 1 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Andrey	V.	Kuzmin	

Residence Information (Select One) US Residency ● Non US Residency Active US Military Service

City	Moscow	Country of Residence	RU

Mailing Address of Inventor:

Address 1	19 Avantgarde St. #35
Address 2	

City	Moscow	State/Province	
Postal Code	125493	Country	RU

Inventor 2 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Mike		Jadon	

Residence Information (Select One) ● US Residency ○ Non US Residency Active US Military Service

City	Manhattan Beach	State/Province	CA	Country of Residence	US

Mailing Address of Inventor:

Address 1	26025 Mureau Rd.
Address 2	

City	Calabasas	State/Province	CA
Postal Code	91302	Country	US

Inventor 3 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Richard	M.	Mathews	

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	2020058 / RMS-02C1C1C14
	Application Number	
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	

Residence Information (Select One) ◉ US Residency ○ Non US Residency Active US Military Service

City	Porter Ranch	State/Province	CA	Country of Residence	US

Mailing Address of Inventor:

Address 1	18810 San Fernando Mission Blvd		
Address 2			
City	Porter Ranch	State/Province	CA
Postal Code	91326	Country	US

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	73091		
Email Address	marc@schuylerlawgroup.com	Add Email	Remove Email

Application Information:

Title of the Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS		
Attorney Docket Number	2020058 / RMS-02C1C1C14	Small Entity Status Claimed	☒
Application Type	Nonprovisional		
Subject Matter	Utility		
Suggested Class (if any)		Sub Class (if any)	
Suggested Technology Center (if any)			
Total Number of Drawing Sheets (if any)	13	Suggested Figure for Publication (if any)	

Publication Information:

☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)

☒ **Request Not to Publish.** I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application **has not and will not** be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

Representative Information:

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	2020058 / RMS-02C1C1C14
	Application Number	
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Either enter Customer Number or complete the Representative Name section below. If both sections are completed the customer Number will be used for the Representative Information during processing.

Please Select One:	● Customer Number	US Patent Practitioner	◯ Limited Recognition (37 CFR 11.9)
Customer Number	73091		

Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, or 365(c) or indicate National Stage entry from a PCT application. Providing this information in the application data sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78.

Prior Application Status	Pending			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	
	Continuation of	15/625931	2017-06-16	

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
15/625931	Continuation of	14/466167	2014-08-22	9727454	2017-08-08

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
14/466167	Continuation of	13/767723	2013-02-14	9652376	2017-05-16

Prior Application Status	Expired			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	
13/767723	non provisional of	61/757464	2013-01-28	

Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button. Add

Foreign Priority Information:

This section allows for the applicant to claim benefit of foreign priority and to identify any prior foreign application for which priority is not claimed. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55(a).

Remove

Application Number	Country	Filing Date (YYYY-MM-DD)	Priority Claimed
			● Yes No

Additional Foreign Priority Data may be generated within this form by selecting the **Add** button. Add

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	2020058 / RMS-02C1C1C14
	Application Number	
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	

Authorization to Permit Access:

Authorization to Permit Access to the Instant Application by the Participating Offices
If checked, the undersigned hereby grants the USPTO authority to provide the European Patent Office (EPO), the Japan Patent Office (JPO), the Korean Intellectual Property Office (KIPO), the World Intellectual Property Office (WIPO), and any other intellectual property offices in which a foreign application claiming priority to the instant patent application is filed access to the instant patent application. See 37 CFR 1.14(c) and (h). This box should not be checked if the applicant does not wish the EPO, JPO, KIPO, WIPO, or other intellectual property office in which a foreign application claiming priority to the instant patent application is filed to have access to the instant patent application. In accordance with 37 CFR 1.14(h)(3), access will be provided to a copy of the instant patent application with respect to: 1) the instant patent application-as-filed; 2) any foreign application to which the instant patent application claims priority under 35 U.S.C. 119(a)-(d) if a copy of the foreign application that satisfies the certified copy requirement of 37 CFR 1.55 has been filed in the instant patent application; and 3) any U.S. application-as-filed from which benefit is sought in the instant patent application. In accordance with 37 CFR 1.14(c), access may be provided to information concerning the date of filing this Authorization.

