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 #
17/877,881	**12/03/2024 07:50:09 PM Z ET**	**42754-53126/US**

Title of Invention

Storage System with Multiplane Segments and Cooperative Flash Management

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	11709772
CONFIRMATION #	3676	FILED BY	Janae Gallun
PATENT CENTER #	68205594	FILING DATE	07/29/2022
CUSTOMER #	758	FIRST NAMED INVENTOR	Andrey V. Kuzmin
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Michael Farn

Documents

TOTAL DOCUMENTS: 1

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
53126 US Change of Correspondence Address.pdf	2	Change of Address	266 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
53126 US Change of Correspondence Address.pdf	39F6442D7297C89D5D68D30DB7D8168E7270ADCE1D04555F7EB741BFA9CB08C0D37026314615744C0C1EAC89E30425245FF5FC22B0B1AB4EC9300AFC6E4A4A13

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/AIA/123 (08-12)
Approved for use through 09/30/2025. 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	11,709,772
Issue Date	July 25, 2023
Application Number	17/877,881
Filing Date	July 29, 2022
First Named Inventor	Andrey V. Kuzmin
Attorney Docket Number	42754-53126/US

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** TX	**ZIP** 75093

Country United States

Telephone	**Email** mjadon@radianmemory.com

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.

☐ If the Patentee was not the applicant for patent (37 CFR 1.42), then a Statement under 37 CFR 3.73(c) (Form PTO/AIA/96 or equivalent) is enclosed or was filed on _____. See 37 CFR 3.71.

☑ Attorney or agent of record. Registration Number 41,015.

☐ Patent practitioner acting in a representative capacity whose correspondence address is the correspondence address of record. Notice has been given to the patentee or owner. Registration Number 41,015.

Signature /Michael W. Farn/

Typed or Printed Name Michael W. Farn

Date Dec 3, 2024	Telephone 650-335-7823

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required, see below*.

☑ *Total of 1 forms are submitted.

A Federal agency may not conduct or sponsor, and a person is not required to respond to, nor shall a person be subject to a penalty for failure to comply with an information collection subject to the requirements of the Paperwork Reduction Act of 1995, unless the information collection has a currently valid OMB Control Number. The OMB Control Number for this information collection is 0651-00**. Public burden for this form is estimated to average ** hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the information collection. Send comments regarding this burden estimate or any other aspect of this information collection, including suggestions for reducing this burden to the Chief Administrative Officer, United States Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450 or email InformationCollection@uspto.gov. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **If filing this completed form by mail, 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. The United States Patent and Trademark Office (USPTO) collects the information in this record under authority of 35 U.S.C. 2. The USPTO's system of records is used to manage all applicant and owner information including name, citizenship, residence, post office address, and other information with respect to inventors and their legal representatives pertaining to the applicant's/owner's activities in connection with the invention for which a patent is sought or has been granted. The applicable Privacy Act System of Records Notice for the information collected in this form is COMMERCE/PAT-TM-7 Patent Application Files, available in the Federal Register at 78 FR 19243 (March 29, 2013). https://www.govinfo.gov/content/pkg/FR-2013-03-29/pdf/2013-07341.pdf

Routine uses of the information in this record may include disclosure to: 1) law enforcement, in the event that the system of records indicates a violation or potential violation of law; 2) a Federal, state, local, or international agency, in response to its request; 3) a contractor of the USPTO having need for the information in order to perform a contract; 4) the Department of Justice for determination of whether the Freedom of Information Act (FOIA) requires disclosure of the record; 5) a Member of Congress submitting a request involving an individual to whom the record pertains, when the individual has requested the Member's assistance with respect to the subject matter of the record; 6) a court, magistrate, or administrative tribunal, in the course of presenting evidence, including disclosures to opposing counsel in the course of settlement negotiations; 7) the Administrator, General Services Administration (GSA), or their designee, during an inspection of records conducted by GSA under authority of 44 U.S.C. 2904 and 2906, in accordance with the GSA regulations and any other relevant (i.e., GSA or Commerce) directive, where such disclosure shall not be used to make determinations about individuals; 8) 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)); 9) the Office of Personnel Management (OPM) for personnel research purposes; and 9) the Office of Management and Budget (OMB) for legislative coordination and clearance.

If you do not furnish the information requested on this form, the USPTO may not be able to process and/or examine your submission, which may result in termination of proceedings, abandonment of the application, and/or expiration of the patent.

Additional Uses

Additional USPTO uses of the information in this record may include disclosure to: 1) the International Bureau of the World Intellectual Property Organization, if the record is related to an international application filed under the Patent Cooperation Treaty; 2) the public i) after publication of the application pursuant to 35 U.S.C. 122(b), ii) after issuance of a patent pursuant to 35 U.S.C. 151, iii) 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, or iv) without publication of the application or patent under the specific circumstances provided for by 37 CFR 1.14(a)(1)(v)-(vii); and/or 3) the National Archives and Records Administration, for inspection of records.

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
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	216166

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 #
/Michael W. Farn/	Michael Farn	41015

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 07/25/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER	
BATAILLE, PIERRE MICHE	

ART UNIT	PAPER NUMBER
2136	

NOTIFICATION DATE	DELIVERY MODE
07/25/2023	ELECTRONIC

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.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PTOC@Fenwick.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.	ISSUE DATE	PATENT NO.
17/877,881	25-Jul-2023	11709772

FENWICK & WEST LLP
SILICON VALLEY CENTER
MOUNTAIN VIEW, CA 94041

EGRANT NOTIFICATION

Your electronic patent grant (eGrant) is now available, which can be accessed via Patent Center at https://patentcenter.uspto.gov

The electronic patent grant is the official patent grant under 35 U.S.C. 153. For more information, please visit https://www.uspto.gov/electronicgrants

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.
17/877,881	07/25/2023	11709772	39471-53126/US	3676

758 7590 07/05/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

ISSUE NOTIFICATION

The projected patent number and issue date are specified above. The patent will issue electronically. The electronically issued patent is the official patent grant pursuant to 35 U.S.C. § 153. The patent may be accessed on or after the issue date through Patent Center at https://patentcenter.uspto.gov/. The patent will be available in both the public and the private sides of Patent Center. Further assistance in electronically accessing the patent, or about Patent Center, is available by calling the Patent Electronic Business Center at 1-888-217-9197.

The USPTO is implementing electronic patent issuance with a transition period, during which period the USPTO will mail a ceremonial paper copy of the electronic patent grant to the correspondence address of record. Additional copies of the patent (i.e., certified and presentation copies) may be ordered for a fee from the USPTO's Certified Copy Center at https://certifiedcopycenter.uspto.gov/index.html. The Certified Copy Center may be reached at (800)972-6382.

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;
James G. Wayda, Laguna Niguel, CA;

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

Radian Memory Systems, Inc., Manhattan Beach, 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.

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 #
17/877,881	**06/06/2023 06:24:19 PM ET**	**39471-53126/US**

Title of Invention

Storage System with Multiplane Segments and Cooperative Flash Management

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	3676	FILED BY	Michael Farn
PATENT CENTER #	62220140	FILING DATE	07/29/2022
CUSTOMER #	758	FIRST NAMED INVENTOR	Andrey V. Kuzmin
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	-

Documents

TOTAL DOCUMENTS: 1

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
53126IF.pdf	1	Issue Fee Payment (PTO-85B)	171 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
53126IF.pdf	E57127A574D2B02DF6A2D11D0A9594B396B10E4A02EDB1FF7 E5E5B555926C09263E6EBDCC46CE2484BBFC77A808F232845 96D81B89E8D5D2398B4EEB6798BE5B

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.

UNITED STATES
PATENT AND TRADEMARK OFFICE

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

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/877,881	06/06/2023 06:24:19 PM ET	39471-53126/US

Title of Invention

Storage System with Multiplane Segments and Cooperative Flash Management

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	3676	FILED BY	Michael Farn
PATENT CENTER #	62220140	AUTHORIZED BY	-
CUSTOMER #	758	FILING DATE	07/29/2022
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Andrey V. Kuzmin

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
DA / 192555	E202366I25168902	Michael Farn

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
2501	UTILITY ISSUE FEE	480.00	1	480.00
			TOTAL AMOUNT:	$480.00

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.

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 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. **Because electronic patent issuance may occur shortly after issue fee payment, any desired continuing application should preferably be filed prior to payment of this issue fee in order not to jeopardize copendency.**

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

758 7590 05/17/2023
FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

TITLE OF INVENTION: Storage System with Multiplane Segments and Cooperative Flash Management

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0.00	$0.00	$480	08/17/2023

EXAMINER	ART UNIT	CLASS-SUBCLASS
BATAILLE, PIERRE MICHE	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 Fenwick & West LLP
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

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)

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

☒ Electronic Payment via Patent Center or 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. 19-2555

5. **Change in 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 to be a notification of loss of entitlement to micro entity status.

❑ Applicant changing to regular undiscounted fee 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 /Michael W. Farn/ Date June 6, 2023

Typed or printed name Michael W. Farn Registration No. 41,015

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

758 7590 05/17/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER
BATAILLE, PIERRE MICHE

ART UNIT	PAPER NUMBER
2136	

DATE MAILED: 05/17/2023

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

TITLE OF INVENTION: Storage System with Multiplane Segments and Cooperative Flash Management

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0.00	$0.00	$480	08/17/2023

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 40% the amount of undiscounted fees, and micro entity fees are 20% the amount of undiscounted 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.

PTOL-85 (Rev. 02/11)

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 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. **Because electronic patent issuance may occur shortly after issue fee payment, any desired continuing application should preferably be filed prior to payment of this issue fee in order not to jeopardize copendency.**

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

758 7590 05/17/2023
FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

TITLE OF INVENTION: Storage System with Multiplane Segments and Cooperative Flash Management

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0.00	$0.00	$480	08/17/2023

EXAMINER	ART UNIT	CLASS-SUBCLASS
BATAILLE, PIERRE MICHE	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)

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

❑ Electronic Payment via Patent Center or 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
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 to be a notification of loss of entitlement to micro entity status.

❑ Applicant changing to regular undiscounted fee 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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 05/17/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER	
BATAILLE, PIERRE MICHE	
ART UNIT	**PAPER NUMBER**
2136	

DATE MAILED: 05/17/2023

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.** 17/877,881	**Applicant(s)** Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	**Art Unit** 2136	**AIA (FITF) Status** Yes

-- 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 04/19/2023.
   ☐ 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 1-30 . 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 04/19/2023.
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 _____.

/PIERRE MICHEL BATAILLE/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 17/877,881	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	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	US-20110022784-A1	01-2011	Yano; Junji	G06F12/0246	711/E12.001
*	B	US-20060106972-A1	05-2006	Gorobets; Sergey A.	G06F12/0246	711/E12.008
*	C	US-9740439-B2	08-2017	Franceschini; Michele M.	G06F3/0619	1/1
	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.

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

CPC							
Symbol						Type	Version
G06F	/	12	/	0246		F	2013-01-01
G06F	/	16	/	1847		I	2019-01-01
G06F	/	11	/	1068		I	2013-01-01
G06F	/	3	/	0611		I	2013-01-01
G06F	/	12	/	10		I	2013-01-01
G06F	/	8	/	654		I	2018-02-01
G06F	/	3	/	0679		I	2013-01-01
G06F	/	3	/	0638		I	2013-01-01
G06F	/	3	/	0688		I	2013-01-01
G06F	/	11	/	1072		I	2013-01-01
G06F	/	2015	/	766		A	2013-01-01
G06F	/	2212	/	7208		A	2013-01-01
G06F	/	2212	/	2022		A	2013-01-01
G06F	/	2212	/	7207		A	2013-01-01

CPC Combination Sets								
Symbol					Type	Set	Ranking	Version
	/		/					

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	05 May 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office — Part of Paper No.: 20230505

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

INTERNATIONAL CLASSIFICATION					
CLAIMED					
G06F	/	12	/	02	
G06F	/	12	/	10	

NON-CLAIMED					
	/		/		

US ORIGINAL CLASSIFICATION	
CLASS	SUBCLASS

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

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	05 May 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office — Part of Paper No.: 20230505

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	**Art Unit**
	PIERRE MICHEL BATAILLE	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
2	1	11	10	20	19	29	28								
3	2	12	11	21	20	30	29								
4	3	13	12	22	21	1	30								
5	4	14	13	23	22										
6	5	15	14	24	23										
7	6	16	15	25	24										
8	7	17	16	26	25										
9	8	18	17	27	26										
10	9	19	18	28	27										

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	05 May 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office — Part of Paper No.: 20230505

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

CPC - Searched*		
Symbol	**Date**	**Examiner**
G06F12/1009, G06F12/0292, G06F9/45545, G06F12/1036, G06F2009/45583	09/26/2022	PMB
G06F3/0679, G06F3/0688, G06F12/10, G06F12/0246, G06F2212/2022, G06F2212/7208, G06F2212/7207, G06F3/0611, G06F3/0638	01/19/2023	PMB

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**
PALM INVENTOR NAME & ASSIGNEE SAEARCH	09/16/2022	PMB
KEYWORD & CONTEXT PE2E SEARCH	09/26/2022	PMB
NON-PATENT LITERATURE SEARCH	09/26/2022	PMB
NON-PATENT LITERATURE SEARCH	01/19/2023	PMB
KEYWORD & CONTEXT PE2E SEARCH	01/19/2023	PMB
KEYWORD & CONTEXT PE2E SEARCH	05/05/2023	PMB
NON-PATENT LITERATURE SEARCH	05/05/2023	PMB

/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

Interference Search			
US Class/CPC Symbol	**US Subclass/CPC Group**	**Date**	**Examiner**
	INTERFERENCE SEARCH INCLUDED IN PE2E SEARCH	01/19/2023	PMB
	INTERFERENCE SEARCH INCLUDED IN PE2E SEARCH	05/05/2023	PMB

/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	

1. A Demand-Based FTL Scheme Using Dualistic Approach on Data Blocks and Translation Blocks

IEEE XploreIEEE Conferences2011-08-0142011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications

2. A commitment-based management strategy for the performance and reliability enhancement of flash-memory storage systems

IEEE XploreIEEE Conferences2009-07-0142009 46th ACM/IEEE Design Automation Conference (Page(s): 858-863)

3. CAST: A page-level FTL with compact address mapping and parallel data blocks

IEEE XploreIEEE Conferences2012-12-0142012 IEEE 31st International Performance Computing and Communications Conference (IPCCC) (Page(s): 142-151)

4. ZFTL: A Zone-based Flash Translation Layer with a two-tier selective caching mechanism

IEEE XploreIEEE Conferences2012-11-0142012 IEEE 14th International Conference on Communication Technology (Page(s): 578-583)

5. A space-efficient caching mechanism for flash-memory address translation

IEEE XploreIEEE Conferences2006-01-014Ninth IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC'06) (Page(s): 8 pp.)

6. A Management Strategy for the Reliability and Performance Improvement of MLC-Based Flash-Memory Storage Systems

IEEE XploreIEEE Periodicals2011-01-274IEEE Transactions on Computers

7. Memory Controller Managed Backing of Superpages

IPCOM000033954DThe IP.com Journal2005-01-064The IP.com Prior Art Database

8. Performance comparison of logical-address-to-physical-address algorithms for non-volatile memory

IEEE XploreIEEE Conferences2003-01-014ASIC, 2003. Proceedings. 5th International Conference on

9. Fast erase algorithm using flash translation layer in NAND-type flash memory

IEEE XploreIEEE Periodicals2011-11-014IEEE Transactions on Consumer Electronics

10. An Efficient FTL Design for Multi-chipped Solid-State Drives

IEEE XploreIEEE Conferences2010-08-0142010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications (Page(s): 237-246)

11. Demand-based block-level address mapping in large-scale NAND flash storage systems

IEEE XploreIEEE Conferences2010-10-0142010 IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS) (Page(s): 173-182)

12. Ozone (O3): An Out-of-Order Flash Memory Controller Architecture

IEEE XploreIEEE Periodicals2010-10-074IEEE Transactions on Computers

13. A Re-configurable FTL (Flash Translation Layer) Architecture for NAND Flash based Applications

IEEE XploreIEEE Conferences2007-05-01418th IEEE/IFIP International Workshop on Rapid System Prototyping (RSP '07) (Page(s): 202-208)

14. High level modelling and performance evaluation of address mapping in NAND flash memory

IEEE XploreIEEE Conferences2009-12-0132009 16th IEEE International Conference on Electronics, Circuits and Systems - (ICECS 2009) (Page(s): 659-662)

15. Conference Record of the Twenty First IEEE Photovoltaic Specialists Conference - 1990 (Cat. No.90CH2838-1)

IEEE XploreIEEE Conferences1990-01-013IEEE Conference on Photovoltaic Specialists (Page(s): 0_1)

16. Design and Analysis of Flash Translation Layers for Multi-Channel NAND Flash-based Storage Devices

IEEE XploreIEEE Periodicals2009-08-013IEEE Transactions on Consumer Electronics

17. Design of a one-transistor-cell multiple-valued CAM

IEEE XploreIEEE Periodicals1996-11-013IEEE Journal of Solid-State Circuits

18. A Workload-Aware Adaptive Hybrid Flash Translation Layer with an Efficient Caching Strategy

IEEE XploreIEEE Conferences2011-07-0132011 IEEE 19th Annual International Symposium on Modelling, Analysis, and Simulation of Computer and Telecommunication Systems (Page(s): 248-255)

19. FLASH modelling for wearleveling algorithms

IEEE XploreIEEE Conferences2011-12-0138th International Conference on High-capacity Optical Networks and Emerging Technologies (Page(s): 267-272)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	1	of	3	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A1	US-11,544,200	01-03-2023	Lercari et al.
	A2	US-11,537,529	12-27-2022	Lercari et al.
	A3	US-11,537,528	12-27-2022	Lercari et al.
	A4	US-11,487,657	11-01-2022	Kuzmin et al.
	A5	US-11,487,656	11-01-2022	Kuzmin et al.
	A6	US-11,449,436	09-20-2022	Lercari et al.
	A7	US-11,416,413	08-16-2022	Lercari et al.
	A8	US-9,489,296	11-08-2016	Tomlin et al.
	A9	US-9,378,149	06-28-2016	Bonwick
	A10	US-2017/0075620 A1	03-16-2017	YAMAMOTO et al.
	A11	US-2016/0162186 A1	06-09-2016	Kashyap et al.
	A12	US-2016/0070496 A1	03-10-2016	Cohen et al.
	A13	US-2016/0011818 A1	01-14-2016	HASHIMOTO et al.
	A14	US-2013/0024460 A1	01-24-2013	Peterson et al.
	A15	US-2013/0019057 A1	01-17-2013	Stephens
	A16	US-2012/0260150 A1	10-11-2012	Cideciyan et al.
	A17	US-2011/0161560 A1	06-30-2011	Hutchison et al.
	A18	US-2011/0082967 A1	04-07-2011	Deshkar et al.
	A19	US-2010/0250837 A1	09-30-2010	Schmidberger et al.
	A20	US-2010/0077136 A1	03-25-2010	Ware et al.
	A21	US-2009/0240903 A1	09-24-2009	Sauber et al.
	A22	US-2009/0182964 A1	07-16-2009	GREINER et al.
	A23	US-2008/0201517 A1	08-21-2008	Kuhne
	A24	US-2005/0141313 A1	06-30-2005	Gorobets et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	2	of	3	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A25	US-2005/0073884 A1	04-07-2005	Gonzalez et al.
	A26	US-2004/0083335 A1	04-29-2004	Gonzalez et al.

FOREIGN PATENT DOCUMENTS					
		Foreign Patent Document			
Examiner Initials*	Cite No.[1]	Country Code[3] – Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	T[6]
	B1	WO 2011/055407 A1	05-12-2011	HITACHI LTD	
	B2	WO 2011/044154 A1	04-14-2011	MARVELL SEMICONDUCTOR INC	

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C1	United States Patent Application No. 17/571,394, filed January 7, 2022, Inventors Andrey V. Kuzmin and James G. Wayda (copy not enclosed).	

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	05/05/2023

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	3	of	3	Attorney Docket Number	39471-53126/US

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).

☐ This application relies, under 35 U.S.C. § 120, on the earlier filing date of prior application No. __, filed __, and the references cited therein are hereby referenced, but are not required to be provided in this application under 37 CFR § 1.98(d).

☐ 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	/Michael W. Farn/	Date (YYYY-MM-DD)	2023-04-19
Name/Print	Michael W. Farn	Registration Number	41,015

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L5	243	(map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ AND (map$3 OR lookup OR look?up OR translat$) near3 address$2 AND (chang$ OR updat$) near4 address$ AND address$ WITH (preserv$ OR reserv$ OR sav$3)	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 09:05 PM
L6	21	L5 AND (chang$ OR preserv$) near5 page near3 address	(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	2023/05/04 09:06 PM
L7	175	(map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ AND (map$3 OR lookup OR look?up OR translat$) near3 address$2 AND (chang$ OR updat$) near4 address$ AND address$ WITH (preserv$ OR reserv$ OR sav$3)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 09:06 PM
L8	18	L7 AND (chang$ OR preserv$) near5 page near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 09:15 PM
L9	12	(translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ WITH (preserv$ OR reserv$ OR sav$3)	(USOCR; FIT (AP, AT, AU, 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	2023/05/04 10:14 PM

			MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L10	8	(translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ WITH (preserv$ OR reserv$ OR sav$3)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 10:14 PM
L11	126	(map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ near5 address$2 AND address$ WITH (preserv$ OR reserv$ OR sav$3)	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 10:18 PM
L12	7	L11 AND (chang$ OR preserv$) near5 page near3 address	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 10:18 PM
L13	0	((map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ near5 address$2 AND address$ WITH (preserv$ OR reserv$ OR sav$3)).pn.	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN,	OR	ON	ON	2023/05/04 10:19 PM

			WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L14	144	(map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ near5 address$2 AND address$ WITH (preserv$ OR reserv$ OR sav$3)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 10:21 PM
L15	2761	L14AND (chang$ OR preserv$) near5 page near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 10:22 PM
L16	37	L14 AND (chang$ OR preserv$) near5 page near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 10:22 PM
L18	160450	(G06F12/0246 G06F12/10 G06F12/02 G06F3/0611 G06F3/0679 G06F3/0638).CPC.	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 11:05 PM
L19	5	L18 AND (translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ WITH (preserv$ OR reserv$) near5 page near3 address$	(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	2023/05/04 11:07 PM
L20	0	L18 AND (translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ WITH (preserv$ OR reserv$) near5 page near3 address$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:07 PM
L21	9	L18 AND (map$4 OR	(USOCR; FIT (AP, AT,	OR	ON	ON	2023/05/04

		translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ AND (preserv$ OR reserv$) near5 page near3 address$	AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				11:08 PM
L22	36	L18 AND (map$4 OR translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ AND (preserv$ OR reserv$) near5 page near3 address$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:08 PM
L23	364	L18 AND (maintain$ OR preserv$) WITH (block OR page) near3 address$ WITH eras$ near5 (page OR block)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:12 PM
L24	79	L23 AND (translat$ OR map$4 OR lookup OR look?up) near3 table WITH eras$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:13 PM
L25	130	L18 AND (maintain$ OR preserv$) WITH (block OR page) near3 address$ WITH eras$ near5 (page OR block)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 11:13 PM
L26	37	L25 AND (translat$ OR map$4 OR lookup OR look?up) near3 table WITH eras$	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL,	OR	ON	ON	2023/05/04 11:13 PM

			PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L27	0	L18 AND (preserv$ near3 page near3 order$ AND (preserv$ OR maintain$) near5 page near3 address)	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 11:23 PM
L28	45	L18 AND (preserv$ near3 page near3 order$ AND (preserv$ OR maintain$) near5 page near3 address)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:23 PM
L29	0	(preserv$ near3 page near3 order$ AND (preserv$ OR maintain$) near5 page near3 address)	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/04 11:24 PM
L30	48	(preserv$ near3 page near3 order$ AND (preserv$ OR maintain$) near5 page near3 address)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:24 PM
L31	5	(preserv$ OR maintain$) near3 page near3 order$ WITH (relocat$ OR backup)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE,	OR	ON	ON	2023/05/04 11:26 PM

			SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L32	14	(preserv$ OR maintain$) near3 page near3 order$ WITH (relocat$ OR backup)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:26 PM
L33	28	("20100077136" "20100125694" "8838937" "8745315" "9251067" "20080270680" "8041884" "20140101369" "9489296" "9268682" "20080177937" "20100030999" "8321652" "8745310" "20140136753" "9507523" "8954694" "20120203958" "7949851" "20090172345" "20070276988" "8909851" "7680977" "20070143569" "7644251" "20100250837" "20110145488" "20120297117").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:28 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	1	((translat$ OR lookup OR look?up) near3 table WITH eras$ near5 address$2 AND page near2 address$ WITH (preserv$ OR reserv$ OR sav$3)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 08:48 PM
N2	2	((map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ near5 address$2 AND address$ WITH (preserv$ OR reserv$ OR sav$3)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 08:54 PM
N3	1	((map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ AND (map$3 OR lookup OR look?up OR	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 09:03 PM

		translat$) near3 address$2 AND (chang$ OR updat$) near4 address$ AND address$ WITH (preserv$ OR reserv$ OR sav$3)).clm.					
N4	2	((map$3 OR lookup OR look?up OR translat$) near3 table WITH eras$ near5 address$2 AND address$ WITH (preserv$ OR reserv$ OR sav$3)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 10:20 PM
N5	0	(preserv$ near3 page near3 order$ AND (preserv$ OR maintain$) near5 page near3 address).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:22 PM
N6	0	((preserv$ OR maintain$) near3 page near3 order$ WITH (relocat$ OR backup)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/04 11:26 PM

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	1	cache NEAR node AND snoop$ near3 (response OR message) AND identifier WITH prefetch$	(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	2023/01/20 11:34 AM
L2	3	cache NEAR node AND snoop$ near3 (response OR message) AND identifier WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 11:35 AM
L4	1	cache near3 node AND snooping near3 message AND prefetch$ near3 (flag OR tag OR identif$) AND prefetch$ near3 operation near5 (need$ OR necess$)	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/20 01:38 PM
L5	1	cache near3 node AND snooping near3 message AND prefetch$ near3 (flag OR tag OR identif$) AND prefetch$ near3 operation near5 (need$ OR necess$)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:38 PM
L6	2	snoop$ near3 message AND prefetch$ AND identifier AND writ$3 AND node WITH prefetch$	(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,	OR	ON	ON	2023/01/20 01:41 PM

			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)				
L7	2	snoop$ near3 message AND prefetch$ AND identifier AND writ$3 AND node WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:41 PM
L8	33	snoop$ near3 (response OR message) AND prefetch$ AND identif$ AND node WITH prefetch$	(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	2023/01/20 01:43 PM
L9	47	snoop$ near3 (response OR message) AND prefetch$ AND identif$ AND node WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:44 PM
L10	73561	(G06F12/0862 G06F12/0238 G06F12/0813 G06F12/0811 G06F12/02 G06F12/08).CPC.	(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	2023/01/20 01:46 PM
L11	1	L10 AND prefetch$ near3 identif$ AND prefetch$ near5 node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5	(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	2023/01/20 01:53 PM

		(response OR result OR message)	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)				
L12	24	L10 AND prefetch$ AND identif$ AND prefetch$ WITH node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(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	2023/01/20 01:53 PM
L13	4	L10 AND prefetch$ near3 identif$ AND prefetch$ near5 node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:54 PM
L14	21	L10 AND prefetch$ AND identif$ AND prefetch$ WITH node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:54 PM
L15	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page) AND (indicat$ OR identi$) AND snoop$3 near4 (response OR result OR message)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/20 02:03 PM

L16	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page) AND (indicat$ OR identi$) AND snoop$3 near4 (response OR result OR message)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 02:03 PM
L17	5	prefetch$ WITH to?be?writ$ near4 (line OR data OR page)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/20 02:04 PM
L18	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 02:04 PM
L21	1	cache NEAR node AND snoop$ near3 (response OR message) AND identifier WITH prefetch$	(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	2023/05/05 11:15 PM
L22	3	cache NEAR node AND snoop$ near3 (response OR message) AND identifier WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L23	1	cache near3 node AND snooping near3 message AND prefetch$ near3 (flag OR tag OR identif$) AND prefetch$ near3 operation near5 (need$ OR necess$)	(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	2023/05/05 11:15 PM

			RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L24	1	cache near3 node AND snooping near3 message AND prefetch$ near3 (flag OR tag OR identif$) AND prefetch$ near3 operation near5 (need$ OR necess$)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L25	2	snoop$ near3 message AND prefetch$ AND identifier AND writ$3 AND node WITH prefetch$	(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	2023/05/05 11:15 PM
L26	2	snoop$ near3 message AND prefetch$ AND identifier AND writ$3 AND node WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L27	33	snoop$ near3 (response OR message) AND prefetch$ AND identif$ AND node WITH prefetch$	(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	2023/05/05 11:15 PM
L28	49	snoop$ near3 (response OR message) AND prefetch$ AND identif$ AND node WITH prefetch$	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L29	75130	(G06F12/0862 G06F12/0238	(US-PGPUB; USPAT; USOCR; FIT (AU, AP,	OR	ON	ON	2023/05/05 11:15 PM

		G06F12/0813 G06F12/0811 G06F12/02 G06F12/08).CPC.	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)				
L30	1	L10 AND prefetch$ near3 identif$ AND prefetch$ near5 node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(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	2023/05/05 11:15 PM
L31	24	L10 AND prefetch$ AND identif$ AND prefetch$ WITH node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(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	2023/05/05 11:15 PM
L32	4	L10 AND prefetch$ near3 identif$ AND prefetch$ near5 node AND prefetch$ near3 (result$ OR message OR response) AND snoop$ near5 (response OR result OR message)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L33	23	L10 AND prefetch$ AND identif$ AND prefetch$ WITH node AND prefetch$ near3	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM

		(result$ OR message OR response) AND snoop$ near5 (response OR result OR message)					
L34	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page) AND (indicat$ OR identi$) AND snoop$3 near4 (response OR result OR message)	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/05 11:15 PM
L35	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page) AND (indicat$ OR identi$) AND snoop$3 near4 (response OR result OR message)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L36	5	prefetch$ WITH to?be?writ$ near4 (line OR data OR page)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/05/05 11:15 PM
L37	1	prefetch$ WITH to?be?writ$ near4 (line OR data OR page)	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:15 PM
L38	27	("0000002"|"11416413"|"11449436"|"11487656"|"11487657"|"11537528"|"11537529"|"11544200"|"20040083335"|"20050073884"|"20050141313"|"20080201517"|"20090182964"|"20090240903"|"20100077136"|"20100250837"|"20110082967"|"20110161560"|	(US-PGPUB; USPAT)	OR	ON	ON	2023/05/05 11:25 PM

		20120260150"|"20130019057"|"20130024460"|"20160011818"|"20160070496"|"20160162186"|"20170075620"|"9378149"|"9489296").PN.					

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	1	((cluster OR cach$3 near2 node) AND snoop$ near3 (response OR message) AND identifier WITH prefetch$).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:36 PM
N2	1	(cache near3 node AND snooping near3 message AND prefetch$ near3 (flag OR tag OR identif$) AND prefetch$ near3 operation near5 (need$ OR necess$)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:38 PM
N3	2	(snoop$ near3 message AND prefetch$ AND identifier AND writ$3 AND node WITH prefetch$).clm	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:41 PM
N4	3	(snoop$ near3 (response OR message) AND prefetch$ AND identif$ AND node WITH prefetch$).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/20 01:44 PM

Doc code: RCEX
Doc description: Request for Continued Examination (RCE)

PTO/SB/30EFS (07-09)
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.

REQUEST FOR CONTINUED EXAMINATION(RCE)TRANSMITTAL
(Submitted Only via EFS-Web)

Application Number	17/877,881	Filing Date	2022-07-29	Docket Number (if applicable)	39471-53126/US	Art Unit	2136
First Named Inventor	Andrey V. Kuzmin			Examiner Name	BATAILLE, PIERRE MICHE		

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 192555

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

☒ Patent Practitioner Signature

Applicant Signature

PTO/SB/30EFS (07-09)
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.

Signature of Registered U.S. Patent Practitioner			
Signature	/Michael W. Farn/	Date (YYYY-MM-DD)	2023-04-19
Name	Michael W. Farn	Registration Number	41015

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.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	1	of	3	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A1	US-11,544,200	01-03-2023	Lercari et al.
	A2	US-11,537,529	12-27-2022	Lercari et al.
	A3	US-11,537,528	12-27-2022	Lercari et al.
	A4	US-11,487,657	11-01-2022	Kuzmin et al.
	A5	US-11,487,656	11-01-2022	Kuzmin et al.
	A6	US-11,449,436	09-20-2022	Lercari et al.
	A7	US-11,416,413	08-16-2022	Lercari et al.
	A8	US-9,489,296	11-08-2016	Tomlin et al.
	A9	US-9,378,149	06-28-2016	Bonwick
	A10	US-2017/0075620 A1	03-16-2017	YAMAMOTO et al.
	A11	US-2016/0162186 A1	06-09-2016	Kashyap et al.
	A12	US-2016/0070496 A1	03-10-2016	Cohen et al.
	A13	US-2016/0011818 A1	01-14-2016	HASHIMOTO et al.
	A14	US-2013/0024460 A1	01-24-2013	Peterson et al.
	A15	US-2013/0019057 A1	01-17-2013	Stephens
	A16	US-2012/0260150 A1	10-11-2012	Cideciyan et al.
	A17	US-2011/0161560 A1	06-30-2011	Hutchison et al.
	A18	US-2011/0082967 A1	04-07-2011	Deshkar et al.
	A19	US-2010/0250837 A1	09-30-2010	Schmidberger et al.
	A20	US-2010/0077136 A1	03-25-2010	Ware et al.
	A21	US-2009/0240903 A1	09-24-2009	Sauber et al.
	A22	US-2009/0182964 A1	07-16-2009	GREINER et al.
	A23	US-2008/0201517 A1	08-21-2008	Kuhne
	A24	US-2005/0141313 A1	06-30-2005	Gorobets et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	2	of	3	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A25	US-2005/0073884 A1	04-07-2005	Gonzalez et al.
	A26	US-2004/0083335 A1	04-29-2004	Gonzalez et al.

FOREIGN PATENT DOCUMENTS					
		Foreign Patent Document			
Examiner Initials*	Cite No.[1]	Country Code[3] – Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	T[6]
	B1	WO 2011/055407 A1	05-12-2011	HITACHI LTD	
	B2	WO 2011/044154 A1	04-14-2011	MARVELL SEMICONDUCTOR INC	

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C1	United States Patent Application No. 17/571,394, filed January 7, 2022, Inventors Andrey V. Kuzmin and James G. Wayda (copy not enclosed).	

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Andrey V. Kuzmin
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2136
				Confirmation Number	3676
				Examiner Name	BATAILLE, PIERRE MICHE
Sheet	3	of	3	Attorney Docket Number	39471-53126/US

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).

☐ This application relies, under 35 U.S.C. § 120, on the earlier filing date of prior application No. __, filed __, and the references cited therein are hereby referenced, but are not required to be provided in this application under 37 CFR § 1.98(d).

☐ 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	/Michael W. Farn/	Date (YYYY-MM-DD)	2023-04-19
Name/Print	Michael W. Farn	Registration Number	41,015

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 May 2011 (12.05.2011)

PCT

(10) International Publication Number
WO 2011/055407 A1

(51) International Patent Classification:
G06F 12/02 (2006.01)

(21) International Application Number:
PCT/JP2009/005859

(22) International Filing Date:
4 November 2009 (04.11.2009)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: **HITACHI, LTD.** [JP/JP]; 6-6, Marunouchi 1-chome, Chiyoda-ku, Tokyo, 1008280 (JP).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **TAKADA, Masanori** [JP/JP]; c/o Hitachi, Ltd., Systems Development Laboratory, 1099, Ohzenji, Asao-ku, Kawasaki-shi, Kanagawa, 2150013 (JP). **KITAHARA, Jun** [JP/JP]; c/o Hitachi, Ltd., Systems Development Laboratory, 1099, Ohzenji, Asao-ku, Kawasaki-shi, Kanagawa, 2150013 (JP).

(74) Agent: **WILLFORT INTERNATIONAL**; Kanda Center Bldg. 5F, 3-2, Kajicho 2-chome, Chiyoda-ku, Tokyo, 1010044 (JP).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: FLASH MEMORY MODULE

[Fig. 6]

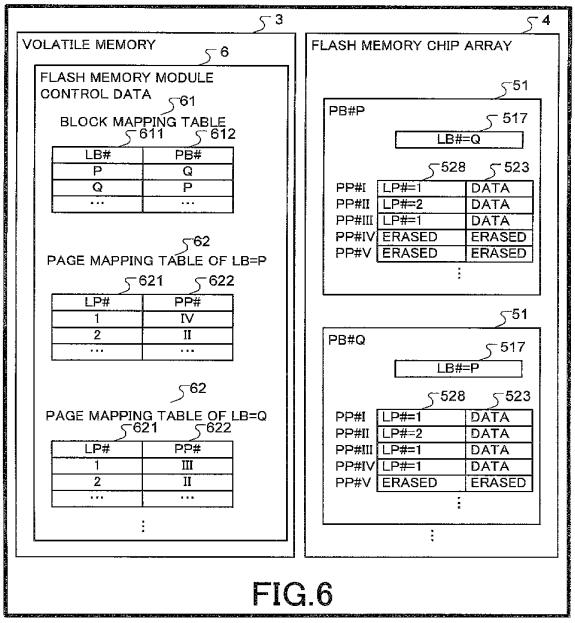


FIG.6

(57) Abstract: Logical/physical conversion information is configured from first conversion information and second conversion information. A controller of a flash memory module restores the first conversion information at boot up time, enables an access command to be received from a host after having restored the first conversion information, and restores the second conversion information after an access command is able to be received.

WO 2011/055407 A1

Published:

— *with international search report (Art. 21(3))*

Description

Title of Invention: FLASH MEMORY MODULE

Technical Field

[0001] The present invention generally relates to a module comprising a flash memory.

Background Art

[0002] Lower-cost, higher-capacity flash memory is coming into widespread use. Thanks to outstanding performance and power consumption features, flash memory use is spreading especially rapidly in modules that place emphasis on these features (for example, storage systems and compact mobile modules).

[0003] A flash memory comprises a plurality of flash memory chips (hereinafter, chips), and each chip comprises a plurality of physical blocks. A physical block is comprised from a plurality of physical pages. The physical block and the physical page are both physical storage areas. For example, the reading and writing of data are carried out in page units, and the erasing of data is carried out in block units.

[0004] The flash memory, in particular, the large-capacity NAND-type flash memory has the following two characteristic features.

[0005] The first is the fact that it is not possible to write new data over data that has already been written to a physical page, and in order to write new data to this physical page, it is necessary to carry out an erase process for the physical block comprising this physical page. Data of a fixed size is written to the erased physical block in order from the start of this block. As described above, the flash memory is limited as to the size and order of a read and a write.

[0006] The other is a wear-out phenomenon. The wear-out phenomenon is one in which the flash memory becomes unable to hold information after erase processing has been carried out more that a certain number of times. The number of erasures is more apt to reach the limit in a case where writes are carried out frequently. In a case where writes concentrate on a specific area, the physical blocks comprising this area reach the end of their service life early on.

[0007] With respect to these problems, for example, Patent Literature 1 discloses a technology in which a controller for controlling the flash memory conceals the size and order limitations of reads and writes. In accordance with Patent Literature 1, an area for appending data is disposed inside the flash memory. Also, for example, Patent Literature 2 discloses a technology for extending the service life of the flash memory by dispersing write destinations.

[0008] These technologies require information (logical/physical conversion information) for making a logical address used by a host to specify an access destination correspond to

a storage-destination flash memory address (physical address). Failure to make this logical/physical conversion data nonvolatile the same as the data being written/read to/from the host will make it impossible to store the data correctly. For example, Patent Literature 3 discloses a technology for storing this logical/physical conversion information together with a user data, and restoring the original logical/physical conversion information from the stored data.

Citation List

Patent Literature

[0009] PTL 1: Japanese Patent Application Laid-open No. 2009-064251
PTL 2: Japanese Patent Application Laid-open No. 2007-265265
PTL 3: Japanese Patent Application Laid-open No. H8-077074

Summary of Invention

Technical Problem

[0010] Ordinarily, a flash memory module boots up when the power is turned ON, and restores the logical/physical conversion information at boot up time. A management information unit is required to restore the logical/physical conversion information. The management information unit, for example, is an information unit that is written into the overhead area of the physical page, and denotes which physical block comprises this physical page, and the logical page inside the logical block to which this physical page corresponds.

[0011] Although the logical/physical conversion information is able to be restored in accordance with the above-mentioned technology, all the management information units must be read in order to perform restoration. For this reason, the restoration of the logical/physical conversion information takes a long time. For example, in a case where the flash memory has a capacity of 256 GB (gigabytes) and the size of the physical page is 4 kB (kilobytes), each management information unit from 64 M (mega) (256 GB divided by 4 kB) of physical pages must be read out. Since the time required for readout is normally 100 microseconds per physical page, the restoration of the logical/physical conversion information requires around 6,400 seconds. This is an extremely long time compared to the time required to boot up a hard disk (for example, around 10 seconds). As flash memory capacity increases, the time required for booting up will most likely increase proportionally.

[0012] An object of the present invention is to reduce the time required to boot up a flash memory module.

Solution to Problem

[0013] The logical/physical conversion information comprises a first conversion information and a second conversion information. A flash memory module controller restores the

first conversion information at boot up time, making the flash memory ready to receive an access command from the host after the first conversion information has been restored, and after the flash memory is ready to receive an access command, the controller restores the second conversion information.

Advantageous Effects of Invention

[0014] The present invention makes it possible to reduce the time required to boot up the flash memory module.

Brief Description of Drawings

[0015] [fig.1]Fig. 1 shows the configuration of a flash memory module related to a first embodiment of the present invention.

[fig.2]Fig. 2 shows the internal configuration of a chip 5.

[fig.3A]Fig. 3A shows the status of a physical block #p in the initialized state.

[fig.3B]Fig. 3B shows an example of the state subsequent to new data being written to the physical block #p.

[fig.3C]Fig. 3C shows the state of the physical block #p subsequent to a merge operation being performed.

[fig.4]Fig. 4 shows the configuration of a flash memory module control data 6.

[fig.5]Fig. 5 shows the configuration of a module control function 7.

[fig.6]Fig. 6 shows an example of the corresponding relationship between tables 61 and 62 inside a volatile memory 3 and the state of a physical block inside a chip array 4.

[fig.7]Fig. 7 shows an example of the configuration of a block mapping table 61.

[fig.8]Fig. 8 shows an example of the configuration of a page mapping table 62.

[fig.9]Fig. 9 shows an example of a scheme for storing data in a physical block 51.

[fig.10]Fig. 10 shows a detailed example of the configuration of a physical page 52.

[fig.11]Fig. 11 shows an example of the configuration of a block metadata 515.

[fig.12]Fig. 12 shows an example of the configuration of a page metadata 526.

[fig.13]Fig. 13 shows an example of the flow of processing carried out by a host read service function 71.

[fig.14]Fig. 14 shows an example of the flow of processing carried out by a host write service function 72.

[fig.15]Fig. 15 shows an example of the flow of processing carried out by a page mapping table reconstruct function 74.

[fig.16]Fig. 16 shows an example of the flow of processing carried out by a power ON function 75.

[fig.17]Fig. 17 shows an example of the flow of processing carried out by an initialize function 76.

[fig.18]Fig. 18 shows an example of an exchange between a controller 2 and the chip array 4 in S726 of the host write service function 72.

[fig.19]Fig. 19 shows an example of the exchanges between the controller 2 and the chip array 4 in S728 through S730 of the host write service function.

[fig.20]Fig. 20 shows an example of the exchanges between the controller 2 and the chip array 4 in S752 of the power ON function 75 and in S743 of the page mapping table reconstruct function 74.

[fig.21]Fig. 21 shows an example of the corresponding relationship between a logical address and a logical block/logical page in the first embodiment of the present invention.

Description of Embodiments

[0016] A flash memory module in which a flash memory module related to the first embodiment of the present invention is used will be explained below by referring to the attached drawings. However, it should be noted that this embodiment is merely an example for realizing the present invention, and does not limit the technical scope of the present invention. Also, the same reference signs will be assigned to common configurations in the respective drawings.

[0017] Fig. 1 shows the configuration of a flash memory module related to the first embodiment of the present invention.

[0018] The flash memory module 1 comprises a flash memory controller (hereinafter, controller) 2, a volatile memory 3, a chip array (one or more flash memories) 4, a flash memory bus 8, and a host interface 9. The flash memory module 1, for example, may be the same shape as a 3.5-inch or 2.5-inch hard disk drive, or may be larger or smaller than same.

[0019] The chip array 4 comprises a plurality of flash memory chips (hereinafter, chip) 5.

[0020] The controller 2 controls the chip array 4, and also controls communications with a host 11. For example, the controller 2 comprises a processor for executing a flash memory control software (a computer program), a memory controller for writing/reading data to/from the chip 5, and a host interface controller for exchanging data with the host 11. In Fig. 1, the controller 2 comprises a single LSI (Large Scale Integration), but, for example, the controller 2 may also be comprised of a plurality of IC (Integrated Circuit) chips, such as an externally connected processor. The flash memory module control function (hereinafter, module control function) 7 of the controller 2 will be explained below.

[0021] The volatile memory 3 is used for storing, from among the information handled by the controller 2, the information required for rapid access. Specifically, for example, the volatile memory 3 stores the above-mentioned flash memory control software, and

a flash memory module control data (hereinafter, module control data) 6 used by this software. A detailed explanation of the module control data 6 will be given below. The volatile memory 3, for example, is a DRAM (Dynamic Random Access Memory). Furthermore, another type of storage medium may be used instead of a volatile memory 3 as long as it is capable of high-speed access.

[0022] The chip 5 is used for storing information written/read to/from the host 11. Data that should be nonvolatile is also stored from among the information required for control. This will be explained in detail below.

[0023] The flash memory bus 8 is used for connecting the controller 2 and the chip 5. In Fig. 1, a plurality of chips 5 are connected to one flash memory bus 8, but another connection scheme may be used.

[0024] The host interface 9 is used to connect the flash memory module 1 to the host 11, and to carry out the exchange of information. For example, a SATA (Serial ATA), SAS (Serial Attached SCSI), FC (Fibre Channel) or other such storage interface may be used as the host interface 9, and a PCI-Express or other such I/O-type interface may also be used.

[0025] The flash memory module 1 may be provided in the storage system. In this case, the storage system comprises a plurality of flash memory modules 1. A RAID group is comprised of two or more memory modules 1. A logical volume is created based on the RAID group. In accordance with this, the host 11 may be a controller (hereinafter, storage controller) of the storage system, or may be a host computer that is connected to the storage system. In the case of the former, the storage controller is able to send an access-destination logical address that accords with an I/O command from the host computer to the flash memory module 1 that manages this logical address, and to access the physical block that corresponds to this logical address.

[0026] Further, the flash memory module 1 may be a SSD (Solid State Drive) or a portable flash memory module. In this case, the host 11 may be a computer.

[0027] Fig. 2 shows the internal configuration of a chip 5. Furthermore, in Fig. 2 and the subsequent drawings, the physical block is abbreviated as "PB", the physical page is abbreviated as "PP", the logical block is abbreviated as "LB" and the logical page is abbreviated as "LP".

[0028] The chip 5 is comprised of a plurality of physical blocks 51. The physical block 51 is comprised of a plurality of physical pages 52.

[0029] The chip 5 has three interfaces, read, write and erase. Specifically, these are as follows.

[0030] (Read) It is possible to read in data from either an entire physical page 52, or a portion of a physical page 52.

[0031] (Write) It is possible to write data to an empty physical page (a physical page, which

has been erased, and on which nothing is written) 52 in order from the start of the physical block 51. Specifically, for example, for physical block 51, in which data is only written on physical page #1, a write is possible to physical page #2, but it is not possible to overwrite the data in physical page #1. It is also not possible to skip physical page #2 and write data to physical page #3.

[0032] (Erase) It is possible to erase data in physical block 51 units. That is, it is not possible to erase only one part inside the physical block 51, and make that part an unwritten state (that is, make it an empty block).

[0033] Normally, the size of the physical block, the size of the physical page, and the size of the read/write unit from the host 11 will differ. For example, as shown in Fig. 21, the physical block is 256 kB, the physical page is 2 kB, and the size of the unit being written/read to/from the host 11 (the access size) is 512 B. That is, a logical block is allocated for each range of logical addresses capable of being specified from the host 11, and a physical block is allocated to the logical block. One or more physical blocks may be allocated to one logical block, but in this embodiment, it is supposed that one physical block is allocated to one logical block. The physical block is configured from a plurality of physical pages. A physical page inside a physical block allocated to a logical block is allocated to each logical page configuring the logical block. The controller 2 is able to specify the logical page of the access destination by performing a predetermined operation based on the logical address specified in the access command from the host 11, and is able to specify the physical page corresponding to the access-destination logical page based on the module control data 6.

[0034] For this reason, a mechanism for making up the difference between the size of the access from the host 11 and the size of the write unit/erase unit (page/block) of the chip 5 is needed for a write.

[0035] Figs. 3A through 3C show examples of the controller 2 operations for concealing the difference between the chip 5 page/block size and the size of the access from the host 11. Furthermore, in Figs. 3A through 3C, a physical block 51 having p as the physical block number (PB#) is given as an example, but the configuration is the same for another physical block 51 as well. The physical block 51 with PB# of p will be referred to as "physical block #p" below.

[0036] A plurality of physical pages 52 inside the physical block #p are divided into a master area and an append area. That is, there are physical pages 52 that belong to the master area and physical pages 52 that belong to the append area in the physical block #p. The data written to each physical page 52 is managed using a logical page number (LP#).

[0037] Fig. 3A shows the status of the physical block #p in the initialized state.

[0038] In the initialized state, the data from a logical page #1 (the logical page having 1 as

the LP#) to a logical page #a are respectively written to each physical page 52 from a physical page #1 (the physical page having 1 as the PP#) to a physical page #a. For the remaining physical pages 52 (that is, from physical page #(a + 1) through physical page #n), the state (erased state) is one in which data has yet to be written in.

[0039] Fig. 3B shows an example of the state subsequent to new data having been written to the physical block #p.

[0040] In this example, the data to be written to logical page #a is stored in the physical page #(a + 1), and the data to be written to logical page #2 is stored in physical page #n. In so doing, it becomes possible to append data to the physical block #p without erasing the physical block #p (strictly speaking, without performing an erase process for the physical block #p) with respect to a write unit request that is smaller than the size of the physical block #p.

[0041] However, according to the example of Fig. 3B, since data is written up to the last physical page #n in the append area, a physical page that is capable of being written to does not exist in the physical block #p. Accordingly, the controller 2 carries out a merge operation (reclamation) that reflects the appended data in the master area. In the merge operation, the controller 2 reads in the latest data of each logical page from the physical block #p, and after carrying out an erase process for the physical block #p, writes the latest data for the logical page corresponding to this physical page in order from the first physical page of the master area. Furthermore, so that the data write is carried out in order from the first physical page to the last physical page, the latest data corresponding to the same logical page exists in either the last physical page or the physical page closest to the last physical page.

[0042] Fig. 3C shows the state of the physical block #p after the merge operation has been carried out.