Applicant Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.	
Applicant	**1**
If the applicant is the inventor (or the remaining joint inventor or inventors under 37 CFR 1.45), this section should not be completed. The information to be provided in this section is the name and address of the legal representative who is the applicant under 37 CFR 1.43; or the name and address of the assignee, person to whom the inventor is under an obligation to assign the invention, or person who otherwise shows sufficient proprietary interest in the matter who is the applicant under 37 CFR 1.46. If the applicant is an applicant under 37 CFR 1.46 (assignee, person to whom the inventor is obligated to assign, or person who otherwise shows sufficient proprietary interest) together with one or more joint inventors, then the joint inventor or inventors who are also the applicant should be identified in this section. Remove	
● Assignee	Legal Representative under 35 U.S.C. 117
Person to whom the inventor is obligated to assign.	Person who shows sufficient proprietary interest
If applicant is the legal representative, indicate the authority to file the patent application, the inventor is:	
Name of the Deceased or Legally Incapacitated Inventor :	
If the Assignee is an Organization check here. ☒	
Organization Name	Radian Memory Systems, Inc.

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	2020058 / RMS-02C1C1C14
	Application Number	
Title of Invention	NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS	

Mailing Address Information:			
Address 1	26025 Mureau Rd.		
Address 2			
City	Calabasas	**State/Province**	CA
Country	US	Postal Code	91302
Phone Number		Fax Number	
Email Address			

Additional Applicant Data may be generated within this form by selecting the Add button. Add

Signature:

Remove

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications

Signature	/Marc Schuyler/			Date (YYYY-MM-DD)	2020-08-19
First Name	Marc	Last Name	Schuyler	Registration Number	35675

Additional Signature may be generated within this form by selecting the Add button. Add

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PTO/AIA/96 (08-12)
Approved for use through 01/31/2013. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

STATEMENT UNDER 37 CFR 3.73(c)

Applicant/Patent Owner: Radian Memory Systems, Inc.

Application No./Patent No.: Herewith (con of 15/625931) Filed/Issue Date: Herewith

Titled: NONVOLATILE MEMORY CONTROLLER HOST-ISSUED ADDRESS DELIMITED ERASURE AND MEMORY CONTROLLER REMAPPING OF HOST-ADDRESS SPACE FOR BAD BLOCKS

Radian Memory Systems, Inc., a Corporation

(Name of Assignee) (Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that, for the patent application/patent identified above, it is (choose **one** of options 1, 2, 3 or 4 below):

1. ☑ The assignee of the entire right, title, and interest.

2. ☐ An assignee of less than the entire right, title, and interest (check applicable box):

   ☐ The extent (by percentage) of its ownership interest is _____%. Additional Statement(s) by the owners holding the balance of the interest must be submitted to account for 100% of the ownership interest.

   ☐ There are unspecified percentages of ownership. The other parties, including inventors, who together own the entire right, title and interest are:

   Additional Statement(s) by the owner(s) holding the balance of the interest must be submitted to account for the entire right, title, and interest.

3. ☐ The assignee of an undivided interest in the entirety (a complete assignment from one of the joint inventors was made). The other parties, including inventors, who together own the entire right, title, and interest are:

   Additional Statement(s) by the owner(s) holding the balance of the interest must be submitted to account for the entire right, title, and interest.

4. ☐ The recipient, via a court proceeding or the like (*e.g.*, bankruptcy, probate), of an undivided interest in the entirety (a complete transfer of ownership interest was made). The certified document(s) showing the transfer is attached.

The interest identified in option 1, 2 or 3 above (not option 4) is evidenced by either (choose **one** of options A or B below):

A. ☐ An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

B. ☑ A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as follows:

1. From: inventors To: Radian Memory Systems, LLC

   The document was recorded in the United States Patent and Trademark Office at

   Reel 029817, Frame 0931, or for which a copy thereof is attached.

2. From: Radian Memory Systems, LLC To: Radian Memory Systems, Inc.

   The document was recorded in the United States Patent and Trademark Office at

   Reel 040579, Frame 0405, or for which a copy thereof is attached.

This collection of information is required by 37 CFR 3.73(b). The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/AIA/96 (08-12)
Approved for use through 01/31/2013. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

STATEMENT UNDER 37 CFR 3.73(c)

3. From: _____ To: _____

The document was recorded in the United States Patent and Trademark Office at

Reel _____, Frame _____, or for which a copy thereof is attached.

4. From: _____ To: _____

The document was recorded in the United States Patent and Trademark Office at

Reel _____, Frame _____, or for which a copy thereof is attached.

5. From: _____ To: _____

The document was recorded in the United States Patent and Trademark Office at

Reel _____, Frame _____, or for which a copy thereof is attached.

6. From: _____ To: _____

The document was recorded in the United States Patent and Trademark Office at

Reel _____, Frame _____, or for which a copy thereof is attached.

☑ Additional documents in the chain of title are listed on a supplemental sheet(s).

☑ As required by 37 CFR 3.73(c)(1)(i), the documentary evidence of the chain of title from the original owner to the assignee was, or concurrently is being, submitted for recordation pursuant to 37 CFR 3.11.

[NOTE: A separate copy (i.e., a true copy of the original assignment document(s)) must be submitted to Assignment Division in accordance with 37 CFR Part 3, to record the assignment in the records of the USPTO. See MPEP 302.08]

The undersigned (whose title is supplied below) is authorized to act on behalf of the assignee.

/Marc P. Schuyler/	19 August 2020
Signature	Date
Marc P. Schuyler	35675
Printed or Typed Name	Title or Registration Number

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PTO/AIA/01 (06-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

DECLARATION (37 CFR 1.63) FOR UTILITY OR DESIGN APPLICATION USING AN APPLICATION DATA SHEET (37 CFR 1.76)

Title of Invention	COOPERATIVE FLASH MEMORY CONTROL

As the below named inventor, I hereby declare that:

This declaration is directed to:

- [■] The attached application, or
- [ ] United States application or PCT international application number _____

filed on _____.

The above-identified application was made or authorized to be made by me.

I believe that I am the original inventor or an original joint inventor of a claimed invention in the application.

I hereby acknowledge that any willful false statement made in this declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

WARNING:

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

LEGAL NAME OF INVENTOR

Inventor: _____ Date (Optional): 02/12/2013

Signature: Andrey V. Kuzmin

Note: An application data sheet (PTO/SB/14 or equivalent), including naming the entire inventive entity, must accompany this form or must have been previously filed. Use an additional PTO/AIA/01 form for each additional inventor.

This collection of information is required by 35 U.S.C. 115 and 37 CFR 1.63. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 1 minute to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/AIA/01 (06-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

DECLARATION (37 CFR 1.63) FOR UTILITY OR DESIGN APPLICATION USING AN APPLICATION DATA SHEET (37 CFR 1.76)

Title of Invention	COOPERATIVE FLASH MEMORY CONTROL

As the below named inventor, I hereby declare that:

This declaration is directed to:

- [■] The attached application, or
- [ ] United States application or PCT international application number _____

  filed on _____.

The above-identified application was made or authorized to be made by me.

I believe that I am the original inventor or an original joint inventor of a claimed invention in the application.

I hereby acknowledge that any willful false statement made in this declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

WARNING:

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

LEGAL NAME OF INVENTOR

Inventor: M. L. M Date (Optional): 2/11/13

Signature: Mike Jadon

Note: An application data sheet (PTO/SB/14 or equivalent), including naming the entire inventive entity, must accompany this form or must have been previously filed. Use an additional PTO/AIA/01 form for each additional inventor.