[0043] The data is written in order from the first physical page of the master area, and all the physical pages comprising the append area are in an erased state. The first physical page #1 through the last physical page #a of the master area correspond to the logical pages #1 through #a. Therefore, for example, the data (the latest data) written in the physical page #1 prior to the merge operation is once again written to physical page #1. This is because the data of the logical page #1 has not been updated. The latest data of the logical page #2 (the data written to the append area) is written to the physical page #2.

[0044] Furthermore, the data write destination in the merge operation may be another empty block (an erased physical block) instead of the same physical block #p. Specifically, for example, the respective latest data corresponding to the respective logical pages inside the physical block #p may be copied to an empty block, and thereafter, an erase process may be carried out with respect to the physical block #p.

[0045] Fig. 4 shows the configuration of the module control data 6.

[0046] The module control data 6 is data used in controlling access to the chip array 4. The module control data 6 comprises a block mapping table 61 comprising information denoting the corresponding relationship between the logical blocks and the physical blocks, and a page mapping table 62 comprising information denoting the corresponding relationship between the logical pages and the physical pages. These tables 61 and 62 will be explained in detail below. Furthermore, the information of the tables 61 and 62 may be managed in a form other than a table.

[0047] Fig. 5 shows the configuration of the module control function 7.

[0048] The module control function 7, for example, is a software (a computer program) that is executed by the processor (a typical microprocessor) of the controller 2. The process carried out by the module control function 7 below is actually performed by the processor that executes this software. At least one part of the module control function 7 may be realized using a hardware circuit. The module control function 7 comprises the host read service function 71, the host write service function 72, the page mapping table reconstruct function 74, the power ON function 75, and the initialize function 76. The functions 71 through 76 will be explained in detail below.

[0049] Fig. 6 shows an example of the corresponding relationship between the tables 61 and 62 inside the volatile memory 3 and the state of the physical block inside the chip array 4.

[0050] In this embodiment, the logical block number and the logical page number are specified based on the logical address specified by the host 11, and access is carried out to the physical page corresponding to the specified logical block number and logical page number. Specifically, for example, in a case where an access from the host 11 to the flash memory module 1 is carried out using a LBA (Logical Block Address) in SCSI, the quotient obtained by dividing the LBA by the capacity of the master area inside the block is the logical block number, and the quotient obtained by dividing the remainder by the page capacity is the logical page number.

[0051] The block mapping table 61 comprises information denoting the corresponding relationship between each logical block number 611 and each physical block number 612. By referring to this table 61, it is possible to determine the physical block 51 in which the access-destination area corresponding to the logical address specified by the host 11 exists.

[0052] The page mapping table 62, for example, exists in each logical block. The page mapping table 62 comprises information denoting the corresponding relationship between each logical page number 621 and each physical page number 622. By referring to this table 62, it is possible to determine the physical page in which the access-destination area corresponding to the logical address specified by the host 11

exists.

[0053] The physical block 51 stores the logical block number 517 of the logical block to which this physical block 51 is allocated. The physical page 52 (for example, this user area) stores data (so-called user data) 523 that is written/read directly to/from the host 11. This physical page 52 (for example, this overhead area) also stores the logical page number 528 corresponding to this physical page 52.

[0054] Specifically, for example, in the overhead area of each physical page 52 there is stored metadata comprising the logical page number 528 corresponding to this physical page 52. As used in this embodiment, "metadata" is not the data written/read directly to/from the host 11, but rather refers to the data required for management purposes. Metadata, for example, is the management information unit.

[0055] For example, in addition to the logical page number 528 corresponding to the first physical page #i, the metadata stored in the overhead area of the first physical page #i may also include the logical block number 517 of the logical block to which is allocated the physical block 51 having this first physical page #i.

[0056] Fig. 7 shows an example of the configuration of the block mapping table 61.

[0057] This table 61 comprises information denoting the corresponding relationship (the block corresponding relationship) between the logical block numbers 611 and the physical block numbers 612. This table 61 is created when the power to the flash memory module 1 is turned ON, and is referred to at the time of a write/read to/from the host 11. This table 61 is updated in a case where the block corresponding relationship has changed. The block corresponding relationship, for example, changes in a case where wear-leveling for suppressing the bias of the number of erasures is carried out. The block corresponding relationship may also change in a case where a merge operation is performed for any physical block 51, and in a case where the page mapping table 62 has been reconstructed.

[0058] For each logical block number 611, the block mapping table 61 comprises a corresponding physical block number 612, and a page mapping table flag 613 showing whether or not a page mapping table 62 is available. In a case where a valid page mapping table 62 is available for a logical block, the value of the flag 613 corresponding to this logical block number 611 is "Y", and in a case where a valid page mapping table 62 is not available for a logical block, the value of the flag 613 is "N". Furthermore, the value of the flag 613, for example, is changed from "Y" to "N" in a case where the physical block allocated to the corresponding logical block has changed to a different physical block (that is, in a case where the merge operation has been carried out). For this reason, in a case where this logical block is to be accessed for the first time since this different physical block has been allocated to the corresponding logical block, the page mapping table 62 for this logical block is reconstructed.

[0059] Fig. 8 shows an example of the configuration of the page mapping table 62.

[0060] This table 62 comprises information denoting the corresponding relationship (page corresponding relationship) between the logical page numbers 621 and the physical page numbers 622. The above-mentioned block mapping table 61 is created when the power to the flash memory module 1 is turned ON, but this table 62 is not created when the power to the flash memory module 1 is turned ON, but rather is created in a case where a logical block specified based on a logical address is the access destination for the first time subsequent to the power to the flash memory module 1 having been turned ON. The table 62 is referred to at the time of a data write/read to/from the host 11, and is updated at the time of a merge operation or a data write from the host 11. The table 62 comprises a corresponding physical page number 612 for each logical page number 611. The table 62 also comprises the number (last appended physical page number) 623 of the physical page to which data was last appended of the plurality of physical pages comprising the append area.

[0061] Fig. 9 shows an example of a scheme for storing data in the physical block 51.

[0062] As described above, the plurality of physical pages 52 included in the physical block 51 are physical pages 52 belonging to the master area and physical pages 52 belonging to the append area. Each physical page 52 comprises an area (overhead area) in which metadata 521 is stored, and a user area, and the user area comprises a plurality of data areas (for example, sectors) 523 in which are stored a plurality of data 523.

[0063] A set of metadata inside the master area, that is, a set of metadata 521 of the physical page 52 belonging to the master area will be called "block metadata" 515 below. By contrast, the individual pieces of metadata 521 inside the append area, that is, the metadata 521 of the physical page 52 belonging to the append area will be called the "page metadata" 526. Incidentally, in a case where the total capacity of the metadata areas inside the master area is greater than the capacity of the block metadata 515 to be stored, at least one physical page 52 inside the master area may not be used to store the block metadata 515.

[0064] Fig. 10 shows a detailed example of the configuration of the physical page 52.

[0065] The physical page 52 comprises a metadata area in which the metadata 521 is stored, an ECC (Error Correcting Code) area in which the ECC 522 of this metadata 521 is stored, a plurality of data areas in which the plurality of data 523 are respectively stored, and an ECC area in which the ECC 524 of these respective data are stored.

[0066] As shown in Fig. 9, the metadata area stores a portion of the block metadata 515 in a case where the physical page 52 comprising this area belongs to the master area, and stores the page metadata 526 in a case where this physical page 52 belongs to the append area.

[0067] By providing the ECC 522 for the metadata 521 (specifically, the portion of the

block metadata 515 and the page metadata 526), it is possible to determine the validity (that is, whether or not an error has occurred) of the metadata 521 even in a case where only the metadata 521 has been read in.

[0068] Fig. 11 shows an example of the configuration of the block metadata 515.

[0069] The block metadata 515 comprises the number 517 of the logical block corresponding to the physical block 51.

[0070] Fig. 12 shows an example of the configuration of the page metadata 526.

[0071] The page metadata 526 comprises the number 528 of the logical page corresponding to the physical page 52 inside the append area. Since the number of the logical page corresponding to the physical page 52 inside the master area is uniquely stipulated, the corresponding logical page 528 need not be stored in the metadata area inside the master area.

[0072] Fig. 13 shows an example of the flow of processing carried out by the host read service function 71.

[0073] This function 71 is called when the controller 2 has received a read request from the host 11, specifies a location on the chip array 4 in which the specified data is stored, and transfers the data to the host 11. The details of this process will be explained on a step-by-step basis below.

[0074] Upon receiving a read request from the host 11, the function 71 determines the logical block number and the logical page number (S711). Specifically, for example, in a case where the logical address specified from the host 11 is an LBA or other linear address, the function 71 is able to let the quotient obtained by dividing the logical address by the logical block capacity be the logical block number, and is able to let the quotient obtained by dividing the remainder by the logical page capacity be the logical page number. The logical block number and the logical page number specified in S711 will be called the "read-source logical block number" and the "read-source logical page number" below.

[0075] Next, the function 71 refers to the block mapping table 61 on the volatile memory 3, and determines the physical block number (the read-source physical block number) 612 corresponding to the read-source logical block number 611 (S712).

[0076] Next, the function 71 checks whether or not a valid page mapping table 62 corresponding to the read-source logical block is available on the volatile memory 3 (S713). Specifically, for example, the function 71 refers to the block mapping table 61, and checks whether or not the flag 613 corresponding to the read-source logical block number is "Y". In a case where a valid page mapping table 62 is not available (S713: NO), the function 71 calls the reconstruct function 74. In accordance with this, a valid page mapping table 62 corresponding to the read-source logical block is constructed. Then, the function 71 proceeds to S715. Furthermore, in a case where a valid page

mapping table 62 is available in S713 (S713: YES), the function 71 proceeds to S715.

[0077] The function 71 refers to the page mapping table 62 corresponding to the read-source logical block, and determines the physical page number (the read-source physical page number) 622 corresponding to the read-source logical page number 621 (S715). In accordance with the above, the read-source physical block and the read-source physical page of the data requested by the host 11 are identified.

[0078] Next, the function 71 reads the data from the identified read-source physical page inside the identified read-source physical block (S716), and responds to the host 11 with the read data (S717).

[0079] In accordance with the above processing, in a case where a read request has been received from the host 11, first, the read-source physical block 51, which is the general storage location, is specified as an access destination, and thereafter, a page mapping table 62 denoting the corresponding relationship between the logical pages and the physical pages is constructed as necessary, and the read-source physical page 52 is specified. For this reason, the information denoting the detailed logical/physical corresponding relationship, that is, the table 62 denoting the corresponding relationship between the logical pages and the physical pages does not need to be created in advance. The block mapping table 61 and the page mapping table 62 are stored on volatile memory 3. The time required to access these tables is shorter than the time required to access the flash memory, thereby making it possible to hold read performance degradation in check.

[0080] Fig. 14 shows an example of the flow of processing carried out by the host write service function 72.

[0081] This function 72 is called when a write request has been received from the host 11, and stores the requested data in a suitable location. The details of this process will be explained on a step-by-step basis below.

[0082] Upon receiving a write request from the host 11, the function 72 determines the logical block number (the write-destination logical block number) and the logical page number (the write-destination logical page number) based on the logical address specified in the write request (S721).

[0083] Next, the function 72 refers to the block mapping table 61, and determines the physical block number (the write-destination physical block number) corresponding to the write-destination logical block number (S722).

[0084] Then, the function 72 refers to the flag 613 corresponding to the write-destination logical block number, and checks whether or not a valid page mapping table 62 corresponding to this logical block number is available (S723). In a case where this table 62 is not available (S723: NO), the function 72 calls the reconstruct function 74. In accordance with this, a page mapping table 62 corresponding to the write-destination

logical block is created. Then, the function 72 proceeds to S725.

[0085] The function 72 checks whether or not there are enough erased pages (empty physical pages) in the write-destination physical block (S725). Specifically, for example, the function 72 refers to the last appended physical page number 623 of the page mapping table 62 corresponding to the write-destination logical page. Then, the function 72 checks the number of empty physical pages inside the write-destination physical block by comparing this page number 623 against the maximum value of the physical page number (the number of the last physical page). The function 72 determines whether or not the amount of data (write data) that accords with the write request from the host 11 exceeds a capacity equivalent to the specified number of empty physical pages.

[0086] In a case where the result of the determination of S725 is affirmative, that is, in a case where there are enough erased pages (S725: YES), the function 72 writes the write data received from the host 11 to the page metadata 526 and to the erased pages (S726). The logical page number determined in S721 is included in the page metadata 526. The function 72, either in S732 or between 74 and S725, determines whether or not the data is stored in the physical page of the same number as the write-destination logical page number (that is, the corresponding physical page inside the master area), and in a case where the data is determined to be stored in this physical page, the function 72 may carry out the determination of S725. In a case where the data is determined not to be stored in this physical page, the function 72 is able to write the write data received from the host 11 to the physical page of the same number as the write-destination logical page number. In the case of a first write to the write-destination physical block, the function 72 may write the write data and the block metadata 515 to this physical block.

[0087] Subsequent to S726, the function 72 updates the page mapping table 62 corresponding to the write-destination logical block (S727). Specifically, for example, the function 72 updates the physical page number 622 of the entry (the entry inside the table 62) comprising the write-destination logical page number to the number of the write-destination physical page in S726, and, in addition, updates the last appended physical page number 623 to the number of the last write-destination physical page in S726. Then the function 72 proceeds to S734.

[0088] By contrast, in a case where there are not enough erased pages in S725 (S725: NO), the merge operation is performed. That is, the function 72 reads in the data (the latest data of the one or more pieces of data corresponding to the logical page) that exists in the write-destination physical block (hereinafter, the old physical block), merges this data with the write data received from the host 11, and creates a set of data to be written to the master area (S728). Next, the function 72 selects one unused physical

block (empty physical block) (S729). In this case, the function 72 may adopt a method of selecting a physical block which is not subjected to erasure so often in order to suppress the bias of the number of erasures among the physical blocks on the chip array 4. Then, the function 72 writes the data created in S728 to the selected physical block as well as to the block metadata 515 (S730). This block metadata 515 comprises the number of the logical block (write-destination logical block number) to which the old physical block is allocated. Then, the function 72 updates the page mapping table 62 corresponding to the write-destination logical block (S731). Specifically, for example, the function 72 performs updating such that the physical pages 622 corresponding to all the logical pages 621 point to the master area, and, in addition, updates the last appended physical page number 623 to the number of the first physical page of the append area. Next, the function 72 updates the block mapping table 61 (S732). Specifically, for example, the function 72 performs updating such that the physical block number 612 corresponding to the write-destination logical block number 611 points to the number of the new physical block written to in S730 (the physical block selected in S729). Next, the function 72 erases the data of the old physical block 51 (S733). Then, the function 72 proceeds to S734.

[0089] Lastly, the function 72 responds to the host 11 to the effect that the write has been completed (S734).

[0090] According to the above processing, in a case where a write request has been received from the host 11, the controller 2 is able to specify the write-destination physical block 51, which is the general storage location. Then, as necessary, the controller 2 is able to create a mapping table 62 denoting the corresponding relationship between the logical pages and the physical pages, and is able to store the data in a suitable location. In both a case where the corresponding relationship between the logical blocks and the physical blocks changes, and a case where the corresponding relationship between the logical pages and the physical pages changes, the controller 2, in addition to writing the write data to the flash memory, is also able to write as metadata information showing which is the write-destination logical block and which is the write-destination logical page. It is therefore possible to write the information for logical/physical conversion to the flash memory without generating excessive accesses. For this reason, it is possible to restore the block mapping table 61 and the page mapping table 62 from the metadata (the block metadata and the page metadata) written in the flash memory even when the volatile memory 3 has been erased due to a loss of power, and the block mapping table 61 and the page mapping table 62 have been erased. For example, restoration may be carried out only for the block mapping table 61, and thereafter, the page mapping table 62 may be restored as needed in accordance with a write access and a read access.

[0091] Fig. 15 shows an example of the flow of processing carried out in accordance with page mapping table reconstruct function 74.

[0092] This function 74 is called in a case where a page mapping table 62 corresponding to the logical block is not available on the volatile memory 3, and creates a page mapping table 62 based on the page metadata 526 on the flash memory. The details of the process will be explained on a step-by-step basis below.

[0093] Upon being called, the function 74 first performs initialization such that the logical page points to the physical page 52 of the master area in a page mapping table 62 corresponding to the called logical block (either the read-source logical block or the write-destination logical block) (S741). Specifically, for example, the function 74 performs the same processing for subsequent logical pages so that the first logical page points to the first physical page, and the second logical page points to the second physical page.

[0094] Next, the function 74 first reads in the page metadata 526 for all the physical pages 52 inside the append area (S742), and acquires the logical page number 528 inside this metadata 526 (S743). The function 74 also confirms that the read-in page metadata 526 is correct (confirms the presence or absence of an error in the metadata 526) by checking the ECC for the read-in page metadata 526 (S744). In a case where this metadata 526 is not correct, the ECC is used to correct this metadata 526. In a case where the data read-source physical page was the page that got erased, the function 74 proceeds to S747 since the page metadata has already been read from all the physical pages inside the append area. In a case where the data read-source physical page was not the page that got erased, appended data exists in this physical page, and the function 74 writes the number of this read-source physical page as the physical page number 622 corresponding to the logical page number specified in S743 for the entry (the entry inside the page mapping table 62) comprising the logical page number specified in S743 (S746). The function 74 repeats the above processing for the other physical pages 52 of the append area (S742), and proceeds to S747 when this processing is finished.

[0095] Lastly, upon finishing the updating of the physical page number 622 of the page mapping table 62, the function 74 updates the last appended physical page number 623 to the number of the last physical page to which the latest data has been written (S747).

[0096] According to the above steps, in a case where there is appended data, the physical page in which the latest data has been appended is associated with each logical page inside a certain specified logical block, and in a case where there is no appended data, the physical page of the master area is associated with each logical page inside the certain specified logical block. As result of this, it is possible to reconstruct (create) a page logical/physical corresponding relationship for one block. Since only the page metadata 526 of the append area may be read in this process, it is possible to minimize

the restoration overhead.

[0097] Furthermore, as has already been explained, the function 74 is called by the host read service function 71 and the host write service function 72 in the case of an access to a logical block for which a page mapping table 62 has not been created. However, when the controller 2 has excess throughput, the processing shown in Fig. 15 may be executed at a suitable time even without a request.

[0098] Fig. 16 shows an example of the flow of processing carried out by the power ON function 75.

[0099] This function 75 is called when the power to the flash memory module 1 is turned ON, and creates the block mapping table 61 on the basis of the metadata on the flash memory. The details of this process will be explained on a step-by-step basis below.

[0100] When the power is turned ON, first, the function 75 carries out the processing of S752 through S754 for all of the physical blocks 51. That is, the function 75 reads out the block metadata 515 from each physical block 51, and extracts the logical block number 517 from this metadata 515 (S752). Next, the function 75 checks to make sure this metadata 515 is correct by checking the ECC for the read-in block metadata 515 (S753). Then, in the block mapping table 61, the function 75 writes the read-source physical block number 612 of the metadata 515 into the entry corresponding to the logical block number 611 read in S752 (S754). The function 75 repeats the above processing for the other physical blocks 51 (S751), and proceeds to S755 when the processing is finished.

[0101] Next, the function 75 sets the page mapping table flags 613 of all the entries in the block mapping table 61 to "N", that is, to the value denoting unavailable (S755)

[0102] Lastly, the function 75 conveys to the host 11 the fact that the flash memory module 1 is ready to receive I/O (S756).

[0103] According to the above steps, the physical blocks corresponding to all the logical blocks inside the flash memory module 1 are identified. As a result, the block logical/physical corresponding relationship may be restored on the basis of information stored in the flash memory. Furthermore, in this process, since it is all right to read only the block metadata 515 stored in the master area, there is no need to read out the data received from the host 11 and the page metadata 526. It is therefore possible to minimize restoration overhead. As a result of this, it is possible to reduce the time required for boot up, thereby making it possible to convey to the host 11 in a short period of time that fact that I/O are ready to be received.

[0104] Fig. 17 shows an example of the flow of processing carried out by the initialize function 76.

[0105] This function 76 is used when the flash memory module 1 is used for the first time, and when erasing already stored data to return to the unused state once again. The

details of this process will be explained on a step-by-step basis below.

[0106] Upon being called, the function 76 first carries out an erase process for all the physical blocks 51 (S761).

[0107] The function 76 selects an unused physical block (an empty physical block) 51 for a single logical block from among the unprocessed logical blocks (S763).

[0108] Next, the function 76 creates the block metadata 515 and the ECC thereof (S764). The number of the above-mentioned single logical block being used in this process is stored in the block metadata 515.

[0109] Next, the function 76 stores the created block metadata 515 and data in the physical pages 52 of the master area (S765). The "data" referred to here may be a fixed value such as that utilized in an HDD format or the like.

[0110] Finally, the function 76 updates the physical block number 612 to the number of the physical block 51 selected in S763 for the block mapping table 61 entry corresponding to the logical block for which this process was carried out (S766).

[0111] The function 76 repeats the above-mentioned S763 through S766 for all the logical blocks (S762).

[0112] According to the above-mentioned process, corresponding physical blocks 51 are allocated to all the logical blocks, and the block metadata 515 comprising the numbers of the logical blocks corresponding to these physical blocks 51 are stored in the allocated physical blocks 51. As a result, it becomes possible to suitably process an access from the host 11 thereafter. It also becomes possible to properly restore the logical/physical conversion information from the flash memory 4 when the power is turned ON.

[0113] Fig. 18 shows an example of the exchange between the controller 2 and the chip array 4 in S726 of the host write service function 72.

[0114] The controller 2 sends the write command together with the data to be written to the physical page 52. The data to be written comprises the page metadata 526 and the ECC 522 thereof, and the data 523 sent from the host and the ECC 524 thereof.

[0115] Since the entire metadata write is carried out concomitantly with the write of the data sent from the host here, the metadata write does not substantially affect the write performance or number of writes-based service life of the flash memory.

[0116] Fig. 19 shows an example of the exchanges between the controller 2 and the chip array 4 in S728 through S730 of the host write service function.

[0117] In this drawing, the physical block a is the old physical block, and the physical block b is the physical block that is to be used anew.

[0118] First, the latest data is read out from the physical page 6 in which the latest data inside the physical block a is stored. This corresponds to the operation carried out in S728.

[0119] Then, a write command for the physical page 52 inside the physical block b (the physical page 52 inside the master area) is sent together with the data to be written to this physical page 52. The data comprises a portion of the block metadata 515 and the ECC 522 thereof, and the data 523 sent from the host 11 and the ECC 524 thereof. This corresponds to S730.

[0120] Since the entire metadata write is carried out concomitantly with the write of the data sent from the host 11 here, the metadata write does not substantially affect the write performance or number of writes-based service life of the flash memory.

[0121] Fig. 20 shows an example of the exchanges between the controller 2 and the chip array 4 in S743 for the page mapping table reconstruct function 74 and S752 for the power ON function 75.

[0122] When the power is turned ON, first the block metadata 515 is read out from the physical block 51. Specifically, the metadata 521 and the ECC 522 thereof are read out from the physical pages 52 of the master area. This process is repeated for all of the physical blocks 52. The preceding corresponds to S752 of the power ON function 75. Then, when the read and construction are finished, the host 11 is notified to the effect that the flash memory module 1 is ready to receive I/O.

[0123] Thereafter, upon receiving an access request (I/O request) from the host 11, the page metadata 526 is read out from the specified physical block 51 on the basis of the logical address specified in this access request. Specifically, the metadata 521 and the ECC 522 thereof are read out from the physical page 52 inside the append area. This process is repeated for the physical pages 52 inside the append area. The preceding corresponds to S743 of the page mapping table reconstruct function 74.

[0124] Only the block metadata 515, that is, only the metadata 521 inside the master area is read out here when the power is turned ON; the page metadata 526 is not read out. For this reason, the time required for boot up is reduced, thereby making it possible to notify the host 11 to the effect that the flash memory module 1 is ready to receive I/O in a short period of time.

[0125] Also, in a case where an access request is received from the host 11, and a page mapping table 62 corresponding to the logical block specified on the basis of the logical address specified in this access request is not available, only the page metadata 526 is read out and the table 62 is created. For this reason, it is possible to restore the information for the logical/physical conversion using a minimal read.

[0126] According to the embodiment described above, it is possible to reduce the affects on the write performance of the flash memory during normal operation while cutting down on the time required for boot up.

[0127] One embodiment of the present invention has been explained above, but it goes without saying that the present invention is not limited to this embodiment, and that

various changes are possible without departing from the gist thereof.

[0128] Furthermore, the page mapping table 62 may be prepared for each plurality of logical blocks instead of for each logical block. For example, when the controller 2 receives an access command, a page mapping table 62 corresponding to the access-destination logical block, which is specified on the basis of the logical address specified in this access command, and the next logical block may be created.

[0129] Also, a single master area and a single append area may be configured using two or more physical blocks. For example, a first physical block may be the master area, and a second physical block may be the append area.

[0130] The block mapping table 61 is constructed on the volatile memory 3, and a plurality of methods are conceivable as this construction method.

[0131] A first method stores a physical block number 612 in the volatile memory 3, and computes an address value on the volatile memory 3 in order to determine the physical block number 612 at access. This address value is determined from the logical block number 611 by adding a value obtained by multiplying the logical block number 611 by the number of bytes of configuration information in one entry of the block mapping table 61 to the start address value on the volatile memory 3 of the block mapping table 61. This address value is uniquely computed. In a case where either a read or write request is received from the host, it is possible to use this address value, access the volatile memory 3, and obtain the information of the physical block number 612 and the flag 613. This method is characterized in that it is possible to shorten the reference time.

[0132] A second method stores the information of the logical block number 611, the physical block number 612 and the flag 613 in the volatile memory 3 of the block mapping table 61 of the first method. This second method has the effect of enhancing reliability by making it possible to check that the address value computed from the logical block number 611 in the first method is correct by comparing this address value to the logical block number 611 stored in the volatile memory 3, and making it possible to detect a controller 2 logical error and a volatile memory 3 software error.

[0133] A third method stores the information of the logical block number 611, the physical block number 612 and the flag 613 in the volatile memory 3, and constructs a block mapping table 61 equivalent to the number of logical blocks being used. In a case where a logical block number 612 is not stored in a physical block, it is determined that the logical block is not being used. In this method, the effect is that the block mapping table 61 may be constructed on the capacity of the logical blocks being used, thereby making it possible to reduce the capacity of the block mapping table 61.

[0134] Also, as for S712 of Fig.13,there are a plurality of methods for determining the physical block number 612 in accordance with the method for constructing the block

mapping table 61. For example, there is a method for determining an address in the block mapping table 61 and finding the physical block number 612 that is stored there by carrying out a computation based on the read-source logical block number 611. In addition, there is also a method for storing the value of the read-source logical block number 611 in the block mapping table 61 as well, and checking to make sure that the computation is correct. Further, there is a method for storing a pair of the values of the read-source logical block number 611 and the physical block number 612 in the block mapping table 61, and using the read-source logical block number 611 to search inside the block mapping table 61 to find the corresponding physical block number 612.

[0135] Also, as for S722 of Fig.14, there are a plurality of methods for determining the physical block number 612 in accordance with the method for constructing the block mapping table 61. There is a method for determining the address in the block mapping table 61 and finding the physical block number that is stored there by carrying out a computation based on the write-destination logical block number, there is also a method for storing the value of the write-destination logical block number in the block mapping table 61 as well, and checking to make sure that the computation is correct, and furthermore, there is a method for storing a pair of the values of the write-destination logical block number and the physical block number in the block mapping table 61, and using the write-destination logical block number to search inside the block mapping table 61 to find the corresponding physical block number.

[0136] When constructing the block mapping table 61 in S754 of Fig.16, there are a plurality of methods for writing data into the block mapping table 61 in accordance with the method for constructing the block mapping table 61.

[0137] A first method determines an address in the block mapping table 61 by carrying out a computation based on a logical block number 517 selected from the metadata 515 and stores the physical block number 611 at this address.

[0138] A second method, in addition to the processing in the first method, also stores the logical block number 517 in the block mapping table 61 for checking the computed address.

[0139] A third method stores a pair comprising the logical block number 517 and the physical block number 611 in the order of the physical blocks 51 from which the block metadata 515 has been read out. In a case where a logical block number 517 is not stored in the physical block 51, that is, a case in which the physical block 51 is not being used, neither the logical block number 517 nor the physical block number 611 is stored in the block mapping table 61.

Reference Signs List

[0140] 1 Flash memory module

Claims

[Claim 1] A flash memory module, comprising:
a flash memory; and
a controller receiving an access command from a host, and, based on logical/physical conversion information denoting a corresponding relationship between a logical address and a physical address of the flash memory, accessing an area in accordance with a physical address corresponding to a logical address specified in the received access command,
wherein the logical/physical conversion information comprises first conversion information and second conversion information, and
wherein the controller restores the first conversion information at boot up time, enables an access command to be received after the first conversion information has been restored, and restores the second conversion information after an access command is able to be received.

[Claim 2] A flash memory module according to claim 1,
wherein the flash memory is a flash memory in which an access is carried out in page units and an erase is carried out in block units,
wherein the flash memory comprises a plurality of physical blocks,
wherein each physical block comprises a plurality of physical pages,
wherein the first conversion information is block mapping information denoting a corresponding relationship between logical blocks and physical blocks,
wherein the second conversion information is page mapping information denoting a corresponding relationship between logical pages and physical pages,
wherein the page mapping information is information restored for each of one or more physical blocks,
in a case where user data is to be written for the first time to an empty physical block, which is a physical block in an erased state, wherein the controller writes a block metadata, which is data comprising an ID of the logical block to which this empty physical block is allocated, to this empty physical block in addition to the user data,
in a case where a certain user data inside the physical block is to be updated, wherein the controller writes, in addition to a post-update user data, a page metadata, which is data comprising an ID of a write-destination logical page of the post-update user data, to a physical page

other than the physical page in which the certain user data is stored, wherein the controller restores the block mapping information at the boot up time by reading in the block metadata from each physical block of the flash memory, and

wherein the controller:

(W) in a case where a write command is received from the host after an access command is able to be received;

(w1) specifies a write-destination logical block and a write-destination logical page based on a logical address specified in the received write command;

(w2) specifies a physical block corresponding to the write-destination logical block based on the block mapping information;

(w3) determines whether or not valid page mapping information corresponding to the write-destination logical block is available;

(w4) in a case where the result of the determination of the (w3) is negative, creates the page mapping information corresponding to the write-destination logical block by reading in the page metadata from the physical page inside the physical block corresponding to the write-destination logical block;

(w5) stores a target user data of the received write command and the page metadata comprising an ID of the write-destination logical page either in the physical page inside the physical block corresponding to the write-destination logical block, or the physical page inside another physical block associated with the write-destination logical block, and updates the page mapping information corresponding to the write-destination logical block, and

(R) in a case where a read command is received from the host after an access command is able to be received;

(r1) specifies a read-source logical block and a read-source logical page based on a logical address specified in the received read command;

(r2) specifies a physical block corresponding to the read-source logical block based on the block mapping information;

(r3) determines whether or not a valid page mapping table corresponding to the read-source logical block is available;

(r4) in a case where the result of the determination of the (r3) is negative, creates the page mapping information corresponding to the read-source logical block by reading in the page metadata from the physical page inside the physical block corresponding to the read-

source logical block; and

(r5) specifies the physical page corresponding to the write-destination logical page based on the page mapping information corresponding to the write-destination logical block, and reads out the user data from the specified physical page.

[Claim 3] A flash memory module according to claim 2,

wherein when storing the block metadata, the controller stores a part of the block metadata that is stored in the physical page together with an ECC (Error Correcting Code),

wherein when storing the page metadata, the controller also stores the ECC in addition to the page metadata,

wherein when reading in the block metadata, the controller determines the presence or absence of an error in each part by using the ECC read in together with the each part of the block metadata, and

wherein when reading in the page metadata, the controller determines the presence or absence of an error in this page metadata by using the ECC read in together with this page metadata.

[Claim 4] A flash memory module according to claim 2,

wherein a master area and an append area are configured from in or more physical blocks, and

wherein the block metadata is stored in the master area, and the page metadata is stored in the append area.

[Claim 5] A flash memory module according to claim 4, wherein the controller, for each logical page, writes a user data, which is not the post-update user data, to the physical page belonging to the master area, and writes the post-update user data and the page metadata comprising an ID of the write-destination logical page of this post-update user data to the physical page belonging to the append area.

[Claim 6] A flash memory module according to claim 1,

wherein the flash memory is a flash memory in which an access is carried out in area units and an erase is carried out in area group units comprising a plurality of areas,

wherein the first conversion information is information denoting a corresponding relationship between logical areas and physical areas, and

wherein the second conversion information is information denoting a corresponding relationship between logical area groups and physical area groups.

[Claim 7] A flash memory module according to claim 6, wherein the second

conversion information is information restored for each of one or more area groups.

[Claim 8] A flash memory module according to claim 7,
wherein the area group is a block, and
wherein the controller receives an access command from the host, and in a case where the valid second conversion information is not available for the block specified based on the logical address specified in this access command, restores the second conversion information for one or more blocks comprising this block.

[Claim 9] A flash memory module according to claim 8,
wherein a master area and an append area are configured in one or more physical blocks, and
wherein the controller:
(X) when writing the user data to the append area, also writes information required to restore the second conversion information to the append area; and
(Y) when writing the user data to the master area, also writes information required to restore the first conversion information to the master area.

[Claim 10] A flash memory module according to claim 7,
wherein the controller stores, in respective one or more physical area groups, first management information comprising information denoting one or more logical area groups to which the one or more physical area groups are allocated,
wherein the controller stores, in a physical area in the physical area group, second management information comprising information denoting a logical area allocated to this physical area,
wherein the controller restores the first conversion information by reading in the first management information from the respective one or more physical area groups, and
wherein the controller restores, for one or more logical area groups, the second conversion information corresponding to the one or more logical area groups by reading in the second management information from each physical area in which the second management information is stored.

[Claim 11] A flash memory module according to claim 10,
wherein when storing the first management information, the controller stores a part of the first management information that is stored in the

physical area together with an ECC (Error Correcting Code),
wherein when storing the second management information, the controller also stores the ECC in addition to the second management information,
wherein when reading in the first management information, the controller determines the presence or absence of an error in each part by using the ECC read in together with the each part of the first management information, and
wherein when reading in the second management information, the controller determines the presence or absence of an error in this second management information by using the ECC read in together with this second management information.

[Claim 12] A flash memory module according to claim 1, wherein the controller restores the second conversion information in a case where the controller receives an access command from the host.

[Claim 13] A flash memory module according to claim 1, wherein the flash memory is configured from one or more physical blocks, the controller is connected to a memory, and the controller reads out a logical address from the physical block at boot up time, and stores the logical address and a physical address that corresponds to this logical address in the memory as the first conversion information.

[Claim 14] A flash memory module according to claim 13, wherein the controller reads out the logical address from the physical block at boot up time, and calculates a first address value corresponding to the logical address storage location from the logical address and a start address value of the memory,
the controller calculates a second address value from the start address value and the logical block address stored in the memory, and
the controller confirms that the first address value and the second address value match.

[Claim 15] A control method of a flash memory module receiving an access command from a host, and, based on logical/physical conversion information denoting a corresponding relationship between a logical address and a physical address of a flash memory, accessing an area in accordance with a physical address corresponding to a logical address specified in the received access command, the method comprising the steps of:
restoring first conversion information of the logical/physical conversion

information at boot up;
notifying the host that an access command is able to be received after having restored the first conversion information; and
restoring second conversion information of the logical/physical conversion information after having notified the host that an access command is able to be received.

[Fig. 1]

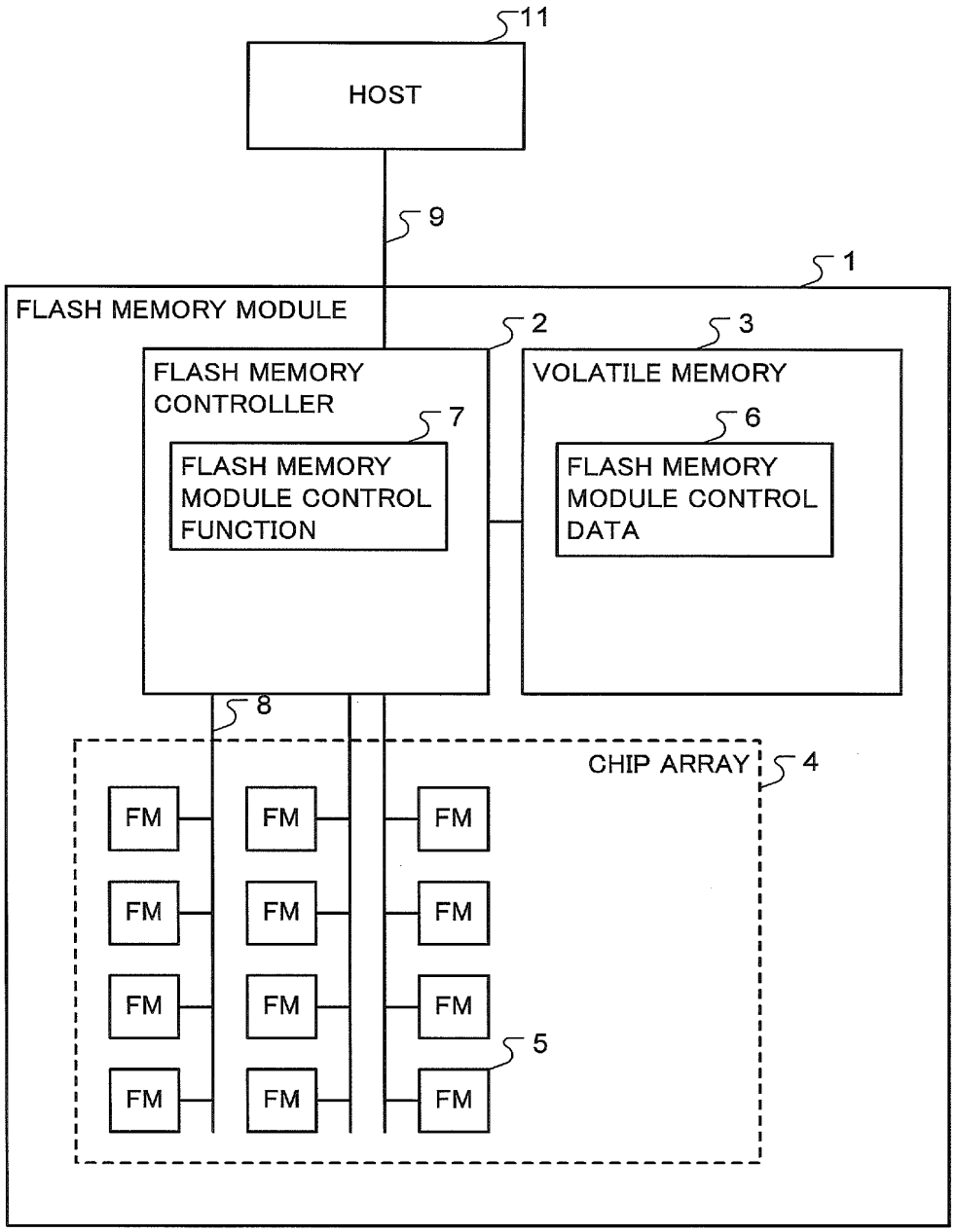

11
HOST
9
1
FLASH MEMORY MODULE
2
FLASH MEMORY CONTROLLER
7
FLASH MEMORY MODULE CONTROL FUNCTION
3
VOLATILE MEMORY
6
FLASH MEMORY MODULE CONTROL DATA
8
CHIP ARRAY
4
FM
5

FIG.1

[Fig. 2]

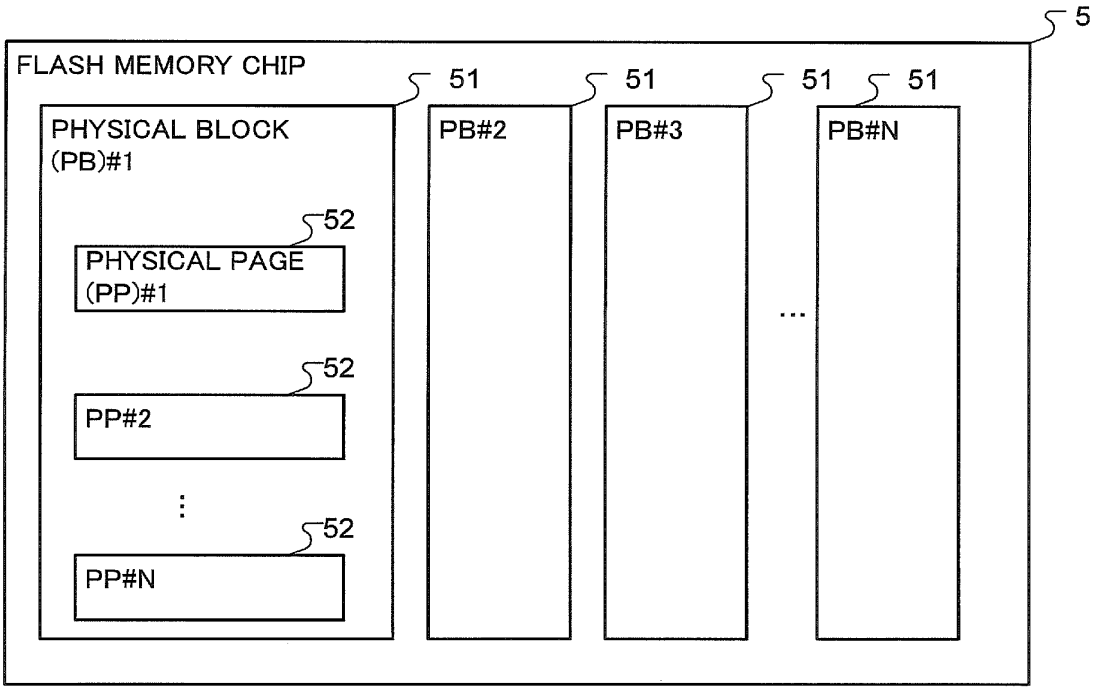


FIG.2

[Fig. 3A]

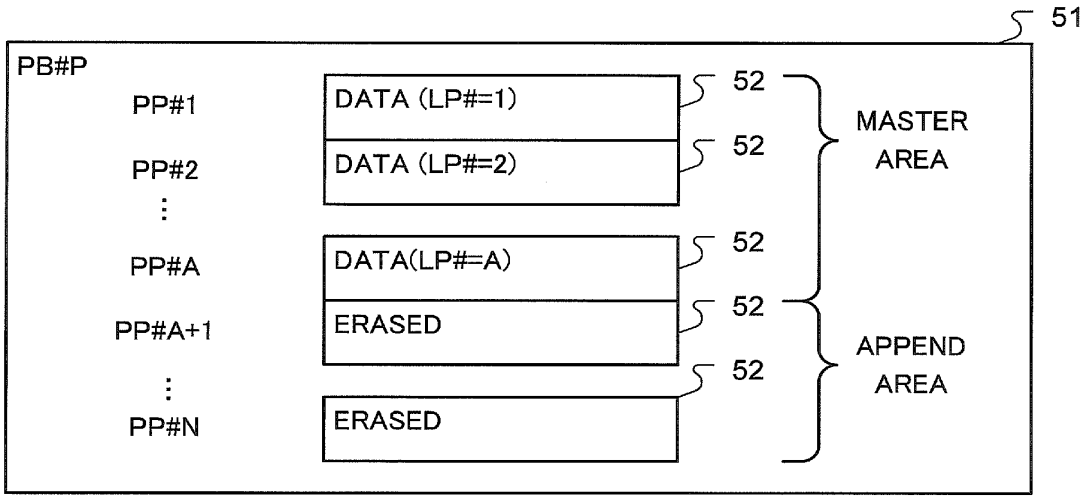


FIG.3A

[Fig. 3B]

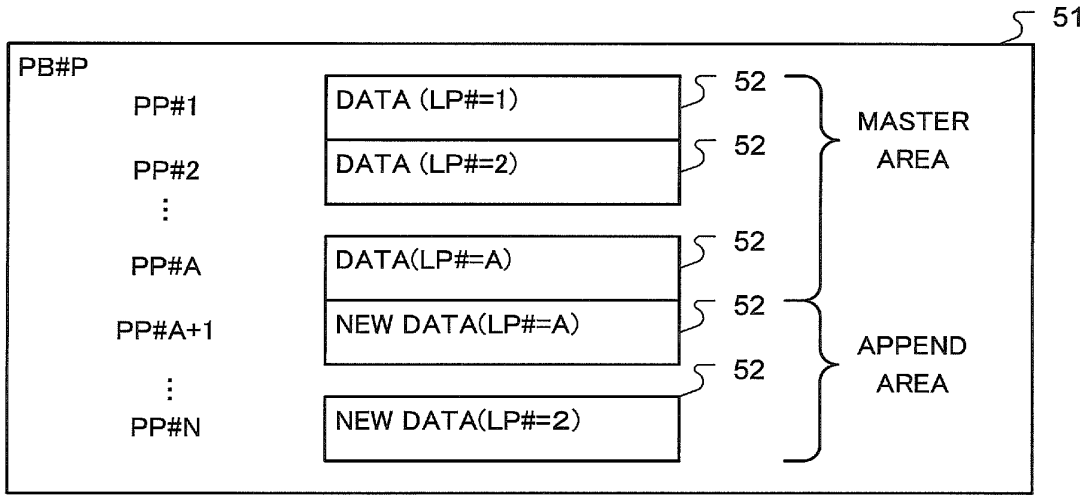


FIG.3B

[Fig. 3C]

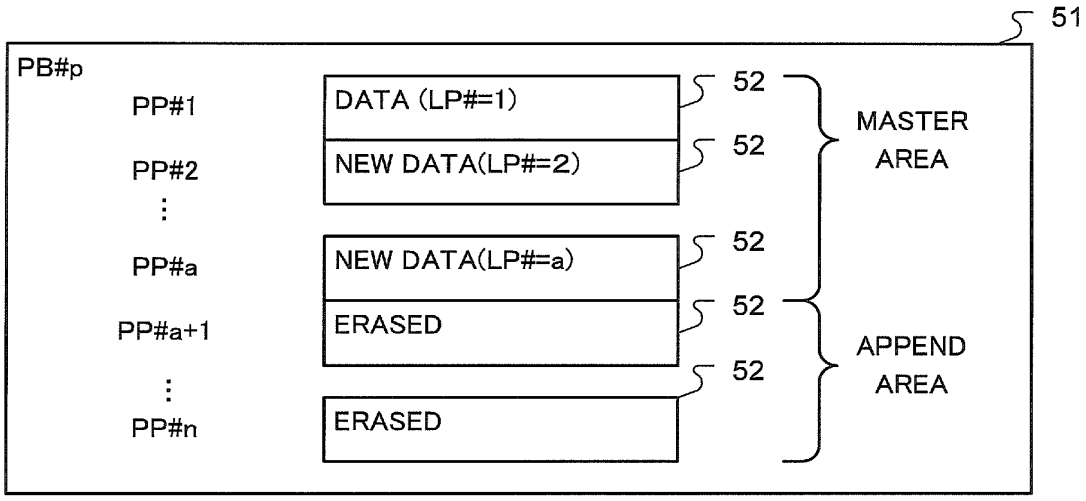


Fig.3C

[Fig. 4]

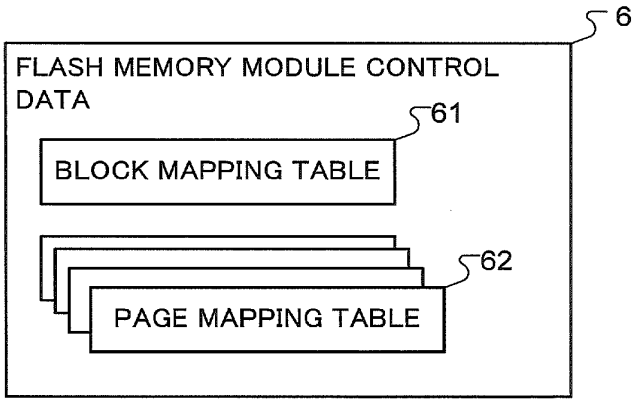


FIG.4

[Fig. 5]

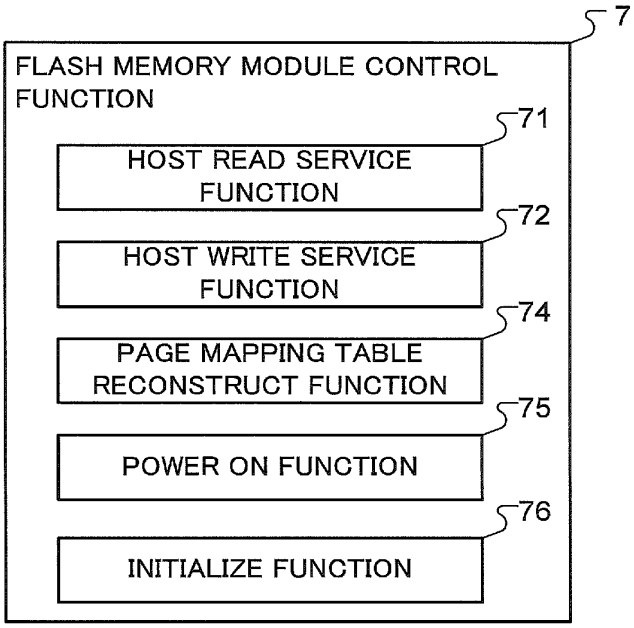


FIG.5

[Fig. 6]