This collection of information is required by 35 U.S.C. 115 and 37 CFR 1.63. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 1 minute to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/AIA/01 (06-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

DECLARATION (37 CFR 1.63) FOR UTILITY OR DESIGN APPLICATION USING AN APPLICATION DATA SHEET (37 CFR 1.76)

Title of Invention	COOPERATIVE FLASH MEMORY CONTROL

As the below named inventor, I hereby declare that:

This declaration is directed to:

- [■] The attached application, or
- [ ] United States application or PCT international application number _____

  filed on _____.

The above-identified application was made or authorized to be made by me.

I believe that I am the original inventor or an original joint inventor of a claimed invention in the application.

I hereby acknowledge that any willful false statement made in this declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

WARNING:

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

LEGAL NAME OF INVENTOR

Inventor: _____ Date (Optional): 2/11/2013

Signature: Richard M. Mathews

Note: An application data sheet (PTO/SB/14 or equivalent), including naming the entire inventive entity, must accompany this form or must have been previously filed. Use an additional PTO/AIA/01 form for each additional inventor.

This collection of information is required by 35 U.S.C. 115 and 37 CFR 1.63. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 1 minute to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Electronically filed

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

INFORMATION DISCLOSURE STATEMENT
McKesson Statement

The present application contains subject matter and/or claims similar to subject matter and/or claims present in one of more of the US patent applications listed below, and/or a continuation of one of the US patent applications listed below. The Examiner is encouraged to review the file wrappers of cases identified below, parents and continuations of the cases identified below, and to consider arguments advanced by US patent examiners and Applicant's remarks. To the extent the Examiner has any questions, he/she is respectfully requested to contact the undersigned using the information set forth below.

Applications identified:

Application No.	***First Inventor***	***Status***
13/767723	Kuzmin	ISSUED - USPT 9652376
14/466167	Kuzmin	ISSUED - USPT 9727454
15/009275	Kuzmin	ISSUED - USPT 9400749
15/625931	Kuzmin	on appeal, appeal docket no. 2019-005230
15/625946	Kuzmin	pending
14/047193	Kuzmin	ISSUED - USPT 9229854
14/951708	Kuzmin	ISSUED - USPT 9519578
15/346641	Kuzmin	ISSUED - USPT 9710377
15/621888	Kuzmin	ISSUED - USPT 10445229
16/570922	Kuzmin	pending
16/751925	Kuzmin	pending
16/783083	Kuzmin	pending
16/794805	Kuzmin	pending
16/833547	Kuzmin	pending
14/848273	Kuzmin	ISSUED - USPT 10642505
14/880529	Lercari	USPT 9542118
15/053372	Lercari	USPT 9588904
15/074778	Lercari	USPT 9785572
15/690006	Lercari	ISSUED - USPT 10642748
15/211927	Jadon	USPT 10552058
16/591735	Jadon	pending
15/211939	Chen	USPT 10552085
16/702736	Chen	pending

16/707934	Lercari	pending
16/779918	Lercari	pending

Attorney For Applicant
Respectfully submitted,
/Marc P. Schuyler/
(408) 655-6375
marc@schuylerlawgroup.com

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

INFORMATION DISCLOSURE STATEMENT

Dear Sir:

In accordance with 37 C.F.R. §§1.97 and 1.98, and in compliance with the duty of disclosure set forth in 37 C.F.R. §1.56, applicant submits attached Form PTO/SB/08a (EFS-Web) for consideration and request the references cited therein be made of record by the U.S. Patent and Trademark Office in the above-captioned application.

Applicant respectfully points out that the submission of the listed references in this Information Disclosure Statement is not an admission that they are prior art or that they are material to patentability of any claims of the application. Also, the submission of this Information Disclosure Statement is not an indication that a search has been made by Applicant.

It is respectfully submitted that copies of all cited materials (and translations, as appropriate) have already been cited in one or more parent applications relied upon for priority, in a manner compliant with 37 C.F.R. §1.98(a-c); these one or more applications are identified below -

USSN 13/767723, USSN 14/466167, USSN 15/625931

Accordingly, per the provisions of 37 C.F.R. §1.98(d), copies of such materials which were previously provided and are not again being provided herewith. To the extent not already submitted in connection with the referenced parent application, NPLs and Foreign References (and translations thereof, as appropriate) are being furnished herewith. Consideration of the foregoing plus the prompt return of a copy of the enclosed Form SB/08A with the Examiner's initials in the left column in accordance with MPEP §609 are respectfully requested.

Respectfully submitted,

August 18, 2020

/Marc P. Schuyler/
Marc P. Schuyler, Reg. No. 35,675
Attorney for Applicant

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Andrey V. Kuzmin
Art Unit	
Examiner Name	unknown
Attorney Docket Number	2020058 / RMS-02C1C1C14

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	8065473	A1	2011-11-22	Ito et al.	
	2	6134631	A	2000-10-17	Jennings III	
	3	7861122	B2	2010-12-28	Cornwell et al.	
	4	5652857	A	1997-07-29	Shimoi et al.	
	5	7096378	B2	2006-08-22	Stence et al.	
	6	7404031	B2	2008-07-22	Oshima	
	7	7702846	B2	2010-04-20	Nakanishi et al.	
	8	7752381	B2	2010-07-06	Wong	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	9	7801561	B2	2010-09-21	Parikh et al.	
	10	7818489	B2	2010-10-19	Karamcheti et al.	
	11	7941692	B2	2011-05-10	Royer et al.	
	12	7970983	B2	2011-06-28	Nochimowski	
	13	7991944	B2	2011-08-02	Lee et al.	
	14	8024545	B2	2011-09-20	Kim et al.	
	15	8074022	B2	2011-12-06	Okin et al.	
	16	8082389	B2	2011-12-20	Fujibayashi	
	17	8086790	B2	2011-12-27	Roohparvar	
	18	5568423	A	1996-10-22	Jou et al.	
	19	7836244	B2	2010-11-16	Kim et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	20	8065471	B2	2011-11-22	Yano et al.	
	21	9229854	B2	2016-01-05	Kuzmin et al.	
	22	5860082	A	1999-01-12	Smith et al.	
	23	7339823	B2	2008-03-04	Nakayama et al.	
	24	7581078	B2	2009-08-25	Ware	
	25	7710777	B1	2010-05-04	Mintierth	
	26	8055833	B2	2011-11-08	Danilak et al.	
	27	8068365	B2	2011-11-29	Kim	
	28	9542118	B1	2017-01-10	Lercari et al.	
	29	9400749	B1	2016-07-26	Kuzmin et al.	
	30	9519578	B1	0216-12-13	Kuzmin et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	31	8001318	B1	2011-08-16	Iyer	
	32	8572331	B1	2013-10-29	Shalvi	
	33	8072463	B1	2011-12-06	Van Dyke	
	34	9335939	B2	2016-05-12	Bennett et al.	
	35	8347042	B2	2013-01-01	You	
	36	9727454	B1	2017-08-08	Kuzmin et al.	
	37	9652376	B1	2017-05-16	Kuzmin et al.	
	38	9710377	B1	2017-07-18	Kuzmin et al.	
	39	10445229	B1	2019-10-15	Kuzmin et al.	
	40	9588904	B1	2017-03-07	Lercari et al.	
	41	9785572	B1	2017-10-10	Lercari et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	42	10552058	B1	2020-02-04	Jadon et al.	
	43	10552085	B1	2020-02-04	Chen et al.	
	44	7406563	A1	2008-07-29	Nagshain	
	45	8645634	B1	2014-02-04	Cox et al.	
	46	9696917	B1	2017-07-04	Sareena et al.	
	47	9846541	B2	2017-12-19	Miyamoto et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20080189485	A1	2008-08-07	Jung et al.	
	2	20110033548	A1	2011-02-10	Kimmel et al.	
	3	20100281230	A1	2010-11-04	Rabii et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	4	20090327602	A1	2009-12-31	Moore et al.	
	5	20110296133	A1	2011-12-01	Flynn et al.	
	6	20120033519	A1	2012-02-09	Confalonieri et al.	
	7	20090172219	A1	2009-07-02	Mardiks	
	8	20090254705	A1	2009-10-08	Abali et al.	
	9	20100042655	A1	2010-02-18	Tse et al.	
	10	20100162012	A1	2010-06-24	Cornwell et al.	
	11	20100329011	A1	2010-12-30	Lee et al.	