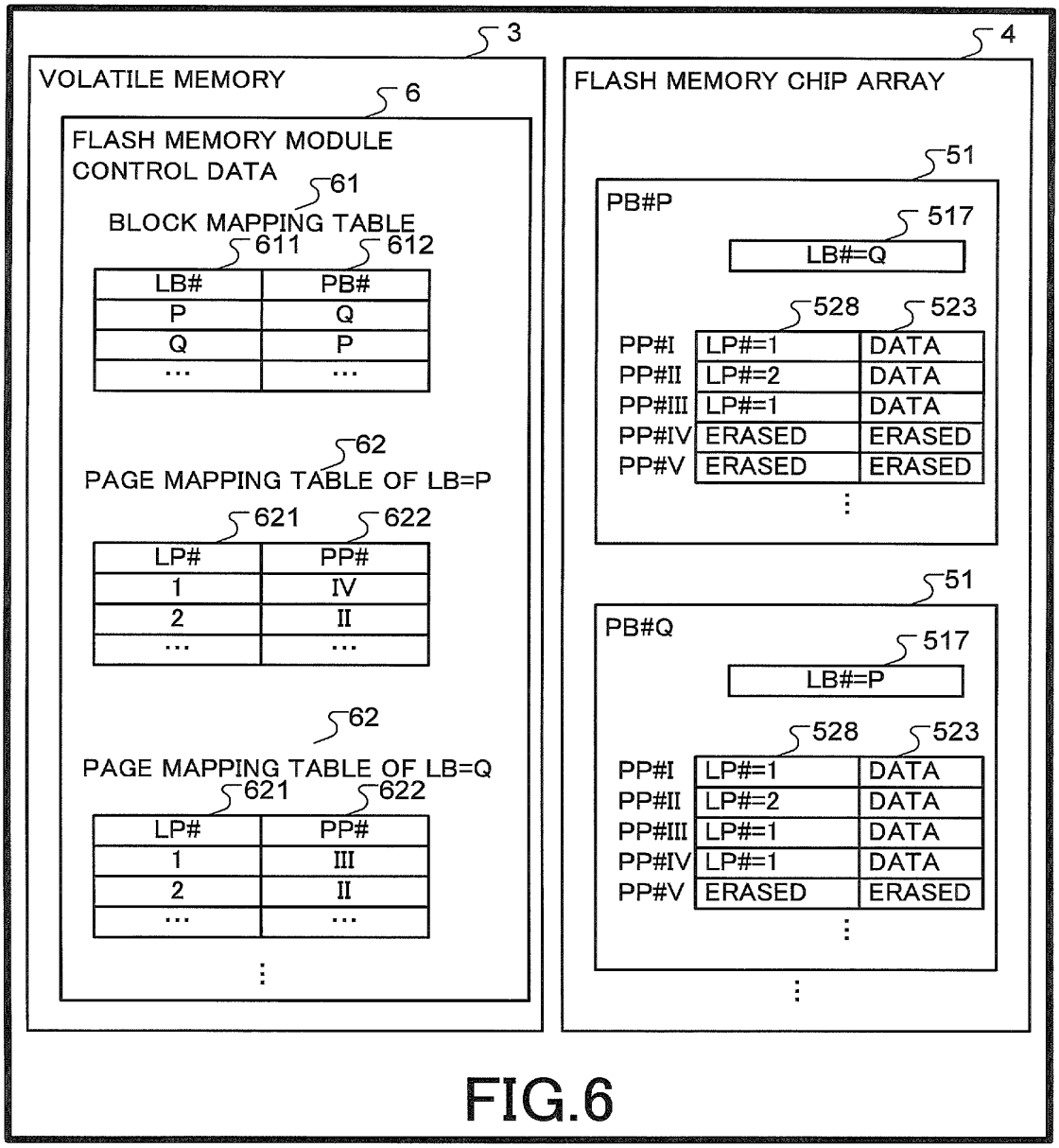

3
VOLATILE MEMORY
6
FLASH MEMORY MODULE CONTROL DATA
61
BLOCK MAPPING TABLE
611
612
LB#
PB#
P
Q
Q
P
62
PAGE MAPPING TABLE OF LB=P
621
622
LP#
PP#
1
IV
2
II
62
PAGE MAPPING TABLE OF LB=Q
621
622
LP#
PP#
1
III
2
II
4
FLASH MEMORY CHIP ARRAY
51
PB#P
517
LB#=Q
528
523
PP#I
LP#=1
DATA
PP#II
LP#=2
DATA
PP#III
LP#=1
DATA
PP#IV
ERASED
ERASED
PP#V
ERASED
ERASED
51
PB#Q
517
LB#=P
528
523
PP#I
LP#=1
DATA
PP#II
LP#=2
DATA
PP#III
LP#=1
DATA
PP#IV
LP#=1
DATA
PP#V
ERASED
ERASED

FIG.6

[Fig. 7]

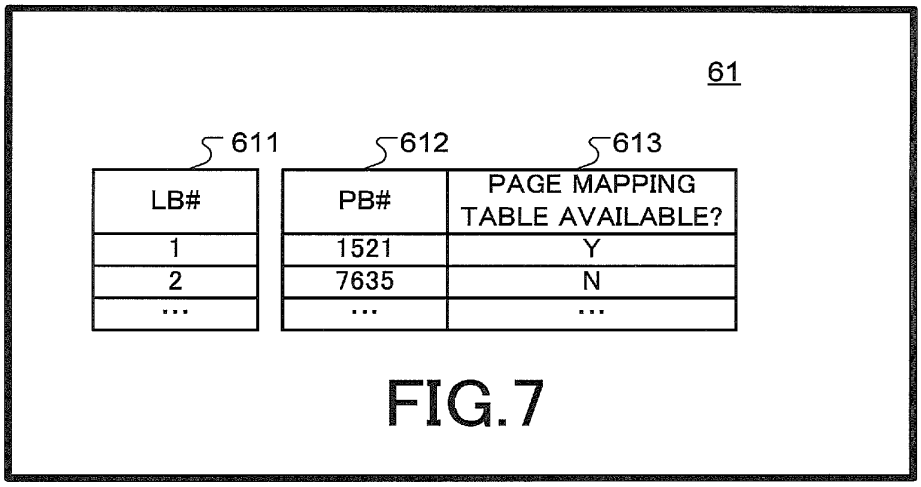


FIG.7

[Fig. 8]

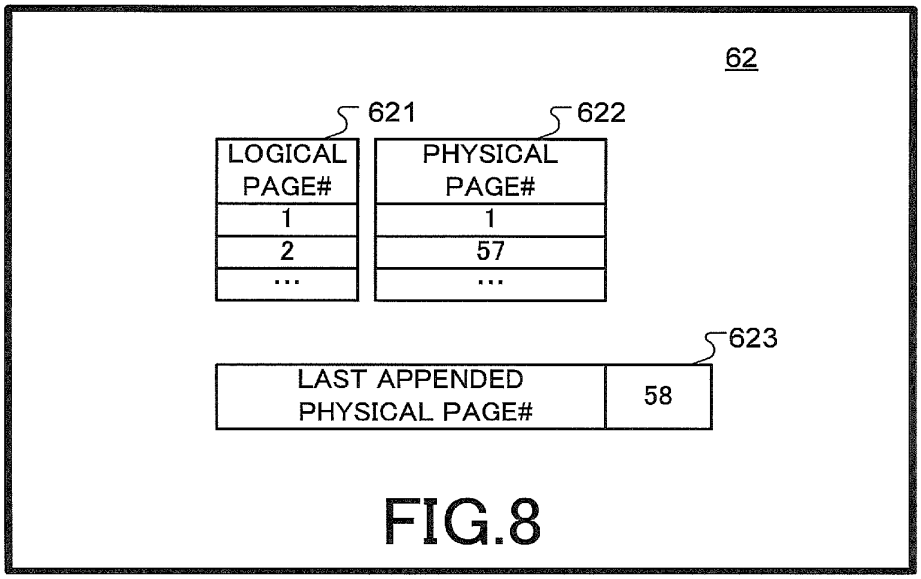


FIG.8

[Fig. 9]

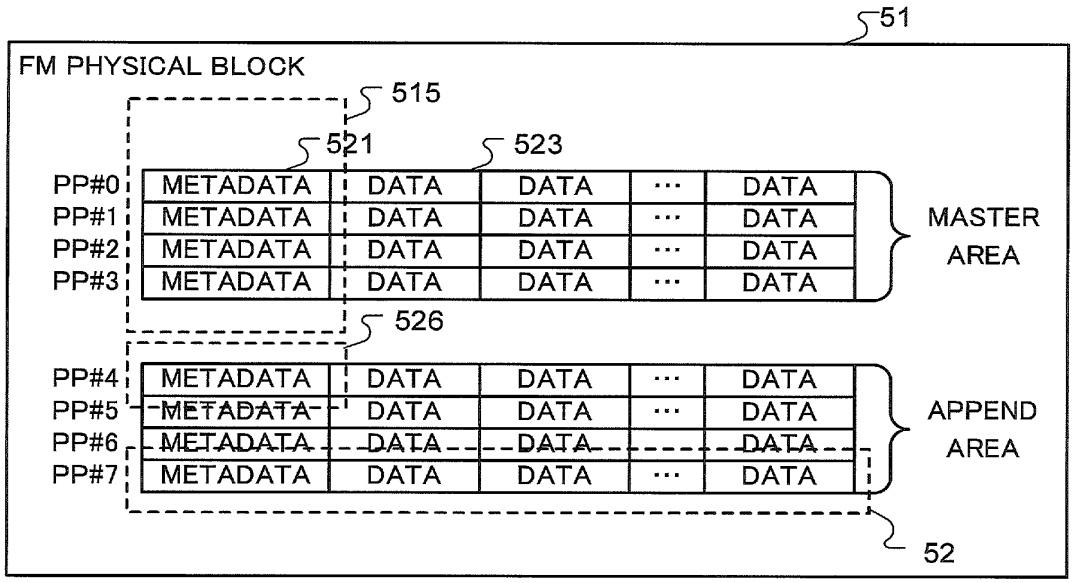


FIG.9

[Fig. 10]

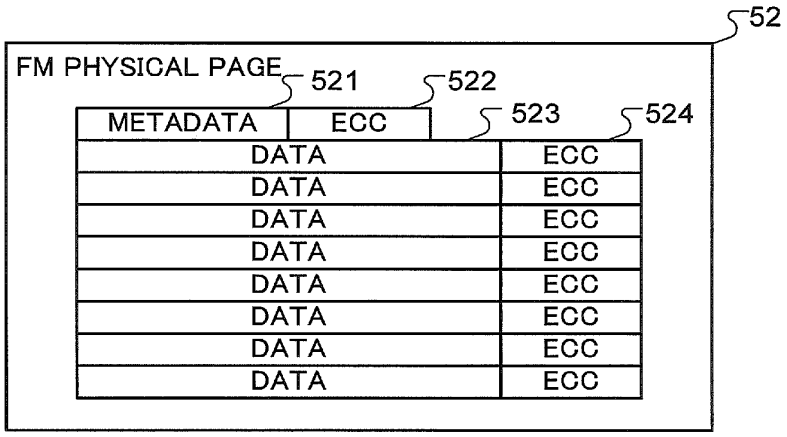


FIG.10

[Fig. 11]

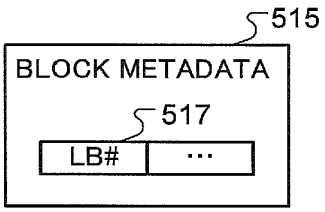


FIG.11

[Fig. 12]

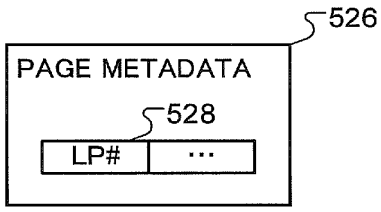


FIG.12

[Fig. 13]

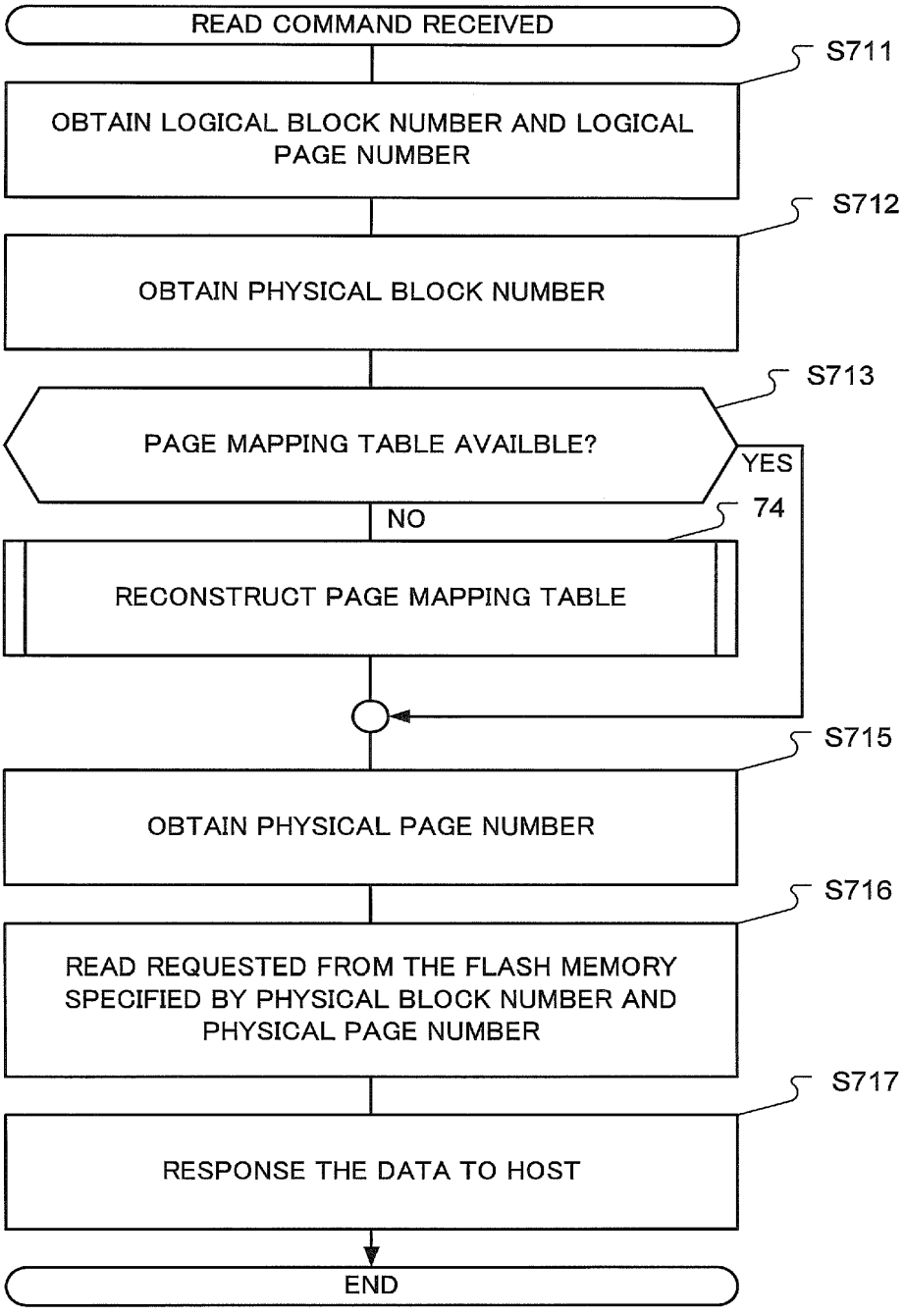

READ COMMAND RECEIVED
S711
OBTAIN LOGICAL BLOCK NUMBER AND LOGICAL PAGE NUMBER
S712
OBTAIN PHYSICAL BLOCK NUMBER
S713
PAGE MAPPING TABLE AVAILBLE?
YES
NO
74
RECONSTRUCT PAGE MAPPING TABLE
S715
OBTAIN PHYSICAL PAGE NUMBER
S716
READ REQUESTED FROM THE FLASH MEMORY SPECIFIED BY PHYSICAL BLOCK NUMBER AND PHYSICAL PAGE NUMBER
S717
RESPONSE THE DATA TO HOST
END
71

FIG.13

[Fig. 14]

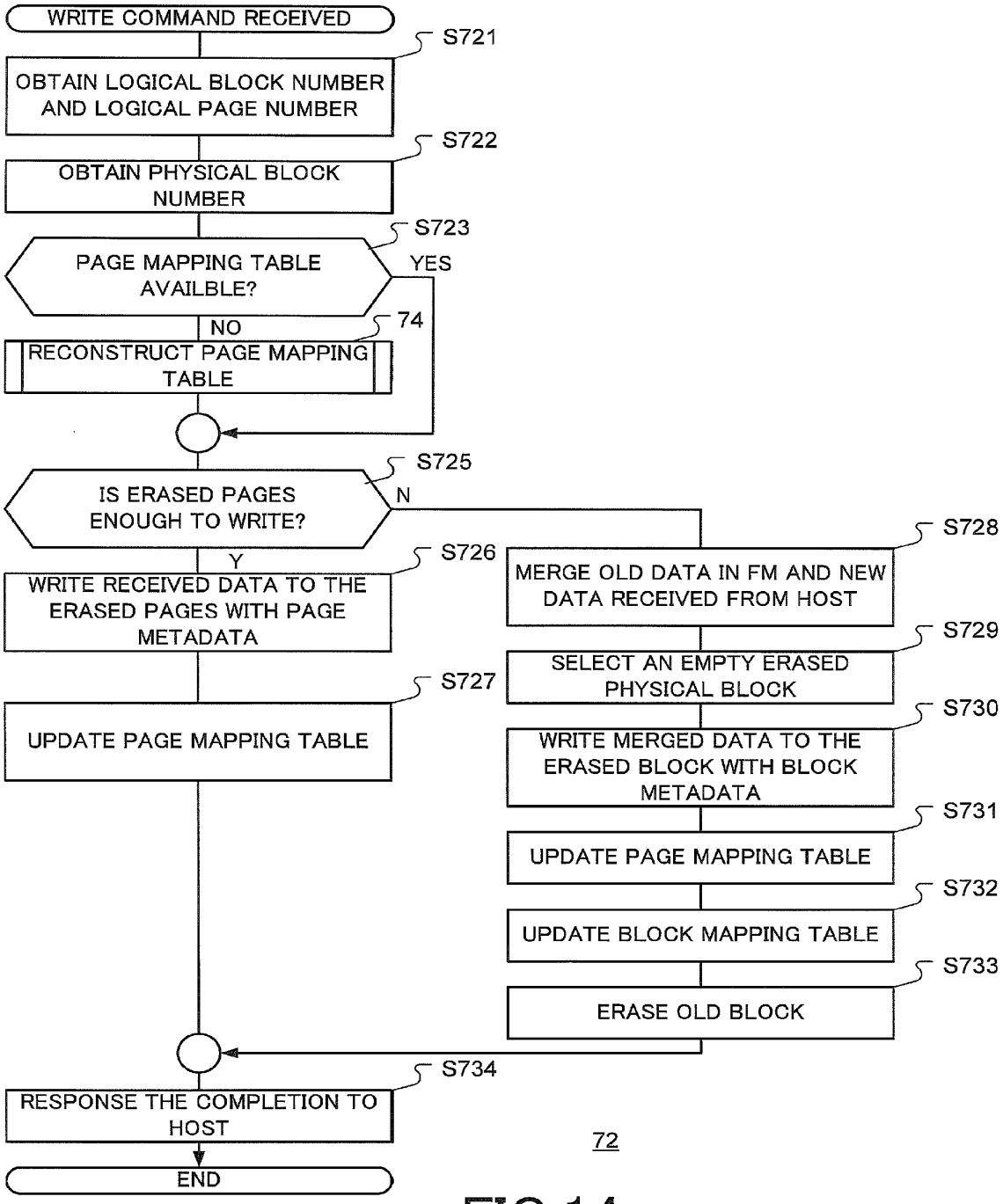

WRITE COMMAND RECEIVED
S721
OBTAIN LOGICAL BLOCK NUMBER AND LOGICAL PAGE NUMBER
S722
OBTAIN PHYSICAL BLOCK NUMBER
S723
PAGE MAPPING TABLE AVAILBLE?
YES
NO
74
RECONSTRUCT PAGE MAPPING TABLE
S725
IS ERASED PAGES ENOUGH TO WRITE?
N
Y
S726
WRITE RECEIVED DATA TO THE ERASED PAGES WITH PAGE METADATA
S727
UPDATE PAGE MAPPING TABLE
S728
MERGE OLD DATA IN FM AND NEW DATA RECEIVED FROM HOST
S729
SELECT AN EMPTY ERASED PHYSICAL BLOCK
S730
WRITE MERGED DATA TO THE ERASED BLOCK WITH BLOCK METADATA
S731
UPDATE PAGE MAPPING TABLE
S732
UPDATE BLOCK MAPPING TABLE
S733
ERASE OLD BLOCK
S734
RESPONSE THE COMPLETION TO HOST
END
72

FIG.14

[Fig. 15]

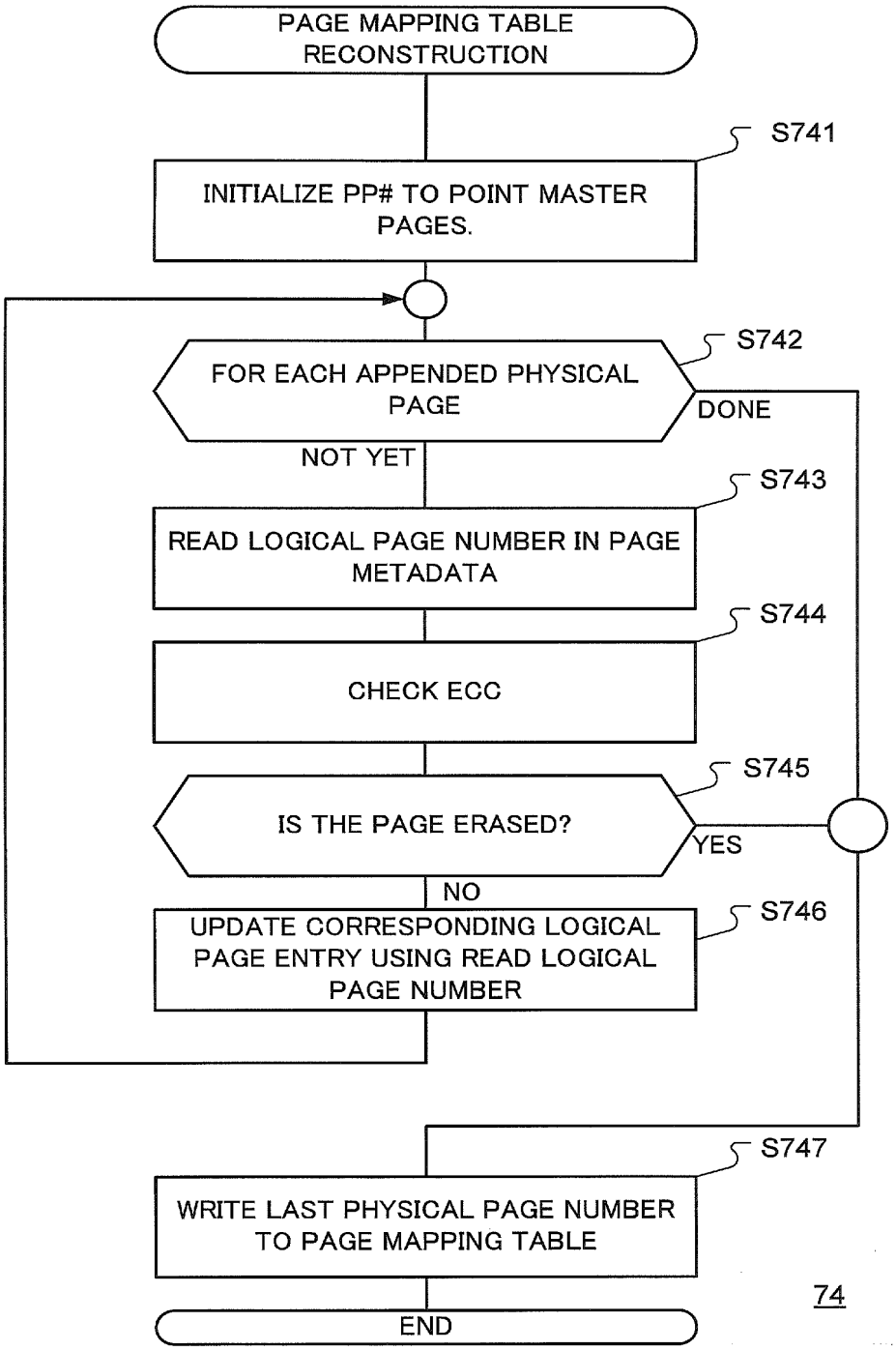

PAGE MAPPING TABLE RECONSTRUCTION
S741
INITIALIZE PP# TO POINT MASTER PAGES.
S742
FOR EACH APPENDED PHYSICAL PAGE
DONE
NOT YET
S743
READ LOGICAL PAGE NUMBER IN PAGE METADATA
S744
CHECK ECC
S745
IS THE PAGE ERASED?
YES
NO
S746
UPDATE CORRESPONDING LOGICAL PAGE ENTRY USING READ LOGICAL PAGE NUMBER
S747
WRITE LAST PHYSICAL PAGE NUMBER TO PAGE MAPPING TABLE
END
74

FIG.15

[Fig. 16]

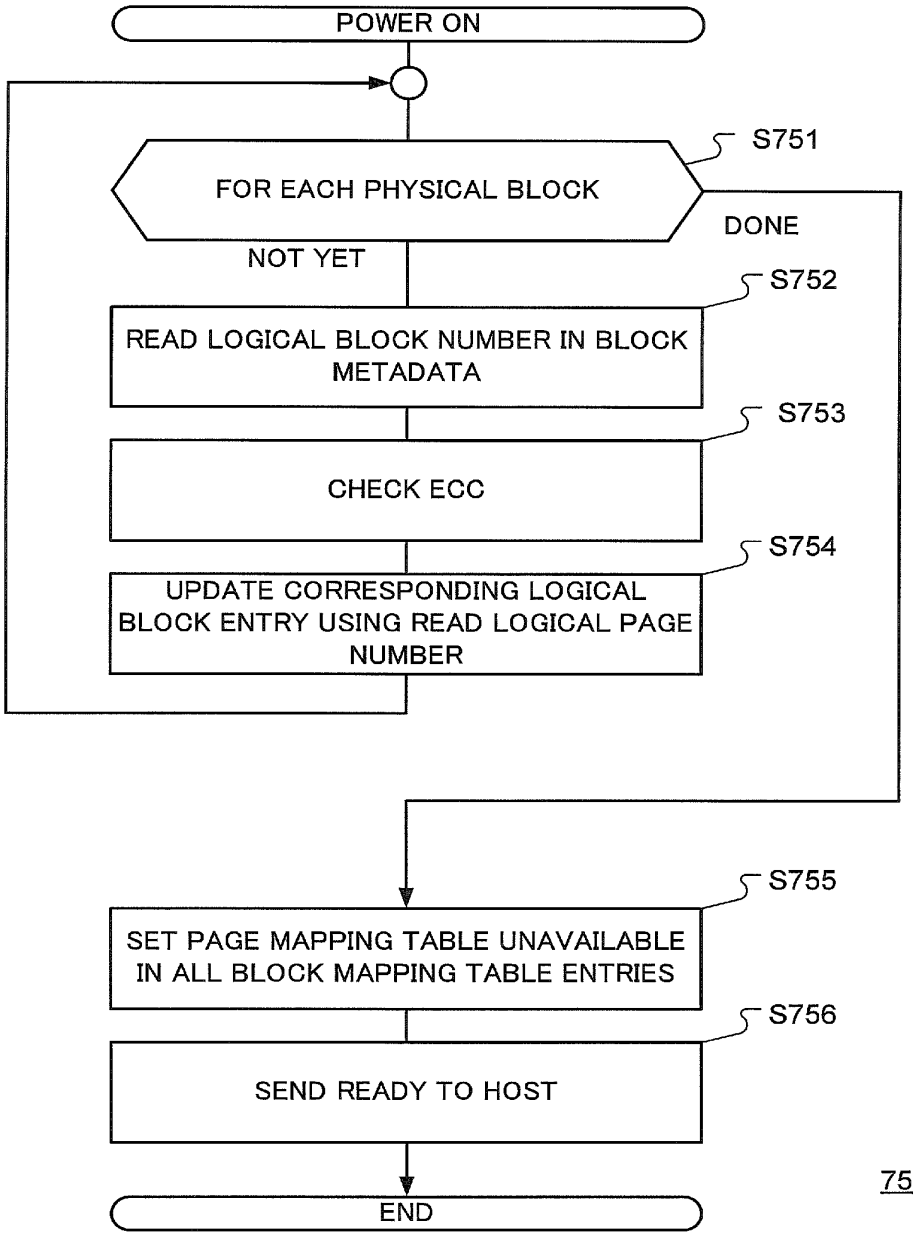

POWER ON
S751
FOR EACH PHYSICAL BLOCK
DONE
NOT YET
S752
READ LOGICAL BLOCK NUMBER IN BLOCK METADATA
S753
CHECK ECC
S754
UPDATE CORRESPONDING LOGICAL BLOCK ENTRY USING READ LOGICAL PAGE NUMBER
S755
SET PAGE MAPPING TABLE UNAVAILABLE IN ALL BLOCK MAPPING TABLE ENTRIES
S756
SEND READY TO HOST
END
75

FIG.16

[Fig. 17]

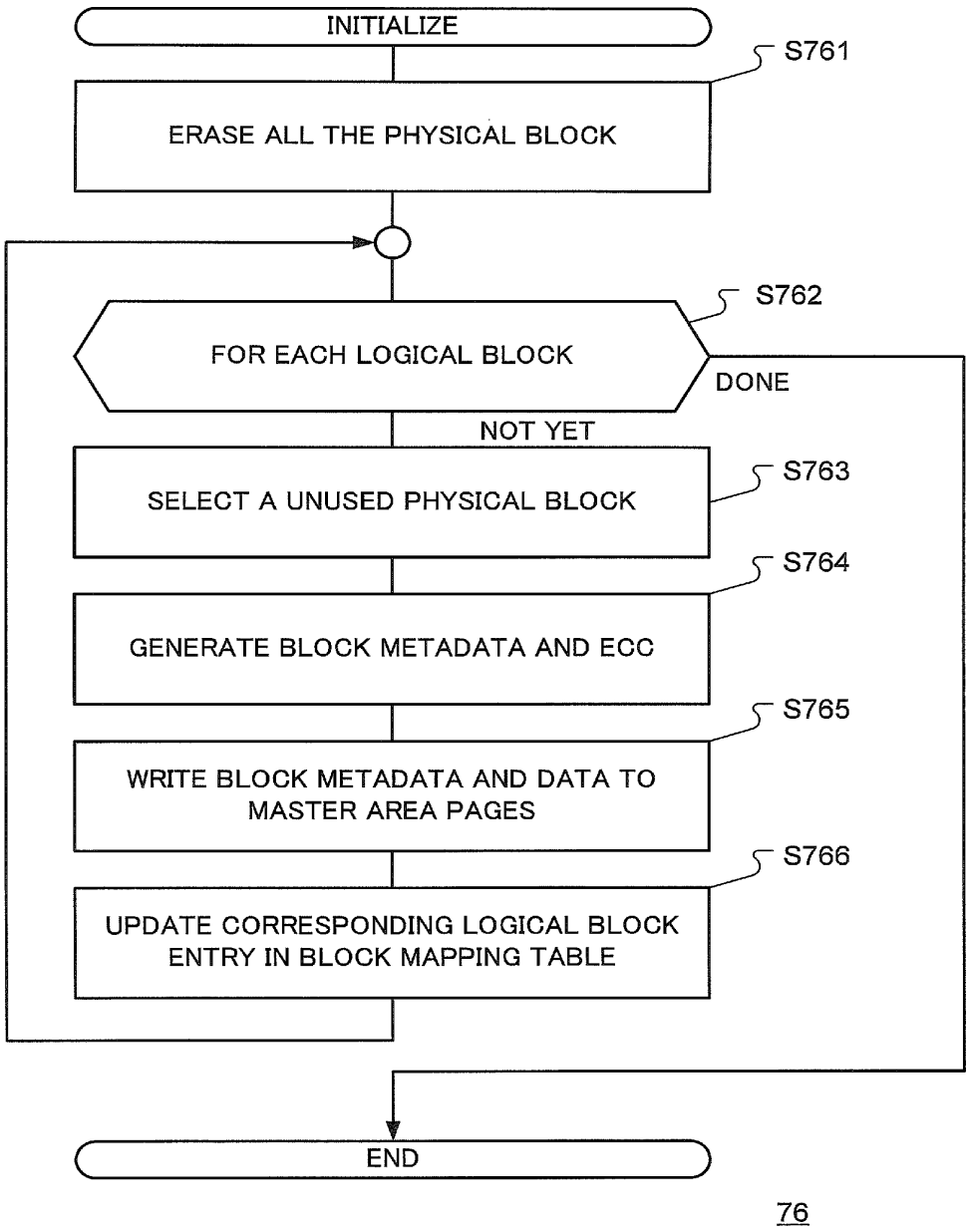

INITIALIZE
S761
ERASE ALL THE PHYSICAL BLOCK
S762
FOR EACH LOGICAL BLOCK
DONE
NOT YET
S763
SELECT A UNUSED PHYSICAL BLOCK
S764
GENERATE BLOCK METADATA AND ECC
S765
WRITE BLOCK METADATA AND DATA TO MASTER AREA PAGES
S766
UPDATE CORRESPONDING LOGICAL BLOCK ENTRY IN BLOCK MAPPING TABLE
END
76

FIG.17

[Fig. 18]

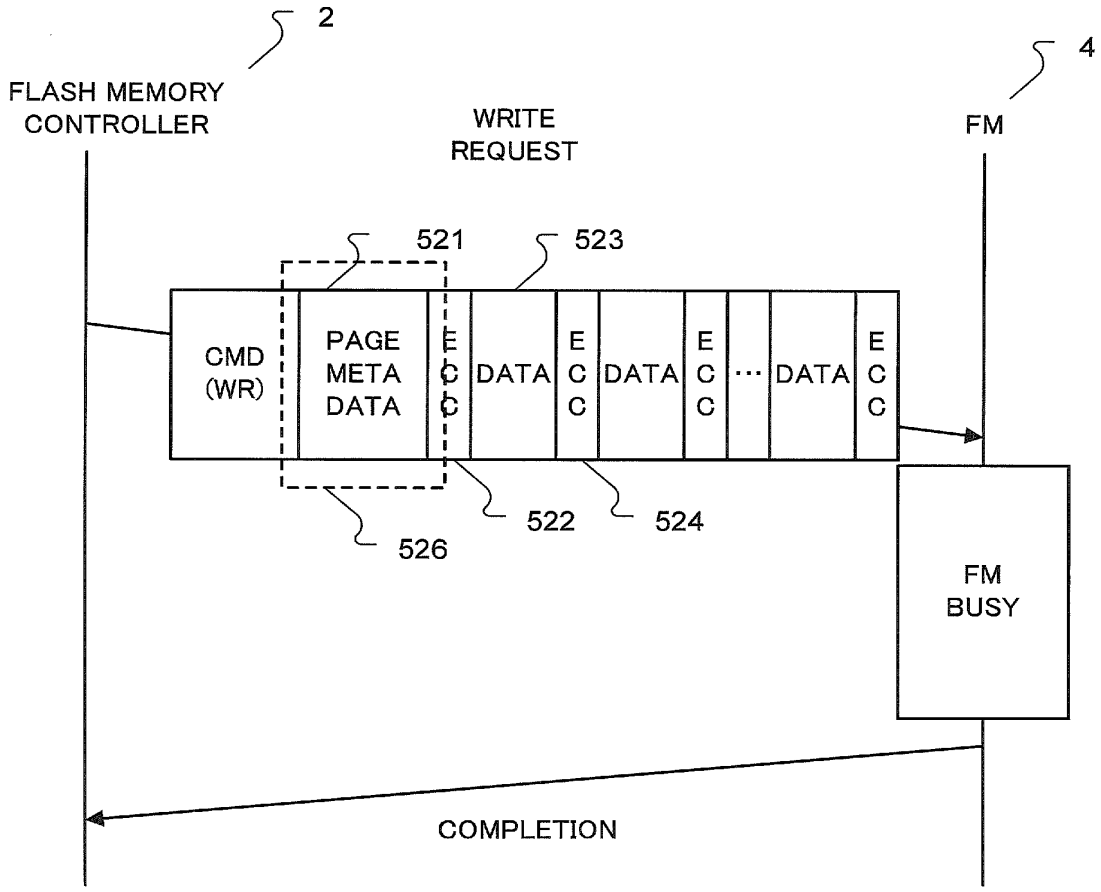

2
FLASH MEMORY CONTROLLER
WRITE REQUEST
4
FM
521
523
CMD (WR)
PAGE META DATA
ECC
DATA
ECC
DATA
ECC
···
DATA
ECC
522
524
526
FM BUSY
COMPLETION

FIG.18

[Fig. 19]

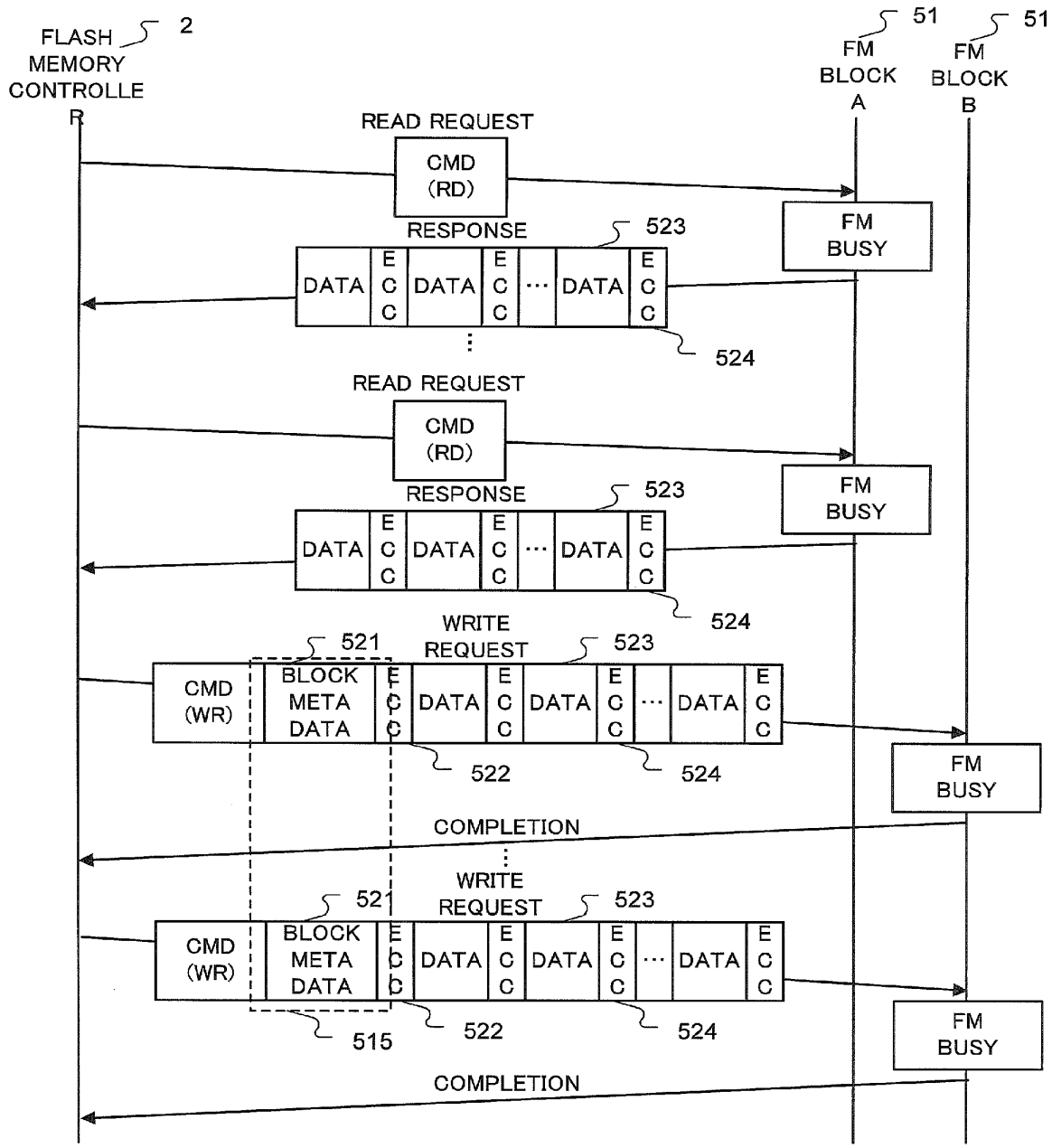

FLASH MEMORY CONTROLLER
2
FM BLOCK A
51
FM BLOCK B
51
READ REQUEST
CMD (RD)
FM BUSY
RESPONSE
523
DATA
ECC
524
READ REQUEST
CMD (RD)
FM BUSY
RESPONSE
523
524
WRITE REQUEST
521
523
CMD (WR)
BLOCK META DATA
522
524
FM BUSY
COMPLETION
WRITE REQUEST
521
523
515
522
524
FM BUSY
COMPLETION

FIG.19

[Fig. 20]

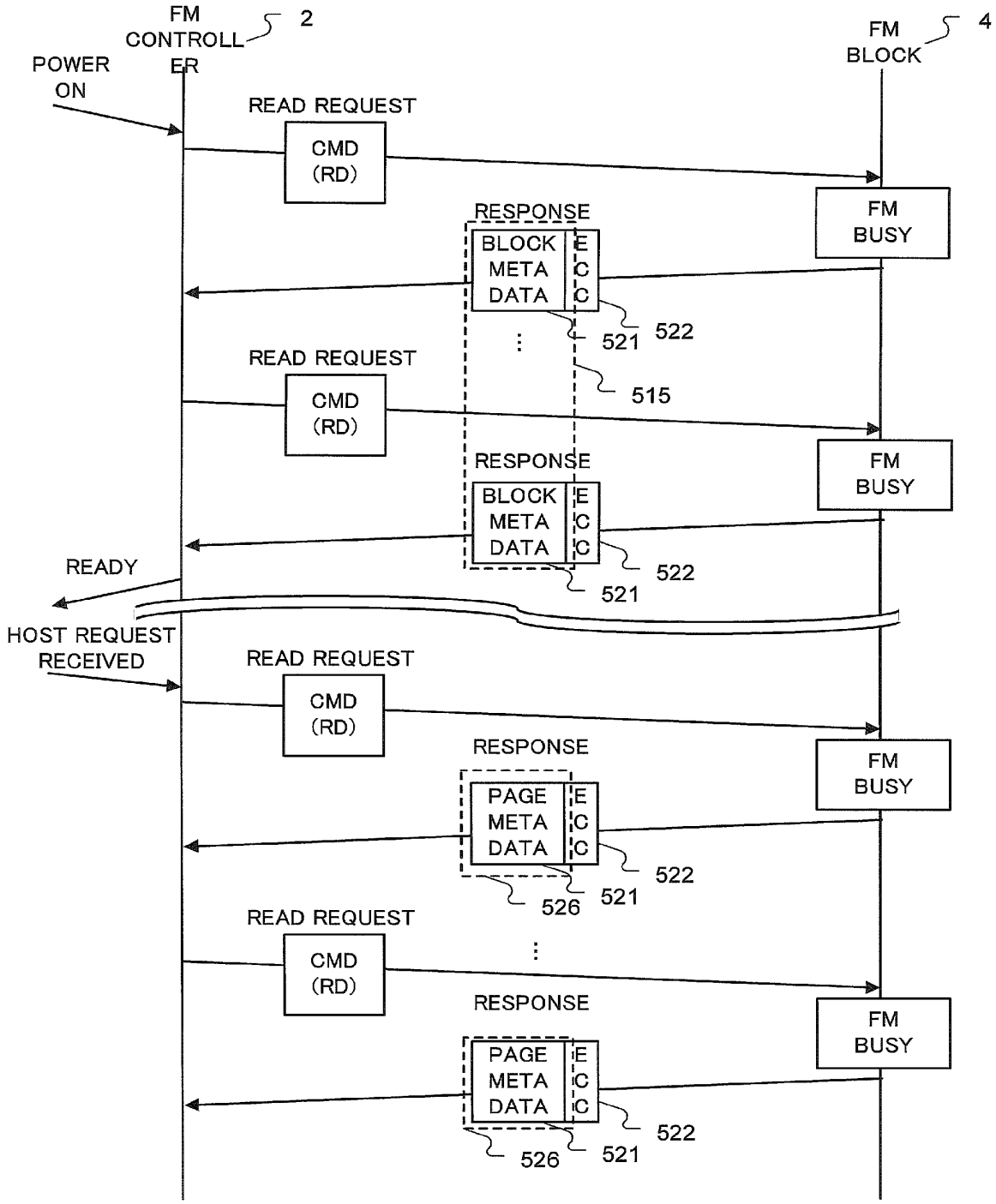

FM CONTROLLER
2
FM BLOCK
4
POWER ON
READ REQUEST
CMD (RD)
FM BUSY
RESPONSE
BLOCK META DATA
ECC
521
522
515
READY
HOST REQUEST RECEIVED
PAGE META DATA
526

FIG.20

[Fig. 21]

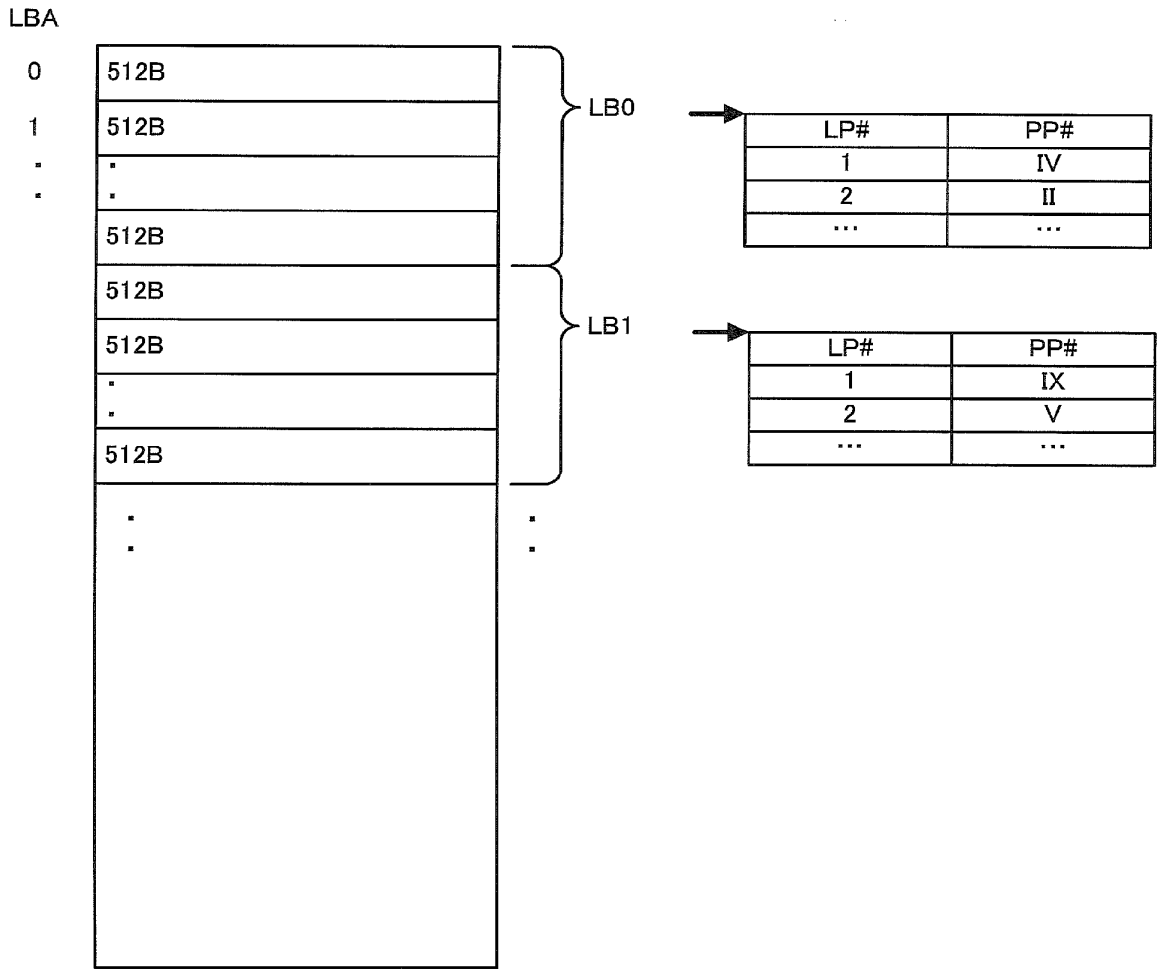

LBA
0
1
512B
512B
512B
512B
512B
512B
512B
LB0
LB1
LP#
PP#
1
IV
2
II
LP#
PP#
1
IX
2
V

FIG.21

INTERNATIONAL SEARCH REPORT

International application No
PCT/JP2009/005859

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F12/02
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/109376 A1 (LIN JIN-SHIN [TW]) 10 June 2004 (2004-06-10) paragraph [0009] - paragraph [0011]	1,6-15
X	US 2007/300037 A1 (ROGERS ANDREW [US] ET AL) 27 December 2007 (2007-12-27) paragraph [0005] - paragraph [0007] figure 1	1,12,15
A	EP 1 939 750 A1 (NETAC TECHNOLOGY CO LTD [CN]) 2 July 2008 (2008-07-02) paragraph [0007] - paragraph [0010]	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
13 July 2010	21/07/2010

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Knutsson, Frédéric

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/JP2009/005859

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004109376 A1	10-06-2004	NONE	
US 2007300037 A1	27-12-2007	NONE	
EP 1939750 A1	02-07-2008	WO 2007033614 A1	29-03-2007
		CN 1936867 A	28-03-2007
		JP 2009510552 T	12-03-2009
		KR 20080067334 A	18-07-2008
		US 2009313417 A1	17-12-2009
		ZA 200803637 A	28-10-2009

Form PCT/ISA/210 (patent family annex) (April 2005)

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 April 2011 (14.04.2011)

PCT

(10) International Publication Number
WO 2011/044154 A1

(51) International Patent Classification:
G06F 12/02 (2006.01) *G06F 12/08* (2006.01)

(21) International Application Number:
PCT/US2010/051514

(22) International Filing Date:
5 October 2010 (05.10.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/248,845 5 October 2009 (05.10.2009) US

(71) Applicant *(for all designated States except US)*: **MARVELL SEMICONDUCTOR, INC.** [US/US]; 5488 Marvell Lane, Santa Clara, California 95054 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **DESHKAR, Shekhar S.** [IN/IN]; Flat No. 11, Bandhangad Apartments, D.P. Road, Aundh, Pune 411007 (IN). **KARMARKAR, Sandeep** [IN/IN]; 401 Sairang Near Paranjape School, Kothrud, Pune 411038 (IN). **PRUTHI, Arvind** [US/US]; 741 Pollard Road, Apt. 1, Los Gatos, California 95032 (US). **JOHRI, Ram Kishore** [US/US]; 1526 Miller Ave., San Jose, California 95129 (US).

(74) Agents: **HUNTER, William E.** et al.; Fish & Richardson P.C., P. O. Box 1022, Minneapolis, Minnesota 55440-1022 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

[Continued on next page]

(54) Title: DATA CACHING IN NON-VOLATILE MEMORY

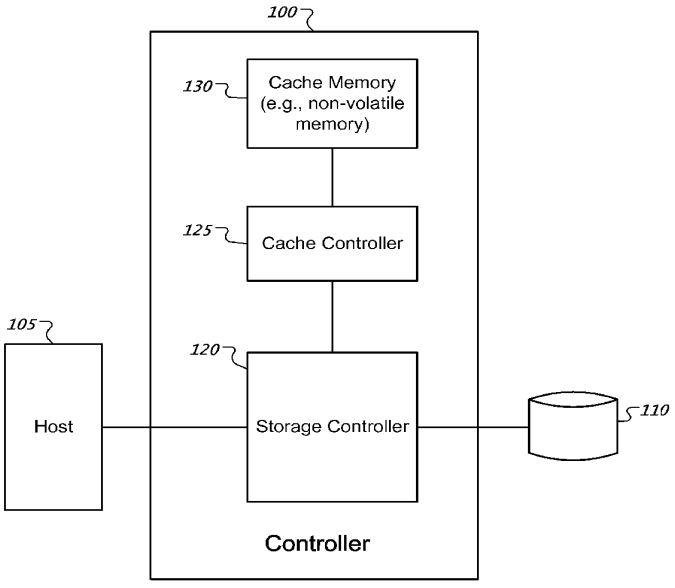


FIG. 1

(57) Abstract: Methods, systems, and apparatus, including computer programs encoded on a computer storage medium, can perform data caching. In some implementations, a method and system include receiving information that includes a logical address, allocating a physical page in a non-volatile memory structure, mapping the logical address to a physical address of the physical page, and writing, based on the physical address, data to the non-volatile memory structure to cache information associated with the logical address. The logical address can include an identifier of a data storage device and a logical page number.

WO 2011/044154 A1

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

DATA CACHING IN NON-VOLATILE MEMORY

CROSS REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of the priority of U.S. Provisional Application Serial No. 61/248,845, filed October 5, 2009 and entitled "Flash Based Cache with a Cache Aware Flash Translation Layer." The content of the above application is incorporated by reference in its entirety.

BACKGROUND

[0002] The present disclosure describes systems and techniques relating to data caching.

[0003] A host system can communicate with a storage system to read and write data. A host system can include a computer system, such as a personal computer, a consumer electronics device, or a server system. A storage system can include one or more data storage devices such as a hard disk drive (HDD) or a solid state drive (SSD) communicating with the host system. In some implementations, a host system includes the storage system, and in other implementations, the storage system is external to the host system. A storage system can perform data caching to decrease average access latency times.

SUMMARY

[0004] The present disclosure includes systems and techniques relating to data caching. According to an aspect of the described systems and techniques, a system includes a host system, a data storage device, a non-volatile memory structure that is configured to cache data with respect to the data storage device, and a controller configured to operate devices such as the data storage device. The controller can receive a logical address from the host system, allocate a physical page in the non-volatile memory structure, store a mapping from the logical address to a physical address of the physical page, receive data associated with the

logical address, and write, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address. The mapping can indicate a cached status of the data associated with the logical address.

[0005] In some implementations, a controller is configured to receive the data associated with the logical address from one of the host system or the data storage device. In some implementations, a controller is configured to invalidate the mapping based on the data associated with the logical address being written to a different physical page. A controller can perform an erasure of a physical page associated with an invalidated mapping to make the physical page available for re-allocation. In some implementations, a controller is configured to invalidate an allocated physical page based a logical address indicated by an invalidate command sequence. In some implementations, a logical address includes an identifier of a data storage device and a logical page number. A system can include two or more data storage devices. The non-volatile memory structure can be configured to cache data with respect to the two or more data storage devices. A mapping can include a data storage device identifier that identifies at least one of the two or more data storage devices. In some implementations, the controller includes a storage controller and a cache controller, and the cache controller is communicatively coupled with the non-volatile memory structure. A non-volatile memory structure can include one or more flash memory structures. A data storage device can include one or more hard disk drives. A system can include caching devices that include non-volatile memory structures. In some implementations, the controller is configured to select a caching device based on the logical address.

[0006] According to another aspect of the described systems and techniques, a method includes receiving, from a host system, information that includes a logical address. The method includes receiving data associated with the logical address from one of the host

system or a data storage device. The method includes allocating a physical page in a non-volatile memory structure. In some implementations, the non-volatile memory structure is configured to cache data with respect to at least one data storage device. The method includes storing a mapping from the logical address to a physical address of the physical page. The method includes writing, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address. The mapping can indicate a cached status of the data associated with the logical address.

[0007] In some implementations, a method includes invalidating the mapping based on the data associated with the logical address being written to a different physical page and performing an erasure of the physical page associated with the invalidated mapping to make the physical page available for re-allocation. Receiving data associated with the logical address can include reading data from the data storage device based on the logical address. In some implementations, a method includes invalidating an allocated physical page based a logical address indicated by an invalidate command sequence.

[0008] The described systems and techniques can be implemented in electronic circuitry, computer hardware, firmware, software, or in combinations of them, such as the structural means disclosed in this specification and structural equivalents thereof. This can include at least one computer-readable medium embodying a program operable to cause one or more data processing apparatus to perform operations described. Thus, program implementations can be realized from a disclosed method, system, or apparatus, and apparatus implementations can be realized from a disclosed system, computer-readable medium, or method. Similarly, method implementations can be realized from a disclosed system, computer-readable medium, or apparatus, and system implementations can be realized from a disclosed method, computer-readable medium, or apparatus.

[0009] An apparatus can include a non-volatile memory structure, circuitry to receive a command sequence that includes a logical address, circuitry to receive data associated with the logical address, circuitry to allocate a physical page in the non-volatile memory structure, circuitry to store a mapping from the logical address to a physical address associated with the physical page, and circuitry to write, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address. The mapping can indicate a cached status of the data associated with the logical address. The logical address can include information that identifies a data storage device and a logical page number.

[0010] In some implementations, an apparatus can include circuitry to receive a read command sequence that includes a logical address, circuitry to identify a physical address associated with the logical address of the read command sequence, and circuitry to read, based on the identified physical address, data from the non-volatile memory structure to provide cached data associated with the logical address of the read command sequence. An apparatus can include circuitry to invalidate an allocated physical page based a logical address indicated by an invalidate command sequence.

[0011] The described systems and techniques can result in a reduced latency storage system. For example, using a mapping that relates a logical address to a physical address of a page on a non-volatile memory can reduce a number of lookups required to read data from a cache data store. In addition, rather than maintaining multiple separate lookup tables, a single lookup table can be employed to more efficiently use memory in a cache data store. This approach to data caching in a storage system can result in reduced latency.

[0012] Details of one or more implementations are set forth in the accompanying drawings and the description below. Other features, objects, and advantages may be apparent from the description and drawings, and from the claims.

DRAWING DESCRIPTIONS

[0013] FIG. 1 shows an example of a storage architecture.

[0014] FIG. 2 shows an example of a cache controller architecture.

[0015] FIG. 3 shows an example of a block and page layout of a non-volatile memory.

[0016] FIG. 4 shows an example of a cache based storage process.

[0017] FIG. 5 shows an example of a mapping table layout.

[0018] FIG. 6 shows an example of a cache aware flash translation layer process.

[0019] FIGS. 7A and 7B show examples of different cache processes.

[0020] FIG. 8 shows an example of a storage system with multiple flash memory structures.

[0021] FIG. 9 shows an example of a storage process.

[0022] Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0023] FIG. 1 shows an example of a storage architecture. A host 105 can access one or more storage devices 110 via a controller 100. A controller 100 can include a storage controller 120, a cache controller 125, and one or more memories such as a cache memory 130. In some implementations, a storage device 110 includes one or more storage elements that are separately addressable by the controller 100.

[0024] A host 105 can send a command sequence such as a write command sequence or a read command sequence to the storage controller 120. A command sequence can include a logical address. In some implementations, a logical address includes a storage device target identifier that is associated with a storage device 110. In some implementations, a command sequence can include a command identifier, a logical address, and a length. In some implementations, a host 105 communicates with a storage controller 120 via an interface such

as a Peripheral Component Interconnect (PCI), Serial Advanced Technology Attachment (SATA), Small Computer System Interface (SCSI), Serial Attached SCSI (SAS), Ethernet, or Universal Serial Bus (USB).

[0025] A storage controller 120 can communicate with a cache controller 125 to cache data received from a storage device 110, cache data received from the host 105, or both. The cache controller 125 can communicate with a cache memory 130, which can include non-volatile memory such as flash memory. In some implementations, the cache controller 125 includes a flash memory controller to control flash memory in the cache memory 130. For example, a storage controller 120 can read data from a storage device 110 and store at least a portion of the read data in a cache memory 130 controlled by a cache controller 125. In another example, a storage controller 120 can receive data from a host 105 for writing to a storage device 110. A storage controller 120 can store write data, received from the host 105, in a cache memory 130 controlled by a cache controller 125. In some implementations, a storage controller 120 stores host write data on a cache memory 130 and is not required to immediately store the host write data on a storage device 110. In some implementations, a storage controller 120 stores host write data on a cache memory 130 and on a storage device 110.

[0026] A host 105 can include one or more processors, memory, and input/output (I/O) connections. In some implementations, a host 105 includes a storage controller 120. For example, a system-on-a-chip (SOC) can include a general purpose processor and one or more controllers (e.g., storage controller, cache controller, or a combination thereof) to access external devices such as hard disk drives (HDDs) or solid-state drives (SSDs). A SSD can include one or more non-volatile memory structures such as a non-volatile memory integrated circuit, e.g., flash chip. A cache controller 125 can include one or more processors and one or more non-volatile memory. In some implementations, a storage controller 120

includes one or more cache controllers 125. In some implementations, a storage controller 120 is integrated with a cache controller 125 and one or more storage devices 110.

[0027] FIG. 2 shows an example of a cache controller architecture. A cache controller 205 can cache data for a storage controller. Caching data can increase performance, e.g., faster read times and faster write times. A cache controller 205 can communicate with one or more non-volatile memory chips such as flash chips 210 to store or retrieve cache data. A cache controller 205 can communicate with a controller memory 215, which stores cache metadata such as address mappings for cached data in the flash chips 210. In some implementations, a controller memory 215 is integrated into a memory region of a flash chip 210, e.g., a portion of memory in a flash chip is configured as a memory for the cache controller 205.

[0028] The cache controller 205 can store information such as cache metadata on the controller memory 215. For example, the cache controller 205 can store one or more mappings in a mapping table 220 to facilitate caching operations. A mapping, stored on the mapping table 220, identifies a relationship between a logical address (LA) and a physical address (PA) corresponding to a physical page stored on at least one of the flash chips 210. A logical address can include a logical page number (LPN) associated with a data location on a storage device. In some implementations, a LPN can be referred to as a logical block address (LBA). In some implementations, a LPN can include an offset and a length. A physical address can include a block identifier and a page identifier.

[0029] In some implementations, a mapping table 220 can include logical addresses that include a LPN and a storage device target identifier that is associated with a data storage device. A storage device target identifier can differentiate identical LPNs. For example, two data storages devices can store different data using identical LPNs. A controller can use a

storage device target identifier to retrieve data associated with a LPN from a corresponding storage device.

[0030] The cache controller 205 can receive a command sequence from a storage controller to retrieve data from cache. The command sequence can identify a logical address. If the cache controller 205 has a cache hit, e.g., the command sequence's logical address is found in the mapping table 220, then the cache controller 205 uses the corresponding physical address to retrieve data from a flash chip 210 corresponding to the physical address. The cache controller 205 can send the retrieved data to the storage controller.

[0031] If the cache controller 205 has a cache miss, e.g., a command's logical address is not found in the mapping table 220, then the cache controller 205 can report the miss to a storage controller. In some implementations, the cache controller 205 can report the miss to a program such as firmware. However, the cache controller 205 can allocate a physical page in a flash chip 210 to store a mapping in the mapping table 220 that identifies a relationship between the command's logical address and a physical address of the allocated physical page. After the storage controller services the cache miss, e.g., retrieves data corresponding to the logical address from a storage device, the cache controller 205 can receive the data and store the data in one or more of the flash chips 210 for a future read access.

[0032] In another example, the cache controller 205 receives a command to write data to the cache. The command identifies a logical address. If the cache controller 205 has a cache hit, the cache controller 205 uses a corresponding physical address to write data to a flash chip 210 corresponding to the physical address. In some implementations, a new location is required for a future write to the same physical page. Therefore, the cache controller 205 can allocate a new physical page and update the mapping table 220 to reflect the new physical page assignment for the logical address. In some implementations, the cache controller 205 can mark the old physical page invalid.

[0033] FIG. 3 shows an example of a block and page layout of a non-volatile memory. A non-volatile memory layout 300 includes one or more blocks 305. A block 305 includes one or more pages 310. For example, a block can be 256 KB in size and a page can be 4 KB in size, and accordingly, a block can include 64 pages. In another example, a block can be 512 KB in size and a page can be 4 KB in size, hence, a block can include 128 pages.

[0034] A storage controller can receive information, such as a command sequence from a host system. A command sequence can include a logical address. For a write command, a command sequence can include data. In some implementations, a logical address includes an identifier of a data storage device and a logical page number. A storage controller can run a cache based storage process. In some implementations, a storage controller operates a cache controller to run at least a portion of a cache based storage process. For example, the storage controller can send a command sequence to a cache controller based upon a command sequence received from the host system.

[0035] FIG. 4 shows an example of a cache based storage process. In some implementations, a cache based storage process includes operations for a storage controller and operations for a cache controller that is in communication with the storage controller. In some implementations, the process runs on a controller such as a storage controller or an integrated storage and cache controller. At 405, a cache based storage process can receive a command sequence that includes a logical address from a host system. In some cases, a command sequence indicates that data are to be retrieved for the host system. In some cases, a command sequence indicates a write operation and includes host write data.

[0036] At 410, the cache based storage process can access a mapping table to determine if a mapping associated with the logical address exists. For example, the process can perform a table lookup based on the logical address. In some implementations, the process uses a hash table to determine whether a mapping associated with the logical address exists. In some

implementations, a cache controller stores a table in a controller memory and uses the table to perform a lookup based on a logical address.

[0037] At 415, if a mapping does not exist, then the process, at 420, can allocate a physical page in a non-volatile memory structure. The process, at 425, can generate a mapping to relate the logical address with a physical address of the physical page. In some implementations, a physical address of a physical page is a starting address of the physical page. At 430, the process can store the mapping in the mapping table. The mapping can be indicative of a cached status of the data associated with the logical address. For example, the existence of a valid mapping table entry can indicate the presence of cached data. At 435, the process can write, based on the physical address, data to the non-volatile memory structure to cache information associated with the logical address. For example, a process can receive a command sequence that includes host write data. The process can cache host write data in the non-volatile memory structure.

[0038] At 415, if the mapping does exist, then the process, at 440, can use the existing mapping to handle a data operation with the non-volatile memory structure. Various examples of data operations include a write operation and a read operation. In some implementations, the process uses the existing mapping to read data from the non-volatile memory structure. In some implementations, the process uses the existing mapping to write data to the non-volatile memory structure. In some implementations, using the existing mapping to write data can include invalidating the existing mapping and creating a new mapping to write data to a different physical location in the non-volatile memory structure.

[0039] FIG. 5 shows an example of a mapping table layout. A mapping table 500 layout is arranged to store one or more entries 505. A mapping table 500 is partitioned into logical addresses 510 and respective physical addresses 515. For example, an entry 505 includes a logical address 510 and a physical address 515. In some implementations, a logical address

includes an identifier such as a target device identifier (e.g., TGT 1, TGT 2) that identifies a target device. For example, a logical address can include a target device identifier and a LPN. With respect to a read command sequence, a controller can read, using a LPN, a page from a device corresponding to a target device identifier included in the sequence. The controller can cache the page. Caching the page can include creating a mapping, in the mapping table 500, that includes the target device identifier and the LPN. In some implementations, an entry 505 includes an active flag to indicate whether the entry 505 is in use. An entry 505 can include a valid flag to indicate whether a physical page associated with a physical address 515 is valid or invalid. For example, an entry 505 is marked as being in use and the valid flag is marked as invalid. An entry 505 that is marked as being in use and the valid flag is marked as valid can indicate a cache status of true.

[0040] A storage system can include one or more non-volatile memories configured to cache information flowing to a storage device or coming from a storage device. Non-volatile memories such as flash memory can be electronically erased and reprogrammed. In some implementations, flash memory can include NAND logic.

[0041] In some implementations, flash memory operates based on erase blocks and pages. For example, a unit of access is a page, whereas a unit of an erase operation is an erase-block. Sizes of blocks and pages can vary with different flash devices. For example, in relation to a flash device included in a large-block flash SSD, an erase block size is between 256KB to 512KB in size and a page is 4KB in size. However, other sizes are possible.

[0042] In some implementations, a flash block, which already contains data, cannot be re-written unless the block is explicitly erased. An erase operation can be time expensive, e.g., can take several milliseconds. In some implementations, for large block NAND devices, the pages within an erase block cannot be written randomly. For example, an erase block includes 100 pages numbered from 1 to 100. In some implementations, if page 20 within the

erase block is written, then pages 1-20 cannot be written to unless the entire block is erased. However, pages after page 20 are write-able.

[0043] In some flash memory designs, a block can be erased only a finite number of times before it becomes unusable due to one or more physical characteristics of flash memory. A flash memory device can use wear-leveling algorithms to distribute a load on multiple flash chips. A flash memory device can use wear-leveling algorithms to make sure that usage of erase blocks within a flash memory is rotated such that no single flash erase block gets burdened with too many write cycles.

[0044] An application may write and subsequently rewrite data to the same logical page. The logical page can be mapped to a physical page stored in a flash memory. Rewriting a page to the same location can require a flash device to read the entire erase block into memory, erase the entire block, rewrite data from memory to pages prior to the page that needs to be updated, update the relevant page, and rewrite the remaining pages in the block to the flash. The operations for rewriting a page to the same location can be expensive in terms of time. Further, there is a possibility of permanent data loss if the flash device is powered off before the rewriting process is completed. However, rewriting in place may not ensure a uniform wear of flash blocks.

[0045] A flash device can re-write a page to a different location to increase operational latency and provide a more distributed wear-leveling of the flash memory. For example, a flash device can mark the page as invalid and write the updated page to a different location. In some implementations, a flash device can use an abstraction layer of logical pages on top of the physical space provided by a flash memory structure. The layer is called as a flash translation layer (FTL). In some implementations, commands sent to the flash devices can use logical addresses to refer to data. In some implementations, a command includes a LPN. The FTL can assign a physical page number to a logical page number using logic internal to

the FTL. For re-writing, the FTL can be configured to re-write pages to a different location. Hence, a flash may have multiple physical pages that are associated with the same logical page of which only one physical page is deemed valid and the rest are deemed invalid. A flash device can use FTL metadata to track which physical locations contain valid data.

[0046] FTL metadata can include a table that stores a mapping table of LPNs and respective physical page locations. In some implementations, a LPN is a logical page address exported via interface such as SAS, SATA, or PCI. The mapping table can maintain the latest physical location that has valid contents for a corresponding LPN. FTL metadata can include a table that describes which physical pages have valid data.

[0047] A flash device can perform garbage collection to reclaim invalidated pages such that they can be reused for a future write operation. In some implementations, FTL logic requires a free block to process an update to a logical page. In some implementations, a processor within the flash controller runs a garbage collection process in the background. A garbage collection process can search for erase blocks and run a de-fragmentation operation before making the page available for write. In some implementations, the process searches for blocks based on a number of invalidated pages contained therein. A de-fragmentation process can collect the valid pages from various blocks in memory, rewrite them into a new erase block, update the metadata mapping, and erase the old blocks. The erased blocks are now available for a new write. A flash device can perform on-demand garbage collection in the event that no free physical pages exist to satisfy a write operation. However, to reduce the likelihood of on-demand garbage collection, a flash controller can declare a lesser amount of storage space than what is physically available.

[0048] A storage system can include a flash device that is configured as a caching device. A caching device can include a cache-aware FTL. Rather than supporting a direct mapping between LPNs and physical storage space, such as when a flash device is used as a target-

device, a caching device can provide caching for multiple target devices with a logical-page range independent of the actual physical space provided by the flash device.

[0049] In some implementations, a caching device can provide commands such as GET and SET commands to manipulate the device. Such commands can accept a target device identifier, e.g., TGT-ID, in addition to a logical page address and transfer length fields. In some implementations, a logical address includes a TGT-ID and a logical page number. The TGT-ID can identify a target device associated with data sent or received via a command. A SET operation can include caching the supplied data, identified by offset and length, for a specified TGT-ID. Caching the supplied data can include storing the supplied data on a flash memory structure. A GET operation can include fetching the data identified by an offset and length specified in the GET call for a specified TGT-ID.

[0050] A cache aware FTL mapping table can include a mapping of an application specified logical address, such as <TGT-ID, LPN>, to a physical location description within a flash memory structure. A physical location description can identify a flash block and a page within the flash block. In some cases, an entry in a mapping table may include a logical address that does not have a corresponding valid physical page (e.g., a physical location description is marked as "invalid"). Such an entry can be eliminated from the mapping table.

[0051] A caching flash device can provide an interface to INVALIDATE a cache mapping. An INVALIDATE command can be used when an application no longer requires a page stored in the cache. A cache aware FTL can track physical page status, e.g., valid or invalid. A cache aware FTL can continue to do garbage collection to maintain a pool of erased blocks.

[0052] In response to a GET call, a cache aware FTL can directly lookup an application supplied logical address in the FTL's internal mappings to check if the data exists within the flash. If not, the FTL can send out an error message.

[0053] In response to a SET call, a cache aware FTL can check if a free page is available to accommodate an incoming write request indicated by the SET call. If a free page does exist, the FTL can transfer the contents of the data to the available page and create a new entry in the FTL's mapping table to indicate the mapping of this new block. The FTL can invalidate an old mapping, if such a mapping exists. If a page is unavailable, the FTL can wait for a garbage collection process to reclaim pages. In some implementations, if a page is unavailable, the FTL can return back with an error message.

[0054] In response to an INVALIDATE call, a cache aware FTL can lookup corresponding logical addresses in the FTL's mapping table, remove these mappings from the mapping table, and can mark corresponding physical pages invalid. A garbage collection process can reclaim pages marked invalid during an INVALIDATE call.

[0055] A caching flash device can support several target devices. While a caching flash device can theoretically provide caching for an unlimited amount of physical space, a notion of a ratio of caching space provided by a flash to the physical space for which the flash acts as a cache is useful to understand the device's performance. In some cases, a smaller ratio may lead to a smaller working set in cache and, hence, results in less effective caching.

[0056] FIG. 6 shows an example of a cache aware flash translation layer process. A caching device can run a cache aware flash translation layer process to handle incoming requests such as a SET call. A storage controller can send a SET call to a caching device to cache application data on the caching device. At 605, the process can receive, via a SET call, information including a logical address and application data. At 610, the process determines whether a mapping corresponding to the logical address exists. If the mapping exists, then, the process, at 615, marks the physical page that is identified by the mapping as invalid. At 620, the process deletes the mapping. In some implementations, the process marks an entry that contains the mapping as invalid.

[0057] At 625, the process determines whether a free physical page exists. If a free physical page does not exist, then the process runs a garbage collection process to free one or more physical pages at 630. If a free physical page does exist, then, at 635, the process allocates the physical page at 635. At 640, the process creates a new mapping to indicate a relationship between the logical address and the allocated physical page. At 645, the process transfers the application data received via the SET call to the allocated physical page on a flash memory structure.

[0058] FIGS. 7A and 7B show examples of different cache processes. The cache process 705 depicted by FIG. 7A includes two look-ups, whereas the cache process 710 depicted by FIG. 7B includes one look-up. A cache process that uses less look-ups can provide better performance over a cache process that uses more look-ups. Examples of better performance include reduced latency, reduced complexity, or both.

[0059] The cache process 705 depicted by FIG. 7A includes receiving a command that includes a logical address, e.g., <TGT-ID, LPN>. The process 705 includes performing a first look-up on the logical address to obtain a LPN of page in a cache memory such as a flash memory. The process 705 includes performing a second look up on the LPN to obtain a physical address of a physical page. The process 705 includes performing a read from or write to the physical page.

[0060] The cache process 710 depicted by FIG. 7B includes receiving a command that includes a logical address. The process 710 includes performing a look-up on the logical address to obtain a physical address of page in a cache memory such as a flash memory. The process 710 includes performing a read from or write to the physical page. The process 710 can access cache-aware FTL metadata that includes a mapping table associating logical addresses, e.g., <TGT-ID, LPN>, with physical addresses of physical pages.

[0061] A cache-aware FTL can respond to commands such as an INVALIDATE command. Responding to such a command can include removing one or more associated mappings (e.g., {<TGT-ID, LPN> <=> physical page}) from the FTL's mapping table. A mapping table may have multiple mappings that are associated with the same logical address as a result of previous page invalidations. Removing such mappings can increase flash memory space utilization and can decrease a garbage collector's processing load.

[0062] In some implementations, a cache controller uses flash memory as a write-back cache. The cache controller can reserve some space in the flash memory as a buffer space. This buffer space can be used to match up a higher incoming I/O rate with a lower cache replacement rate. In some implementations, a FTL can reserve space in the flash memory for a garbage collection process to allow the process to catch up with an incoming I/O data stream if such a scenario arises.

[0063] A cache-aware FTL can reserve less space when compared to a FTL that is not cache-aware. This is because as caching logic starts invalidating cache blocks, the corresponding physical pages also start becoming invalid, which hence become candidates for erasing. A garbage collector can directly use these invalid pages to produce pages for allocation. However, in some implementations, a cache-aware FTL is not required to reserve space.

[0064] In some implementations, a storage controller can use two or more caching devices to cache data from storage devices. A storage controller can distribute data caching across multiple caching devices. In some implementations, a storage controller can use a hash function that maps the tuple <TGT-ID, offset, length> to a cache device. The cache device that is identified by an output of the hash function can be used for operations (e.g., GET, SET, INVALIDATE) for that tuple.

[0065] FIG. 8 shows an example of a storage system with multiple flash memory structures. A storage system can include data storage devices 810a, 810b such as a HDD, SSD, or a combination thereof. A controller 815 can write and read data to the data storage devices 810a, 810b. A controller 815 can communicate with one or more caching devices 820. A caching device 820 includes memory to cache data such as one or more flash memory structures, e.g., one or more flash chips 845. The caching device 820 can include interface logic 825, a processor 830, a random access memory (RAM) 835, and a buffer manager 840

[0066] FIG. 9 shows an example of a storage process. At 905, a storage controller can receive a command sequence that includes a logical address (e.g., <TGT-ID, LPN>). At 910, a storage controller can select a caching device based on the logical address. Selecting a caching device can include applying a hash function to the logical address to produce an identifier of a caching device. At 915, the storage controller can send the logical address and associated information to the selected caching device to handle the command sequence. At 920, if there is a cache read miss, then at 925, the storage controller can read data associated with the LPN from the data storage device identified by the TGT-ID. At 930, the storage controller can operate the caching device to write the read data to a flash memory structure. At 935, the storage controller can send a response to the command sequence. For a cache read miss, the response can include data retrieved from the storage device.

[0067] At 920, if there is not a cache read miss, then the storage controller can send a response to the command sequence at 935. For a write command sequence, the response can include a status, e.g., write was successful.

[0068] A few embodiments have been described in detail above, and various modifications are possible. The disclosed subject matter, including the functional operations described in this specification, can be implemented in electronic circuitry, computer hardware, firmware, software, or in combinations of them, such as the structural means

disclosed in this specification and structural equivalents thereof, including potentially a program operable to cause one or more data processing apparatus to perform the operations described (such as a program encoded in a computer-readable medium, which can be a memory device, a storage device, a machine-readable storage substrate, or other physical, machine-readable medium, or a combination of one or more of them).

[0069] The term "data processing apparatus" encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them.

[0070] A program (also known as a computer program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages, and it can be deployed in any form, including as a stand alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A program does not necessarily correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[0071] While this specification contains many specifics, these should not be construed as limitations on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments. Certain features that are described in this

specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[0072] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments.

[0073] Other embodiments fall within the scope of the following claims.

CLAIMS

WHAT IS CLAIMED IS:

1. A system comprising:

a host system;

a data storage device;

a non-volatile memory structure that is configured to cache data with respect to the data storage device; and

a controller configured to (i) operate the data storage device, (ii) receive a logical address from the host system, (iii) allocate a physical page in the non-volatile memory structure, (iv) store a mapping from the logical address to a physical address of the physical page, (v) receive data associated with the logical address, and (vi) write, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address, wherein the mapping indicates a cached status of the data associated with the logical address.

2. The system of claim 1, wherein the controller is configured to receive the data associated with the logical address from one of the host system or the data storage device.

3. The system of claim 1, wherein the controller is configured to (i) invalidate the mapping based on the data associated with the logical address being written to a different physical page and (ii) perform an erasure of the physical page associated with the invalidated mapping to make the physical page available for re-allocation.

4. The system of claim 1, wherein the controller is configured to invalidate an allocated physical page based a logical address indicated by an invalidate command sequence.

5. The system of claim 1, comprising:

two or more data storage devices including the data storage device,

wherein the non-volatile memory structure is configured to cache data with respect to the two or more data storage devices, and wherein the mapping includes a data storage device identifier that identifies at least one of the two or more data storage devices.

6. The system of claim 1, wherein the controller includes a storage controller and a cache controller, wherein the cache controller is communicatively coupled with the non-volatile memory structure.

7. The system of claim 1, wherein the non-volatile memory structure includes a flash memory structure, and wherein the data storage device includes a hard disk drive.

8. The system of claim 1, comprising:

a first caching device that includes the non-volatile memory structure; and

a second caching device,

wherein the controller is configured to select a caching device based on the logical address.

9. An apparatus, comprising:

a non-volatile memory structure;

circuitry to receive a command sequence that includes a logical address, wherein the logical address includes (i) information that identifies a data storage device and (ii) a logical page number;

circuitry to receive data associated with the logical address;

circuitry to allocate a physical page in the non-volatile memory structure;

circuitry to store a mapping from the logical address to a physical address associated with the physical page; and

circuitry to write, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address, wherein the mapping indicates a cached status of the data associated with the logical address.

10. The apparatus of claim 9, comprising:

circuitry to receive a read command sequence that includes a logical address;

circuitry to identify a physical address associated with the logical address of the read command sequence; and

circuitry to read, based on the identified physical address, data from the non-volatile memory structure to provide cached data associated with the logical address of the read command sequence.

11. The apparatus of claim 9, comprising:

circuitry to invalidate an allocated physical page based a logical address indicated by an invalidate command sequence.

12. The apparatus of claim 9, wherein the non-volatile memory structure comprises a flash memory structure.

13. A method, comprising:

receiving, from a host system, information that includes a logical address;

receiving data associated with the logical address from one of the host system or a data storage device;

allocating a physical page in a non-volatile memory structure, wherein the non-volatile memory structure is configured to cache data with respect to at least one data storage device;

storing a mapping from the logical address to a physical address of the physical page; and

writing, based on the physical address, the data associated with the logical address to the non-volatile memory structure to cache the data associated with the logical address, wherein the mapping indicates a cached status of the data associated with the logical address.

14. The method of claim 13, comprising:

invalidating the mapping based on the data associated with the logical address being written to a different physical page; and

performing an erasure of the physical page associated with the invalidated mapping to make the physical page available for re-allocation.

15. The method of claim 13, wherein receiving data associated with the logical address comprises reading data from the data storage device based on the logical address.

16. The method of claim 13, comprising:

invalidating an allocated physical page based a logical address indicated by an invalidate command sequence.

17. The method of claim 13, wherein the non-volatile memory structure is configured to cache data with respect to two or more data storage devices, and wherein the logical address includes a target device identifier that identifies at least one of the two or more data storage devices.

18. The method of claim 13, comprising:

selecting a caching device from two or more caching devices based on the logical address, wherein writing data to the non-volatile memory structure comprises communicating with the selected caching device.

19. The method of claim 13, wherein selecting the caching device comprises applying a hash function to the logical address to produce an identifier of a caching device.

20. The method of claim 13, wherein writing data to the non-volatile memory structure comprises writing to a flash memory structure.

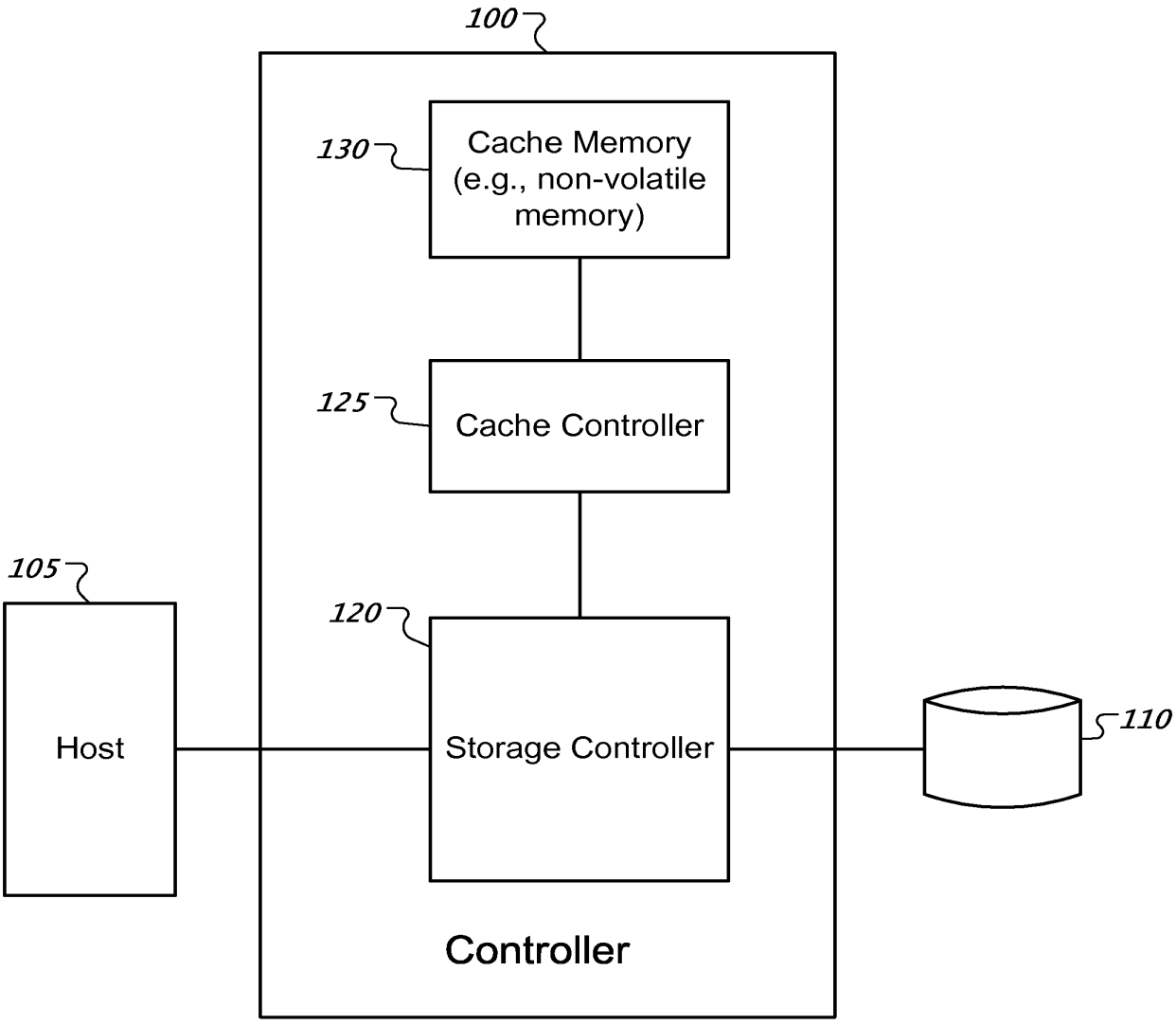

100
130
Cache Memory (e.g., non-volatile memory)
125
Cache Controller
105
120
Host
Storage Controller
110
Controller

FIG. 1

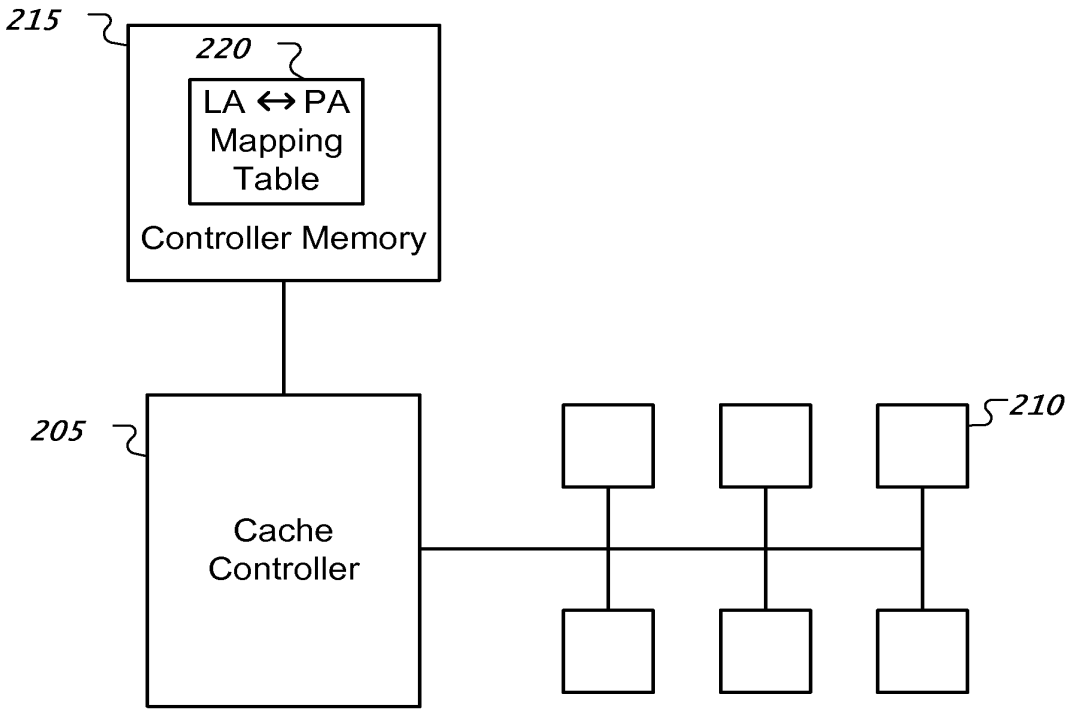

215
220
LA ↔ PA
Mapping
Table
Controller Memory
205
Cache
Controller
210

FIG. 2

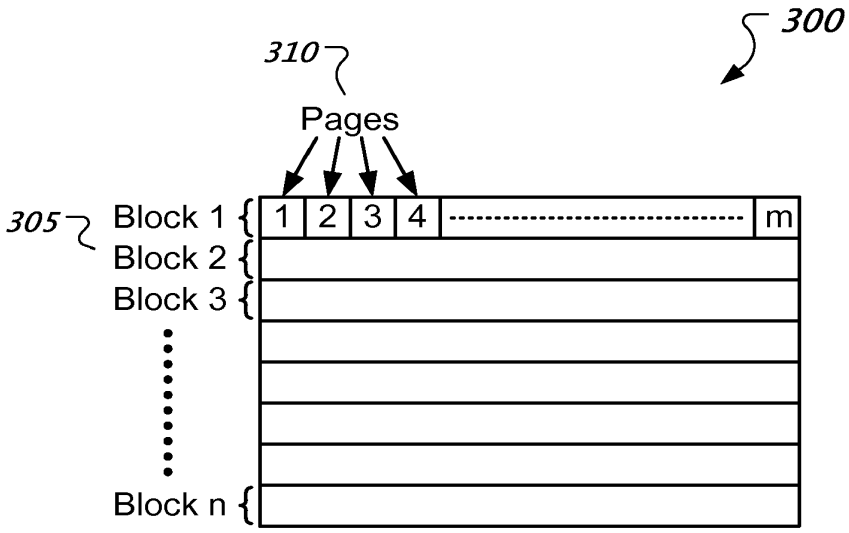


FIG. 3

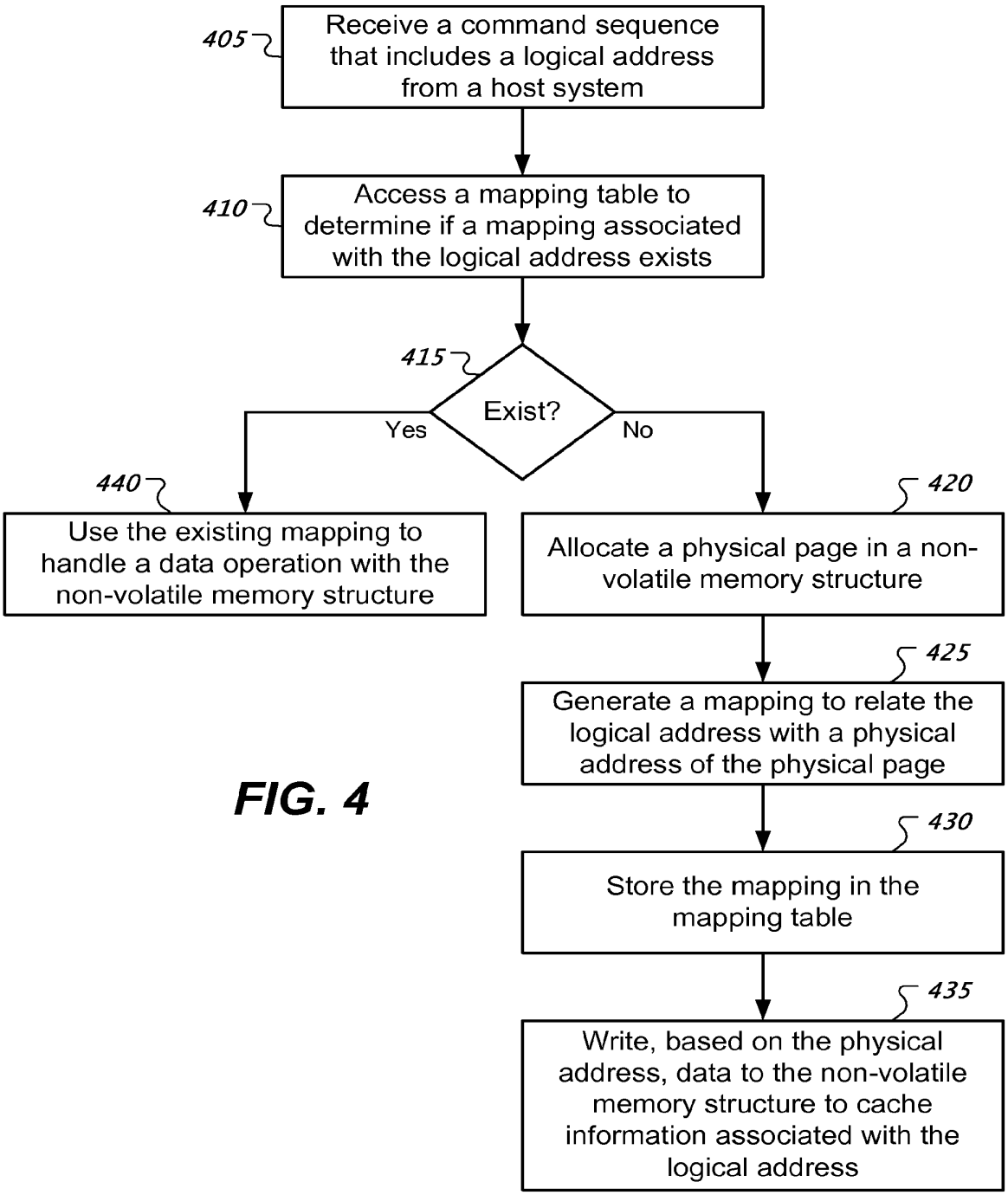

405
Receive a command sequence that includes a logical address from a host system
410
Access a mapping table to determine if a mapping associated with the logical address exists
415
Exist?
Yes
No
440
Use the existing mapping to handle a data operation with the non-volatile memory structure
420
Allocate a physical page in a non-volatile memory structure
425
Generate a mapping to relate the logical address with a physical address of the physical page
430
Store the mapping in the mapping table
435
Write, based on the physical address, data to the non-volatile memory structure to cache information associated with the logical address

FIG. 4

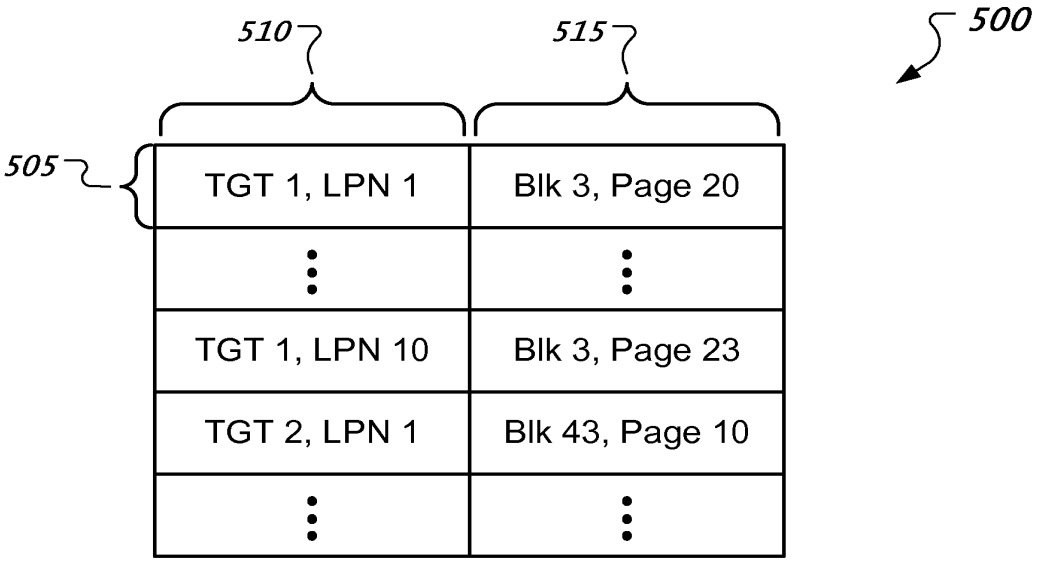
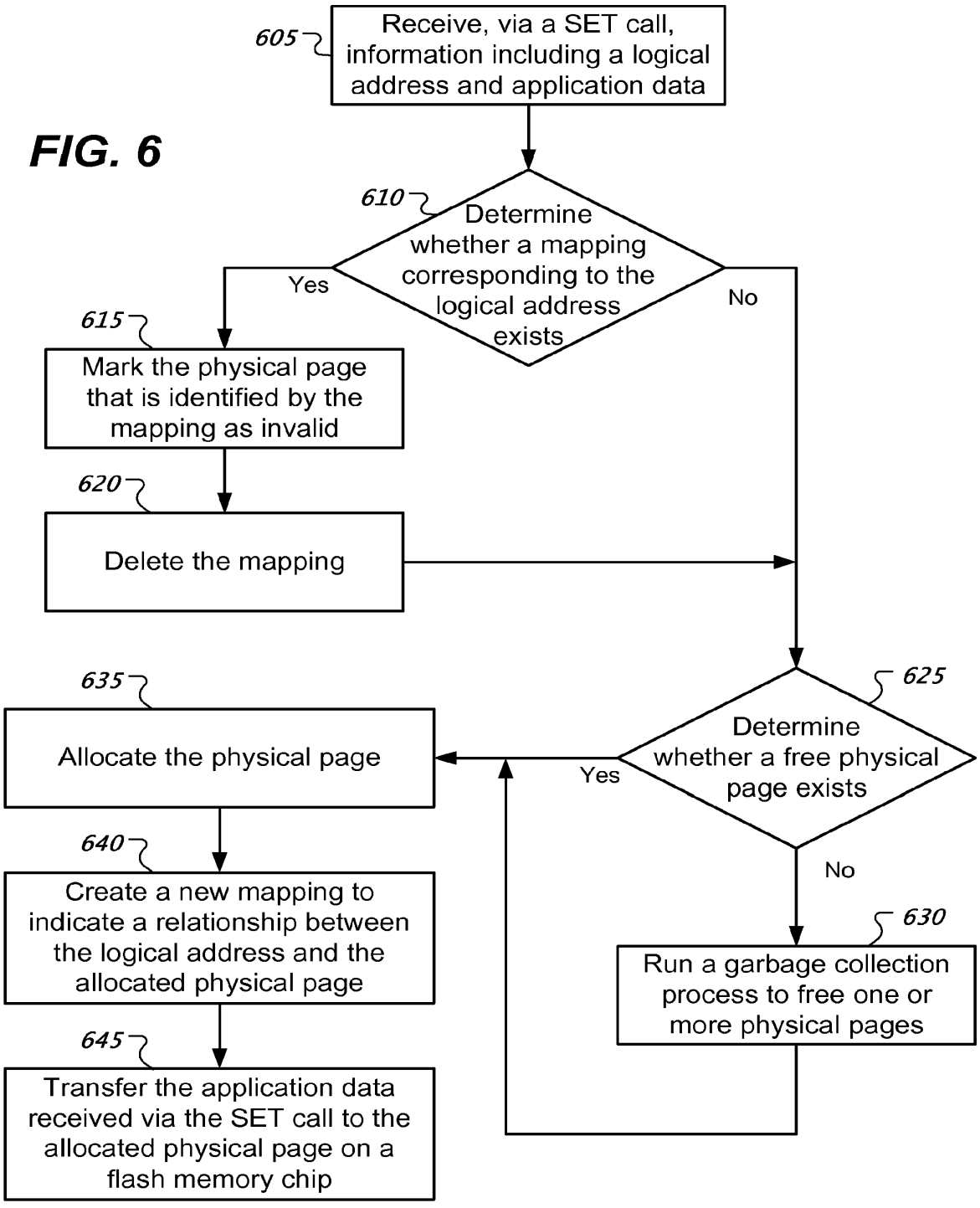

510
515
500
505
TGT 1, LPN 1
Blk 3, Page 20
TGT 1, LPN 10
Blk 3, Page 23
TGT 2, LPN 1
Blk 43, Page 10

FIG. 5

FIG. 6


605
Receive, via a SET call, information including a logical address and application data
610
Determine whether a mapping corresponding to the logical address exists
Yes
No
615
Mark the physical page that is identified by the mapping as invalid
620
Delete the mapping
625
Determine whether a free physical page exists
Yes
No
635
Allocate the physical page
640
Create a new mapping to indicate a relationship between the logical address and the allocated physical page
645
Transfer the application data received via the SET call to the allocated physical page on a flash memory chip
630
Run a garbage collection process to free one or more physical pages

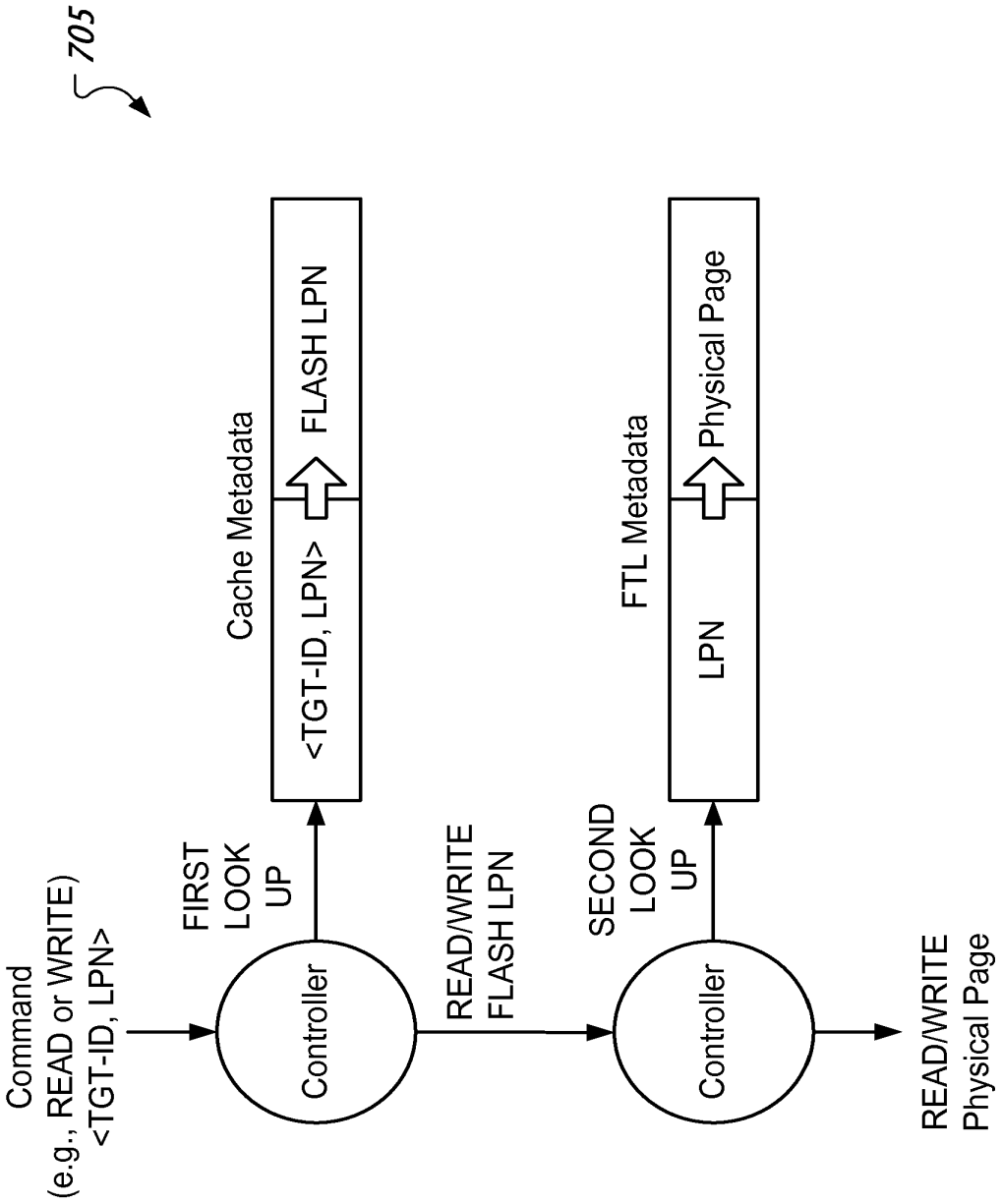

705
Command
(e.g., READ or WRITE)
<TGT-ID, LPN>
Controller
FIRST
LOOK
UP
Cache Metadata
<TGT-ID, LPN>
FLASH LPN
READ/WRITE
FLASH LPN
Controller
SECOND
LOOK
UP
FTL Metadata
LPN
Physical Page
READ/WRITE
Physical Page