	12	20110055445	A1	2011-03-03	Gee et al.	
	13	20110296133	A1	2011-10-20	Soran et al.	
	14	20100115172	A1	2010-05-06	Gillingham et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	15	20080147964	A1	2008-06-19	Chow et al.	
	16	20060022171	A1	2006-10-05	Maeda et al.	
	17	20090036163	A1	2009-02-05	Kimbrell	
	18	20090300015	A1	2009-12-03	Kazan et al.	
	19	20140215129	A1	2014-07-31	Kuzmin et al.	
	20	20070233939	A1	2007-10-04	Kim	
	21	20130166825	A1	2013-06-27	Kim et al.	
	22	20110238943	A1	2011-09-29	Devendran et al.	
	23	20090089490	A1	2009-04-02	Ozawa et al.	
	24	20030065866	A1	2003-04-03	Spencer	
	25	20090172250	A1	2009-07-02	Allen et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	26	20130007343	A1	2013-01-03	Rub	
	27	20140208004	A1	2014-07-24	Cohen	
	28	20120131381	A1	2012-05-24	Eleftheriou	
	29	20090083478	A1	2009-03-26	Kunimatsu	
	30	20120066441	A1	2012-03-15	Weingarten	
	31	20090271562	A1	2009-10-29	Sinclair	
	32	20110197023	A1	2011-08-11	Iwamitsu et al.	
	33	20130290619	A1	2013-10-31	Knight	
	34	20080155204	A1	2008-06-26	Qawami et al.	
	35	20090089482	A1	2009-04-02	Traister	
	36	20150324264	A1	2015-11-12	Vidypoornachy et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	37	20050144413	A1	2005-06-30	Kuo et al.	
	38	20070283428	A1	2007-12-06	Ma et al.	
	39	20090172257	A1	2009-07-02	Prins et al.	
	40	20100287217	A1	2010-11-11	Borchers et al.	
	41	20120059972	A1	2013-01-10	Chen	
	42	20130013852	A1	2013-01-10	Hou et al.	
	43	20130297852	A1	2013-11-07	Fai et al.	
	44	20140101371	A1	2014-04-10	Nguyen et al.	
	45	20150113203	A1	2015-04-23	Dancho et al.	
	46	20150261456	A1	2015-09-17	Alcantara et al.	
	47	20140122781	A1	2014-05-01	Smith et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	48	20130124793	A1	2013-05-16	Gyl et al.	
	49	20130019062	A1	2013-01-17	Bennett et al.	
	50	20100262761	A1	2010-10-14	Borchers et al.	
	51	20070058431	A1	2007-03-15	Chung et al.	
	52	20090198946	A1	2009-08-06	Ebata	
	53	20140189209	A1	2014-07-03	Sinclair et al.	
	54	20120204079	A1	2012-08-09	Takefman et al.	
	55	20070260811	A1	2007-11-08	Merry JR et al.	
	56	20080082596	A1	2008-04-03	Gorobets	
	57	20110161784	A1	2011-06-30	Sellinger et al.	
	58	20100161882	A1	2010-06-24	Stern et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	59	20150212938	A1	2015-06-30	Chen et al.	
	60	20130111295	A1	2013-05-02	Li et al.	
	61	20160018998	A1	2016-01-21	Mohan et al.	
	62	20160357462	A1	2016-12-15	Nam et al.	
	63	20130073816	A1	2013-03-21	Seo et al.	
	64	20150149789	A1	2015-05-28	Seo et al.	
	65	20160364179	A1	2016-12-15	Tsai et al.	
	66	20030028733	A1	2003-02-06	Tsunoda et al.	
	67	20100191779	A1	2010-01-29	Hinrichs	
	68	20090046533	A1	2009-02-19	Jo	
	69	20150347296	A1	2015-12-03	Kotte et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	70	20100241866	A1	2010-09-23	Rodorff	
	71	20130242425	A1	2013-09-19	Zayas et al.	
	72	20110314209	A1	2011-12-22	Eckstein	
	73	20100182838	A1	2010-07-22	Kim et al.	
	74	20170031699	A1	2017-02-02	Banerjee et al.	
	75	20150193148	A1	2015-07-09	Miwa et al.	
	76	20150134930	A1	2015-05-14	Huang et al.	
	77	20130124793	A1	2013-05-16	Gyl et al.	
	78	20140297949	A1	2014-10-02	Nagawaka	
	79	20150347041	A1	2015-12-03	Kotte et al.	
	80	20150067297	A1	2015-03-05	Arroyo et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	81	20160342509	A1	2016-11-24	Kotte et al.	
	82	20140189209	A1	2014-07-03	Sinclair et al.	
	83	20070046681	A1	2007-03-01	Nagashima	
	84	20140208062	A1	2014-07-24	Cohen	
	85	20120159039	A1	2012-06-21	Kegel et al.	
	86	20110276756	A1	2011-11-10	Bish et al.	
	87	20080147964	A1	2008-06-19	Chow et al.	
	88	20150324264	A1	2015-11-12	Chinnakkonda et al.	
	89	20160026564	A1	2016-01-28	Manning	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	1	2009100149	WO	A1	2009-08-13	Shaeffer et al.	entire doc.	