FIG. 7A

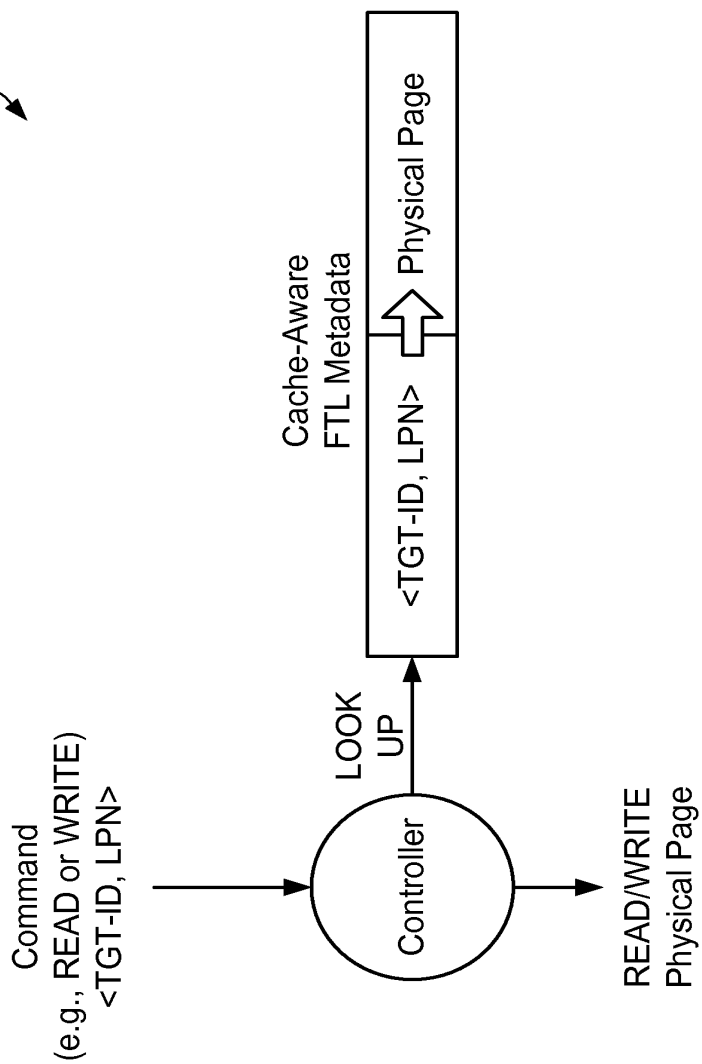

710
Command
(e.g., READ or WRITE)
<TGT-ID, LPN>
Controller
LOOK
UP
Cache-Aware
FTL Metadata
<TGT-ID, LPN>
Physical Page
READ/WRITE
Physical Page

FIG. 7B

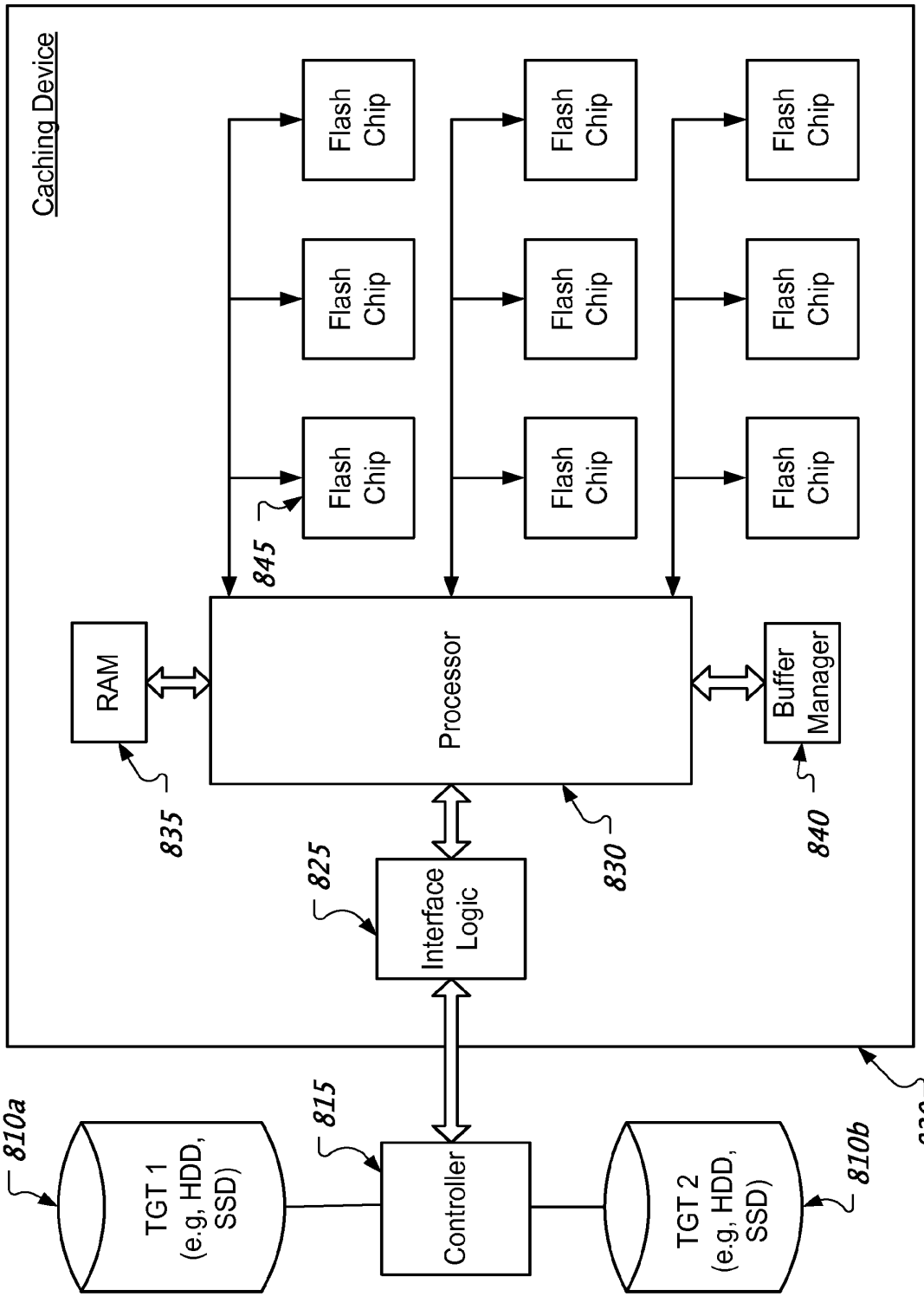

Caching Device
Flash Chip
845
RAM
Processor
Buffer Manager
835
840
825
Interface Logic
830
820
810a
TGT 1 (e.g, HDD, SSD)
815
Controller
TGT 2 (e.g, HDD, SSD)
810b

FIG. 8

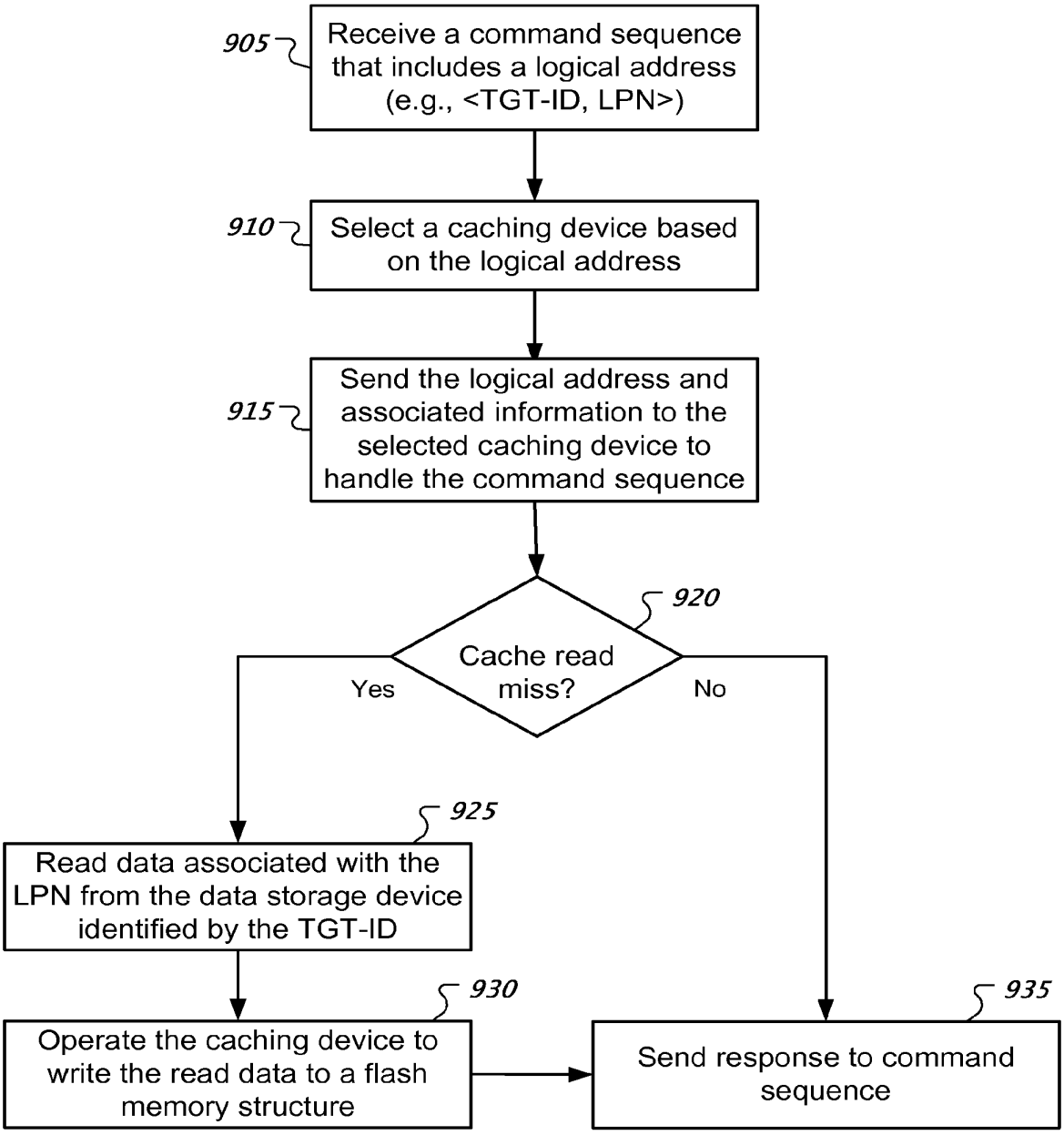

905
Receive a command sequence that includes a logical address (e.g., <TGT-ID, LPN>)
910
Select a caching device based on the logical address
915
Send the logical address and associated information to the selected caching device to handle the command sequence
920
Cache read miss?
Yes
No
925
Read data associated with the LPN from the data storage device identified by the TGT-ID
930
Operate the caching device to write the read data to a flash memory structure
935
Send response to command sequence

FIG. 9

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/051514

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F12/02 G06F12/08
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2007/146845 A2 (BITMICRO NETWORKS INC [US]; BRUCE REY H [US]; MATEO NOEME P [PH]; NITE) 21 December 2007 (2007-12-21) page 3, line 1 - line 17 page 13, line 1 - line 5 page 19, line 13 - line 19 page 21, line 12 - page 23, line 4 figure 2 -----	1-20
X	EP 1 804 168 A1 (SAMSUNG ELECTRONICS CO LTD [KR]) 4 July 2007 (2007-07-04) paragraph [0009] paragraph [0018] - paragraph [0023] paragraph [0031] ----- -/--	1-4, 6-16, 18-20

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
25 January 2011	07/02/2011

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Knutsson, Frédéric

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/051514

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009/193182 A1 (NITTA TATSUO [JP]) 30 July 2009 (2009-07-30) paragraph [0022] - paragraph [0030] paragraph [0038] - paragraph [0040] -----	1-4, 6-16, 18-20

Form PCT/ISA/210 (continuation of second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2010/051514

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2007146845 A2	21-12-2007	NONE	
EP 1804168 A1	04-07-2007	CN 1991790 A	04-07-2007
		JP 2007179545 A	12-07-2007
		KR 20070068796 A	02-07-2007
		US 2007150654 A1	28-06-2007
US 2009193182 A1	30-07-2009	CN 101499036 A	05-08-2009
		JP 2009181314 A	13-08-2009

Form PCT/ISA/210 (patent family annex) (April 2005)

Electronic Patent Application Fee Transmittal

Application Number:	17877881
Filing Date:	29-Jul-2022
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Filer:	Michael Wayne Farn/Kim Dean
Attorney Docket Number:	39471-53126/US

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($)
Miscellaneous:				
RCE- 1ST REQUEST	2801	1	544	544
		Total in USD ($)		**544**

Electronic Acknowledgement Receipt	
EFS ID:	47865236
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn
Filer Authorized By:	
Attorney Docket Number:	39471-53126/US
Receipt Date:	19-APR-2023
Filing Date:	29-JUL-2022
Time Stamp:	13:58:31
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	DA
Payment was successfully received in RAM	$544
RAM confirmation Number	E20234ID59345435
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	Request for Continued Examination (RCE)	53126_US_RCE_2023-04-19.pdf	697685 25cb7c884c9cc59007bd158a94d9b31a872d3ed0	no	3
Warnings:					
Information:					
2	Information Disclosure Statement (IDS) Form (SB08)	53126_US_IDS_2023-04-19.pdf	230321 4a24e0ca2ef01dbec331b76ad9631f0565912cc0	no	3
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	Foreign Reference	WO2011055407A1.pdf	2132199 8b6000fd2b52bb32e8a5840467221dda8a99ea98	no	47
Warnings:					
Information:					
4	Foreign Reference	WO2011044154A1.pdf	1277986 b340b5efbf021492a801215bf3b3fe5602e3ca67	no	40
Warnings:					
Information:					
5	Fee Worksheet (SB06)	fee-info.pdf	38015 8607c300c4816d71b2726ebf7cbc16bb4c06848b	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	4376206		

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 17/877,881	Filing Date 07/29/2022	☐ 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 $240 =	
☐ 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	04/19/2023	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 30	Minus	** 30	= 0		x $40 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $192 =	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 L TURNER/
*** 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.

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 17/877,881	Filing Date 07/29/2022	☐ 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 $240 =	
☐ 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	04/19/2023	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 30	Minus	** 30	= 0		x $40 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $192 =	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 L TURNER/
*** 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

NOTICE OF ALLOWANCE AND FEE(S) DUE

758 7590 02/08/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER
BATAILLE, PIERRE MICHE

ART UNIT	PAPER NUMBER
2136	

DATE MAILED: 02/08/2023

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

TITLE OF INVENTION: Storage System with Multiplane Segments and Cooperative Flash Management

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0.00	$0.00	$480	05/08/2023

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 40% the amount of undiscounted fees, and micro entity fees are 20% the amount of undiscounted 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)

758 7590 02/08/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

TITLE OF INVENTION: Storage System with Multiplane Segments and Cooperative Flash Management

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0.00	$0.00	$480	05/08/2023

EXAMINER	ART UNIT	CLASS-SUBCLASS
BATAILLE, PIERRE MICHE	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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 02/08/2023

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER	
BATAILLE, PIERRE MICHE	
ART UNIT	**PAPER NUMBER**
2136	

DATE MAILED: 02/08/2023

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.

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.** 17/877,881	**Applicant(s)** Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	**Art Unit** 2136	**AIA (FITF) Status** Yes

-- 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 01/06/2023 & 01/24/2023.
   ☐ 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 1-30 . 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 _____.

/PIERRE MICHEL BATAILLE/
Primary Examiner, Art Unit 2136

Notice of Pre-AIA or AIA Status

The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

Allowable Subject Matter

Claims 1-30 are allowed.

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

The present invention features:

solid state storage drive (SSD) comprising: a host interface configured to communicate with a host using commands from a Non-Volatile Memory Express (NVMe) standard; non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages; logic coupled to the host interface and to the flash memory, wherein the logic comprises at least one of hardware logic and software logic, and communicates with the host via the host interface using NVMe commands.

Referenced prior art include:

US 11,307,995 teaching storage device comprising: flash memory having physical erase units; and circuitry to, for each of incoming data write requests: derive, from incoming address information, first, second and third address portion identifying an addressed block; and the circuitry to identify one of the physical erase units that

corresponds to the identified logical erase unit, and program associated data into the identified one of the physical erase units.

US 20130166825 (KIM et al) teaching method of controlling a non-volatile memory device having multiple planes including sequentially transmitting data sets stored at respective buffer addresses to page buffers, respectively, of the planes, and programming in parallel at least two data sets stored in respective page buffers to memory cells of the non-volatile memory device.

US 7,631,138 (Gonzalez et al) teaching memory system having non-volatile memory cells arranged in blocks as a unit of erase, pages therein as a unit of data programming and reading, and planes of a plurality of blocks that are independently accessible,; determining that more than said given number of one or more units of data having consecutive logical addresses are being received, writing the more than said given number of units of data in parallel into pages within two or more blocks of one of the metablocks in two or more planes.

US 9,710,377(Wayda et al) teaching managing multi-drive, multi-die or multi-plane NAND flash memory wherein a host directly assigns physical addresses and performs logical-to-physical address translation in a manner that reduces or eliminates the need for executing multiple of the same functions.

US 2015/0134930 (Huang et al) teaching memory-mapping an information unit (such as a file) into virtual memory such that the information unit is stored on a bulk-erase type block storage device using a single level of address translation, eliminating

the need for the storage device to perform separate and independent address mappings.

US 20120198129 (VanAKEN et al) teaching system including a controller and a plurality of nonvolatile memory modules wherein each module is configured to perform a management function with respect to the module at least partially based on a parameter, the parameter and modules to support multiple forms of concurrency with respect to data programming involving the modules.

US 20120260150 (Cideciyan et al) teaching methods and apparatus provided for controlling data management operations including storage of data in solid state storage of a solid state storage system wherein Input data is stored in successive groups of data write locations in the solid state storage, each group comprising a set of write locations in each of a plurality of logical subdivisions of the solid state storage.

US 2013/0024460 (Peterson et al) teaching method including writing an index segment associated with a plurality of data packets to the log-based structure, the index segment comprising index entries for determining logical identifiers of the data packets and recording, on the storage media, information indicating where the index segment is written on the storage medium.

US 2005/0141313 (Gorobets et al) teaching non-volatile memory constituted from a set of memory planes, each having its own set of read/write circuits so that the memory planes can operate in parallel, the memory further organized into erasable blocks, each for storing a logical group of logical units of data.

However, none of the prior art of record teaches or renders obvious the features as recited in the claims, taken in light of the disclosure, specifically:

> logic operates to: divide the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in at least one zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses; in response to a write command from the host to write data within the range of logical block addresses of a specified zone, perform a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes; generate metadata based on accessing and maintenance of erase units in the specified zone; and based on the metadata, make a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone.

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

Any inquiry concerning this communication or earlier communications from the examiner should be directed to PIERRE MICHEL BATAILLE whose telephone number is (571)272-4178. The examiner can normally be reached Monday - Thursday 7-6 ET.

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.

/PIERRE MICHEL BATAILLE/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 17/877,881	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	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	US-20120198129-A1	08-2012	Van Aken; Stephen P.	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.

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

CPC											
Symbol									Type	Version	
G06F	/	12	/	0246					F	2013-01-01	
G06F	/	16	/	1847					I	2019-01-01	
G06F	/	11	/	1068					I	2013-01-01	
G06F	/	3	/	0611					I	2013-01-01	
G06F	/	12	/	10					I	2013-01-01	
G06F	/	8	/	654					I	2018-02-01	
G06F	/	3	/	0679					I	2013-01-01	
G06F	/	3	/	0638					I	2013-01-01	
G06F	/	3	/	0688					I	2013-01-01	
G06F	/	11	/	1072					I	2013-01-01	
G06F	/	2015	/	766					A	2013-01-01	
G06F	/	2212	/	7208					A	2013-01-01	
G06F	/	2212	/	2022					A	2013-01-01	
G06F	/	2212	/	7207					A	2013-01-01	

CPC Combination Sets											
Symbol									Type	Set	Ranking	Version
	/		/									

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	26 January 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office — Part of Paper No.: 20230119

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

INTERNATIONAL CLASSIFICATION					
CLAIMED					
G06F	/	12	/	02	
G06F	/	12	/	10	

NON-CLAIMED					
	/		/		

US ORIGINAL CLASSIFICATION	
CLASS	SUBCLASS

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

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	26 January 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office — Part of Paper No.: 20230119

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	**Art Unit**
	PIERRE MICHEL BATAILLE	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
2	1	11	10	20	19	29	28								
3	2	12	11	21	20	30	29								
4	3	13	12	22	21	1	30								
5	4	14	13	23	22										
6	5	15	14	24	23										
7	6	16	15	25	24										
8	7	17	16	26	25										
9	8	18	17	27	26										
10	9	19	18	28	27										

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	30	
/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	26 January 2023	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2B

U.S. Patent and Trademark Office Part of Paper No.: 20230119

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

CPC - Searched*		
Symbol	**Date**	**Examiner**
G06F12/1009, G06F12/0292, G06F9/45545, G06F12/1036, G06F2009/45583	09/26/2022	PMB
G06F3/0679, G06F3/0688, G06F12/10, G06F12/0246, G06F2212/2022, G06F2212/7208, G06F2212/7207, G06F3/0611, G06F3/0638	01/19/2023	PMB

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**
PALM INVENTOR NAME & ASSIGNEE SAEARCH	09/16/2022	PMB
KEYWORD & CONTEXT PE2E SEARCH	09/26/2022	PMB
NON-PATENT LITERATURE SEARCH	09/26/2022	PMB
NON-PATENT LITERATURE SEARCH	01/19/2023	PMB
KEYWORD & CONTEXT PE2E SEARCH	01/19/2023	PMB

/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner
	INTERFERENCE SEARCH INCLUDED IN PE2E SEARCH	01/19/2023	PMB

/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	

1. Hydra: A Block-Mapped Parallel Flash Memory Solid-State Disk Architecture

IEEE XploreIEEE Periodicals2010-06-253IEEE Transactions on Computers

2. Ozone (O3): An Out-of-Order Flash Memory Controller Architecture

IEEE XploreIEEE Periodicals2010-10-073IEEE Transactions on Computers

3. Performance comparison of logical-address-to-physical-address algorithms for non-volatile memory

IEEE XploreIEEE Conferences2003-01-013ASIC, 2003. Proceedings. 5th International Conference on

4. Essential roles of exploiting internal parallelism of flash memory based solid state drives in high-speed data processing

IEEE XploreIEEE Conferences2011-02-0132011 IEEE 17th International Symposium on High Performance Computer Architecture (Page(s): 266-277)

5. Optimized structure for fast data storage to non-volatile memory with minimum chip size

IEEE XploreIEEE Conferences2004-01-013Proceedings 7th International Conference on Signal Processing, 2004. Proceedings. ICSP '04. 2004.

6. A space-efficient flash translation layer for CompactFlash systems

IEEE XploreIEEE Periodicals2002-05-013IEEE Transactions on Consumer Electronics

7. Flash memory-based storage device for mobile embedded applications

IEEE XploreIEEE Conferences2007-10-0132007 IEEE International Conference on Systems, Man and Cybernetics (Page(s): 3698-3702)

8. Architecture exploration of flash memory storage controller through a cycle accurate profiling

IEEE XploreIEEE Periodicals2011-11-013IEEE Transactions on Consumer Electronics

9. A Management Strategy for the Reliability and Performance Improvement of MLC-Based Flash-Memory Storage Systems

IEEE XploreIEEE Periodicals2011-01-273IEEE Transactions on Computers

10. Conference Record of the Twenty First IEEE Photovoltaic Specialists Conference - 1990 (Cat. No.90CH2838-1)

IEEE XploreIEEE Conferences1990-01-013IEEE Conference on Photovoltaic Specialists (Page(s): 0_1)

11. Dataset Management-Aware Software Architecture for Storage Systems Based on SSDs

IEEE XploreIEEE Conferences2012-06-0132012 IEEE Seventh International Conference on Networking, Architecture, and Storage (Page(s): 286-292)

12. CAST: A page-level FTL with compact address mapping and parallel data blocks

IEEE XploreIEEE Conferences2012-12-0132012 IEEE 31st International Performance Computing and Communications Conference (IPCCC) (Page(s): 142-151)

13. Detecting Solid-State Disk Geometry for Write Pattern Optimization

IEEE XploreIEEE Conferences2011-08-0132011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications

14. nand Flash Memory and Its Role in Storage Architectures

IEEE XploreIEEE Periodicals2008-11-013Proceedings of the IEEE

15. Accelerating enterprise solid-state disks with non-volatile merge caching

IEEE XploreIEEE Conferences2010-08-013International Conference on Green Computing (Page(s). 203-214)

16. POSIX and the Verification Grand Challenge: A Roadmap

IEEE XploreIEEE Conferences2008-03-01313th IEEE International Conference on Engineering of Complex Computer Systems (iceccs 2008) (Page(s): 153-162)

17. Physically addressed queueing (PAQ): Improving parallelism in solid state disks

IEEE XploreIEEE Conferences2012-06-0132012 39th Annual International Symposium on Computer Architecture (ISCA) (Page(s): 404-415)

18. Parameter-Aware I/O Management for Solid State Disks (SSDs)

IEEE XploreIEEE Periodicals2012-04-203IEEE Transactions on Computers

19. WAFTL: A workload adaptive flash translation layer with data partition

IEEE XploreIEEE Conferences2011-05-0132011 IEEE 27th Symposium on Mass Storage Systems and Technologies (MSST) (Page(s): 1-12)

20. A Hybrid Approach to NAND-Flash-Based Solid-State Disks

IEEE XploreIEEE Periodicals2010-08-203IEEE Transactions on Computers

21. An Efficient FTL Design for Multi-chipped Solid-State Drives

IEEE XploreIEEE Conferences2010-08-0132010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications (Page(s): 237-246)

22. A file-system-aware FTL design for flash-memory storage systems

IEEE XploreIEEE Conferences2009-04-0132009 Design, Automation & Test in Europe Conference & Exhibition (Page(s): 393-398)

23. Virtual storage architecture guide (VSAG)

IEEE XploreIEEE Conferences1995-01-013Proceedings of IEEE 14th Symposium on Mass Storage Systems (Page(s): 312-326)

24. Special Issues in Flash

IEEE XploreIEEE Conferences2008-11-0132008 IEEE/ACM International Conference on Computer-Aided Design (Page(s): 821-826)

25. High level modelling and performance evaluation of address mapping in NAND flash memory

IEEE XploreIEEE Conferences2009-12-0132009 16th IEEE International Conference on Electronics, Circuits and Systems - (ICECS 2009) (Page(s). 659-662)

26. Cross-Layer Optimizations in Solid-State Drives

IEEE XploreIEEE Periodicals2011-12-013IEEE Embedded Systems Letters

27. Characterizing flash memory: Anomalies, observations, and applications

IEEE XploreIEEE Conferences2009-12-0132009 42nd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO) (Page(s): 24-33)

28. Memory management scheme for cost-effective disk-on-modules in consumer electronics devices

IEEE XploreIEEE Periodicals2008-11-013IEEE Transactions on Consumer Electronics

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	15074	(flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 08:32 PM
L2	20	L1 AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(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	2023/01/19 08:33 PM
L3	4781	(flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:33 PM
L4	72	L3 AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:33 PM
L5	383	(flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND (configuration OR	(USOCR; FIT (AP, AT, AU, 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	2023/01/19 08:36 PM

		maintenance) near5 (flash OR plane OR storage OR memory) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L6	2	L5 AND erase$ near3 unit AND physical near3 plane AND logical	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 08:37 PM
L7	544	(flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND (configuration OR maintenance) near5 (flash OR plane OR storage OR memory) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:37 PM
L8	45	L7 AND erase$ near3 unit AND physical near3 plane AND logical	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:37 PM
L9	2	logical near3 address AND erase$ near3 unit AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 write AND multi?plane near4 write	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT;	OR	ON	ON	2023/01/19 09:49 PM

			IBM_TDB)				
L10	10	logical near3 address AND erase$ near3 unit AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 write AND multi?plane near4 write	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 09:49 PM
L11	0	(different adj1 plane OR multi?plane) WITH eras$ near3 unit AND metadata WITH eras$ AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 writ$3 AND multi?plane near4 writ$3	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 10:03 PM
L12	10	(different adj1 plane OR multi?plane) WITH eras$ near3 unit AND metadata WITH eras$ AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 writ$3 AND multi?plane near4 writ$3	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 10:03 PM
L13	1	(maintenance OR reconfigur$) WITH eras$3 near3 unit WITH (plane OR zone) AND (multi?plane OR different adj1 (plane OR flash)) near5 eras$3	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 10:06 PM
L14	36	(maintenance OR reconfigur$) WITH eras$3 near3 unit WITH	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 10:06 PM

		(plane OR zone) AND (multi?plane OR different adj1 (plane OR flash)) near5 eras$3					
L15	36	(maintenance OR reconfigur$) WITH eras$3 WITH (plane OR zone) AND (multi?plane OR different adj1 (plane OR flash)) near5 eras$3	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 10:10 PM
L16	3	(maintenance OR reconfigur$) WITH eras$3 WITH (plane OR zone) AND (multi?plane OR different adj1 (plane OR flash)) near5 eras$3	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 10:11 PM
L17	25	("7913061" "20120198128" "8626989" "8443263" "20110161784" "20070245070" "20110231713" "7870328" "20140201437" "20120151118" "20120198129" "8527693" "9182924" "8504798" "20050141313" "20090019217").pn.	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 10:14 PM
L19	16	("7913061" "20120198128" "8626989" "8443263" "20110161784" "20070245070" "20110231713" "7870328" "20140201437" "20120151118" "20120198129" "8527693" "9182924" "8504798" "20050141313" "20090019217").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 10:14 PM
L20	161593	(G06F3/0611 G06F3/0638	(US-PGPUB; USPAT; USOCR; FIT (AP, AT,	OR	ON	ON	2023/01/19 10:59 PM

		G06F3/0679 G06F3/0688 G06F12/10 G06F12/0246 G06F2212/2022 G06F2212/7208 G06F2212/7207).CPC.	AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L21	61	L20 AND (flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND (configuration OR maintenance) near5 (flash OR plane OR storage OR memory) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 11:00 PM
L22	154	L20 AND (flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND (configuration OR maintenance) near5 (flash OR plane OR storage OR memory) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:00 PM
L23	45	L22 AND erase$ near3 unit AND physical near3 plane AND logical	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:01 PM
L24	10	(different adj1 plane OR multi?plane) WITH eras$ AND metadata WITH eras$ AND physical near3 plane AND logical near3 controller AND physical near3 controller AND	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:01 PM

		(flash OR plane) near4 writ$3 AND multi?plane near4 writ$3					
L25	0	(different adj1 plane OR multi?plane) WITH eras$ AND metadata WITH eras$ AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 writ$3 AND multi?plane near4 writ$3	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2023/01/19 11:02 PM
L26	10	L20 AND (different adj1 plane OR multi?plane) WITH eras$ AND metadata WITH eras$ AND physical near3 plane AND (flash OR plane) near4 writ$3 AND multi?plane near4 writ$3	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:03 PM
L27	31	L20 AND (different adj1 plane OR multi?plane) WITH eras$ AND metadata WITH eras$ AND physical near3 plane	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:03 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	0	(((flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:34 PM
N2	1	((flash OR plane OR storage OR memory) near5 (zone OR divid$ OR subdivid$ OR	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 08:36 PM

		sub?divid$ OR subset OR sub?set) AND (configuration OR maintenance) near5 (flash OR plane OR storage OR memory) AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth).clm.					
N3	0	(logical near3 address AND erase$ near3 unit AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 write AND multi?plane near4 write).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 09:49 PM
N4	0	((different adj1 plane OR multi?plane) WITH eras$ AND metadata WITH eras$ AND physical near3 plane AND logical near3 controller AND physical near3 controller AND (flash OR plane) near4 writ$3 AND multi?plane near4 writ$3).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/19 11:02 PM

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	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US

CONFIRMATION NO. 3676

758
FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

POA ACCEPTANCE LETTER

OC000000137256878

Date Mailed: 01/26/2023

NOTICE OF ACCEPTANCE OF POWER OF ATTORNEY

This is in response to the Power of Attorney filed 01/20/2023.

The Power of Attorney in this application is accepted. Correspondence in this application will be mailed to the above address as provided by 37 CFR 1.33.

Questions about the contents of this notice and the requirements it sets forth should be directed to the Office of Data Management, Application Assistance Unit, at **(571) 272-4000** or **(571) 272-4200** or **1-888-786-0101**.

/ctuazon/

Doc Code: DIST.E.FILE **Document Description: Electronic Terminal Disclaimer - Filed**	PTO/SB/26 U.S. Patent and Trademark Office Department of Commerce

Electronic Petition Request	**TERMINAL DISCLAIMER TO OBVIATE A DOUBLE PATENTING REJECTION OVER A "PRIOR" PATENT**
Application Number	17877881
Filing Date	29-Jul-2022
First Named Inventor	Andrey Kuzmin
Attorney Docket Number	39471-53126/US
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management

☒ Filing of terminal disclaimer does not obviate requirement for response under 37 CFR 1.111 to outstanding Office Action

☒ This electronic Terminal Disclaimer is not being used for a Joint Research Agreement.

Owner	Percent Interest
Radian Memory Systems, LLC	100%

The owner(s) with percent interest listed above in the instant application hereby disclaims, except as provided below, the terminal part of the statutory term of any patent granted on the instant application which would extend beyond the expiration date of the full statutory term of prior patent number(s)

11487657

as the term of said prior patent is presently shortened by any terminal disclaimer. The owner hereby agrees that any patent so granted on the instant application shall be enforceable only for and during such period that it and the prior patent are commonly owned. This agreement runs with any patent granted on the instant application and is binding upon the grantee, its successors or assigns.

In making the above disclaimer, the owner does not disclaim the terminal part of the term of any patent granted on the instant application that would extend to the expiration date of the full statutory term of the prior patent, "as the term of said prior patent is presently shortened by any terminal disclaimer," in the event that said prior patent later:
- expires for failure to pay a maintenance fee;
- is held unenforceable;
- is found invalid by a court of competent jurisdiction;
- is statutorily disclaimed in whole or terminally disclaimed under 37 CFR 1.321;
- has all claims canceled by a reexamination certificate;
- is reissued; or
- is in any manner terminated prior to the expiration of its full statutory term as presently shortened by any terminal disclaimer.

◉ Terminal disclaimer fee under 37 CFR 1.20(d) is included with Electronic Terminal Disclaimer request.

○ I certify, in accordance with 37 CFR 1.4(d)(4), that the terminal disclaimer fee under 37 CFR 1.20(d) required for this terminal disclaimer has already been paid in the above-identified application.	
Applicant claims the following fee status: ◉ Small Entity ○ Micro Entity ○ Regular Undiscounted I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.	
THIS PORTION MUST BE COMPLETED BY THE SIGNATORY OR SIGNATORIES I certify, in accordance with 37 CFR 1.4(d)(4) that I am: ◉ An attorney or agent registered to practice before the Patent and Trademark Office who is of record in this application Registration Number 41015 ○ A sole inventor ○ A joint inventor; I certify that I am authorized to sign this submission on behalf of all of the inventors as evidenced by the power of attorney in the application ○ A joint inventor; all of whom are signing this request	
Signature	/Michael W. Farn/
Name	Michael W. Farn

*Statement under 37 CFR 3.73(b) is required if terminal disclaimer is signed by the assignee (owner).
Form PTO/SB/96 may be used for making this certification. See MPEP § 324.

Electronic Patent Application Fee Transmittal

Application Number:	17877881
Filing Date:	29-Jul-2022
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Filer:	Michael Wayne Farn
Attorney Docket Number:	39471-53126/US

Filed as Small Entity

Filing Fees for Utility under 35 USC 111(a)

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
STATUTORY OR TERMINAL DISCLAIMER	2814	1	170	170
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**170**

Doc Code: DISQ.E.FILE
Document Description: Electronic Terminal Disclaimer – Approved

Application No.: 17877881

Filing Date: 29-Jul-2022

Applicant/Patent under Reexamination: Kuzmin

Electronic Terminal Disclaimer filed on January 24, 2023

☒ APPROVED

This patent is subject to a terminal disclaimer

☐ DISAPPROVED

Approved/Disapproved by: Electronic Terminal Disclaimer automatically approved by EFS-Web

U.S. Patent and Trademark Office

Electronic Acknowledgement Receipt	
EFS ID:	47421445
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn
Filer Authorized By:	
Attorney Docket Number:	39471-53126/US
Receipt Date:	24-JAN-2023
Filing Date:	29-JUL-2022
Time Stamp:	19:00:39
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	DA
Payment was successfully received in RAM	$170
RAM confirmation Number	E20231NJ00374663
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	Terminal Disclaimer-Filed (Electronic)	eTerminal-Disclaimer.pdf	41302 aa5bd22bac0472ae8a661319293dc9a3f1f0e7f9	no	2
Warnings:					
Information:					
2	Fee Worksheet (SB06)	fee-info.pdf	38235 aab1c343e4e025faa71ce00b98ed85957098be65	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	79537		

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: PA..
Document Description: Power of Attorney

PTO/AIA/82A (07-13)
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.

TRANSMITTAL FOR POWER OF ATTORNEY TO ONE OR MORE REGISTERED PRACTITIONERS

NOTE: This form is to be submitted with the Power of Attorney by Applicant form (PTO/AIA/82B) to identify the application to which the Power of Attorney is directed, in accordance with 37 CFR 1.5, unless the application number and filing date are identified in the Power of Attorney by Applicant form. If neither form PTO/AIA/82A nor form PTO/AIA82B identifies the application to which the Power of Attorney is directed, the Power of Attorney will not be recognized in the application.

Application Number	17/877,881
Filing Date	July 29, 2022
First Named Inventor	Kuzmin, Andrey V.
Title	Storage System with Multiplane Segments and Cooperative Flash Management
Art Unit	2136
Examiner Name	Pierre Michel Batallie
Attorney Docket Number	39471-53126/US

SIGNATURE of Applicant or Patent Practitioner

Signature	/Michael W. Farn/	Date (Optional)	
Name	Michael W. Farn	Registration Number	41015
Title (if Applicant is a juristic entity)	Attorney of Applicant		
Applicant Name (if Applicant is a juristic entity)	Radian Memory Systems, Inc.		

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ *Total of one forms are submitted.

This collection of information is required by 37 CFR 1.131, 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.

DocuSign Envelope ID: 7362DC0B-D20E-4FE4-A09E-531BC5AC0144

Document Description: Power of Attorney

PTO/AIA/82B (07-13)
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

POWER OF ATTORNEY BY APPLICANT

I hereby revoke all previous powers of attorney given in the application identified in either the attached transmittal letter or the boxes below.

Application Number	Filing Date

(Note: The boxes above may be left blank if information is provided on form PTO/AIA/82A.)

☑ I hereby appoint the Patent Practitioner(s) associated with the following Customer Number as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above: 00758

OR

☐ I hereby appoint Practitioner(s) named in the attached list (form PTO/AIA/82C) as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the patent application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above. (Note: Complete form PTO/AIA/82C.)

Please recognize or change the correspondence address for the application identified in the attached transmittal letter or the boxes above to:

☑ The address associated with the above-mentioned Customer Number

OR

☐ The address associated with Customer Number:

OR

☐ Firm or Individual Name					
Address					
City		State		Zip	
Country					
Telephone		Email			

I am the Applicant (if the Applicant is a juristic entity, list the Applicant name in the box):

Radian Memory Systems, Inc.

☐ Inventor or Joint Inventor (title not required below)

☐ Legal Representative of a Deceased or Legally Incapacitated Inventor (title not required below)

☑ Assignee or Person to Whom the Inventor is Under an Obligation to Assign (provide signer's title if applicant is a juristic entity)

☐ Person Who Otherwise Shows Sufficient Proprietary Interest (e.g., a petition under 37 CFR 1.46(b)(2) was granted in the application or is concurrently being filed with this document) (provide signer's title if applicant is a juristic entity)

SIGNATURE of Applicant for Patent

The undersigned (whose [DocuSigned by: 37B5BBC872A047F...] is authorized to act on behalf of the applicant (e.g., where the applicant is a juristic entity).

Signature		Date (Optional)	
Name	Michael Jadon		
Title	CEO		

NOTE: Signature - This form must be signed by the applicant in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ Total of one forms are submitted.

This collection of information is required by 37 CFR 1.131, 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.

Electronic Acknowledgement Receipt

EFS ID:	47406802
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn/Nicole Walsh
Filer Authorized By:	Michael Wayne Farn
Attorney Docket Number:	39471-53126/US
Receipt Date:	20-JAN-2023
Filing Date:	29-JUL-2022
Time Stamp:	21:50:23
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	53126_US_POA.pdf	247642 59b23400dc9c8f7e845d313cfd4d2b09bbb9b09b	no	2

Warnings:

Information:	
Total Files Size (in bytes):	247642

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.

Electronic Acknowledgement Receipt

EFS ID:	47332628
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn
Filer Authorized By:	
Attorney Docket Number:	39471-53126/US
Receipt Date:	06-JAN-2023
Filing Date:	29-JUL-2022
Time Stamp:	18:25:22
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/Request for Reconsideration-After Non-Final Rejection	53126_US_AMENDMENT.pdf	103285 6da49a90530f3ac7672ea28fe47039a50aeadc80	no	12

Warnings:

Information:	
Total Files Size (in bytes):	103285

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

FIRST NAMED INVENTOR:	Andrey V. Kuzmin
APPLICATION NO.:	17/877,881
TITLE:	Storage System with Multiplane Segments and Cooperative Flash Management
FILING DATE:	July 29, 2022
EXAMINER:	Bataille, Pierre Miche
GROUP ART UNIT:	2136
CONFIRMATION NO.:	3676
ATTY. DKT. NO.:	39471-53126/US

CERTIFICATE OF TRANSMISSION			
I hereby certify under 37 C.F.R. § 1.8(a)(ii) that this correspondence is being deposited or transmitted in accordance with 37 C.F.R. § 1.8(a)(i) on or before the local date shown below.			
Signature:	/Michael W. Farn/		
Typed or Printed Name:	Michael W. Farn, Reg. No.: 41,015	Dated:	Jan 6, 2023

RESPONSE AND AMENDMENT A

The following is in response to the Office Action mailed October 7, 2022.

AMENDMENTS TO THE CLAIMS

All pending claims, along with a current status of each, are reproduced below. If indicated, please amend the claims accordingly.

1. (Currently amended) A solid state storage drive (SSD) comprising:

a host interface configured to communicate with a host using commands from a Non-Volatile Memory Express (NVMe) standard;

non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages;

logic coupled to the host interface and to the flash memory, wherein the logic comprises at least one of hardware logic and software logic, and communicates with the host via the host interface using NVMe commands;

wherein the logic ~~and the flash memory~~ operates ~~are operable~~ to:

divide the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in at least one zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses;

in response to a write command from the host to write data within the range of logical block addresses of a specified zone, perform a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes;

generate metadata based on accessing and maintenance of erase units in the specified zone; and

based on the metadata, make a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone.

2. (Currently amended) The SSD of claim 1 further comprising:

additional non-volatile memory, wherein the logic further operates ~~is further operable~~ to store the generated metadata in the additional non-volatile memory.

3. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to compare the metadata with at least one threshold, and wherein making the notification available to the host is dependent on the comparison.

4. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to determine, based on the metadata, that maintenance is recommended for the specified zone and to identify the zone start logical block address associated with the specified zone for which maintenance is recommended; wherein determining that maintenance is recommended for the specified zone, identifying the zone start logical block address associated with the specified zone, and making the notification available to the host are all unsolicited by the host.

5. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to receive a command from the host; and, in response to said command, transmit

to the host (a) information representing a list of host accessible zones including a zone start logical block address for each listed zone, and (b) a host accessible capacity for each listed zone.

6. (Currently amended) The SSD of claim 1 wherein for at least one write command addressed within the range of logical block addresses of the specified zone, the specified zone contains an integer number of erase units in the NAND flash memory, and the logic operates ~~is operable~~ to generate an address offset from the write command which corresponds to a location within a sequential range of locations that corresponds to the integer number of erase units.

7. (Currently amended) The SSD of claim 1 wherein:

the logic further operates ~~is further operable~~ to receive read commands and write commands from the host, and to execute the read commands and the write commands by transferring associated data between the host and the NAND flash memory; and

each read command directed to the SSD is accompanied by a logical read address, each logical read address has a first address portion which identifies one of the zones according to a value of the first portion, and a second address portion which identifies a physical memory location in the SSD in a manner that depends on the first address portion.

8. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to:

store information identifying a number of zones comprising the NAND flash memory and to transmit, to the host, information identifying the number of zones and sizes of the zones; and

receive, from the host, data access commands which are dependent, for the specified zone, on the information transmitted to the host identifying the number and the sizes of the zones.

9. (Currently amended) The SSD of claim 8 wherein the logic further operates ~~is further operable~~ to:

in response to a maintenance command from the host directed to the specified zone, issue erase commands to the erase units in the specified zone, wherein the issued erase commands are based on the size of the specified zone in accordance with the information transmitted to the host.

10. (Original) The SSD of claim 1 wherein the flash memory comprises multiple dies of flash memory and the erase units on different planes are also in different dies.

11. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to, upon the performance of the multi-plane write, update the metadata to indicate an amount of unerased host accessible space in the specified zone.

12. (Currently amended) The SSD of claim 11 wherein the logic further operates ~~is further operable~~ to: upon the performance of the multi-plane write, detect that the specified zone contains no erased host accessible space; and, in response to said detection, transition the specified zone to a full state.

13. (Currently amended) The SSD of claim 12 wherein the logic further operates ~~is further operable~~ to receive a command from the host; and, in response to said command, transmit

to the host information representing the full state of the specified zone and an amount of erased host accessible space in the specified zone that is available for writing by the host.

14. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to receive a maintenance command directed to the specified zone and, in response to the maintenance command, erase host accessible space only within the specified zone and transition the specified zone to an empty state.

15. (Currently amended) The SSD of claim 1 wherein the metadata is dependent on an erase count for the specified zone; and the logic further operates ~~is further operable~~ to compare the metadata with at least one threshold for erase count, wherein making the notification available to the host is dependent on the comparison.

16. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to:

in response to a maintenance command from the host directed to the zone start logical block address of the specified zone, perform an erase of erase units in the specified zone, and transition a state of the specified zone to an empty state whereby all host accessible pages of the specified zone are available for writing by the host.

17. (Original) The SSD of claim 16 wherein the maintenance command from the host is in response to the notification that recommends performing maintenance on the specified zone.

18. (Currently amended) The SSD of claim 16 wherein the metadata is dependent on erase counts for erase units in the specified zone; and the logic further operates ~~is further operable~~ to update the metadata responsive to performance of the erase.

19. (Original) The SSD of claim 16 wherein performing the erase of erase units in the specified zone comprises performing the erase on erase units of different planes in the specified zone.

20. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to, in response to a read command from the host to read data from sequential addressing of the specified zone, perform a multi-plane read of said data from erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes.

21. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to, in response to a write command from the host to the zone start logical block address of the specified zone: assign a logical block address within the range of logical block addresses of the specified zone, transfer data from the host to the specified zone at the assigned logical block address, and return the assigned logical block address to the host.

22. (Currently amended) The SSD of claim 1 wherein the metadata is dependent on wear of the erase units and the logic further operates ~~is further operable~~ to, in response to a command from the host, transmit to the host information based on such metadata.

23. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to, in response to a command from the host, transmit to the host information dependent on wear for the specified zone.

24. (Currently amended) The SSD of claim 1 wherein the metadata is dependent on a time period since data was written to the specified zone; and the logic further operates ~~is further operable~~ to compare the metadata with at least one threshold, wherein making the notification available to the host is dependent on the comparison.

25. (Original) The SSD of claim 24 wherein comparing the metadata with at least one threshold and making the notification available to the host are unsolicited by the host.

26. (Currently amended) The SSD of claim 1 wherein logic further operates ~~is further operable~~ to, in response to a command from the host, transmit to the host information dependent on a time period since data was written to the specified zone.

27. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to detect a failure condition of one of the erase units of the specified zone and transmit to the host the zone start logical block address and a state of the specified zone.

28. (Currently amended) The SSD of claim 1 wherein the logic further operates ~~is further operable~~ to detect a failure condition of one of the erase units of the specified zone and, in response to a write command from the host within the range of logical block addresses of the specified zone, transmit to the host a notification of a write failure.

29. (Original) The SSD of claim 1 wherein the metadata is dependent on read errors detected for the specified zone.

30. (Currently amended) A method for providing a host with access to an SSD that contains logic and non-volatile memory comprising NAND flash memory, the logic comprising circuity and/or a processor executing instructions, and the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages, the method comprising, at the SSD:

dividing the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in each zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses;

in response to a write command from the host to write data to the range of logical block addresses of a specified zone, performing a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes;

generating metadata based on accessing and maintenance of erase units in the specified zone; and

based on the metadata, making a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone;

wherein the SSD and the host communicate over a host interface, using commands from a Non-Volatile Memory Express (NVMe) standard.

REMARKS

Claims 1–30 were pending in this application, and claims 1 and 30 were the independent claims.

STATEMENT OF SUBSTANCE OF INTERVIEW

Applicant's representative Mike Farn (Reg. No. 41,015) conducted a telephonic interview with Examiner Bataille on Dec 19, 2022. During the interview, pending claims 1 and 30, and the rejections to those claims under 35 U.S.C. § 112 and § 101 were discussed. An agreement was reached that the claim amendments herein would overcome these rejections.

OBJECTIONS TO THE CLAIMS

Claims 1-30 were objected to because of various informalities, which are addressed as follows:

In claim 1, "flash memory" is deleted from the phrase "wherein the logic and the flash memory are operable to."

In all applicable claims, "is operable to" is replaced by "operates to."

Claim 30 is amended so that the logic is always tangible.

REJECTIONS UNDER 35 U.S.C. § 112(b) or pre-AIA 35 U.S.C. § 112 second paragraph

Claims 1–30 were rejected under 35 U.S.C. § 112(b) or 35 U.S.C. § 112 (pre-AIA), second paragraph, for not specifically pointing out and distinctly claiming the subject matter that the Applicant regards as the invention. As agreed with the Examiner, these rejections are overcome by replacing "is operable to" with "operates to."

REJECTIONS UNDER 35 U.S.C. § 101

Claim 30 was rejected under 35 U.S.C. § 101 as directed to non-statutory subject matter. In view of the following, these rejections should be withdrawn. As agreed with the Examiner, this rejection is overcome by specifying that the logic is tangible.

DOUBLE PATENTING REJECTION

Claims 1-30 are provisionally rejected on the ground of nonstatutory double patenting as being unpatentable overclaims 1-23 and 25-28 of copending Application No. 17/563,481. If the claims are deemed otherwise allowable, the Examiner is requested to contact the attorney of record, and Applicant will then file an e-terminal disclaimer.

CONCLUSION

Based on the foregoing, the application is in condition for allowance of all claims, and a Notice of Allowance is respectfully requested. If the examiner believes for any reason direct contact would help advance the prosecution of this case to allowance, the examiner is encouraged to telephone the undersigned at the number given below.

If extensions of time are necessary to prevent abandonment of this application, then such extensions of time are hereby petitioned under 37 CFR 1.136(a), and any fees required therefore are hereby authorized to be charged to our Deposit Account 19-2555.

Respectfully submitted,

Dated: Jan 6, 2023

/ Michael W. Farn /

Michael W. Farn, Reg. No. 41,015
FENWICK & WEST LLP
801 California Street
Mountain View, CA 94041
Phone: (650) 335-7823
Fax: (650) 938-5200

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 17/877,881	Filing Date 07/29/2022	☐ 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 $240 =	
☐ 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	01/06/2023	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 30	Minus	** 30	= 0		x $40 =	0
	Independent (37 CFR 1.16(h))	* 2	Minus	*** 3	= 0		x $192 =	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".	/ROSS W BROWN/
*** 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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 12/27/2022

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER	
BATAILLE, PIERRE MICHE	

ART UNIT	PAPER NUMBER
2136	

NOTIFICATION DATE	DELIVERY MODE
12/27/2022	ELECTRONIC

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.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PTOC@Fenwick.com

Applicant-Initiated Interview Summary	**Application No.** 17/877,881	**Applicant(s)** Kuzmin et al.		
	Examiner PIERRE MICHEL BATAILLE	**Art Unit** 2136	**AIA (First Inventor to File) Status** Yes	**Page** 1 of 2

All Participants (applicant, applicants representative, PTO personnel)	**Title**	**Type**
PIERRE MICHEL BATAILLE	Primary Examiner	Telephonic
MICHAEL W FARN		

Date of Interview: 19 December 2022

Issues Discussed:

35 U.S.C. 101

Applicant and examiner discussed the rejection under 101 and came with agreement to amend claim 30 by incorporating logic to execute programming instructions, in order to overcome the rejection.

35 U.S.C. 112

Applicant and examiner discussed the rejection under 112 and came with agreement to amend the claims replacing the phrase "operable to" with actual functional language. Applicant is to provide an amendment for the examiner to look at before filing official response.

Non-statutory Double Patenting

Applicant and examiner discussed the double patenting rejection and proposed filing a terminal disclaimer to overcome the rejection.

/PIERRE MICHEL BATAILLE/ 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.

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 10/07/2022

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER	
BATAILLE, PIERRE MICHE	

ART UNIT	PAPER NUMBER
2136	

NOTIFICATION DATE	DELIVERY MODE
10/07/2022	ELECTRONIC

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.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PTOC@Fenwick.com

Office Action Summary	Application No. 17/877,881	Applicant(s) Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	Art Unit 2136	AIA (FITF) Status Yes

-- 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 07/29/2022.

☐ 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) 1-30 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) 1-30 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 07/29/2022 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 08/02/2022.

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

4) ☐ Other: _____.

Notice of Pre-AIA or AIA Status

1. Page 4 of the Application Data Sheet (ADS) states: "This application (1) claims priority to or the benefit of an application filed before March 16, 2013 and (2) also contains, or contained at any time, a claim to a claimed invention that has an effective filing date on or after March 16, 2013. Therefore, based on the statement under 37 CFR 1.55 or 1.78, the present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

- Claims 1-30 are pending in the application under prosecution and have been examined.
- The specification has not been checked to the extent necessary to determine the presence of all possible minor errors.
- The specification should be amended to reflect the status of all related application, whether patented or abandoned. Therefore, applications noted by their serial number and/or attorney docket number should be updated with correct serial number and patent number if patented.
- The first instance of all acronyms or abbreviation should be spelled out for clarity, whether or not considered well known in the art.

In the response to this Office action, the Examiner respectfully requests that support be shown for language added to any original claims on amendment and any new claims. That is, indicate support for newly added claim language by specifically pointing to page(s) and line

numbers in the specification and/or drawing figure(s). This will assist the Examiner in prosecuting this application.

37 C.F.R. § 1.83(a) requires the Drawings to illustrate or show all claimed features. Applicant must clearly point out the patentable novelty that they think the claims present, in view of the state of the art disclosed by the references cited or the objections made, and must also explain how the amendments avoid the references or objections. See 37 C.F.R. § 1.111(c).

Claim Objections

2. Claims 1-30 are objected to because of the following informalities:

- Claim 1 states: "wherein the logic and the flash memory are operable to".

It is unclear how the flash memory performs the recited functions: divide …, write …, generate …, and make modification…

- The clause "operable to" is relative. There is no clear indication that the functions are executed.
- Claims 2-29 (dependent directly or indirectly upon claim 1) repeat the recitation "the logic further operable". There is no clear indication that the functions are executed in the claims.
- Claim 30 is abstract idea because there is no logic to perform the functions, as recited.

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 1-30 are rejected under 35 U.S.C. 112(b) or 35 U.S.C. 112 (pre-AIA), second paragraph, as failing to set forth 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.

The presence of the clause "operable to" in "wherein the logic and the flash memory are operable to … " renders the claims indefinite.

In the case of "operable to" repeated throughout the claims, there is no affirmative limitations because they are merely indicating the recited logic and flash memory might be used or operable.

Claim Rejections - 35 USC § 101

5. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claim 30 is rejected under 35 U.S.C. 101 because the claimed invention (is directed to a judicial exception (i.e., a law of nature, a natural phenomenon, or an abstract idea) without significantly more. Claim 30 is directed to element for or function:

"method for providing a host with access to an SSD that contains non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units

and each erase unit comprising a plurality of pages, the method comprising, at the SSD: dividing the flash memory into zones; …

performing a multi-plane write of said data to erase units on different planes …

generating metadata based on accessing and maintenance …"

making a notification available to the host that recommends performing maintenance …"

The claim does not include additional elements that are sufficient to amount to significantly more than the judicial exception because the claim direct toward an abstract idea of using a function, without corresponding hardware to carry the function. See MPEP 2106 II.

Double Patenting

6. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on nonstatutory double patenting provided the reference application or patent either is shown to be commonly owned with the examined application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement. See MPEP § 717.02 for applications subject to examination under the first inventor to file provisions of the AIA as explained in MPEP § 2159. See MPEP § 2146 *et seq.* for applications not subject to examination under the first inventor to file provisions of the AIA. A terminal disclaimer must be signed in compliance with 37 CFR 1.321(b).

The USPTO Internet website contains terminal disclaimer forms which may be used. Please visit www.uspto.gov/patent/patents-forms. The filing date of the application in which the form is filed determines what form (e.g., PTO/SB/25, PTO/SB/26, PTO/AIA/25, or PTO/AIA/26) should be used. A web-based eTerminal Disclaimer may be filled out completely online using web-screens. An eTerminal Disclaimer that meets all requirements is auto-processed and approved immediately upon submission. For more information about eTerminal Disclaimers, refer to www.uspto.gov/patents/process/file/efs/guidance/eTD-info-I.jsp.

7. Claims 1-30 are provisionally rejected on the ground of nonstatutory double patenting as being unpatentable over claims 1-23 and 25-28 of copending Application No. 17/563,481. Although the claims at issue are not identical, they are not patentably distinct from each other because: claims 1-23 and 25-286 of application (481) anticipate claims 1-30 of the instant application

This is a provisional nonstatutory double patenting rejection.

Claim 1 (17/877,881)	Claim 1 (17/563,481)
A solid state storage drive (SSD) comprising:	A storage apparatus comprising:
a host interface configured to communicate with a host using commands from a Non-Volatile Memory Express (NVMe) standard; non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages;	at least one host processor; and one or more storage drives, a given one of the one or more storage drives comprising flash memory having planes, each of the planes having erase units, each of the erase units being associated with exactly one of the planes, the flash memory being organized so as to have subdivisions corresponding to respective, mutually-exclusive sets of the erase units, each of the mutually-exclusive sets comprising erase units that are in respective ones of the planes
logic coupled to the host interface and to the flash memory, wherein the logic comprises at least one of hardware logic and software logic, and communicates with the host via the host interface using NVMe	wherein the given one of the one or more storage drives further comprises circuitry to transmit to the at least one host processor information representing one or more of the subdivisions of the given one of the one or more storage drives, to identify associated

commands; wherein the logic and the flash memory are operable to: divide the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in at least one zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses; in response to a write command from the host to write data within the range of logical block addresses of a specified zone, perform a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes; generate metadata based on accessing and maintenance of erase units in the specified zone; and	subdivision size and associated subdivision logical address, store metadata respective to the subdivisions of the given one of the one or more storage drives, identify a need for maintenance for one of the subdivisions of the given one of the one or more storage drives dependent on the respective metadata, and a logical address associated with the need for maintenance in the one of the subdivisions, and transmit to the at least one host processor the logical address associated with the need for maintenance; and wherein the at least one host processor is to transmit a maintenance command to the given one of the one or more storage drives, the maintenance command being addressed to the one of the subdivisions for which the need for maintenance was identified, to cause the given one of the one or more storage drives to responsively erase two or

based on the metadata, make a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone.	more of the erase units corresponding to the one of the subdivisions for which the need for maintenance was identified, including erase units in at least two different ones of the planes

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Y. -H. Chang, W. -L. Lu, P. -C. Huang, L. -J. Lee and T. -W. Kuo, "An Efficient FTL Design for Multi-chipped Solid-State Drives," 2010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications, 2010, pp. 237-246.

US 11,307,995 teaching storage device comprising: flash memory having physical erase units; and circuitry to, for each of incoming data write requests: derive, from incoming address information, first, second and third address portion identifying an addressed block; and the circuitry to identify one of the physical erase units that corresponds to the identified logical erase unit, and program associated data into the identified one of the physical erase units.

US 20130166825 (KIM et al) teaching method of controlling a non-volatile memory device having multiple planes including sequentially transmitting data sets stored at respective buffer addresses to page buffers, respectively, of the planes, and

programming in parallel at least two data sets stored in respective page buffers to memory cells of the non-volatile memory device.

US 7,631,138 (Gonzalez et al) teaching memory system having non-volatile memory cells arranged in blocks as a unit of erase, pages therein as a unit of data programming and reading, and planes of a plurality of blocks that are independently accessible,; determining that more than said given number of one or more units of data having consecutive logical addresses are being received, writing the more than said given number of units of data in parallel into pages within two or more blocks of one of the metablocks in two or more planes.

US 9,710,377(Wayda et al) teaching managing multi-drive, multi-die or multi-plane NAND flash memory wherein a host directly assigns physical addresses and performs logical-to-physical address translation in a manner that reduces or eliminates the need for executing multiple of the same functions.

US 2015/0134930 (Huang et al) teaching memory-mapping an information unit (such as a file) into virtual memory such that the information unit is stored on a bulk-erase type block storage device using a single level of address translation, eliminating the need for the storage device to perform separate and independent address mappings.

Conclusion

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to PIERRE MICHEL BATAILLE whose telephone number is (571)272-4178. The examiner can normally be reached Monday - Thursday 7-6 ET.

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.

/PIERRE MICHEL BATAILLE/
Primary Examiner, Art Unit 2136

Notice of References Cited	Application/Control No. 17/877,881	Applicant(s)/Patent Under Reexamination Kuzmin et al.	
	Examiner PIERRE MICHEL BATAILLE	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	US-7631138-B2	12-2009	Gonzalez; Carlos J.	G06F12/0246	711/5
*	B	US-9710377-B1	07-2017	Kuzmin; Andrey V.	G06F8/654	1/1
*	C	US-11307995-B1	04-2022	Lercari; Robert	G06F3/064	1/1
*	D	US-20130166825-A1	06-2013	KIM; Jin Yeong	G06F12/0246	711/E12.008
*	E	US-20150134930-A1	05-2015	Huang; Jian	G06F3/0664	711/206
	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	Y. -H. Chang, W. -L. Lu, P. -C. Huang, L. -J. Lee and T. -W. Kuo, "An Efficient FTL Design for Multi-chipped Solid-State Drives," 2010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications, 2010, pp. 237-246.
	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.

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	**Art Unit**
	PIERRE MICHEL BATAILLE	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	09/26/2022								
	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	✓								
	26	✓								
	27	✓								
	28	✓								
	29	✓								
	30	✓								

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/877,881	Kuzmin et al.
	Examiner	Art Unit
	PIERRE MICHEL BATAILLE	2136

CPC - Searched*		
Symbol	Date	Examiner
G06F12/1009, G06F12/0292, G06F9/45545, G06F12/1036, G06F2009/45583	09/26/2022	PMB

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
PALM INVENTOR NAME & ASSIGNEE SAEARCH	09/16/2022	PMB
KEYWORD & CONTEXT PE2E SEARCH	09/26/2022	PMB
NON-PATENT LITERATURE SEARCH	09/26/2022	PMB

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

/PIERRE MICHEL BATAILLE/ Primary Examiner, Art Unit 2136	

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	3	("20200201789" "20200169383" "20200125501").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 05:38 AM
L2	15647	"Non- Volatile Memory Express" OR NVMe	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:34 AM
L3	120	Andrey near1 KUZMIN.in. James near1 WAYDA.in.	(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	2022/09/26 06:35 AM
L4	0	L3 AND L2 AND multiplane AND common near3 page near2 address	(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	2022/09/26 06:37 AM
L5	0	L3 AND L2 AND multiplane AND common near3 page near2 address	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:37 AM
L6	2	L3 AND L2 AND multiplane	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:37 AM
L7	2	L3 AND multiplane	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:39 AM
L8	682	eras$ near3 (block OR unit) WITH two near3 (plane OR storage OR memory) AND logical near3 address	(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	2022/09/26 06:59 AM

			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)				
L9	24	L8 AND eras$ near5 address AND common near3 address	(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	2022/09/26 07:00 AM
L10	3	L8 AND eras$ WITH address WITH common near3 address	(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	2022/09/26 07:00 AM
L11	494	eras$ near3 (block OR unit) WITH two near3 (plane OR storage OR memory) AND logical near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 07:00 AM
L12	18	L11 AND eras$ near5 address AND common near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 07:00 AM
L13	15	L11 AND eras$ WITH address WITH common near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 07:01 AM
L14	36	17/563,467 17/563,481 16/794,805 16/570,922 15/621,888 16/751,925	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 05:14 PM
L15	38	("11334479" "9710377" "11216365" "11314636" "9229854" "9519578" "10445229" "11188457"	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 05:23 PM

		"11288203" "8954708" "11003586" "20130166825" "20180203774" "10248515" "20210405898" "11269781" "11237978" "20160070507" "20220269440" "20180336139" "10915458" "11226903" "11137920" "8504798" "11100006" "20140258596" "10642748" "20180121134" "20110107013" "20220214808" "20130132646" "9189389" "9053009" "20090216938" "11347657" "10394489" "11416413" "11275695").pn.					
L16	21	("8954708" "20130166825" "8504798" "20110107013" "20130132646" "9053009" "20090216938" "8316175" "20050144358" "8117380" "20070245070" "7870328" "20100277968" "20100042774" "8400814" "20130173847" "8775722" "20090019217" "7913061" "20050141313" "20100023672").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 05:28 PM
L19	1627	logical near3 address WITH (plane OR range OR zone) AND (storage OR drive) near3 (divid$ OR subdivid$ OR sub?divid$ OR subdivis$)	(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	2022/09/26 06:55 PM

			VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L20	2	L19 AND generat$ near3 metadata AND maintenan$ near3 (need OR necessit$)	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/26 06:56 PM
L21	1598	logical near3 address WITH (plane OR range OR zone) AND (storage OR drive) near3 (divid$ OR subdivid$ OR sub?divid$ OR subdivis$)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:57 PM
L22	10	L21 AND generat$ near3 metadata AND maintenan$ near3 (need OR necessit$)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:58 PM
L23	0	(common OR single) near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND generat$ near3 metadata AND maintenan$ near3 (need OR necessit$)	(USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/26 08:22 PM
L24	9	(common OR single) near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND generat$ near3 metadata AND maintenan$ near3 (need OR necessit$)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:23 PM
L25	0	(common OR single)	(USOCR; FIT (AP, AT,	OR	ON	ON	2022/09/26

		near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND logical near4 (page OR block) near3 address	AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				08:25 PM
L26	36	(common OR single) near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND logical near4 (page OR block) near3 address	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:25 PM
L27	0	eras$ near3 unit WITH different adj2 (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/26 08:31 PM
L28	0	eras$ near3 unit WITH different adj2 (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:32 PM
L29	102	eras$ near3 unit WITH (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:32 PM

L30	13	eras$ near3 unit WITH (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ AND (storage OR drive OR memory))	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:39 PM
L31	47220	G06F12/1009.CPC. G06F12/0292.CPC. G06F9/45545.CPC. G06F12/1036.CPC. G06F2009/45583.CPC.	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/26 09:03 PM
L32	6	L31 AND eras$ near3 unit WITH (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/26 09:03 PM
L33	35	L31 AND eras$ near3 unit WITH (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 09:03 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	27	(eras$ near3 (block OR	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26

		unit) WITH two near3 (plane OR storage OR memory) AND logical near3 address).clm.					06:57 AM
N3	61	(logical near3 address WITH (plane OR range OR zone) AND (storage OR drive) near3 (divid$ OR subdivid$ OR sub?divid$ OR subdivis$)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 06:57 PM
N4	0	((common OR single) near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND generat$ near3 metadata AND maintenan$ near3 (need OR necessit$)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:22 PM
N5	0	((common OR single) near3 address WITH (write OR erase) near6 (multi?plane OR multi?zone OR multiplane OR multizone) AND logical near4 (page OR block) near3 address).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:24 PM
N6	329	((common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:28 PM
N7	0	(eras$ near3 unit WITH different adj2 (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$ and1 (storage OR drive OR memory))).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:29 PM
N8	10	(eras$ near3 unit WITH (storage OR memor$) AND (common OR single) near3 address WITH (write OR erase) near6 (multiblock OR multi?block OR multi$	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/26 08:33 PM

		and1 (storage OR drive OR memory))).clm.					

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	1	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A1	US 11,334,479	05-17-2022	Kuzmin et al.
	A2	US 11,321,237	05-03-2022	Lercari et al.
	A3	US 11,314,636	04-26-2022	Kuzmin et al.
	A4	US 11,275,695	03-15-2022	Lercari et al.
	A5	US 11,269,781	03-08-2022	Lercari et al.
	A6	US 11,226,903	01-18-2022	Lercari et al.
	A7	US 11,216,365	01-04-2022	Kuzmin et al.
	A8	US 11,188,457	11-30-2021	Kuzmin et al.
	A9	US 11,175,984	11-16-2021	Lercari et al.
	A10	US 11,163,492	11-02-2021	Kamran et al.
	A11	US 11,130,169	09-28-2021	Zumbusch et al.
	A12	US 11,023,387	06-01-2021	Lercari et al.
	A13	US 11,023,386	06-01-2021	Lercari et al.
	A14	US 11,023,315	06-01-2021	Jadon et al.
	A15	US 11,003,586	05-11-2021	Lercari et al.
	A16	US 10,977,188	04-13-2021	Lercari et al.
	A17	US 10,956,082	03-23-2021	Chen et al.
	A18	US 10,915,458	02-09-2021	Lercari et al.
	A19	US 10,884,915	01-05-2021	Kuzmin et al.
	A20	US 10,838,853	11-17-2020	Kuzmin et al.
	A21	US 10,642,748	05-05-2020	Lercari et al.
	A22	US 10,642,505	05-05-2020	Kuzmin et al.
	A23	US 10,552,085	02-04-2020	Chen et al.
	A24	US 10,552,058	02-04-2020	Jadon et al.

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	2	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A25	US 10,445,229	10-15-2019	Kuzmin et al.
	A26	US 10,063,379	08-28-2018	Pattanaik et al.
	A27	US 9,846,541	12-19-2017	Miyamoto et al.
	A28	US 9,785,572	10-10-2017	Lercari et al.
	A29	US 9,727,454	08-08-2017	Kuzmin et al.
	A30	US 9,710,377	07-18-2017	Kuzmin et al.
	A31	US 9,696,917	07-04-2017	Sareena et al.
	A32	US 9,652,376	05-16-2017	Kuzmin et al.
	A33	US 9,588,904	03-07-2017	Lercari et al.
	A34	US 9,542,118	01-10-2017	Lercari et al.
	A35	US 9,519,578	12-13-2016	Kuzmin et al.
	A36	US 9,400,749	07-26-2016	Kuzmin et al.
	A37	US 9,335,939	05-10-2016	Bennett et al.
	A38	US 9,229,854	01-05-2016	Kuzmin et al.
	A39	US 8,645,634	02-04-2014	Cox et al.
	A40	US 8,572,331	10-29-2013	Butterworth et al.
	A41	US 8,539,197	09-17-2013	Marshall et al.
	A42	US 8,347,042	01-01-2013	You
	A43	US 8,086,790	12-27-2011	Roohparvar
	A44	US 8,082,389	12-20-2011	Fujibayashi et al.
	A45	US 8,074,022	12-06-2011	Okin et al.
	A46	US 8,072,463	12-06-2011	Van Dyke et al.
	A47	US 8,068,365	11-29-2011	Kim
	A48	US 8,065,473	11-22-2011	Ito et al.

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	3	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A49	US 8,065,471	11-22-2011	Yano et al.
	A50	US 8,055,833	11-08-2011	Danilak et al.
	A51	US 8,024,545	09-20-2011	Kim et al.
	A52	US 8,001,318	08-16-2011	Iyer et al.
	A53	US 7,991,944	08-02-2011	Lee et al.
	A54	US 7,970,983	06-28-2011	Nochimowski
	A55	US 7,941,692	05-10-2011	Royer et al.
	A56	US 7,861,122	12-28-2010	Cornwell et al.
	A57	US 7,836,244	11-16-2010	Kim et al.
	A58	US 7,818,489	10-19-2010	Karamcheti et al.
	A59	US 7,801,561	09-21-2010	Parikh et al.
	A60	US 7,752,381	07-06-2010	Wong
	A61	US 7,710,777	05-04-2010	Montierth et al.
	A62	US 7,702,846	04-20-2010	Nakanishi et al.
	A63	US 7,581,078	08-25-2009	Ware
	A64	US 7,406,563	07-29-2008	Nagshain
	A65	US 7,404,031	07-22-2008	Oshima
	A66	US 7,339,823	03-04-2008	Nakayama et al.
	A67	US 7,096,378	08-22-2006	Stence et al.
	A68	US 6,134,631	10-17-2000	Jennings
	A69	US 5,860,082	01-12-1999	Smith et al.
	A70	US 5,652,857	07-29-1997	Shimoi et al.
	A71	US 5,568,423	10-22-1996	Jou et al.
	A72	US-2020/0368491 A1	11-26-2020	Poltorak

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	4	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A73	US-2017/0031699 A1	02-02-2017	Banerjee et al.
	A74	US-2016/0364179 A1	12-15-2016	Tsai et al.
	A75	US-2016/0357462 A1	12-08-2016	Nam et al.
	A76	US-2016/0026564 A1	01-28-2016	Manning
	A77	US-2016/0019159 A1	01-21-2016	UEDA et al.
	A78	US-2016/0018998 A1	01-21-2016	Mohan et al.
	A79	US-2015/0378886 A1	12-31-2015	Nemazie et al.
	A80	US-2015/0355965 A1	12-10-2015	Peddle
	A81	US-2015/0347296 A1	12-03-2015	Kotte et al.
	A82	US-2015/0324264 A1	11-12-2015	Chinnakkonda et al.
	A83	US-2015/0261456 A1	09-17-2015	Alcantara et al.
	A84	US-2015/0212938 A1	07-30-2015	CHEN et al.
	A85	US-2015/0193148 A1	07-09-2015	MIWA et al.
	A86	US-2015/0149789 A1	05-28-2015	SEO et al.
	A87	US-2015/0134930 A1	05-14-2015	Huang et al.
	A88	US-2015/0113203 A1	04-23-2015	Dancho et al.
	A89	US-2014/0297949 A1	10-02-2014	NAKAGAWA
	A90	US-2014/0215129 A1	07-31-2014	Kuzmin et al.
	A91	US-2014/0208062 A1	07-24-2014	COHEN
	A92	US-2014/0208004 A1	07-24-2014	Cohen et al.
	A93	US-2014/0189209 A1	07-03-2014	Sinclair et al.
	A94	US-2014/0122781 A1	05-01-2014	Smith et al.
	A95	US-2014/0101371 A1	04-10-2014	Nguyen et al.
	A96	US-2013/0332656 A1	12-12-2013	KANDIRAJU et al.

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	5	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A97	US-2013/0326117 A1	12-05-2013	Aune
	A98	US-2013/0297852 A1	11-07-2013	Fai et al.
	A99	US-2013/0290619 A1	10-31-2013	Knight
	A100	US-2013/0275682 A1	10-17-2013	Ramanujan et al.
	A101	US-2013/0242425 A1	09-19-2013	ZAYAS et al.
	A102	US-2013/0166825 A1	06-27-2013	KIM et al.
	A103	US-2013/0138868 A1	05-30-2013	Seroff et al.
	A104	US-2013/0124807 A1	05-16-2013	Nielsen et al.
	A105	US-2013/0124793 A1	05-16-2013	Gyl et al.
	A106	US-2013/0111298 A1	05-02-2013	Seroff et al.
	A107	US-2013/0111295 A1	05-02-2013	Li et al.
	A108	US-2013/0073816 A1	03-21-2013	SEO et al.
	A109	US-2013/0073793 A1	03-21-2013	YAMAGISHI et al.
	A110	US-2013/0019062 A1	01-17-2013	Bennett et al.
	A111	US-2013/0013852 A1	01-10-2013	Hou et al.
	A112	US-2013/0007343 A1	01-03-2013	Rub et al.
	A113	US-2012/0204079 A1	08-09-2012	Takefman et al.
	A114	US-2012/0198129 A1	08-02-2012	Van Aken et al.
	A115	US-2012/0159039 A1	06-21-2012	Kegel et al.
	A116	US-2012/0131381 A1	05-24-2012	Eleftheriou et al.
	A117	US-2012/0131270 A1	05-24-2012	HEMMI
	A118	US-2012/0079174 A1	03-29-2012	Nellans et al.
	A119	US-2012/0066441 A1	03-15-2012	Weingarten
	A120	US-2012/0059972 A1	03-08-2012	Chen

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	6	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A121	US-2012/0054419 A1	03-01-2012	Chen et al.
	A122	US-2012/0033519 A1	02-09-2012	Confalonieri et al.
	A123	US-2011/0314209 A1	12-22-2011	Eckstein
	A124	US-2011/0296133 A1	12-01-2011	Flynn et al.
	A125	US-2011/0276756 A1	11-10-2011	Bish et al.
	A126	US-2011/0238943 A1	09-29-2011	DEVENDRAN et al.
	A127	US-2011/0197023 A1	08-11-2011	Iwamitsu et al.
	A128	US-2011/0161784 A1	06-30-2011	Selinger et al.
	A129	US-2011/0153911 A1	06-23-2011	Sprouse et al.
	A130	US-2011/0055445 A1	03-03-2011	Gee et al.
	A131	US-2011/0033548 A1	02-10-2011	LAI et al.
	A132	US-2010/0329011 A1	12-30-2010	Lee et al.
	A133	US-2010/0312976 A1	12-09-2010	KANEDA
	A134	US-2010/0287327 A1	11-11-2010	Li et al.
	A135	US-2010/0287217 A1	11-11-2010	Borchers et al.
	A136	US-2010/0281230 A1	11-04-2010	Rabii et al.
	A137	US-2010/0262773 A1	10-14-2010	Borchers et al.
	A138	US-2010/0262765 A1	10-14-2010	CHEON et al.
	A139	US-2010/0262762 A1	10-14-2010	Borchers et al.
	A140	US-2010/0262761 A1	10-14-2010	Borchers et al.
	A141	US-2010/0241866 A1	09-23-2010	Rodorff
	A142	US-2010/0191779 A1	07-29-2010	Hinrichs
	A143	US-2010/0182838 A1	07-22-2010	KIM et al.
	A144	US-2010/0162012 A1	06-24-2010	Cornwell et al.

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	7	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A145	US-2010/0161882 A1	06-24-2010	Stern et al.
	A146	US-2010/0115172 A1	05-06-2010	GILLINGHAM et al.
	A147	US-2010/0042655 A1	02-18-2010	Tse et al.
	A148	US-2009/0327602 A1	12-31-2009	Moore et al.
	A149	US-2009/0300015 A1	12-03-2009	Kazan et al.
	A150	US-2009/0271562 A1	10-29-2009	Sinclair
	A151	US-2009/0254705 A1	10-08-2009	Abali et al.
	A152	US-2009/0198946 A1	08-06-2009	Ebata
	A153	US-2009/0172257 A1	07-02-2009	Prins et al.
	A154	US-2009/0172250 A1	07-02-2009	Allen et al.
	A155	US-2009/0172219 A1	07-02-2009	MARDIKS
	A156	US-2009/0089490 A1	04-02-2009	OZAWA et al.
	A157	US-2009/0089482 A1	04-02-2009	Traister
	A158	US-2009/0083478 A1	03-26-2009	Kunimatsu et al.
	A159	US-2009/0046533 A1	02-19-2009	Jo
	A160	US-2009/0036163 A1	02-05-2009	Kimbrell
	A161	US-2008/0320476 A1	12-25-2008	Wingard et al.
	A162	US-2008/0189485 A1	08-07-2008	Jung et al.
	A163	US-2008/0155204 A1	06-26-2008	Qawami et al.
	A164	US-2008/0147964 A1	06-19-2008	Chow et al.
	A165	US-2008/0126690 A1	05-29-2008	Rajan et al.
	A166	US-2008/0082596 A1	04-03-2008	Gorobets
	A167	US-2008/0034153 A1	02-07-2008	Lee et al.
	A168	US-2007/0283428 A1	12-06-2007	Ma et al.

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	8	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A169	US-2007/0260811 A1	11-08-2007	Merry et al.
	A170	US-2007/0233939 A1	10-04-2007	KIM
	A171	US-2007/0168321 A1	07-19-2007	Saito et al.
	A172	US-2007/0143569 A1	06-21-2007	Sanders et al.
	A173	US-2007/0058431 A1	03-15-2007	Chung et al.
	A174	US-2007/0046681 A1	03-01-2007	Nagashima
	A175	US-2007/0019481 A1	01-25-2007	Park
	A176	US-2006/0022171 A1	02-02-2006	Leist
	A177	US-2005/0144413 A1	06-30-2005	Kuo et al.
	A178	US-2003/0188032 A1	10-02-2003	Solomon et al.
	A179	US-2003/0065866 A1	04-03-2003	Spencer
	A180	US-2003/0028733 A1	02-06-2003	Tsunoda et al.

FOREIGN PATENT DOCUMENTS					
		Foreign Patent Document			
Examiner Initials*	Cite No.[1]	Country Code[3] – Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	T[6]
	B1	WO-2009/0100149 A1	08-01-2009	RAMBUS, INC.	

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C1	Altera Corp, et al., "Hybrid Memory Cube" specification, 2012, 122 pages.	
	C2	Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, and Vijayan Prabhakaran, "ResearchRemoving The Costs of Indirection in Flash-based SSDs with NamelessWrites," Jun. 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.	

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	9	of	11	Attorney Docket Number	39471-53126/US

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C3	C. Kuo, J. Hsieh and L. Chang, "Detecting Solid-State Disk Geometry for Write Pattern Optimization," 2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications, 2011, pp. 89-94.	
	C4	Cornwell, "Anatomy of a solid state drive," Oct. 17, 2012, ACMQUEUE, 13 pages.	
	C5	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," Apr. 10, 2010, MSST '10 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1-12.	
	C6	Garth Gibson, Greg Ganger, "Principles of Operation for Shingled Disk Devices," Canregie Mellon Parallel Data Laboratory, CMU-PDL-11-107, Apr. 2011, 9 pages.	
	C7	High-Speed NAND Flash: Design Considerations to Maximize Performance, Flash Memory Summit, Santa Clara, CA USA Aug. 11, 2009, , Robert Pierce, pp. 1-19.	
	C8	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, May 6-10, 2013, 5 pages.	
	C9	Intel Corp, PCI-SIG SR-IOV Primer--An Introduction to SR-IOV Technology,: 321211-002, Revision 2.5, Jan. 2011, 28 pages.	
	C10	J. Kim, D. Jung, J. Kim and J. Huh, "A methodology for extracting performance parameters in solid state disks (SSDs)," 2009 IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems, London, 2009, pp. 1-10.	
	C11	JEDEC Standard, JESD229, Wide IO, Dec. 2011, 74 pages.	
	C12	John D. Strunk, "Hybrid Aggregates: Combining SSDs and HDDs in a single storage pool," Dec. 15, 2012, ACM SIGOPS Operating Systems Review archive, vol. 46 Issue 3, Dec. 2012, pp. 50-56.	
	C13	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, ASPDAC 2008, 26 pages.	
	C14	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.	
	C15	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, Feb. 23, 2010.	
	C16	NAND 201: An Update on the Continued Evolution of NAND Flash, Jim Cooke, Micron White Paper, Sep. 6, 2011, pp. 1-10.	
	C17	NVM Express, V. 1.2.1, 217 pages, Jun. 3, 2016.	
	C18	NVM Express, Version 1.0b, Jul. 12, 2011, pp. 1-126, published at http://www.nvmexpress.org/resources/ by the NVM Express Work Group.	

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	10	of	11	Attorney Docket Number	39471-53126/US

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C19	Ohad, Rodeh, "IBM Research Report Defragmentation Mechanisms for Copy-on-Write File-systems," IBM white paper, Apr. 26, 2010, 10 pages, available at domino.watson.IBM.com/library/CyberDig.nsf/papers/298A0EF3C2CDB17B8525770- 70056B41F/$File/rj 10465.pdf.	
	C20	Open NAND Flash Interface (ONFI), specification, version 2.0, 174 pages, Feb. 27, 2008.	
	C21	Open NAND Flash Interface (ONFI), specification, version 3.1, 296 pages Sep. 19, 2012.	
	C22	Optimizing NAND Flash Performance, Flash Memory Summit, Santa Clara, CA USA Aug. 2008, Ryan Fisher, pp. 1-23.	
	C23	Ouyang et al., "SDF: Software-defined flash for web-scale internet storage systems," Computer Architecture News--ASPLOS '14, vol. 42 Issue 1, Mar. 2014, 14 pages.	
	C24	S Im and D. Shin, "Flash-Aware RAID Techniques for Dependable and High-Performance Flash Memory SSD," in IEEE Transactions on Computers, vol. 60, No. 1, pp. 80-92, Jan. 2011.	
	C25	Spansion SLC NAND Flash Memory for Embedded, data sheet, S34ML01G1, S34ML02G1, S34ML04G1, Sep. 6, 2012, pp. 1-73.	
	C26	Stan Park and Kai Shen, Department of Computer Science, University of Rochester, "FIOS: A Fair, Efficient Flash I/O Scheduler," Feb. 23, 2012, pp. 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.	
	C27	W. Lafi, D. Lattard and A. Jerraya, "High level modelling and performance evaluation of address mapping in NAND flash memory," 2009 16th IEEE International Conference on Electronics, Circuits and Systems--(ICECS 2009), Yasmine Hammamet, 2009, pp. 659-662.	
	C28	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, Apr. 14, 2014, 14 pages.	
	C29	Xiangyong Ouyangyz, David Nellansy, Robert Wipfely, David Flynny, Dhabaleswar K. Pandaz, "Beyond Block I/O Rethinking Traditional Storage Primitives," Aug. 20, 2011, published at http://www.sciweavers.org/read/beyond-block-i-o-rethinking-traditional-st- orage-primitives-327868, by Fusion IO and the Ohio State University.	
	C30	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, Feb. 7, 2012, pp. 1-16.	

EXAMINER SIGNATURE	
Examiner Signature	/PIERRE MICHEL BATAILLE/
Date Considered	09/16/2022

*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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	11	of	11	Attorney Docket Number	39471-53126/US

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).

☒ This application relies, under 35 U.S.C. § 120, on the earlier filing date of prior application No. 16/570,922 filed September 13, 2019, and prior application No. 16/751,925 filed January 24, 2020, and the references cited therein **B1 and C1-C30** are hereby referenced, but are not required to be provided in this application under 37 CFR § 1.98(d).

☐ 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	/Michael W. Farn/	Date (YYYY-MM-DD)	2022-08-02
Name/Print	Michael W. Farn	Registration Number	41,015

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /P.B/

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	3	logical WITH physical AND controller WITH interleav$ WITH write WITH read AND eras$ AND maintenance near3 (need$ OR necess$) AND metadata	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 06:51 PM
L2	10	logical WITH physical AND controller WITH interleav$ WITH write WITH read AND eras$ AND maintenance near3 (need$ OR necess$) AND metadata	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 06:51 PM
L3	9858	(flash OR storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 08:22 PM
L4	17	L3 AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(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	2022/09/12 08:24 PM

			IBM_TDB)				
L5	3070	(flash OR storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:25 PM
L6	63	L5 AND (memory OR storage) near3 (space OR area) AND reserv$ near3 (size OR capacity) AND bandwidth	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:25 PM
L7	23	((flash OR storage OR memory) near5 maintenance AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)) AND reserv$ near3 (size OR capacity) AND bandwidth	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 08:28 PM
L8	96	((flash OR storage OR memory) near5 maintenance AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)) AND reserv$ near3 (size OR capacity) AND bandwidth	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:28 PM
L9	35	L8 AND erase$ near3 unit AND physical near3 plane AND logical	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:30 PM
L10	625	(flash OR storage OR memory) WITH plane AND (memory OR storage) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH,	OR	ON	ON	2022/09/12 09:26 PM

			TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L11	0	L10 AND eras$ WITH host WITH size WITH logical near2 address$	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 09:27 PM
L12	2	L10 AND eras$ WITH size WITH logical near2 address$	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 09:27 PM
L13	426	(flash OR storage OR memory) WITH plane AND (memory OR storage) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:27 PM
L14	25	L13 AND eras$ WITH size WITH logical near2 address$	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:28 PM
L15	2	maintenance near3 (need$ OR necess$) WITH metadata AND wear near3 level$ AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND eras$ near3 (bulk OR unit OR block)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL,	OR	ON	ON	2022/09/12 09:30 PM

			PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L16	5	maintenance near3 (need$ OR necess$) WITH metadata AND wear near3 level$ AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND eras$ near3 (bulk OR unit OR block)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:31 PM
L17	2	maintenance near3 (need$ OR necess$) WITH metadata AND wear near3 level$ AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND logical WITH physical	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 09:34 PM
L18	5	maintenance near3 (need$ OR necess$) WITH metadata AND wear near3 level$ AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND logical WITH physical	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:34 PM
L19	43	maintenance near5 (identif$ OR need$ OR necess$) WITH (wear near2 level$ OR eras$) AND (storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set)	(USOCR; FIT (AP, AT, AU, 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, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	OR	ON	ON	2022/09/12 09:57 PM
L20	37	maintenance near5 (identif$ OR need$ OR	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:57 PM

		necess$) WITH (wear near2 level$ OR eras$) AND (storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set)					
L23	26	("8626989" "20120198128" "20120198129" "8595415" "20090187798" "7813212" "20130166825" "8954708" "7913061" "20100306482" "8533405" "7509471" "7139864" "7631162" "20030065899" "6898662" "20120198186" "10048884" "8239643" "20050141313" "8103841" "20090019217" "8051257" "20070113029" "7793068" "20070113030").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 10:16 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	0	(logical WITH physical AND controller WITH interleav$ WITH write WITH read AND eras$ AND maintenance near3 (need$ OR necess$) AND metadata).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 06:50 PM
N2	12	((flash OR storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND maintenance near5 (identif$ OR need$ OR necess$)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:21 PM
N4	0	(((flash OR storage OR memory) near5 maintenance AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set)	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 08:27 PM

		AND maintenance near5 (identif$ OR need$ OR necess$)) AND reserv$ near3 (size OR capacity) AND bandwidth).clm.					
N5	0	(maintenance near3 (need$ OR necess$) WITH metadata AND wear near3 level$ AND (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set) AND eras$ near3 (bulk OR unit OR block)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:29 PM
N6	1	(maintenance near5 (identif$ OR need$ OR necess$) WITH (wear near2 level$ OR eras$) AND (storage OR memory) near5 (divid$ OR subdivid$ OR sub?divid$ OR subset OR sub?set)).clm.	(US-PGPUB; USPAT)	OR	ON	ON	2022/09/12 09:56 PM

High-performance NAND flash controller exploiting parallel out-of-order command execution

IEEE XploreIEEE Conferences2010-04-013Proceedings of 2010 International Symposium on VLSI Design, Automation and Test (Page(s): 160-163)

A parallel flash translation layer based on page group-block hybrid-mapping method

IEEE XploreIEEE Periodicals2012-05-013IEEE Transactions on Consumer Electronics

A technique to improve garbage collection performance for NAND flash-based storage systems

IEEE XploreIEEE Periodicals2012-05-013IEEE Transactions on Consumer Electronics

Exploiting Internal Parallelism of Flash-based SSDs

IEEE XploreIEEE Periodicals2010-01-013IEEE Computer Architecture Letters

NANDFlashSim: Intrinsic latency variation aware NAND flash memory system modeling and simulation at microarchitecture level

IEEE XploreIEEE Conferences2012-04-0132012 IEEE 28th Symposium on Mass Storage Systems and Technologies (MSST) (Page(s): 1-12)

Commands scheduling optimized flash controller for high bandwidth SSD application

IEEE XploreIEEE Conferences2012-10-0132012 IEEE 11th International Conference on Solid-State and Integrated Circuit Technology (Page(s): 1-3)

Optimized structure for fast data storage to non-volatile memory with minimum chip size

IEEE XploreIEEE Conferences2004-01-013Proceedings 7th International Conference on Signal Processing, 2004. Proceedings. ICSP '04, 2004.

Formalising Flash Memory: First Steps

IEEE XploreIEEE Conferences2007-07-01312th IEEE International Conference on Engineering Complex Computer Systems (ICECCS 2007) (Page(s): 251-260)

An Efficient FTL Design for Multi-chipped Solid-State Drives

IEEE XploreIEEE Conferences2010-08-0132010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications (Page(s): 237-246)

ZFTL: A Zone-based Flash Translation Layer with a two-tier selective caching mechanism

IEEE XploreIEEE Conferences2012-11-0132012 IEEE 14th International Conference on Communication Technology (Page(s): 578-588)

F-KDB: An K-D-B Tree Implementation over Flash Memory

IEEE XploreIEEE Conferences2010-06-0132010 10th IEEE International Conference on Computer and Information Technology (Page(s): 635-642)

FLASH modelling for wearleveling algorithms

IEEE XploreIEEE Conferences2011-12-0138th International Conference on High-capacity Optical Networks and Emerging Technologies (Page(s): 267-272)

- High level modelling and performance evaluation of address mapping in NAND flash memory
  IEEE XploreIEEE Conferences2009-12-0132009 16th IEEE International Conference on Electronics, Circuits and Systems - (ICECS 2009) (Page(s): 659-662)

- A Demand-Based FTL Scheme Using Dualistic Approach on Data Blocks and Translation Blocks
  IEEE XploreIEEE Conferences2011-08-0132011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications

- A commitment-based management strategy for the performance and reliability enhancement of flash-memory storage systems
  IEEE XploreIEEE Conferences2009-07-0132009 46th ACM/IEEE Design Automation Conference (Page(s): 858-863)

- A Management Strategy for the Reliability and Performance Improvement of MLC-Based Flash-Memory Storage Systems
  IEEE XploreIEEE Periodicals2011-01-273IEEE Transactions on Computers

- nand Flash Memory and Its Role in Storage Architectures
  IEEE XploreIEEE Periodicals2008-11-013Proceedings of the IEEE

- Cross-Layer Optimizations in Solid-State Drives
  IEEE XploreIEEE Periodicals2011-12-013IEEE Embedded Systems Letters

- Design of a solid-state file using flash EEPROM
  IEEE XploreIBM Periodicals1995-09-013IBM Journal of Research and Development

- Software Based NAND Flash Management Techniques
  IEEE XploreIEEE Conferences2009-04-0132009 International Conference on Computing, Engineering and Information (Page(s): 168-171)

- CAST: A page-level FTL with compact address mapping and parallel data blocks
  IEEE XploreIEEE Conferences2012-12-0132012 IEEE 31st International Performance Computing and Communications Conference (IPCCC) (Page(s): 142-151)

- Advanced Nonvolatile Memory Designs and Technologies
  IEEE XploreWiley-IEEE eBooks2002-01-013Wiley-IEEE Press 2002

- NAND Flash Memory-Based SSD Controller Using Full Page-Associative Block Recycling Mechanism
  IEEE XploreIEEE Conferences2011-05-0132011 3rd IEEE International Memory Workshop (IMW) (Page(s): 1-4)

- Ozone (O3): An Out-of-Order Flash Memory Controller Architecture
  IEEE XploreIEEE Periodicals2010-10-073IEEE Transactions on Computers

☐ Fast erase algorithm using flash translation layer in NAND-type flash memory

IEEE XploreIEEE Periodicals2011-11-01IEEE Transactions on Consumer Electronics

Bibliographic Data

Application No: 17/877,881

Foreign Priority claimed: ○ Yes ◉ No

35 USC 119 (a-d) conditions met: ☐ Yes ☑ No ☐ Met After Allowance

Verified and Acknowledged: /PIERRE MICHEL BATAILLE/ (Examiner's Signature) | (Initials)

Title: Storage System with Multiplane Segments and Cooperative Flash Management

FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
07/29/2022	711	2136	39471-53126/US
RULE			

APPLICANTS

Radian Memory Systems, Inc., Manhattan Beach, CA,

INVENTORS

Andrey V. Kuzmin, Moscow, RUSSIAN FEDERATION

James G. Wayda, Laguna Niguel, CA, UNITED STATES

CONTINUING DATA

This application is a CON of 17563481 12/28/2021

17563481 is a CON of 16751925 01/24/2020 PAT 11314636

16751925 is a CON of 16570922 09/13/2019 PAT 11188457

16570922 is a CON of 15621888 06/13/2017 PAT 10445229

15621888 is a CON of 15346641 11/08/2016 PAT 9710377

15346641 is a CON of 14951708 11/25/2015 PAT 9519578

14951708 is a CON of 14047193 10/07/2013 PAT 9229854

14047193 is a CIP of 13767723 02/14/2013 PAT 9652376

13767723 has PRO of 61757464 01/28/2013

FOREIGN APPLICATIONS

IF REQUIRED, FOREIGN LICENSE GRANTED**

08/10/2022

**** SMALL ENTITY ****

STATE OR COUNTRY

RUSSIAN FEDERATION

ADDRESS

FENWICK & WEST LLP

SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041
UNITED STATES

FILING FEE RECEIVED

$3,500

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.
17/877,881	07/29/2022	Andrey V. Kuzmin	39471-53126/US	3676

758 7590 08/30/2022

FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

EXAMINER
CENTRAL, DOCKET

ART UNIT	PAPER NUMBER
OPAP	

NOTIFICATION DATE	DELIVERY MODE
08/30/2022	ELECTRONIC

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.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PTOC@Fenwick.com

Decision Granting Request for Prioritized Examination (Track I)	**Application No.** 17/877,881	**Applicant(s)** Kuzmin et al.	
	Examiner FIKIRTE A GEREMEW	**Art Unit** OMBL	**AIA (FITF) Status** Yes

1. THE REQUEST FILED 29 July 2022 IS **GRANTED** .

The above-identified application has met the requirements for prioritized examination

A. ☑ for an original nonprovisional application (Track I).
B. ☐ for an application undergoing continued examination (RCE).

2. **The above-identified application will undergo prioritized examination**. The application will be accorded special status throughout its entire course of prosecution until one of the following occurs:

A. filing a **petition for extension of time** to extend the time period for filing a reply;

B. filing an **amendment to amend the application to contain more than four independent claims, more than thirty total claims** , or a multiple dependent claim;

C. filing a **request for continued examination** ;

D. filing a notice of appeal;

E. filing a request for suspension of action;

F. mailing of a notice of allowance;

G. mailing of a final Office action;

H. completion of examination as defined in 37 CFR 41.102; or

I. abandonment of the application.

Telephone inquiries with regard to this decision should be directed to FIKIRTE GEREMEW at (703) 756-1930. In his/her absence, calls may be directed to Petition Help Desk at (571) 272-3282.