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T5
	1	NVM Express, Version 1.0b, July 12, 2011, pages 1-126, published at http://www.nvmexpress.org/resources/ by the NVM Express Work Group.	
	2	John D. Strunk, "Hybrid Aggregates: Combining SSDs and HDDs in a single storage pool," December 15, 2012, ACM SIGOPS Operating Systems Review archive, Volume 46 Issue 3, December 2012, Pages 50-56.	
	3	Yiying Zhang, Leo Prasath Arulraj, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Computer Sciences Department, University of Wisconsin-Madison, "De-indirection for Flash-based SSDs with NamelessWrites," published at https://www.usenix.org/conference/fast12/de-indirection-flash-based-ssds-nameless-writes, 2-7-2012, pages 1-16.	
	4	Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, and Vijayan Prabhakaran, "ResearchRemoving The Costs Of Indirection in Flash-based SSDs with NamelessWrites," 6-22-2010, pp. 1-5, published at www.cs.wisc.edu/wind/Publications/hotstorage10-nameless.pdf by Computer Sciences Department, University of Wisconsin-Madison and Microsoft Research.	
	5	Stan Park and Kai Shen, Department of Computer Science, University of Rochester, "FIOS: A Fair, Efficient Flash I/O Scheduler," 2-23-2012, pages 1-15, published at www.usenix.org/event/fast12/tech/full_papers/Park.pdf by the Advanced Computing Systems Association, Fast'12, 10th Usenix Conference On File And Storage Technologies, San Jose.	
	6	Eric Seppanen, Matthew T. O'Keefe, David J. Lilja, Department of Electrical and Computer Engineering, University of Minnesota, "High Performance Solid State Storage Under Linux," 4-10-2010, MSST '10 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), Pages 1-12	
	7	Xiangyong Ouyangyz, David Nellansy, Robert Wipfely, David Flynny, Dhabaleswar K. Pandaz, "Beyond Block I/O: Rethinking Traditional Storage Primitives," 8-20-2011, published at http://www.sciweavers.org/read/beyond-block-i-o-rethinking-traditional-storage-primitives-327868, by Fusion IO and the Ohio State University	
	8	Intel Corp, "PCI-SIG SR-IOV Primer - An Introduction to SR-IOV Technology,: 321211-002, Revision 2.5, January 2011, 28 pages	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	9	Open NAND Flash Interface (ONFI), specification, version 2.0, 174 pages, 02-27-2008	
	10	Open NAND Flash Interface (ONFI), specification, version 3.1, 296 pages 09-19-2012	
	11	NVM Express, V. 1.2.1, 217 pages, 06/03/2016	
	12	Garth Gibson, Greg Ganger, "Principles of Operation for Shingled Disk Devices," Canregie Mellon Parallel Data Laboratory, CMU-PDL-11-107, April 2011, 9 pages	
	13	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, ASPDAC 2008, 26 pages	
	14	Hua Wangx, Ping Huangxz, Shuang Hex, Ke Zhoux, Chunhua Lix, and Xubin He, "A Novel I/O Scheduler for SSD with Improved Performance and Lifetime," Mass Storage Systems and Technologies (MSST), 2013 IEEE 29th Symposium on, 6-10 May 2013, 5 pages.	
	15	Altera Corp. et al., "Hybrid Memory Cube" specification, 2012, 122 pages.	
	16	JEDEC Standard, JESD229, Wide IO, December 2011, 74 pages.	
	17	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, 978-1-4244-1922-7/08, 2008 IEEE, 6 pages	
	18	Optimizing NAND Flash Performance, Flash Memory Summit, Santa Clara, CA USA August 2008, Ryan Fisher, pp. 1-23.	
	19	High-Speed NAND Flash: Design Considerations to Maximize Performance, Flash Memory Summit, Santa Clara, CA USA August 11,2009, , Robert Pierce, pp. 1-19.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