/FIKIRTE A GEREMEW/ PROGRAM SUPPORT ASSISTANT, OMBL	

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
17/877,881	07/29/2022	2185	1330	39471-53126/US	30	2

CONFIRMATION NO. 3676

758
FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041

FILING RECEIPT

OC000000135737807

Date Mailed: 08/11/2022

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;
James G. Wayda, Laguna Niguel, CA;

Applicant(s)

Radian Memory Systems, Inc., Manhattan Beach, CA;

Power of Attorney: None

Domestic Priority data as claimed by applicant

This application is a CON of 17/563,481 12/28/2021
which is a CON of 16/751,925 01/24/2020 PAT 11314636
which is a CON of 16/570,922 09/13/2019 PAT 11188457
which is a CON of 15/621,888 06/13/2017 PAT 10445229
which is a CON of 15/346,641 11/08/2016 PAT 9710377
which is a CON of 14/951,708 11/25/2015 PAT 9519578
which is a CON of 14/047,193 10/07/2013 PAT 9229854
which is a CIP 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: Yes

Permission to Access Search Results: Yes

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: 08/10/2022

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 17/877,881**

Projected Publication Date: Request for Non-Publication Acknowledged

Non-Publication Request: Yes

Early Publication Request: No

** **SMALL ENTITY** **

Title

Storage System with Multiplane Segments and Cooperative Flash Management

Preliminary Class

710

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: Yes

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: 17/877,881

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	80		N/A	
SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A	350		N/A	
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A	400		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	30 minus 20 =	* 10	x 50 =	500	OR		
INDEPENDENT CLAIMS (37 CFR 1.16(h))	2 minus 3 =	*	x 240 =	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	1330		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.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	1	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A1	US 11,334,479	05-17-2022	Kuzmin et al.
	A2	US 11,321,237	05-03-2022	Lercari et al.
	A3	US 11,314,636	04-26-2022	Kuzmin et al.
	A4	US 11,275,695	03-15-2022	Lercari et al.
	A5	US 11,269,781	03-08-2022	Lercari et al.
	A6	US 11,226,903	01-18-2022	Lercari et al.
	A7	US 11,216,365	01-04-2022	Kuzmin et al.
	A8	US 11,188,457	11-30-2021	Kuzmin et al.
	A9	US 11,175,984	11-16-2021	Lercari et al.
	A10	US 11,163,492	11-02-2021	Kamran et al.
	A11	US 11,130,169	09-28-2021	Zumbusch et al.
	A12	US 11,023,387	06-01-2021	Lercari et al.
	A13	US 11,023,386	06-01-2021	Lercari et al.
	A14	US 11,023,315	06-01-2021	Jadon et al.
	A15	US 11,003,586	05-11-2021	Lercari et al.
	A16	US 10,977,188	04-13-2021	Lercari et al.
	A17	US 10,956,082	03-23-2021	Chen et al.
	A18	US 10,915,458	02-09-2021	Lercari et al.
	A19	US 10,884,915	01-05-2021	Kuzmin et al.
	A20	US 10,838,853	11-17-2020	Kuzmin et al.
	A21	US 10,642,748	05-05-2020	Lercari et al.
	A22	US 10,642,505	05-05-2020	Kuzmin et al.
	A23	US 10,552,085	02-04-2020	Chen et al.
	A24	US 10,552,058	02-04-2020	Jadon et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	2	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A25	US 10,445,229	10-15-2019	Kuzmin et al.
	A26	US 10,063,379	08-28-2018	Pattanaik et al.
	A27	US 9,846,541	12-19-2017	Miyamoto et al.
	A28	US 9,785,572	10-10-2017	Lercari et al.
	A29	US 9,727,454	08-08-2017	Kuzmin et al.
	A30	US 9,710,377	07-18-2017	Kuzmin et al.
	A31	US 9,696,917	07-04-2017	Sareena et al.
	A32	US 9,652,376	05-16-2017	Kuzmin et al.
	A33	US 9,588,904	03-07-2017	Lercari et al.
	A34	US 9,542,118	01-10-2017	Lercari et al.
	A35	US 9,519,578	12-13-2016	Kuzmin et al.
	A36	US 9,400,749	07-26-2016	Kuzmin et al.
	A37	US 9,335,939	05-10-2016	Bennett et al.
	A38	US 9,229,854	01-05-2016	Kuzmin et al.
	A39	US 8,645,634	02-04-2014	Cox et al.
	A40	US 8,572,331	10-29-2013	Butterworth et al.
	A41	US 8,539,197	09-17-2013	Marshall et al.
	A42	US 8,347,042	01-01-2013	You
	A43	US 8,086,790	12-27-2011	Roohparvar
	A44	US 8,082,389	12-20-2011	Fujibayashi et al.
	A45	US 8,074,022	12-06-2011	Okin et al.
	A46	US 8,072,463	12-06-2011	Van Dyke et al.
	A47	US 8,068,365	11-29-2011	Kim
	A48	US 8,065,473	11-22-2011	Ito et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	3	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A49	US 8,065,471	11-22-2011	Yano et al.
	A50	US 8,055,833	11-08-2011	Danilak et al.
	A51	US 8,024,545	09-20-2011	Kim et al.
	A52	US 8,001,318	08-16-2011	Iyer et al.
	A53	US 7,991,944	08-02-2011	Lee et al.
	A54	US 7,970,983	06-28-2011	Nochimowski
	A55	US 7,941,692	05-10-2011	Royer et al.
	A56	US 7,861,122	12-28-2010	Cornwell et al.
	A57	US 7,836,244	11-16-2010	Kim et al.
	A58	US 7,818,489	10-19-2010	Karamcheti et al.
	A59	US 7,801,561	09-21-2010	Parikh et al.
	A60	US 7,752,381	07-06-2010	Wong
	A61	US 7,710,777	05-04-2010	Montierth et al.
	A62	US 7,702,846	04-20-2010	Nakanishi et al.
	A63	US 7,581,078	08-25-2009	Ware
	A64	US 7,406,563	07-29-2008	Nagshain
	A65	US 7,404,031	07-22-2008	Oshima
	A66	US 7,339,823	03-04-2008	Nakayama et al.
	A67	US 7,096,378	08-22-2006	Stence et al.
	A68	US 6,134,631	10-17-2000	Jennings
	A69	US 5,860,082	01-12-1999	Smith et al.
	A70	US 5,652,857	07-29-1997	Shimoi et al.
	A71	US 5,568,423	10-22-1996	Jou et al.
	A72	US-2020/0368491 A1	11-26-2020	Poltorak

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	4	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A73	US-2017/0031699 A1	02-02-2017	Banerjee et al.
	A74	US-2016/0364179 A1	12-15-2016	Tsai et al.
	A75	US-2016/0357462 A1	12-08-2016	Nam et al.
	A76	US-2016/0026564 A1	01-28-2016	Manning
	A77	US-2016/0019159 A1	01-21-2016	UEDA et al.
	A78	US-2016/0018998 A1	01-21-2016	Mohan et al.
	A79	US-2015/0378886 A1	12-31-2015	Nemazie et al.
	A80	US-2015/0355965 A1	12-10-2015	Peddle
	A81	US-2015/0347296 A1	12-03-2015	Kotte et al.
	A82	US-2015/0324264 A1	11-12-2015	Chinnakkonda et al.
	A83	US-2015/0261456 A1	09-17-2015	Alcantara et al.
	A84	US-2015/0212938 A1	07-30-2015	CHEN et al.
	A85	US-2015/0193148 A1	07-09-2015	MIWA et al.
	A86	US-2015/0149789 A1	05-28-2015	SEO et al.
	A87	US-2015/0134930 A1	05-14-2015	Huang et al.
	A88	US-2015/0113203 A1	04-23-2015	Dancho et al.
	A89	US-2014/0297949 A1	10-02-2014	NAKAGAWA
	A90	US-2014/0215129 A1	07-31-2014	Kuzmin et al.
	A91	US-2014/0208062 A1	07-24-2014	COHEN
	A92	US-2014/0208004 A1	07-24-2014	Cohen et al.
	A93	US-2014/0189209 A1	07-03-2014	Sinclair et al.
	A94	US-2014/0122781 A1	05-01-2014	Smith et al.
	A95	US-2014/0101371 A1	04-10-2014	Nguyen et al.
	A96	US-2013/0332656 A1	12-12-2013	KANDIRAJU et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	5	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A97	US-2013/0326117 A1	12-05-2013	Aune
	A98	US-2013/0297852 A1	11-07-2013	Fai et al.
	A99	US-2013/0290619 A1	10-31-2013	Knight
	A100	US-2013/0275682 A1	10-17-2013	Ramanujan et al.
	A101	US-2013/0242425 A1	09-19-2013	ZAYAS et al.
	A102	US-2013/0166825 A1	06-27-2013	KIM et al.
	A103	US-2013/0138868 A1	05-30-2013	Seroff et al.
	A104	US-2013/0124807 A1	05-16-2013	Nielsen et al.
	A105	US-2013/0124793 A1	05-16-2013	Gyl et al.
	A106	US-2013/0111298 A1	05-02-2013	Seroff et al.
	A107	US-2013/0111295 A1	05-02-2013	Li et al.
	A108	US-2013/0073816 A1	03-21-2013	SEO et al.
	A109	US-2013/0073793 A1	03-21-2013	YAMAGISHI et al.
	A110	US-2013/0019062 A1	01-17-2013	Bennett et al.
	A111	US-2013/0013852 A1	01-10-2013	Hou et al.
	A112	US-2013/0007343 A1	01-03-2013	Rub et al.
	A113	US-2012/0204079 A1	08-09-2012	Takefman et al.
	A114	US-2012/0198129 A1	08-02-2012	Van Aken et al.
	A115	US-2012/0159039 A1	06-21-2012	Kegel et al.
	A116	US-2012/0131381 A1	05-24-2012	Eleftheriou et al.
	A117	US-2012/0131270 A1	05-24-2012	HEMMI
	A118	US-2012/0079174 A1	03-29-2012	Nellans et al.
	A119	US-2012/0066441 A1	03-15-2012	Weingarten
	A120	US-2012/0059972 A1	03-08-2012	Chen

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	6	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A121	US-2012/0054419 A1	03-01-2012	Chen et al.
	A122	US-2012/0033519 A1	02-09-2012	Confalonieri et al.
	A123	US-2011/0314209 A1	12-22-2011	Eckstein
	A124	US-2011/0296133 A1	12-01-2011	Flynn et al.
	A125	US-2011/0276756 A1	11-10-2011	Bish et al.
	A126	US-2011/0238943 A1	09-29-2011	DEVENDRAN et al.
	A127	US-2011/0197023 A1	08-11-2011	Iwamitsu et al.
	A128	US-2011/0161784 A1	06-30-2011	Selinger et al.
	A129	US-2011/0153911 A1	06-23-2011	Sprouse et al.
	A130	US-2011/0055445 A1	03-03-2011	Gee et al.
	A131	US-2011/0033548 A1	02-10-2011	LAI et al.
	A132	US-2010/0329011 A1	12-30-2010	Lee et al.
	A133	US-2010/0312976 A1	12-09-2010	KANEDA
	A134	US-2010/0287327 A1	11-11-2010	Li et al.
	A135	US-2010/0287217 A1	11-11-2010	Borchers et al.
	A136	US-2010/0281230 A1	11-04-2010	Rabii et al.
	A137	US-2010/0262773 A1	10-14-2010	Borchers et al.
	A138	US-2010/0262765 A1	10-14-2010	CHEON et al.
	A139	US-2010/0262762 A1	10-14-2010	Borchers et al.
	A140	US-2010/0262761 A1	10-14-2010	Borchers et al.
	A141	US-2010/0241866 A1	09-23-2010	Rodorff
	A142	US-2010/0191779 A1	07-29-2010	Hinrichs
	A143	US-2010/0182838 A1	07-22-2010	KIM et al.
	A144	US-2010/0162012 A1	06-24-2010	Cornwell et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	7	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A145	US-2010/0161882 A1	06-24-2010	Stern et al.
	A146	US-2010/0115172 A1	05-06-2010	GILLINGHAM et al.
	A147	US-2010/0042655 A1	02-18-2010	Tse et al.
	A148	US-2009/0327602 A1	12-31-2009	Moore et al.
	A149	US-2009/0300015 A1	12-03-2009	Kazan et al.
	A150	US-2009/0271562 A1	10-29-2009	Sinclair
	A151	US-2009/0254705 A1	10-08-2009	Abali et al.
	A152	US-2009/0198946 A1	08-06-2009	Ebata
	A153	US-2009/0172257 A1	07-02-2009	Prins et al.
	A154	US-2009/0172250 A1	07-02-2009	Allen et al.
	A155	US-2009/0172219 A1	07-02-2009	MARDIKS
	A156	US-2009/0089490 A1	04-02-2009	OZAWA et al.
	A157	US-2009/0089482 A1	04-02-2009	Traister
	A158	US-2009/0083478 A1	03-26-2009	Kunimatsu et al.
	A159	US-2009/0046533 A1	02-19-2009	Jo
	A160	US-2009/0036163 A1	02-05-2009	Kimbrell
	A161	US-2008/0320476 A1	12-25-2008	Wingard et al.
	A162	US-2008/0189485 A1	08-07-2008	Jung et al.
	A163	US-2008/0155204 A1	06-26-2008	Qawami et al.
	A164	US-2008/0147964 A1	06-19-2008	Chow et al.
	A165	US-2008/0126690 A1	05-29-2008	Rajan et al.
	A166	US-2008/0082596 A1	04-03-2008	Gorobets
	A167	US-2008/0034153 A1	02-07-2008	Lee et al.
	A168	US-2007/0283428 A1	12-06-2007	Ma et al.

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	8	of	11	Attorney Docket Number	39471-53126/US

U.S. PATENT DOCUMENTS				
		Document No.		
Examiner Initials*	Cite No.[1]	Number – Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document
	A169	US-2007/0260811 A1	11-08-2007	Merry et al.
	A170	US-2007/0233939 A1	10-04-2007	KIM
	A171	US-2007/0168321 A1	07-19-2007	Saito et al.
	A172	US-2007/0143569 A1	06-21-2007	Sanders et al.
	A173	US-2007/0058431 A1	03-15-2007	Chung et al.
	A174	US-2007/0046681 A1	03-01-2007	Nagashima
	A175	US-2007/0019481 A1	01-25-2007	Park
	A176	US-2006/0022171 A1	02-02-2006	Leist
	A177	US-2005/0144413 A1	06-30-2005	Kuo et al.
	A178	US-2003/0188032 A1	10-02-2003	Solomon et al.
	A179	US-2003/0065866 A1	04-03-2003	Spencer
	A180	US-2003/0028733 A1	02-06-2003	Tsunoda et al.

FOREIGN PATENT DOCUMENTS					
		Foreign Patent Document			
Examiner Initials*	Cite No.[1]	Country Code[3] – Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	T[6]
	B1	WO-2009/0100149 A1	08-01-2009	RAMBUS, INC.	

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C1	Altera Corp, et al., "Hybrid Memory Cube" specification, 2012, 122 pages.	
	C2	Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, and Vijayan Prabhakaran, "ResearchRemoving The Costs of Indirection in Flash-based SSDs with NamelessWrites," Jun. 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.	

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	9	of	11	Attorney Docket Number	39471-53126/US

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C3	C. Kuo, J. Hsieh and L. Chang, "Detecting Solid-State Disk Geometry for Write Pattern Optimization," 2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications, 2011, pp. 89-94.	
	C4	Cornwell, "Anatomy of a solid state drive," Oct. 17, 2012, ACMQUEUE, 13 pages.	
	C5	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," Apr. 10, 2010, MSST '10 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1-12.	
	C6	Garth Gibson, Greg Ganger, "Principles of Operation for Shingled Disk Devices," Canregie Mellon Parallel Data Laboratory, CMU-PDL-11-107, Apr. 2011, 9 pages.	
	C7	High-Speed NAND Flash: Design Considerations to Maximize Performance, Flash Memory Summit, Santa Clara, CA USA Aug. 11, 2009, , Robert Pierce, pp. 1-19.	
	C8	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, May 6-10, 2013, 5 pages.	
	C9	Intel Corp, PCI-SIG SR-IOV Primer--An Introduction to SR-IOV Technology,: 321211-002, Revision 2.5, Jan. 2011, 28 pages.	
	C10	J. Kim, D. Jung, J. Kim and J. Huh, "A methodology for extracting performance parameters in solid state disks (SSDs)," 2009 IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems, London, 2009, pp. 1-10.	
	C11	JEDEC Standard, JESD229, Wide IO, Dec. 2011, 74 pages.	
	C12	John D. Strunk, "Hybrid Aggregates: Combining SSDs and HDDs in a single storage pool," Dec. 15, 2012, ACM SIGOPS Operating Systems Review archive, vol. 46 Issue 3, Dec. 2012, pp. 50-56.	
	C13	Li-Pin Chang, "Hybrid Solid State Disks: Combining Heterogeneous NAND Flash in Large SSDs," National Chiao-Tung University, Taiwan, ASPDAC 2008, 26 pages.	
	C14	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.	
	C15	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, Feb. 23, 2010.	
	C16	NAND 201: An Update on the Continued Evolution of NAND Flash, Jim Cooke, Micron White Paper, Sep. 6, 2011, pp. 1-10.	
	C17	NVM Express, V. 1.2.1, 217 pages, Jun. 3, 2016.	
	C18	NVM Express, Version 1.0b, Jul. 12, 2011, pp. 1-126, published at http://www.nvmexpress.org/resources/ by the NVM Express Work Group.	

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	10	of	11	Attorney Docket Number	39471-53126/US

OTHER REFERENCES – NON-PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	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, page(s), volume-issue number(s), publisher, city and/or country where published	T[6]
	C19	Ohad, Rodeh, "IBM Research Report Defragmentation Mechanisms for Copy-on-Write File-systems," IBM white paper, Apr. 26, 2010, 10 pages, available at domino.watson.IBM.com/library/CyberDig.nsf/papers/298A0EF3C2CDB17B8525770- 70056B41F/$File/rj 10465.pdf.	
	C20	Open NAND Flash Interface (ONFI), specification, version 2.0, 174 pages, Feb. 27, 2008.	
	C21	Open NAND Flash Interface (ONFI), specification, version 3.1, 296 pages Sep. 19, 2012.	
	C22	Optimizing NAND Flash Performance, Flash Memory Summit, Santa Clara, CA USA Aug. 2008, Ryan Fisher, pp. 1-23.	
	C23	Ouyang et al., "SDF: Software-defined flash for web-scale internet storage systems," Computer Architecture News--ASPLOS '14, vol. 42 Issue 1, Mar. 2014, 14 pages.	
	C24	S Im and D. Shin, "Flash-Aware RAID Techniques for Dependable and High-Performance Flash Memory SSD," in IEEE Transactions on Computers, vol. 60, No. 1, pp. 80-92, Jan. 2011.	
	C25	Spansion SLC NAND Flash Memory for Embedded, data sheet, S34ML01G1, S34ML02G1, S34ML04G1, Sep. 6, 2012, pp. 1-73.	
	C26	Stan Park and Kai Shen, Department of Computer Science, University of Rochester, "FIOS: A Fair, Efficient Flash I/O Scheduler," Feb. 23, 2012, pp. 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.	
	C27	W. Lafi, D. Lattard and A. Jerraya, "High level modelling and performance evaluation of address mapping in NAND flash memory," 2009 16th IEEE International Conference on Electronics, Circuits and Systems--(ICECS 2009), Yasmine Hammamet, 2009, pp. 659-662.	
	C28	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, Apr. 14, 2014, 14 pages.	
	C29	Xiangyong Ouyangyz, David Nellansy, Robert Wipfely, David Flynny, Dhabaleswar K. Pandaz, "Beyond Block I/O Rethinking Traditional Storage Primitives," Aug. 20, 2011, published at http://www.sciweavers.org/read/beyond-block-i-o-rethinking-traditional-st- orage-primitives-327868, by Fusion IO and the Ohio State University.	
	C30	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, Feb. 7, 2012, pp. 1-16.	

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] Applicant's unique citation designation number (option). [2] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [3] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] 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)				Complete if Known	
				Application Number	17/877,881
				Filing Date	July 29, 2022
				First Named Inventor	Kuzmin, Andrey V.
				Applicant	Radian Memory Systems, Inc.
				Art Unit	2185
				Confirmation Number	3676
				Examiner Name	Not Yet Assigned
Sheet	11	of	11	Attorney Docket Number	39471-53126/US

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).

☒ This application relies, under 35 U.S.C. § 120, on the earlier filing date of prior application No. 16/570,922 filed September 13, 2019, and prior application No. 16/751,925 filed January 24, 2020, and the references cited therein **B1 and C1-C30** are hereby referenced, but are not required to be provided in this application under 37 CFR § 1.98(d).

☐ 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	/Michael W. Farn/	Date (YYYY-MM-DD)	2022-08-02
Name/Print	Michael W. Farn	Registration Number	41,015

Electronic Acknowledgement Receipt

EFS ID:	46324263
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn/Ally White
Filer Authorized By:	Michael Wayne Farn
Attorney Docket Number:	39471-53126/US
Receipt Date:	02-AUG-2022
Filing Date:	
Time Stamp:	14:55:54
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	Information Disclosure Statement (IDS) Form (SB08)	53126_US_IDS.pdf	292509 7fa4763d7c5f1500cb58bc89cbf01931516d859c	no	11

Warnings:

Information:	
This is not an USPTO supplied IDS fillable form	
Total Files Size (in bytes):	292509

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: ECOMM.AUTH/ECOMM.WTDW
Doc Description: Internet Communications Authorized/Internet Communications Authorization Withdrawn

PTO/SB/439 (11-15)

AUTHORIZATION FOR INTERNET COMMUNICATIONS IN A PATENT APPLICATION OR REQUEST TO WITHDRAW AUTHORIZATION FOR INTERNET COMMUNICATIONS

Application No.	Filed herewith
Filing Date	Filed herewith
First Named Inventor	Andrey V. Kuzmin
Art Unit	Not yet assigned
Examiner Name	Not yet assigned
Practitioner Docket No.	39471-53126/US

To: Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

I. To authorize permission for Internet Communications.

☑ Recognizing that Internet communications are not secure, I hereby authorize the USPTO to communicate with the undersigned and practitioners in accordance with 37 CFR 1.33 and 37 CFR 1.34 concerning any subject matter of this application via video conferencing, instant messaging, or electronic mail. I understand that a copy of these communications will be made of record in the application file. (MPEP 502.03)

II. To withdraw authorization for Internet Communications.

☐ The authorization given on _____, to the USPTO to communicate with the undersigned and any practitioner in accordance with 37 CFR 1.33 and 37 CFR 1.34 concerning any subject matter of this application via Internet communications such as video conferencing, instant messaging, or electronic mail is hereby withdrawn. I understand that the withdrawal is effective when approved rather than when received.

I am the

☐ applicant.

☑ attorney or agent of record. Registration number _____.

☐ attorney or agent acting under 37 CFR 1.34. Registration number _____.

/Michael W. Farn/ — Signature

July 29, 2022 — Date

Michael W. Farn — Typed or printed name

650.335.7823 — Telephone Number

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications. Juristic entities must be represented by a patent practitioner (see 37 CFR 1.31, which is applicable to any paper filed on or after September 16, 2012, that is presented on behalf of a juristic entity, regardless of application filing date). Submit multiple forms if more than one signature is required, see below*.

☑ * Total of one forms are submitted.

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	MULTI-ARRAY OPERATION SUPPORT AND RELATED DEVICES, SYSTEMS AND SOFTWARE

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 14/047193

filed on October 7, 2013.

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: Andrey V. Kuzmin Date (Optional) 11/5/13

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	MULTI-ARRAY OPERATION SUPPORT AND RELATED DEVICES, SYSTEMS AND SOFTWARE

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: James G. Wayda Date (Optional): 10/6/2013

Signature: James G. Wayda

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.

Doc Code: TRACK1.REQ
Document Description: TrackOne Request

PTO/AIA/424 (04-14)

CERTIFICATION AND REQUEST FOR PRIORITIZED EXAMINATION UNDER 37 CFR 1.102(e) (Page 1 of 1)

First Named Inventor:	Andrey V. Kuzmin	Nonprovisional Application Number (if known):	Not yet assigned
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management		

APPLICANT HEREBY CERTIFIES THE FOLLOWING AND REQUESTS PRIORITIZED EXAMINATION FOR THE ABOVE-IDENTIFIED APPLICATION.

1. The processing fee set forth in 37 CFR 1.17(i)(1) and the prioritized examination fee set forth in 37 CFR 1.17(c) have been filed with the request. The publication fee requirement is met because that fee, set forth in 37 CFR 1.18(d), is currently $0. The basic filing fee, search fee, and examination fee are filed with the request or have been already been paid. I understand that any required excess claims fees or application size fee must be paid for the application.

2. I understand that the application may not contain, or be amended to contain, more than four independent claims, more than thirty total claims, or any multiple dependent claims, and that any request for an extension of time will cause an outstanding Track I request to be dismissed.

3. The applicable box is checked below:

I. ☑ **Original Application (Track One) - Prioritized Examination under § 1.102(e)(1)**

i. (a) The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a). This certification and request is being filed with the utility application via EFS-Web.

---OR---

(b) The application is an original nonprovisional plant application filed under 35 U.S.C. 111(a). This certification and request is being filed with the plant application in paper.

ii. An executed inventor's oath or declaration under 37 CFR 1.63 or 37 CFR 1.64 for each inventor, or the application data sheet meeting the conditions specified in 37 CFR 1.53(f)(3)(i) is filed with the application.

II. ☐ **Request for Continued Examination - Prioritized Examination under § 1.102(e)(2)**

i. A request for continued examination has been filed with, or prior to, this form.
ii. If the application is a utility application, this certification and request is being filed via EFS-Web.
iii. The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a), or is a national stage entry under 35 U.S.C. 371.
iv. This certification and request is being filed prior to the mailing of a first Office action responsive to the request for continued examination.
v. No prior request for continued examination has been granted prioritized examination status under 37 CFR 1.102(e)(2).

Signature /Michael W. Farn/	Date July 29, 2022
Name (Print/Typed) Michael W. Farn	Practitioner Registration Number 41,015

Note: *This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required.**

☑ *Total of ONE forms are submitted.

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.

STORAGE SYSTEM WITH MULTIPLANE SEGMENTS AND COOPERATIVE FLASH MANAGEMENT

[0001] This document is a continuation of US Application No. 17/563481, filed on December 28, 2021, which in turn is a continuation of US Application No. 16/751,925, filed on January 24, 2020 (issued on April 26, 2022 as US Patent 11314636), which in turn is a continuation of US Application No. 16/570922, filed on September 13, 2019 (issued on November 30, 2021 as US Patent 11188457), which in turn is a continuation of US Application No. 15/621888, filed on June 13, 2017 (issued on October 15, 2019 as US Patent 10445229), which in turn is a continuation of US Application No. 15/346341, filed on November 8, 2016 (issued on July 18, 2017 as US Patent 9710377), which in turn is a continuation of US Application No. 14/951708, filed on November 25, 2015 (issued on December 13, 2016 as US Patent 9519578), which in turn is a continuation of US Application No. 14/047193, filed on October 7, 2013 (issued on January 5, 2016 as US Patent 9229854), which in turn is a continuation in-part of US Utility Patent Application No. 13/767723, filed on February 14, 2013 (issued on May 16, 2017 as US Patent 9652376), which in turn claims priority to US Provisional Patent Application No. 61/757464, filed on January 28, 2013. Each of the aforementioned patent applications is hereby incorporated by reference.

[0002] This disclosure relates to storage systems and, more specifically, to the architecture and operation of storage systems that utilize multi-die and/or multi-plane memory.

Background

[0003] NAND flash memory is one memory technology with granularity issues and latency issues that can constrain system performance. A conventional NAND flash memory device is typically characterized by programming and erasing latencies in the hundreds of microseconds, and little flexibility in the quantum of write read data and write data exchanged with memory. Thus, use of this type of memory is often characterized by substantial bus idle time and retrieval of excess data. Further, a program/erase ("P/E") asymmetry in NAND flash memory can further complicate latency, because data cannot be written to a previously-used memory location until an impacted erase unit ("EU") has been erased; in turn, an EU cannot be erased until all previously-written physical pages within that EU have been released, with still-active data being moved or scrapped as necessary. These and other maintenance operations, typically transparently managed by a memory controller, can result in delays occasioned by memory controller competition with host commands for memory bandwidth. Each of these issues is not necessarily unique to NAND flash memory and, to a lesser or greater extent, can affect different types of volatile and non-volatile memory.

[0004] To make more efficient use of memory, designers have increasingly turned to bus management techniques such as interleaving requests to multiple devices to improve bus utilization and to hide (or conceal) latency. Thus, for example, latency associated with a second memory access can be masked against the latency associated with a first, overlapping (parallel) memory access. With NAND flash memory in particular, multi-plane designs have emerged which permit accesses to data stored in parallel planes or arrays (typically two), but subject to certain addressing and/or timing restrictions. That is, because these devices typically use common input/output (IO) circuitry and address circuitry to handle requests, overlapping multi-plane access has traditionally been limited to situations where a common or base address is used for all planes.

[0005] Unfortunately, it is difficult to realize the benefits of multi-plane or multi-die architecture at an application or operating system level. That is to say, logical addresses typically arrive at the memory controller from the host as a stream of random accesses, and are sequentially assigned to first available physical space by the memory controller; as memory, particularly flash memory, is erased and recycled via read, write, wear leveling, garbage collection and other processes, sequential logical addresses become scattered throughout physical memory space. For multi-plane memory therefore, there is no practical mechanism for the host or memory controller to group related data in a manner geared for multi-plane access (i.e., there is no guarantee available physical addresses used for related data will be consistent with device multi-plane addressing restrictions). Thus, in practice, the benefits of multi-plane and multi-die capabilities tend to be restricted to multi-plane writes by a memory controller of coincidentally-related data uniquely for the purpose of bandwidth management, with read access by either the host and memory controller being inefficiently performed and largely restricted to independent, single page accesses. For both multi-die and multi-plane designs, as channel bandwidth continues to improve, the issues associated with a logical-to-physical translation later at the memory controller can be expected to create additional latency.

[0006] A need therefore exists for improvements that provide additional flexibility to systems and application designers. Ideally, such improvements would provide flexibility in storing and accessing multiple pages of data, for example, across multiple dies or planes. Applied to multi-plane memory, a need exists for techniques that permit effective use of multi-plane capabilities notwithstanding device IO and addressing restrictions. The present invention satisfies these needs and provides further, related advantages.

Brief Description Of The Drawings

[0007] FIG. 1A is an illustrative diagram showing one or more hosts 103a/103b, a memory controller 105 and memory 107. These elements can be implemented together as a memory system having multiple dies and/or multiple planes.

[0008] FIG. 1B is a block diagram showing techniques associated with multi-plane or multi-die memory requests that originate from the host.

[0009] FIG. 2A is an illustrative diagram of a host 201.

[0010] FIG. 2B is a block diagram showing host configuration and management of tables used to perform multi-plane or multi-die space assignment, reads, writes and associated management functions.

[0011] FIG. 3A is an illustrative diagram of a system having a memory controller 303, a host 305 and a memory 307.

[0012] FIG. 3B is a block diagram of a solid-state drive (SSD) having a memory controller and NAND flash memory 337.

[0013] FIG. 3C is a block diagram of a memory controller.

[0014] FIG. 3D is a block diagram of memory controller logic used in supporting cooperative functions.

[0015] FIG. 4 is a flow diagram that illustrates host tasks in a cooperative memory management scheme.

[0016] FIG. 5 is a flow diagram that illustrates host tasks in a cooperative memory management scheme.

[0017] FIG. 6 a diagram showing examples of information that can be kept by a memory controller for each of plural physical subdivisions of memory.

[0018] FIG. 7 shows a process associated with wear-aware writes.

[0019] FIG. 8 shows a process associated with a delegated copy operation.

[0020] FIG. 9 shows a process associated with an explicit erase operation (or relocation of data and recycling of previously used memory space).

[0021] FIG. 10A shows a process associated with host-owned defect management.

[0022] FIG. 10B shows a process associated with shared defect management.

[0023] FIG. 10C shows a process associated with memory controller-owned defect management.

[0024] FIG. 11A shows a process associated with host-owned and shared garbage collection.

[0025] FIG. 11B shows a process associated with memory controller-managed garbage collection.

[0026] FIG. 12A shows a process associated with host-owned and shared wear leveling.

[0027] FIG. 12B shows a process associated with memory controller-managed wear leveling.

[0028] 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).

[0029] FIG. 14 illustrates a block diagram of a storage server having multiple solid state drives (SSDs) and hard disk drives (HDDs).

[0030] FIG. 15 illustrates a block diagram of functions performed a storage server operating system.

[0031] FIG. 16 illustrates a block diagram of storage management tasks used by a storage server.

[0032] FIG. 17 illustrates how space reclamation is applied to mitigate P/E asymmetry.

[0033] FIG. 18 illustrates a garbage collection process.

[0034] FIG. 19 illustrates a process associated with wear leveling.

[0035] FIG. 20A shows one possible format for a multi-plane request from the host to memory controller; the request can be a write request having chained or fused single plane commands (Cmd0 and Cmd1) and respective data for writing into like-memory locations in planes of memory.

[0036] FIG. 20B shows another possible format for a multi-plane request from host to memory controller; unlike the command of FIG. 20A, the command (MP/MD) of FIG. 20B is a command that specifies a base address that will be applied to each plane used to service the request.

[0037] FIG. 20C shows a possible format for a multi-plane read request, having chained or fused single plane commands.

[0038] FIG. 20D shows a possible format for two single plane read requests from the memory controller to memory (separated by time "t2"), which is optionally performed in a response to a multi-plane command received by the memory controller from the host.

[0039] FIG. 21 shows a block diagram associated with a multi-plane read request from a host.

[0040] FIG. 22 shows a block diagram associated with a multi-plane write request from a host.

[0041] FIG. 23 shows a block diagram associated with a multi-plane deallocate (or page release) request from a host.

[0042] FIG. 24 shows a block diagram associated with a multi-plane erase request from a host.

[0043] FIG. 25 shows a block diagram associated with consolidation of multiple single planes of data to a multi-plane block of data, orchestrated by a host.

[0044] 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

[0045] This disclosure provides improvements in memory controller design, memory system

design, host design, and related management methods. A memory controller maintains information pertaining to memory geometry, including dies present and/or whether memory supports multi-plane access and/or other extended capabilities. As a non-limiting example, a memory controller can provide data to the host which identifies multi-plane capabilities of a memory die. Optionally, the memory controller and/or host maintain information that that tracks each physical unit of memory managed by the memory controller. This information permits a host to track physical space allocation across multiple dies or planes. Optionally, a logical-to-physical (L2P) translation layer can reside primarily or entirely on a host. A host can thus be provided with the capability to control physical address assignment to organize memory layout (e.g., to direct new writes of data based on application or system level needs); for example, certain data types (e.g., consecutive media data) can be organized and stored directly by the host in a manner consistent with multi-plane device addressing restrictions. Note that the improvements discussed herein are not necessarily limited to NAND flash memory dies or devices; as one none limiting example, these techniques can also be applied to other memory forms such as shingled magnetic drives. Also, while these teachings are described in the context of multi-array memory with cross-array addressing restrictions, these teachings can be applied more generally in the case of multiple plane memory storage, e.g., to memory devices lacking addressing restrictions, multiple planes, multiple dies or other specific memory characteristics discussed herein.

[0046] More specifically, in one implementation, a memory controller exports memory array/die/device geometry, and therefore has an option to educate one or more hosts on multi-array, multi-die or multi-plane capabilities of each managed memory array. This export can be effectuated, for example, in response to an inquiry command from a host operating system during a device discovery phase. Then, with insight into geometry, a host can organize address space in a manner geared for improved memory access, e.g., according to a policy aimed at IO optimization for managing multiple or linked pages. Again, the actual presence in a system of particular memory geometries (e.g., multiple planes) is not a prerequisite for the host to apply the disclosed techniques.

[0047] In applying these techniques, where allocation requests allow, the host assigns physical space for writes so as to facilitate both immediate write accesses and later multiple page read access, all with the benefit of the improved throughput. For example, later accesses can be performed in a manner unencumbered by memory controller address translation requirements; for multi-plane memory and shingled drives, data placement can be pre-planned in a manner consistent with any inter-plane addressing restrictions for multi-array accesses. As should therefore be apparent, these

techniques therefore provide operating systems and applications with significantly greater ability to utilize the full capabilities of certain (e.g., complex) memory architectures.

[0048] Note the symmetry benefit that differentiates this approach from a conventional approach where a flash translation layer (FTL) is relied on at the memory controller to remap logical writes to first-available physical space - when the host plans data access to be at known physical locations (i.e., across multiple devices, tracks, dies, planes, etc., where access can be overlapped), it manages data placement in a manner that it can later readily retrieve related information. This can be performed in a manner where the need for (and delay occasioned by) memory controller address translation or cross-array addressing restrictions (if present) become moot. The same planning and management benefits apply to many commands, for instance in the case of NAND flash, for writes, deallocate (page release), erase and potentially other commands. Importantly, while techniques presented below facilitate an architecture with a minimized FTL or no FTL at all in the memory controller, it is emphasized that reduced FTL practices are not required for all embodiments; that is, many of the techniques presented herein can also be used in implementations where the memory controller includes an FTL, such as in the context of a traditional NAND flash architecture.

[0049] In one embodiment, the memory controller/host interaction is augmented with specific commands directed to multi-array (e.g., multi-plane, die or track) IO capabilities. This command structure then permits the host to issue multi-array access commands via as a single operation rather than as a series of disassociated, single array commands. For example, such a command structure can be supported using an application program interface ("API") on the host side, and one or more extensions to a conventional command structure (e.g., NVMe for NAND flash memory) to support multi-plane operations. Utilizing physical addressing consistent with pertinent device addressing restrictions, this permits effective multi-array storage, retrieval, release and erasure notwithstanding any device IO or addressing restrictions.

[0050] This disclosure is organized as follows. A memory controller which maintains an information repository for running queries or otherwise to provide visibility to a host as to physical space management, and related system considerations will first be introduced. These capabilities help support direct use of physical addressing, referenced above. Second, some system options that leverage the capabilities of such a memory controller will then be discussed; for example, such a design optionally permits a host to direct certain memory management processes (e.g., garbage collection and wear leveling for NAND flash) in a manner that reduces competition of memory controller management

functions with host-originated access requests. Note that the described devices, systems and techniques are exemplary, e.g., the presented techniques can be applied to host design, memory device or die design, to a memory controller or memory system, and to other memory technologies as described. Note also that the various embodiments and FIGS. are not mutually-exclusive, i.e., teachings described in connection with one FIG. for one embodiment of a component (e.g., a memory controller) are generally applicable to other embodiments of that component (for example, described in connection with a different FIG.), or with reciprocal components (such as a host). Finally, implementation and use of one or more memory devices, efficient interleaving across devices, dies or planes, a command infrastructure and some optional system operations will then be discussed.

I. Introduction.

[0051] FIG. 1A provides a diagram 101 depicting interaction between one or more hosts 103a/103b, a memory controller 105 and memory 107. Note that two hosts are shown in the depicted system, one (103b) in dashed lines to indicate its optional presence. While the techniques herein can be implemented with any number of hosts, a typical implementation features a single host 103a, a memory controller 105 and memory 107 managed by the memory controller. These elements can be implemented together as a memory system in which the managed memory has multiple tracks, dies or planes (109a...109c) suitable for parallel (or pipelined) access. Note that the host can include any form of processor, for example, a personal computer or smartphone CPU, or the CPU of a storage manager (e.g., of a network ready storage appliance). A dashed line 110 indicates that in some environments, the host's interaction with memory is governed by external communications (e.g., with one or more clients, not separately shown in FIG. 1A); for example, a network-ready device can support external communications (e.g., over the Internet or a wide area network or "WAN") where memory accesses are responsive to requests or other interactions with the clients.

[0052] As mentioned previously, the memory controller 105 optionally stores data 111 representing geometry of managed memory, such as particulars regarding the depicted multiple tracks/dies/planes 109a...109c, such as number of tracks/dies/planes, volume of each such unit, addressing limitations and so forth. This information can also be made known *a priori* to the host, e.g., in an application-specific design such as where a host is preconfigured for operation with a specific memory device or devices. Whether already known by the host or stored by the memory controller 105 and sent to the host, the host 103a makes use of this information in building a table 112 which it uses to plan space assignment and the writing of data to the managed memory 108. As one relatively simple

example, a host interacting with two-plane NAND flash memory can allocate blocks of storage for individual pages of data or for multiple pages of related data. The host would first ascertain the number or amount of 2-page data storage blocks needed (that should be reserved), and then would configure priority/usage information (e.g., rules) for use during run-time to perform space assignment and to perform maintenance operations. Note that strictly speaking, a configuration step as just referenced is not strictly required, i.e., a host could simply reserve physical addresses for linked or related data, and could then assign addresses as needed until the reserved space is exhausted. Whether initially dead-reckoned or variably determined, run-time management can be used to help adjust space allocation. For example, for a multi-plane embodiment, if single-plane space were to run low, the host could change the criteria used to determine which data will be stored across dies or planes as related data. Alternatively, the host could perform garbage collection, page release and/or erase operations so as to recycle currently unused space that could then be used for single-plane writes. If addresses reserved for multi-plane writes were to run low, the host could perform garbage collection on physical addresses assigned to multi-plane space or could change the threshold at which *n*-plane space is assigned. Once again, these functions are merely options, e.g., a host could simply assign related data locations if any such locations are available. Reserved space can also be dynamically varied during run-time if desired. For example, the host could free physical addresses reserved for multi-plane space for single plane writes, or vice-versa, depending on dynamic conditions. Clearly many examples exist. Again, while this example is rooted in the use of multi-plane operations, the host can use similar management techniques in managing operations across tracks or dies or in otherwise managing memory space allocated to multiple pages; for example, a host could assign the same physical address (or address offset) in each of multiple pages for related data that may be read at the same time, though this is not required in all embodiments. FIG. 1A also highlights signaling link 113 to indicate that host-to-memory communications can be in a variety of forms including serial-differential, network-based, packet based, parallel, single-ended, point-to-point or other communications formats. Without limitation, in one embodiment, the host 103a and the memory controller 105 are in the form of respective integrated circuits (ICs), coupled by conductive-path serial, differential links, with communications optionally compliant with a version of NVMe.

[0053] Based on information stored in the table 112, the host plans data storage in a manner conducive to true, host-initiated multi-track, multi-page, multi-plane or multi-die reads, that is, such that successive read requests are issued by the host with the direct physical addresses for related data. This permits the host to plan for pipelined use of a memory bus in a manner that optimizes the retrieval of

related data needed by the host for a particular task. For example, a host needing to read linked data from multiple dies can issue a series of read requests to the memory controller that can immediately be passed on to the pertinent memory die. Even for embodiments where the memory controller cooperates in defect management (and performs very limited L2P translation for this purpose), the bypass of binary search trees and other complex translation functions permits streamlined interleaving across dies or devices. For multi-plane access (irrespective of whether also performed in the context of multi-die access), a single command or fused series of commands can be used by host and memory controller to access data that has been stored in consideration of device multi-array address restrictions; thus, one command, or a series of chained commands, can be issued to retrieved linked data from effectively "paired" physical addresses without offending native device IO restrictions or addressing restrictions. In at least one embodiment, the issuance of a multi-plane or multi-track request involves specifying a common, base (logical or physical) address used to access the related data in each respective die or plane. For architectures providing some flexibility in address restrictions (e.g., where certain NAND flash multi-plane architectures permit a limited "plane" address to differ from a specified EU/page base address), the request can specify an plane address offset to the base address. In one embodiment, a single command or fused series of commands from the host can be used to manage multi-die access.

[0054] For NAND flash memory, this structure facilitates expanded use of interleaving and/or multi-plane access for many different types of commands. For example, as physical pages of memory are released (or "deallocated") for data stored across dies or planes as a unit, the host can issue a request (e.g., one or multiple commands) that specifies release of a physical address (or offset) in each one of respective dies or planes. Memory devices supporting this capability can update page allocation tables and/or stored metadata to reflect this release, or can send messages (acks) to the host to track and update this information. In an embodiment where the memory controller tracks this information, the memory controller can then notify the host of the need to erase an erase unit (EU) in question (e.g., once page utilization has satisfied a predetermined garbage collection or erasure threshold or in response to a reclaimable space query from the host). Similarly, the host can issue multi-plane (or multi-die) erase operations, or can defragment data, essentially identifying separated (but commonly accessed) data and relocating that data to a striped (multi-plane or multi-die) configuration. The host can initiate these operations and then track the resulting impact on physical address assignment. Thus, this structure permits the host to issue multi-page writes without substantial latency at the memory controller for address translation, and it also permits the host to plan multi-page access in a manner

where related data storage is planned. Note that page release, erasure and other functions will be further discussed below in connection with FIGS. 23-25. Note once again that while many embodiments will be discussed below in the context of NAND flash memory, the techniques provided herein are equally applicable to other forms of memory, for example, in dealing with cross-track addressing restrictions for shingled solid state magnetic drives (e.g., HDDs). In a shingled magnetic drive system, a hard disk controller is constrained by having to write individual overlapping magnetic tracks together as a group (e.g., with a track grouping separated by a guard band), but by being able to read tracks individually. The techniques described further below, for example, relating to host-owned or cooperative physical space assignment, garbage collection, defect management, data relocation, cold data management and multi-page writes and reads, are equally applicable to such systems, and indeed, any technology characterized by a program/erase (P/E) asymmetry.

[0055] Returning to FIG. 1A, a variety of techniques can be utilized to extend operation to multiple hosts such as depicted optional host 103b. For example, without limitation, the system could assign dedicated logical and/or physical space allocated to each host, and rely on memory controller arbitration over competing host commands. Other techniques also exist. Extension of techniques disclosed herein to a multiple host environment is within the level of ordinary skill in the art, and will be discussed below sparingly.

[0056] FIG. 1B is a block diagram showing one method of management of multi-page memory requests that originate from the host. This method is generally identified by numeral 151 in FIG. 1B. As indicated by dashed-line block 153, a memory controller optionally stores configuration data for managed memory for provision to the host (i.e., to one or more hosts as appropriate). As noted, this information can include information on array size, number of devices, dies and planes supported, and other information such as manufacturer, use of multi-level cells, and so forth. Note that there is no requirement that managed memory devices or arrays have to be uniform, e.g., it is possible for a memory controller to manage two dies, one supporting only single plane operations and the other supporting two-plane operations. Per numerals 155 and 157, the host retrieves this information and optionally uses it in planning logical and/or physical space requirements.

[0057] During run-time, as the need to write data to memory arises, the host uses this information (and, optionally, stored priority information) to allocate space for each new write (159). However, rather than necessarily allocating this space sequentially, the host can allocate space in a manner where respective address locations are geared for later multi-plane access. Nearly any form of linked data can

be stored in these respective linked locations, for example contiguous media data, data and its corresponding error information, or indeed, any other form of related information that may need to be accessed in succession or together. In an embodiment where the host directly assigns physical addresses, the memory controller optionally has no FTL function (which exists on the host only), and thus the host in this embodiment assigns physical memory space and issues write (and read/deallocate/erase and/or other) requests in a manner that directly specifies physical address as a host-command field (i.e., a command-specified operand). This is represented by function block 163. In a variant of this embodiment, discussed further below, the memory controller can perform limited L2P mapping (e.g., it can detect memory defects, mark blocks as "bad" and assign a write in progress to a substitute free space without immediately alerting the host); the memory controller in such an implementation intercepts reads or other operations to "bad" address blocks only. Over time, as wear leveling is performed and bad block information is reported to the host and bad blocks avoided in future physical space assignments, the host understanding of valid physical space once again eliminates the need for temporary translation at the memory controller. In a second example, a host can issue commands to the memory controller for delegated copy of data to reclaim space. As will be discussed below, the memory controller relocates pages from a first data block (which is then reclaimable) to a second block (while preserving page order within relocated blocks); a limited translation mechanism can then be employed so as to eliminate any need for the host to update its address references. Note that even in such embodiments, the memory controller can use very simple translation mechanisms (e.g., detection of at most a small number of addresses, and an ensuing address modification step), and does not need an extensive search tree or complex processing to find the pertinent physical address.

[0058] For flash memory in particular, periodic maintenance such as wear leveling and garbage collection must typically be performed, per numeral 165. Performance of these operations on a host-owned, memory controller-owned, shared basis will be variously discussed below. The host issues multi-plane requests consistent with its role in maintenance. Finally, as mentioned, with judicious address assignment at the outset (either with original writes and/or as a result of periodic maintenance), the host is in a position to issue true multi-plane read requests, where related data is requested from respective planes in a manner that *a priori* is consistent with any native multi-plane address restrictions. These functions are variously represented by numerals 167 and 171 in FIG. 1B.

[0059] Note also that while the discussion of FIG. 1B was rooted in the context of multi-plane access, similar techniques can be applied more generally to multi-page (e.g., multi-die or multi-device access). In a multi-page configuration, direct physical addressing is applied by the host to substantially

simplify the address translation task at the memory controller, permitting much greater flexibility in interleaving requests and providing for improved bus utilization.

[0060] With some basic implementation options thus introduced, this disclosure will now discuss host operation in greater detail, with reference to FIGS. 2A and 2B. Note that a host can be implemented using hardware logic and/or software logic (e.g., as circuitry running instructions that control the operation of general purpose hardware). It is also possible to implement such logic as instructions stored on non-transitory machine-readable media that, when accessed and executed by a general purpose machine (e.g., such as a CPU) control that general purpose machine to implement the functions of such logic. Each of these operations is represented by the depicted FIGS.

[0061] FIG. 2A generally shows a host 201. The host optionally has a first interface 203a to interact with one or more clients, for example, over a WAN or local area network, and a second interface 203b to interact with one or more memory controllers (e.g., one for each drive if multiple drives are present). Whether or not interacting with a client, the host typically includes an operating system 205, which periodically needs to send data to storage and/or to retrieve that data. Request generation logic 207 is therefore called upon to generate a command to be sent to memory to request the storage or retrieval of data as appropriate. Typically, a request is formatted to specify one or more commands (e.g., a write command), an associated address, and optionally a data operand (e.g., in the case of a write request). The addressing format varies depending on implementation, but typically in a multi-plane context will include multiple fields which specify (for example), a drive identifier (if multiple memory controllers are present in the system), a device identifier, a die identifier, a plane identifier, and an address offset corresponding to a specific page number in a specific array. Typically, each field includes one or more bits, with a single bit being needed for example to specify one of two planes, and with many bits (e.g., 16 or more bits) being potentially needed to specify other fields (such as page or row address). Other address fields can also or instead be used depending on implementation, for example, bank number, level of a multilevel system, and so forth, and more than two planes can be used, depending on implementation.

[0062] To support multi-page and multi-plane capabilities, a host is typically configured in a manner that plans ahead for such multi-array allocation and support. To this effect, the host 201 includes logic 209 for initially configuring such access and for dynamically supporting such access (for example, by reallocating space, changing assignment rules/policies, and relocating data as necessary). As mentioned previously, examples of different maintenance schemes will be thoroughly discussed in the sections

below. In turn, logic 209 uses this information to both allocate space as appropriate (using table 211) and also to determine any sorting information that will be used during run-time to assign new writes to cross-die/cross-plane linked memory space as a matter of priority. Sort logic (212) can store in a register this sorting or priority information in the form of a series of related fields, for example, that index data type or a data usage characteristic and provide a field that assists with determining where new writes should be placed. For example, if the host detects the need to store multiple pages of media information, the sort logic could examine its priority information and determine that such media information is preferentially associated with multi-plane writes (across dies or planes as appropriate). Thus, the sort logic 212 assigns (logical or physical) address space in a manner that accords priority for this information as necessary for the host to assign physical space for cross-die or cross-plane storage. In one embodiment, multiple pages of data are sent as part of a request that can include commands simultaneously directed to multiple arrays, or a series of commands intended for single arrays, specifying related addresses, a common base address, an address offset, or a combination of these things. The formulated requests are then sent to a request queue 213; the request queue verifies that transmitted requests have been received and/or completed by the memory controller, and performs related error detection and correction. As an example, in some implementations, the request queue 213 can utilize an acknowledgement process to track the receipt and processing of commands. In another embodiment, tracked commands can be deleted from the queue when it is determined, for example, that error-free read data has been procured by the host from managed memory. Clearly, many examples are possible.

[0063] FIG. 2B provides some additional detail on one possible method for configuring host operations in a NAND flash system. FIG. 2B is generally divided into upper and lower halves (252/271) representing configuration and run-time functions, respectively. First, the host periodically (such as at system startup, at first operation or at another checkpoint or interval) retrieves information regarding memory system configuration, such as the number of devices, dies per device, LUN organization, whether multiple planes are supported, and so forth, as required for the particular implementation. This is represented by numeral 253 in FIG. 2B. As noted by optional function block 255, in one embodiment, this call for data is implemented by an NVMe-compatible Get Features command, directed to a memory controller. Note that it is possible to otherwise retrieve configuration data from the memory controller, or alternatively, memory geometry can be determined *a priori* and stored local to the host. The host then proceeds to plan space assignment, e.g., physical space assignment as represented by function block 257. If multiple dies are present (in a manner supporting parallel access)

or if multiple planes are present, as indicated by decision block 259, the host proceeds to identify system requirements (e.g., to plan memory organization as previously introduced) and arrives at a usage priority, per numerals 261 and 263. For any desired sorting order, the host can establish specific classes of access (e.g., single plane versus two plane, three plane, four plane, and so forth) as pertinent to the application, all per numerals 264 and 265. With memory space planned out, the host can store information for use during run-time as appropriate. Multi-modal operation can also be used, e.g., in which the memory controller and/or host uses a form or class of multi-plane addressing in first mode, and elects to not use this form or class of addressing in a second mode. If there is to be no orchestrated multi-plane or multi-die access, the host still performs certain configuration functions as appropriate; for example, per the discussion below, the host can still track and assign physical space and share in memory management to a selected degree. That is, whether accesses are uniquely single plane, multi-die or multi-plane, the host can still participate in one or more of defect management, data reclamation, data relocation, release of stale space, L2P table management and other physical space management functions. This helps reduce memory controller competition with host-initiated requests for memory bandwidth and provides for reduced latency and increased efficiency in memory access.