	20	NAND 201: An Update on the Continued Evolution of NAND Flash, Jim Cooke, Micron White Paper, September 6, 2011, pp. 1-10.	
	21	Spansion SLC NAND Flash Memory for Embedded, data sheet, S34ML01G1, S34ML02G1, S34ML04G1, Sept. 6, 2012, pp. 1-73.	
	22	Wang et al., "An Efficient Design and Implementation of LSM-Tree based Key-Value Store on Open Channel SSD,"EuroSys '14 Proceedings of the Ninth European Conference on Computer Systems, Article No. 16, 2014-04-14, 14 pages.	
	23	Ouyang et al., "SDF: Software-defined flash for web-scale internet storage systems," Computer Architecture News - ASPLOS '14, Volume 42 Issue 1, March 2014, 14 pages.	
	24	Macko et al., "Tracking Back References in a Write-Anywhere File System,"FAST'10 Proceedings of the 8th USENIX conference on File and storage technologies, 14 pages, 2-23-2010	
	25	Ohad, Rodeh, "IBM Research Report Defragmentation Mechanisms for Copy-on-Write File-systems," IBM white paper, April 26, 2010, 10 pages, available at domino.watson.ibm.com/library/CyberDig.nsf/papers/298A0EF3C2CDB17B852577070056B41F/$File/rj10465.pdf	

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Andrey V. Kuzmin
	Art Unit	
	Examiner Name	unknown
	Attorney Docket Number	2020058 / RMS-02C1C1C14

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Marc P. Schuyler/	Date (YYYY-MM-DD)	2020-08-18
Name/Print	Marc P. Schuyler	Registration Number	35675

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.