II. Architecture Of An Exemplary, Cooperative Flash Controller.

[0064] Optional features mentioned above are supported by structure where the memory controller also stores information specific to each of plural subdivisions of memory (e.g., for each physical memory unit managed by the host, in on-board registers reserved for this purpose). The memory controller makes data based on this stored information accessible to the host. For example, the memory controller can store a table with information organized by 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 (e.g., either as requested by the host or on an unsolicited basis). In one embodiment, this is achieved by storing per-physical-storage unit metadata for each unit managed by the memory controller. The memory controller advantageously includes logic that processes host queries and that provides data back to the host responsive to those queries. The data can either be provided 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.

[0065] These structures and techniques help reduce control bandwidth competition with host-initiated reads and writes and can help minimize write amplification. 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. 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 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). Further, as noted earlier, eliminating the latency associated with memory controller address translation (and/or defect management processes) permits the host to more effectively structure multi-die and multi-plane accesses to memory.

[0066] FIGS. 3A and 3B are used to introduce one full-featured design of a memory controller, that is, that supports host tracking of physical address space and multimodal configuration that supports different modes of host/controller allocation of responsibility for maintenance operations. The capabilities permit one circuit design to support many different implementations. For example, while the memory controller of FIGS. 3A and 3B can be adapted to either host-owned or memory controller-owned defect management (as described below), a memory controller supporting multi-plane, track, die or device writes and reads need not implement any or all of such responsibilities or functions. This is to say, the memory controller embodiments described below are examples only.

[0067] FIG. 3A illustrates one embodiment of a storage system 301 and associated memory controller 303, host 305 and memory 307. In the illustrated embodiment, the memory controller is structured to cooperate with the host 305 in the control of the memory 307. The memory controller 303 has at least one first interface 309 to exchange commands and data with the host. Although two such interfaces and corresponding transmission paths are seen in FIG. 3A, 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 303 to assist in memory management functions. As before, the commands and signaling protocol can be 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 307 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 303. 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. Similarly, in embodiments where the memory is a magnetic disk media (e.g., shingled drives), each subdivision can include a "band" of tracks (e.g., having a group of tracks that must be erased at-once).

[0068] The memory controller tracks physical subdivisions using metadata retained in internal storage 311. 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 313, the storage retains this information for each subdivision of the memory governed by the memory controller, in this case, for a physical subdivision of the memory 307. In embodiments where the memory 307 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 313). Note that for flash memory, each EU can also correspond to multiple pages, as indicated by numeral 315 (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 317 that is operable to send to a host either some or all of the "raw" metadata retained in the storage 311, or derived or processed information based that storage 311. 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 309, via a dedicated connection, or via a backplane connection.

[0069] Several configurations are also represented by the embodiment of FIG. 3A. First, as represented by numeral 321, the memory controller 303 can be designed as a standalone integrated circuit with the host 305 and the memory implemented as one or more discrete integrated circuits (e.g.,

the host in the form of a host processor). For example, dashed-line box 323 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. Second, as represented by dashed-line box 323, the memory controller 303 can instead be co-packaged or otherwise combined with the memory 307 as a storage subsystem. That is, it is also possible (as represented by dotted-line box 325) to aggregate the memory controller 303, the host 305 and the memory 307 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). In either implementation, per reference numeral 327, 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.

[0070] FIG. 3B shows a solid-state drive (SSD) having a memory controller 330 and NAND flash memory comprising one or more NAND flash memory devices 337. The flash memory controller 330 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 332, which receives commands from the host and fulfills those commands under the auspices of firmware. This logic and firmware will be further exemplified below, but for purposes of FIG. 3B, it is noted that this logic relies on internal memory 333 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 334. 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, or other networking formats. Each of these standards provide for commands to be sent from an initiator (such as the host) to a specific target (such as a memory device or memory controller). 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.

[0071] To perform input/output (IO) operations, controller firmware interacts with a low-level flash memory interface 335 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) 337, each coupled to the flash memory interface 335 via a multidrop channel. Each device 337 includes a multiplex interface 338 that connects to one or more co-packaged dies 339. Each die can have one or more planes 340, each with independent control and data registers 341, so that each die is capable of performing multiple IO commands simultaneously. These registers can be delegated complex commands (e.g., multi-operation 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. Note that a LUN can be striped across multiple planes (or dies) if desired for the particular implementation. For example, using the techniques of this disclosure, it is possible to write a LUN across planes by ensuring physical addressing of pages written to respective planes in a manner consistent with device multi-plane addressing limitations.

[0072] As mentioned, in this embodiment, 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 (which are designed to assist the host with management of the bandwidth-consuming functions). 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 335, the number of a device 337 connected to that channel, the number of a die 339 within that device 337, the number of a plane 340 located on the die 339, the location of a block within the die 339, and the location of a page within the block. Thus, physical address depending on embodiment can include a quadruple of channel, device, die and logic unit number (LUN).

[0073] FIG. 3C 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 345. As before, a host interface 349 is used to exchange commands and data with a host, and a flash interface 351 is used to issue commands to and exchange data with one or more flash memory devices (not shown in FIG. 3C). 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. 3C, 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.

[0074] The host interface 349 separates controller commands from any received packets (as necessary), and routes these commands over a command bus 352 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.

[0075] A memory operations logic block 357 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. In one embodiment, the memory operations logic block 357 performs control over these various phases of programming using buffers 359. Since a memory device can include multiple planes (as discussed above), the memory command processing logic block 359 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 355 (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).

[0076] Commands relating to the return of query information or the setting of an operating mode are sent to query/management logic block 361. Generally speaking, the host can request (a) return of raw metadata for the entire memory space managed by the memory controller, or for specific address ranges or EU ranges, or (b) other information derived from processing or filtering of that metadata by the memory controller. For example, 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. If pertinent to an embodiment supporting multi-plane writes and reads, logic on the host can automatically add "linked EU's" (i.e., EUs in counterpart planes to EU candidates) as part of such a list, or alternatively, can otherwise filter EUs having multi-plane data so as to not include them in such a list. Since the purpose of such a garbage collection 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 361 uses internal registers 363 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 having the described structure. 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 363.

[0077] A media icon 364 is depicted to indicate the optional use of software or firmware by the memory controller. The memory operations and query/management logic blocks 357 and 361 and registers 363 are all depicted within a dashed-line box 365 denoting that, in one implementation, these elements can reside on a single die (e.g., a single processor or coprocessor); in such an embodiment, the media icon 364 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 (365) 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.

[0078] The metadata storage 355 is indicated as separate from dashed-line box 365, 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 through power cycling. In another embodiment, the metadata storage can exist in SRAM (e.g., internal to optional die 365), 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 366, the metadata storage can be read directly by the host, i.e., via a form of commands for direct memory access (DMA). 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.

[0079] FIG. 3D shows a block diagram 367 of a cooperative memory controller in expanded detail; more particularly, FIG. 3D is used to show how logic functions can be implemented using hardware and firmware logic 369. This logic 369 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 is the substantial reduction or omission of an FTL layer or its equivalent, as mentioned, this is not required for all embodiments. In another embodiment, the memory controller implements comprehensive cooperative functions that help reduce the need for complex L2P translation processes, as referenced above. In a cooperative or shared management scheme, the host can retrieve per-memory unit physical address status and, with this information, directly address specified physical locations – as mentioned, this substantially reduces or obviates the need for complex translation processes resident at the memory controller.

[0080] For the embodiment of FIG. 3D, logic 369 is seen to have a number of basic function blocks, including interface logic 371 to interact with the host using packetized commands and responses, logic 373 used for local metadata management, command processing logic 377 used for query processing and other management functions, and IO scheduling logic 381 used to manage memory transactions (e.g., read, program, and erase operations). As noted, even in an embodiment where it is desired to

substantially reduce the translation layer, a memory controller can still optionally implement some address translation, for example, for defect remapping, transparent data relocation and other functions. This will be described below. The metadata management logic 373 maintains locally-stored information in the form of metadata 378, as mentioned, for each physical unit of memory (e.g., each EU) 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 alluded to earlier, 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 378 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.

[0081] 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 transparently remap specific data over to other memory space. This operation could be invoked in implementations, for example, where direct physical addressing is used, but where the memory controller performs defect management or is tasked with delegated data copy (transfer) to free up 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 378. The limited address translation scheme is then used such that future reads specifying the remapped EU are intercepted by the memory controller logic using locally-stored metadata 378 and redirected to the proper physical location. Note that such a limited FTL can be made quite manageable in overhead if an optional practice is used to preserve page address from the old EU when writing data into a destination EU. 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). As will be discussed elsewhere, a limited FTL even in this situation is optionally used, e.g., the memory controller could be tasked with performing some space assignment tasks, but immediately reports back to the host P2L information for the assigned space, thereby enabling the host to update its addressing scheme to thereafter use direct physical addressing by the host (i.e., future requests are directly addressed to the remapped space). Other alternatives are also possible.

[0082] The command processing logic 377 receives commands from the host directed to general configuration of memory operations and for queries. Generally speaking, this logic manages and updates the metadata 378 and runs queries on the metadata, returning information to the host as appropriate via the host interface logic 371. 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 378 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.

[0083] 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 possesses dedicated logic infrastructure to perform WL, GC, DM and IM specific functions (379, 380, 381 and 382, 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 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 and/or reclamation, 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 translation overhead, provides for faster, more consistent response, and facilitates multiple drive storage aggregates based on solid state drives (SSDs) as well as mixed or heterogeneous systems that combine SSDs with other memory types.

[0084] To assist with host scheduling of 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 NAND flash embodiment of FIG. 3D, 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 377 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 address space (i.e., "free" or "virgin" space that is already erased and ready to receive write data), candidates for erase (i.e., space that did have at least some data no longer in use, that is, consisting of at least some released space and possibly some free/virgin space but that can be erased to thereby provide a contiguous block of free space), candidates for data relocation and garbage collection (e.g., space that is mostly released with only a small portion still used to service data read requests that could be reclaimed once the data is moved or consolidated elsewhere), candidates for cold data relocation (e.g., moving low turnover data to facilitate wear leveling), or more general functions.

[0085] For both embodiments that use wear-aware writes as well as those that do not, the memory controller can include wear leveling logic 379. 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 367 can generate alerts when predetermined wear or data age 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 383 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 384), 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 dies, planes or drives.

[0086] Copy-on-write processes 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. 3D therefore uses garbage collection logic 380 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 (in use) data, or is released (has pages not in active use but that remains unerased). Garbage collection involves accounting for released pages, for example, using release accounting logic 385 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 data 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 or "in use" 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, erased or otherwise not in use. For example, if most pages in a given EU were 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.

[0087] 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 385. 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 385 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 385 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 386 being delegated specific tasks by the host (such as the identification of EUs to the host for relocation of data, or delegated 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 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 369 processes this command and, once the command is completed, returns the freed EU to a pool of available EUs for future data allocation.

[0088] Write and erase errors discovered by the flash memory controller are handled by defect management flash management logic 381. Pages found defective due to write error are remapped by defect remapping logic 387, with the subject write operation retried transparent to the host. The original page in error is marked as defective or "bad" and added to a bad block list or map to avoid further use of associated physical pages. Unit relocation logic 388 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.

[0089] While flash memory typically provides strong error detection and correction (EDC), the memory controller may also provide onboard data integrity management logic 382. 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 390 using mechanisms similar to wear leveling and garbage-collection relocation processes described above.

[0090] 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 381 detects completion of the command. Pertinent information is added by metadata management logic 373 to the stored metadata 378 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 378 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.

[0091] Note that, depending on embodiment, the 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 386 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. 3D presents a single memory controller architecture adaptable to multiple, different implementations, but this is not required for all embodiments.

[0092] 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.

[0093] As discussed above, a cooperative 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 dynamically multimodal, i.e., supported in a single memory controller design and configured *in situ* depending on policy or other system parameters for memory controller-owned processing, host-owned processing or cooperative processing. 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 system with this memory controller might not implement this type of cooperation between host and memory system – in such a system, the memory controller can be programmed (e.g., using a command register or mode register) to turn this management feature 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.

[0094] When configured to provide cooperative management 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.

[0095] 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 of FIG. 4 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 of SCSI, for example, using a vendor-specific extension or a vendor specific NVMe administrative command (or similar command for other protocols). 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.

[0096] As an example of management action, a host can initiate an explicit erase operation that specifies an EU or equivalent 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 or a multi-die 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 such a configuration, commands from the host can be interleaved such that as read and write operations are performed in one die or 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 devices or drives are connected point-to-point with a host.

[0097] 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 response 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 (e.g., for an address range or

for all management memory). 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).

[0098] 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 tracked, e.g., with those memory controller or host calculating time for the EU. 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), that are valid (with actively-used data), or that are 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 EU also tracked as part of the metadata.

[0099] 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 Management Tasks.

[0100] Flash management tasks discussed below can be grouped according to three primary categories: Defect Management, Wear Leveling Management, and Garbage Collection Management. Note again that these management tasks are not necessarily unique to flash memory and that techniques disclosed herein can be applied to other memory forms as appropriate. A memory controller with hardware to support all forms of memory controller-host interaction described in this disclosure has logic that is multimodal or can be customized support management schemes specific to the desired implementation. That is, optionally, 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 structures 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 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 initiated by the host, i.e., such that tasks are schedule so that there is no competition.

1. Capabilities.

a. Wear-Aware Writes.

[0101] 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 (free) 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, space reclamation and/or erase operations at a time convenient to the host.

[0102] 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 via 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" as mentioned earlier 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 embodiments where a memory controller is to perform its own defect management (i.e., marking the intended physical address as "bad" with data relocation) or is to otherwise perform delegated relocation or copy without immediately reporting remapped (L2P or P2L) information back to the host, L2P translation is advantageously restricted to a few, notable situations. For a NVMe compatible implementation, these processes can once again be implemented with a vendor specific command or API. 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.

[0103] 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 (e.g., reserving physical address space for multi-page writes and reads can inhibit symmetric wear). 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.

b. Page Underutilization And Stale Space Reclamation.

[0104] To address page utilization issues, including the presence of stale (unused) data, the host in one embodiment can advantageously enquire at any time as to the extent of page utilization and/or aged data present in a flash device or SSD. A similar analogy exists for enquiries related to tracks in a band of shingled magnetic media. First, note that updates of memory content in such memory are 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 this operation can be performed directly by the host, i.e., with the host reading data from a first physical location, modifying that data, writing the modified data directly to a second physical location based on a listing of available space, releasing the first physical location and updating metadata for the second physical location. Over time, EUs can acquire greater and greater numbers of released pages, and so, experience low page utilization in terms of a decreased population of still-referenced pages. 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 do 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 dies or 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 plane, die or 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.

[0105] As with the various schemes presented above, depending on implementation, 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 or SCSI 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 satisfies 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, sets registers and initiates routines as necessary to synchronously or asynchronously provide the requested data.

[0106] 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 a 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 dies, planes or drives, a host can choose the most suitable destination for data relocation, potentially including another die, plane or 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 drive 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.

[0107] 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 or band determined to have one or more defective memory cells. Alternatively, such an operation could be performed if page utilization for (e.g.) 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. Also, in a multi-die (or multi-plane) context, such an operation can be initiated by the host for the purpose of balancing released/free space across dies or planes, i.e., to maintain a reserve of space in each array so as to support ensuing multi-die or multi-plane writes. 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 to the memory controller 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 frees 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.

[0108] 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.

[0109] 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 not 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 first from the controller to the host and then back from the host from the controller.

ii. Explicit Erase.

[0110] 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 dies, planes or drives, a host can hide (stack) an explicit erase to one of these structures behind data operations to another. Note that in a multidrop or other shared command path system, such an embodiment can interleave commands across multiple dies or SSDs, such that explicit erase operations are performed in one or more dies or SSDs while data is exchanged in the performance of a write or read command in a different die or SSD (e.g., with data being exchanged over a shared data path).

[0111] 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, data relocation command (e.g., delegated copy, referenced above) or a space reclamation command can be performed as part of a garbage collection operation.

[0112] 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 (*0-n*) for which 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 page utilization below a specific threshold (e.g., 0-20% of that EU is in active use,

or #pages released >60%, representing a possible space reclamation candidate). 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 from an EU selected for an erase to a specified target, with all constituent pages of that EU then being released or otherwise unused. Then, per numeral 915, the host issues an explicit erase command. Note that for embodiments that use reserved physical space that are preallocated for multi-page writes, the host can include logic that ensures consistency for erased units; that is, if one EU associated with multi-page writes is erased, it may be desired to erase counterpart EUs (e.g., in adjacent planes for multi-plane writes), and host software can be configured to automatically check related or linked space. The memory controller can manage the erase operation (e.g., as a sequence of P/V cycles applied to a substrate well of affected memory) and return a code to the host confirming successful erase for each affected EU (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).

[0113] 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.

[0114] To facilitate cooperative memory 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.

[0115] A cooperative memory controller and cooperative host can generally support one or more of 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.

[0116] 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. 3C can be used for

this purpose. Note that in each mode or scheme discussed, the memory controller can still 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.

[0117] For host-owned defect management, units marked unusable by the chip or device manufacturer are identified during first time device initialization. Host software is then expected to query the memory controller or the memory for this defect list using the method provided by the controller.

[0118] 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.

[0119] Subsequent write errors that occur can be incrementally reported back to the host software without further memory controller involvement in directly managing those defects. 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.

[0120] 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.

[0121] 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 die, 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.

[0122] In addition to prioritization, as indicated earlier, it can be desired to factor whether affected memory is used for multi-page writes and reads. For example, if one EU (storing at least some multi-page data) is determined to be a candidate for data relocation, then perhaps any "paired" EUs in

adjacent planes should be subject to the same scrutiny; the host optionally includes software to manage this task. Alternatively, if a separate pool or class of multi-page space is to be kept on-hand, a separate "spare capacity" process can be used for each such pool or class. Specifically, for multi-plane or multi-die operations, a process can be invoked to ensure that each plane/die has at least a minimum pool of free space, so as to avoid inhibiting writes to multi-page space.

ii. Shared Defect Management.

[0123] 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 detect defects in accessed memory and can maintain a defect list across power failures, reporting this information to the host as needed. Upon detection of a defect, the memory controller reports this information to the host and optionally marks physical blocks associated with the error as bad. If the memory controller is operated in a cooperative wear leveling mode, then depending on embodiment, the memory controller will not suggest a defective area for allocation.

[0124] 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 (and to update local information that tracks defects). 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.

[0125] 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.

[0126] 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, 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, 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.

[0127] 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.

[0128] Optionally, 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 can be 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.

[0129] 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 based on correspondence of a read address with physical address of the defective "bad" block; the memory

controller transparently obtains the remapped physical 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.

[0130] When spare area reserved for defect remapping is exhausted (or falls below a predetermined threshold), 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.

[0131] Note that if spare capacity is completely exhausted, this even can threaten and potentially halt normal device operations. 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.

[0132] 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, and is not unique to flash memory. Some 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%.

[0133] 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.

[0134] 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 candidate unit selection, and relocation of valid (active) data, and also typically therefore assumes responsibility for released space accounting. 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 and to apply any garbage collection candidate identification logic to ensure the desired amount of free units or available capacity; the host (or optionally the memory controller) performs released space accounting, to track page utilization for each erase unit. Such 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 once the last remaining unreleased 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.

[0135] 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 plane, die or device in general. This latter information could be used by a host, for example, in selecting a specific one of multiple dies, devices or 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.

[0136] 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/free 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 each time it detects the specified condition, notifying host software of the condition and delivering condition specific information (including any address range in question).

[0137] 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 locally stored information (e.g., metadata, 1111).

[0138] Once again, if multi-page (e.g., multi-plane, multi-die or multi-track) reads and writes are to be supported, the garbage collection tasks can be performed for each "linked" or "paired" array, e.g., to ensure consistency across dies or planes in a manner that ensures sufficient available capacity.

iii. Memory Controller-Owned Garbage Collection.

[0139] Memory controller-owned garbage collection typically requests some type of L2P tracking local to the memory controller and it therefore entails different considerations than 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.

[0140] 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 (including avoiding L2P overhead), will typically elect the clear advantages presented by the host-owned and shared configurations discussed above.

c. Wear leveling Management.

[0141] 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.

[0142] 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.

[0143] 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 "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. Once again, if reserved space is to be used to support multi-page writes and reads of different page sizes, it can be advantageous to support either a separate wear leveling process for each class of reserved space (e.g., for 2-page writes), or alternatively, to periodically change the physical

addresses used for these operations (e.g., so as to periodically swap EUs for single plane writes with EUs paired for multi-page writes, thereby "precessing" space assignment for each reserved class through available physical space).

[0144] 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 use 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); it can also issue a delegated copy command as discussed above (1217), for example, for relocations within or across planes or dies within a given SSD. 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.

[0145] 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 minimized, greatly reducing the possibility of memory controller task competition with host requests.

ii. Controller-Owned Wear Leveling.

[0146] 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 updates its addressing information to implement 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.

[0147] 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.

[0148] 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.

[0149] 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, the physical address allocation 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).

[0150] In another embodiment, the memory controller further cooperates with the host by keeping track of backward address translations. 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, it can request the controller to return, for each block range suggested in the query response, 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. 3C.

[0151] 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.

[0152] 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.

[0153] Such a process is generally designated by reference numeral 1301 in FIG. 13. In this 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, physical pages, bands, tracks, etc., 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 updates/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 (or uses greatly simplified mechanisms with substantially simplified translation). Thus, for delegated data assignment (or the other mentioned processes), the memory controller can support resident space assignment, while at the same time providing the benefits of access to specific physical addresses dictated by the host.

[0154] 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 can update its address references 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.

[0155] Those skilled in the art will recognize that other combinations of sole/shared management schemes other than those shown above are also possible.

[0156] With basic implementation of a cooperative 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.

[0157] FIGS. 14-19 are used to illustrate functions introduced above in an exemplary storage system. For purposes of this discussion, it will be assumed that a storage system or subsystem includes plural storage modules or drives, one or more of which is based on NAND flash memory solid-state devices (i.e., integrated circuits). Again, these techniques may be extended to systems not rooted in NAND flash memory. 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.

[0158] 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 can be 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. These same techniques can also be applied to other P/E asymmetric technologies. 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.

[0159] FIG. 14 illustrates an exemplary system 1401. In particular, a storage server 1403 receives requests to read 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 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.

[0160] 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 (e.g., within a die or across dies of a drive) and potentially across 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).

[0161] 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.

[0162] The storage server in this particular example 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.

[0163] 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 dies or 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 moved 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.

[0164] 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) is 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. Space Reclamation.

[0165] Host-assisted space reclamation uses metadata from a memory controller to reclaim and consolidate space owing to P/E asymmetry, 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 a physical subdivision 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 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 physical subdivision units and its constituent elements (e.g., pages) are only marked as free once the erasure has successfully completed and confirmed by return of a completion code by the particular memory controller.

[0166] FIG. 17 generally shows two methods, including a method of automatic erasure each time a page is released (this method is represented by solid lines only), and a method of batched space reclamation appropriate for address ranges, specified "candidate" subdivisions, or for a drive as a whole. The latter method will be described first, utilizing the example where flash EUs represent the tracked physical subdivisions. 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 and/or free (i.e., not yet written to); as indicated by steps 1705, 1707 and 1711, the method effectively 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, but has been used to store data, at step 1713 the page is marked as released rather than free. The bitmap is then checked (1715) to determine whether it is the last page of that EU; for EUs that are determined to be previously used but empty, i.e., consisting only of pages that are released or of pages that are released or are virgin (not written to since the last erase of the EU), the method continues at 1719, with an explicit erase operation being called and the drive summary information being updated (1721); otherwise, the method ends for that EU at step 1723, and the method then proceeds to the next EU (1703). 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.

[0167] 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 emptied (i.e., consisting only of released or released and free pages) (1715); if the EU is not fully emptied, the method ends. If the EU is fully emptied, 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.

[0168] Note once again that the operation of FIG. 17 can be implemented as a host query, for example, using a suitably-designed API. The host can query the memory controller as to reclaimable space, meaning EUs (in a flash embodiment) where at least one page has been written to that EU but where there are no longer any pages that are still "valid." The host optionally specifies addresses (e.g., a

device, a range of EUs, etc.) for which it wishes to ascertain reclaimable space, and the memory controller responds with a list of EUs that can be erased and used in new space assignment. In one embodiment, the host can query the memory controller to independently determine each of free space (erased space), reclaimable space, space with low page utilization, and so forth, each using a separate API. Also, blocks of memory (e.g., EUs) can be ranked by the memory controller according to any desired criteria; for example, it is desirable for some implementations to sort space by a per-block metric of released space to overall space (or equivalently, released to free space) such that blocks with small proportions of free space or contiguous free space are reclaimed first. Other options are also possible.

B. Garbage Collection.

[0169] In one embodiment, garbage collection is performed to reclaim space of subdivisions where the amount of released space relative to in-use or free space exceeds a predetermined threshold. Garbage collection is achieved by copying valid data (that is, unreleased pages) to another memory location to fully empty source locations in a manner such that the original space is reclaimable. 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 a block or subdivision (e.g., a flash EU) has been relocated, all portions of a partially used subdivision are thereby released or are otherwise free, and the subdivision 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 instead be initiated when this ratio exceeds 1:5 (e.g., at least twenty percent of valid space in an EU has been released), depending on dynamic system conditions. Such dynamic control can be implemented by policy module (1619 from FIG. 16). Many such examples are possible.

[0170] 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.

Note that the discussed method can be applied more generally to any form of space consolidation or reclamation (localized or otherwise), as well as potentially other management functions. The method starts at step 1803. While such an alert remains in effect (1805), the host software identifies a specific physical subdivision under consideration (e.g., from a prioritized list of EUs) and then loops from one subdivision to another (1808) through a range of the subdivisions. 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 subdivision in that list. Once the condition is cleared, the process stops at step 1807.

[0171] 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 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-mostly or other read-write characteristics). In addition, the use of reserved space for multi-page writes (or the fact that resident data is paired with a subdivision on another die or plane) can be factored into this screening process. If a candidate is not appropriate, the host software then proceeds to examine the next candidate in the list, per numeral 1815. If a 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 or selected memory space. The process of FIG. 17 can then be called to fully erase the pertinent subdivision, e.g., flash 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.

[0172] 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 memory class, plane, die, device, 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 becomes 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. Also, as noted previously, if an embodiment supports multi-page, multi-plane, multi-die or multi-track writes, this also can be factored into the process.

[0173] 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 significantly less worn out than to mean 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.

[0174] 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.

[0175] 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).

[0176] 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 can optionally as part of this process determine whether the candidate is associated with multipage data or a preallocated class of multi-page address space. 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 plane, die or 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.

[0177] 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.

[0178] 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 311 from FIG. 3A) 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. Commanding Interleaved And Multi-Array Operations.

[0179] An exemplary flash-based SSD can have 1 to 16 integrated circuits (ICs), each composed of 1 to 16 dies. Each die on each IC has independent IO circuitry that can be concurrently operated with other dies that are present, i.e., subject to any bus constraints (e.g., the rate at which commands can be directed to respective dies over a shared bus). Each die can generally be operated in parallel for the purpose of interleaving. In some embodiments, each die includes a cache for queuing commands for pipelined operation. The memory controller examples described above generally enable the host (and any related software, OS or application level) to fully utilize the chip/die level parallelization potential by exporting the physical geometry of managed memory, including device composition. This can be performed for example by a synchronous command from the host, such as using an extension of the NVMe Identify Array command. The array geometry data, made available to the host at a configuration phase, lets the host perform space allocation and IO scheduling so as to attain the maximum degree of parallelization and to optimize IO bandwidth.

[0180] The use of direct physical addressing by the host (i.e., based on host-assigned physical addresses) helps minimize or eliminate the need for an FTL at the memory controller, and thus affords greater control over interleaving/pipelining where a memory controller manages a number of dies. With this capability, and with support for host-initiated maintenance operations (e.g., garbage collection,

erase, maintenance operations, defect management and so forth), the host is able to ensure that adequate free space is available in each memory array and to generally streamline operations. In one example, a host can stripe related data across memory dies and, because the host performs physical address assignment, can issue commands after L2P translation has already been performed. Thus, the memory controller is not encumbered by latencies associated with L2P translation and, as mentioned, this structure helps minimize memory controller maintenance functions which could otherwise compete with host-requested read and write operations. The host can also better interleave different command types across dies (read versus write or erase, or deallocate, for example), as timing is not encumbered by controller-resident translation latencies.

[0181] Referring briefly back to FIG. 2B (numeral 265), the host has the option of establishing (pre-allocating) classes of reserved space, for example, for 1-page single-plane access, 2-page multi-plane access, 3-page multi-plane access, and so forth. In a general multi-page context (e.g., with a single die address space or multiple dies), the host can simply track L2P addressing and assign whatever physical address space is available in different respective dies and use the L2P table to fetch related data from each respective die, according to a separate (independent) physical address for each die; however, the host can also choose to assign physical addresses such that related data is stored at the same address or a similar address in each die. The selection of a common or similar address for related data stored in different arrays is less important in the multi-die context, but is typically important in the multi-plane context (e.g., where devices are often characterized by native addressing restrictions for multi-plane access).

[0182] A conventional multi-plane NAND flash die might have multiple data registers, one per plane, with a single address register shared by all constituent planes. In the two or four plane case, prevalent in the available NAND flash dies, addressed EUs must be at respective addresses that differ by, e.g., a least significant bit, since the shared address register typically allows for a single bit address extension only (i.e., for the adjacent plane), with addressed pages (for programming) using the page address within each EU. These addressing limitations of some legacy designs can constrain the ability to use multi-plane capabilities for reasons already mentioned. Note that similar limitations also apply to magnetic shingle drives and more generically, to multi-page management and access, and that, once again, multi-plane NAND flash memory are used as the non-limiting discussion example.

[0183] To address this, the host of this disclosure uses insight into physical space allocation within each plane of multi-plane memory and assigns related data the same (or highly similar) addresses in

each plane. This same process of address assignment can also optionally be performed across dies. The ability to choose physical addresses permits the host to effectively direct multi-plane writes and reads as long as the assigned addresses (for related, respective data to be obtained from the planes) are consistent with any device addressing limitations for multiple plane use. That is, contrary to chip and die level parallelism where concurrently running operations can generally be completely independent, multi-plane operations are advantageously constrained to prearranged locations that are consistent with any pertinent device addressing limitations. Host-selection of physical addressing thus permits the host to direct a "true" multi-plane write or read operation without being affected by legacy multi-plane device addressing restrictions. That is, for example, the host can ensure that a write of related data across planes (a) receives the same EU address for each plane (i.e., for the write of respective pages of data), and (b) the same address offset within each EU. The host can assign such writes if the same physical-unit address in respective planes is available, but as mentioned earlier, in one embodiment, the host prearranges classes of space for multi-plane writes. For example, the host can "reserve" a specific range of like-EU addresses in each plane for multi-plane writes and then assign related data EU addresses for respective planes and page address offsets within these EUs, as consistent with device addressing restrictions. Harkening back to numeral 265 of FIG. 2B, the host optionally has (for an embodiment having more than two planes) the option of reserving physical space for respective classes of writes where 1, 2, 3, 4 or another number of planes will be written to in parallel. The host determines operating system or application needs (or uses other criteria to prearrange space) and then it assigns space as needed. As noted elsewhere in this description, and as denoted by numerals 273, 275, 277, 278 and 279 of FIG. 2B, the host factors the existence of such prearranged space reservations into defect management, data relocation (e.g., hot/cold space management and garbage collection), page release and erase, space reclamation and other maintenance operations as needed; for example, if a hypothetical host employs three classes of prearranged space, the host might use three (respective) separate processes or function calls to ensure persistent availability of a threshold amount of "free" (e.g., erased) space, one for each class. The host also updates L2P mapping on a dynamic basis, and (in one embodiment) can dynamically reallocate the amount of space reserved for each class according to need. If no space is available, or due to other factors (e.g., OS preference for single plane writes), the host can elect to discontinue or suspend multi-plane writes or other multi-plane capabilities for any prearranged class of memory. As a non-limiting example of this principle, a host might determine that multi-plane addressing needs represent a small set of data, or relate to low priority data, and so may discontinue multi-plane writes.

[0184] Note that as mentioned earlier the controller support may be selective according to implementation: for instance, only multi-plane writes may be supported, or the controller may provide a full range of support including multi-plane read, deallocate, erase and potentially other operations as well. As also indicated just above, a host preferably reserves the right to issue single plane requests (i.e., 1 page only) to multi-plane memory, to enhance flexibility in memory transaction granularity. To this effect, a host interacting with multi-plane memory will typically establish at least two classes of access (corresponding to the classes of reserved space) and will use a dedicated command or addressing structure to identify each (single or multi-) class of write, read or other command to an affected memory die. That is, an addressed memory die typically examines a field in the command structure to determine whether a command is for a single page or EU in a single plane, or whether multiple planes are affected. In one implementation, a memory controller use a custom set of commands to effectuate multi-plane operations. However, such operations can also be effectuated using the existing NVMe fused IO operations framework. This permits the host to issue multi-plane IO requests as what is essentially a single "group" request rather than as a series of independent requests that could have a multi-plane relationship lost in connection with scheduling in the OS or host IO subsystem.

[0185] To begin, array geometry is first discovered. These steps, and the configuration of multi-plane access have already been outlined in connection with FIG. 2B. These steps can optionally be performed in a manner that is NVMe-compliant. During a device discovery phase, the host first queries the memory controller using the vendor-specific Identify Array extension to the NVMe Identify command and, based on the response, notes whether a given die is multi-plane. Typically, a device (IC) with multiple dies will have the same configuration for each die. The data structure set forth in Table 1, below, can be used for this purpose.

Table 1: Identify Array Data Structure

Bytes	**Description**
0:0	**Channels**: This field specifies the number of channels present in the array.
1:1	**Targets**: This field specifies the number of targets per channel present in the array.
2:2	**LUNs**: This field specifies the number of logical units (LUNs) per target present in the array.
3:3	**Planes:** This field specifies the number of planes per LUN. If the NAND chips that the device is equipped with do not support multi-plane operations, this field is set to 1.
5:4	**Erase Units**: This field specifies the total number of Erases (EUs) per LUN present in the array.

Bytes	Description
6:6	**Pages:** This field specifies the total number of pages per Erase Unit present in the array.
8:7	**Page size**: This field specifies the page size, in bytes, of the NAND chips comprising the array.
10:9	**Defective units**: This field specifies the number of known defective Erase Units in the array.
12:11	**Reserved units**: This field specifies the number of Erase Units in the array reserved for defect management purposes.

[0186] If a particular die/device has multiple planes, the host further interrogates the memory controller (e.g., with the NVMe Get Features command) to determine which flash memory operations other than a multi-plane write operation are supported. This or an equivalent command is also preferably used to learn each multi-plane addressing mode supported by the device (i.e., any device-specific multi-plane addressing restrictions). The supported operations can be returned in the bit-encoded Dword 0 of the completion queue entry for the command. The bit field definitions are described in Table 2, below.

Table 2: Supported multi-plane operation encoding in the completion queue entry Dword 0

Bit	Description
31:04	Reserved
03:03	Set to 0/1 if the device does not support/supports multi-plane deallocate operation
02:02	Set to 0/1 if the device does not support/supports multi-plane erase operation
01:01	Set to 0/1 if the device does not support/supports multi-plane read operation
00:00	Set to 0/1 if the device does not support/supports multi-plane write operation

[0187] Then, as noted in FIG. 2B, when allocating physical space, the host where possible prearranges (reserves) co-located space in multiple planes' erase units so that ranges allocated in each individual unit of the same LUN have the same offsets within the erase unit or, more generally, so as to account for multi-plane addressing limitations regarding the erase unit(s) addressed by a multi-plane operation. Again, as referenced earlier, the host in some implementations interacts with a memory controller (and/or SPD) to determine drive capabilities, including device and memory layout and any addressing limitations. An API can be used on the host end for this purpose, with a then-educated host using provided information to plan memory assignment. Write operations that address co-located space on multiple dies are then issued by the host to the memory controller as what in effect is a single request to each die. Finally, when such data is to be read back, the host issues a multi-plane read command instead of multiple, independent read commands directed to single planes with the pertinent

physical address(es). As to additional multi-plane commands, memory devices are preferably used that support multi-plane deallocate (page release) and erase commands. As noted earlier, host-owned, shared or other forms of sole or cooperative management can optionally be used consistent with these multi-plane functions. Use of these management functions helps support continual host-awareness of physical space status, such the host is in an independent position to assess space needed for multi-plane writes and to perform maintenance on an as-needed basis.

[0188] To provide support for transaction scheduling in a conventional host, an API can be added to join multiple single-plane IO requests for related data into what is in effect a single, multi-plane operation and to effectuate pertinent scheduling in the host or OS transaction queue. This can be performed by providing for use of a request structure that distinguishes multi-page commands from single-plane commands, or alternatively, by a structure that permits chaining of multiple single-plane IO requests. These commands are then scheduled by the host as an integral unit and sent to the memory controller consecutively (i.e., they not subject to scheduling reorganization). NVMe provides a command structure that is consistent with the use of such single-plane IO requests. That is, NVMe provides for two individual commands to be fused, and requires matching address fields for these fused commands. A vendor-specific modification of this structure permits use of a first fused command field (bit code "01b") as describing *first and subsequent* fused operations in a sequence of fused commands, and a second fused command (bit code "10b") as describing the *terminal* operation of the fused sequence. Note once again that while select embodiments are described in the context of NVMe (e.g., in the context of GetFeatures, fused and other commands), other commands can also be used to provide these same capabilities relative to other storage and networking protocols; even if a particular protocol (e.g., the most recent SCSI versions) do not provide for linked commands, it is otherwise possible to pipeline or otherwise effectively issue multi-page commands using techniques referenced below.

[0189] A fused command structure is depicted in FIG. 20A by reference numeral 2001. This is to say, a first write command (cmd0) specifies a first physical address (PA0) in a first plane and includes a particular bit field that indicates that it is a non-terminal entry of a sequence of fused commands (that are to be performed in sequence). A second write command (cmd1) immediately follows and contains necessary information to specify a physical address (PA1) in a second plane that is consistent with addressing restrictions for the particular device. Note that while a command structure for two planes is illustrated, by extension, any number of commands can be fused together (e.g., successive commands for four planes). Each command specifies a single address with a subfield of the address indicating a

single plane, and consequently, is structured as a single plane command. For each command (other than the last command), the fused command field is set to also indicate that the command is part of the sequence of commands (but is not "last"). The terminal command of the sequence has a different value in the fused command field, indicating it is the last command in the sequence. The write commands are then followed by write data for each plane in succession. An NVMe-compliant memory controller recognizes these commands, and optionally does not process the first command until the last command in the sequence has been received. As an example, the memory controller does not write data to the medium (or return data to the host) until the entire sequence of fused commands has been received. Note that as referenced earlier, in one embodiment, these commands are scheduled (e.g., for a particular drive destination) and are sent in the form of packets to the particular drive. Also as earlier noted, the routing structure can be consistent with a signaling standard, such as a version of PCIE, SATA, or a different signaling standard. If desired, these communications can be routed from the host point-to-point to the memory controller, or can be routed via a switch, bridge, hub or a multidrop bus, or via another interconnection mechanism.

[0190] FIG. 20B shows a single-command equivalent to the fused commands of FIG. 20A. In particular, a request 2003 in effect specifies a multi-plane command (MP) and a base address. If the memory die in question supports any type of addressing offset between planes, the request can also specify an address offset for the second (or each additional) plane. Alternatively, if supported, the command structure can include separate address fields for each plane. Once again, write data (as appropriate) for each plane can be sent together with the commands.

[0191] FIG. 20C shows the fusing of multiple commands in the read direction. In the same manner as described above, a first read command (cmd0, 2005) specifies a physical address and a bit field indicating it is a non-terminal command of a command sequence. Additional single-plane commands such as "cmd1" follows this command and are effective to specify address in a respective plane (e.g., either specifying an implied offset associated with plane address or specifying a complete physical address field corresponding to the respective plane). As before, each command in the sequence carries a field indicating whether it is a non-terminal or terminal member of the sequence, and the memory controller does not respond with read data (such as read data 2009 for a first one of the commands) until after the last command in the fused sequence has been received (i.e., as denoted by a delay "t" seen in FIG. 20C). The memory controller issues the reads requests to the managed memory as appropriate, and returns the read data as respective, sequential pages of data (2009, 2011).

[0192] Note that memory controller-to-memory interaction does not have to be structured in this same manner. That is, the memory controller may have its own transaction queue that receives commands, schedules them, and then issues them to memory as appropriate. Thus, as indicated in FIG. 20D, the memory controller, even if it receives fused commands, may elect to forward commands to a memory die (or to multiple dies) as a series of single plane commands. Thus, for example, the memory controller can issue a first command to memory (cmd0, 2013), corresponding to the first command received from the host (cmd0, 2005, from FIG. 20C), and receive read data 2015 from the memory (following latency "t1"); the memory controller can then, after delay "t2" issue a second command to memory (cmd1, 2017), corresponding to the second command received from the host (cmd1, 2007, from FIG. 20C) and receive read data 2019 from the memory (again, following latency "t1"). In this case, the interval "t" from FIG. 20C would optionally correspond to the time following the memory controller's receipt of the last fused command (cmd1, 2007) through a time necessary for the controller to obtain all sets of read data (2015, 2019) from the managed memory, such that the respective read data could be sent consecutively (without bus idle time). The memory controller could then reply with the data, in what is effectively a single frame of multi-plane data.

[0193] FIGS. 21-25 are used to discuss in detail the performance of various multi-plane requests from the host's perspective. In a NVMe-compatible environment, each of these requests can be issued using the fused command field framework just described. Naturally, a different command structure can be utilized in different implementations, in compliance with other standards or existing standards as they evolve.

[0194] FIG. 21 depicts flow associated with a multi-plane (MP) read request, generally designated using reference numeral 2101. More specifically, the host first issues a read request (2103); as mentioned, a single multi-array request can be used for some embodiments and fused or chained commands can be used for others. As depicted in dashed-line block 2105, these commands can optionally specify a base address and an address extension for accessing a second plane. As also depicted by dashed-line block 2107, optionally, each command specifies a physical address to the memory controller, that is, notwithstanding the issues of wear (and wear-leveling and garbage collection) typically needed for flash memory maintenance. The memory controller then reads the requested data from the associated addresses in respective arrays (2109). As mentioned in connection with FIG. 20D, the memory controller-to-memory interaction can be in the form of single-plane accesses or multi-plane accesses as appropriate (2111). The memory controller and/or host, to the extent they track per-unit data for units of physical memory, can update metadata as appropriate (2113); for

example, metadata can be updated to reflect last-read characteristics and other metadata useful in determining whether data is "hot" or "cold." Either independently or as part of his process, the read data is transmitted to the host (2115), optionally via packetized, serial transmission (as mentioned).

[0195] FIG. 22 depicts flow associated with a MP write request, generally represented by numeral 2201. As depicted, the host issues a multi-plane write request to the memory controller, per numeral 2203. As before, this request can optionally be a set of fused commands that specify a common (base) address, with or without plane-address extensions (2205). For hosts that monitor and assign physical addresses (2207), this address is used in lieu of logical address (e.g., this corresponds to the logical address field referenced by the NVMe standard). The memory controller receives and acts on this physical address without address translation (or with limited address translation, if the memory controller owns defect management), per numeral 2209. As before, single-plane or multi-plane accesses can be used by the memory controller to perform the requested write operations (2211). As part of this process, the memory controller and/or host perform defect management (and any associated L2P remapping) if issues occur and update per-physical-unit metadata (2213). In some embodiments, the memory controller returns a write acknowledgement (WA) to the host in connection with these operations. Once again, all host-memory controller communications are optionally performed using packetized, serial transmission.

[0196] FIG. 23 depicts flow (2301) associated with a MP deallocate command. That is, the command framework introduced above for multi-plane writes can also be used to ask (via a single request) that the memory controller mark linked or paired pages of data in respective planes as "released." As noted earlier, once all pages in an EU have been released, that EU can be erased and recycled. Support for a MP deallocate command permits simultaneous release of parallel physical storage units across planes such as previously allocated by the host in appreciation of the multi-plane device capabilities, and so, permits release and recycle with fewer commands and reduced latency. Per numeral 2303, the host issues such a request, for example, as a multi-plane command or set of fused commands (depending on embodiment). Once again, per numerals 2305 and 2307, these commands can specify a base address (or common address) and can specify physical addresses that pass un-translated by the memory controller. The memory controller receives the request(s) and updates its per-physical-storage-unit metadata to indicate release of the affected pages for each die or plane (2309). As indicated by dashed-line (optional function) block 2313, the memory controller can use the processes described earlier (e.g., asynchronous alerts to the host) to indicate that an EU corresponding to a released page is a candidate

for erasure or garbage collection; this condition can be determined as a function of any desired threshold (e.g., no more valid pages in EU, EU page utilization below a defined threshold, and so forth).

[0197] FIG. 24 depicts flow (2401) associated with a MP erase request, in many ways similar to the flow of the MP deallocate command just described. The host first issues a MP erase request (2403), for example, based on an asynchronous alert or responsive to a host query. Alternatively for embodiments where the host tracks page utilization by physical unit, the need for erasure can be prompted by a host software process. The host as part of this request informs the memory controller of a base address to be applied to all pertinent planes to erase affected EUs (2405). This request can include physical EU address as a command operand, as denoted by optional function block 2407. This request can also be sent as a series of fused commands, as indicated. Upon receiving the host command(s) (e.g., multi-plane or fused), the memory controller erases all affected EUs (2409). Depending on embodiment, the memory controller can then either update its own stored data repository (metadata), or can send information (e.g., as part of an ACK) to the host, specifying that the host should update its stored metadata to indicate status of the affected EUs, and update listings of available space, per numerals 2411 and 2413.

[0198] FIG. 25 depicts flow (2501) associated with a defragment operation initiated by the host. Host software first determines the need for defragmentation, as represented by numeral 2503. As implied by this label, the host can determine that data stored in disparate locations should be stored together for more efficient access, e.g., as respective pages of multi-plane data. Per numerals 2505 and 2507, the host first commands reads of the fragmented data from the pertinent locations. Either the host or memory controller, depending on embodiment can then update stored metadata to record release of the affected pages (2509). Then, the retrieved data is written via a multi-plane write operation to a new destination, for example, using a base address common to each affected memory plane (2511). Finally, the host and/or memory controller update both address translation tables (the host, if using physical location addressing) and metadata for the physical storage units destinations of the newly written data. Note that if supported by the memory controller, the entire sequence can be performed as a delegated atomic operation using a single sequence of fused commands (i.e., without retrieved data being sent from the memory controller to the host).

[0199] Note that this operation can also be used in any multi-page die context, such as a multi-die context also referenced by function block 2511. For example, a host wished to redistribute data across multiple dies (for potentially faster, interleaved access) can use the method generally depicted in FIG. 25

for doing this. Because each die is typically addressed as a response to a separate host command, in some embodiments, it is not necessary to have related data stored at like-physical addresses in different dies (e.g., any desired physical address can be used). In some embodiments, custom multi-die commands can be used, for example, to direct concurrent ("group") write/read/deallocate/erase/or other functions across multiple dies, for example, using a common physical address or address offset.

[0200] As should be apparent from the description above, the disclosed techniques can provide substantial improvements in memory system operation, particularly for flash memory. The use of physical-unit addressing by the host permits a host to structure data operations to enable multi-page write, read and other operations (e.g., at the system level). Whether or not multiple planes of memory are present in a system, this structure also permits the host to enhance interleaving across multiple pages/dies/devices of memory by permitting the host to schedule interleaved requests in a manner unencumbered by the requirements of full, conventional L2P translation at the memory controller.

VI. Conclusion.

[0201] 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 by bundling them in a common 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 types, but it is possible to have a single tier of memory (e.g., all NAND-flash), or three or more tiers of memory of different types. 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.

[0202] 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, magnetic media (such as the aforementioned shingled disk drives) 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 single-level cell (SLC) NAND-flash memory, but it should be equally possible to apply the teachings herein to multi-level cell (MLC) NAND-flash memory, and to SLC or MLC NOR-flash memory designs.

[0203] 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 a complex translation layer. 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 translation layer. For example, it is possible to combine cooperative garbage collection and space reclamation processes with a conventional translation layer. Advantageously, by providing ready-host visibility into physical subdivision-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 a translation layer.

[0204] One implementation provides a memory controller to control the performance of memory transactions directed to multi-plane memory. The memory controller has interface circuitry (e.g., one or two interfaces) sufficient to communicate with at least one host and with the multi-plane memory. As before, the memory controller can optionally be coupled to a host via a first signaling link (e.g., via a first interface, using packetized communications) and via a second signaling link to the multi-plane memory device. Multiple or shared link designs can also be used. Also, in one design, the memory controller and multi-plane memory can be bundled together on a single IC with no dedicated high speed serial link

between them (e.g., as might otherwise be the case for separate ICs). Further, another design presents a multiple-IC system where controller and multiple memory devices managed by that controller are associated with respective ICs; each memory device can be in the form of a single or multi-die package, with 2, 4, 8 or a different number of dies per package in the latter case. Any number of these dies (e.g., zero, one, many or all) can be configured to have one or more multi-plane flash memory arrays.

[0205] The memory controller can include logic (e.g., hardware logic or software logic) operable to exchanges commands and data via the interface circuitry. This logic in one embodiment is operable to receive multi-page commands from the host, such that the host specifically determines when a multi-page operation (as opposed to a single page operation) should be effected. As discussed earlier, in such an embodiment, the host optionally can mix single page and multi-page commands depending on need and, preferably, has a command structure (e.g., using an NVMe compliant architecture as discussed) that permits exchange of single plane and multi-plane commands with the memory controller. Again, per the discussion earlier, a multi-page command can be configured as a series of "fused" single-page commands, where the memory controller receives all of the commands at once or in close succession and then instructs performance without awaiting completion of a request for a first one of the pages.

[0206] In one aspect of the invention, the logic of the memory controller is capable of receiving and interpreting a multi-page (or multi-array) read request from a host. Note that the actual control over multi-array memory can optionally involve single-page commands between the memory controller and addressed memory device, as referenced earlier.

[0207] In another aspect of the invention, the logic is capable of receiving and interpreting a multi-page write request. Once again, the actual control over multi-array memory can optionally involve single-page commands between the memory controller and addressed memory device.

[0208] In still a further aspect of the invention, the logic is capable of receiving and interpreting a multi-page deallocate (page release) command, or alternatively (or in addition), a multi-page erase command.

[0209] These features can be used together or individually, e.g., this disclosure in one embodiment provides a memory controller capable of receiving multi-page read, write, deallocate and erase commands from a host. In another embodiment, a memory controller can receive a multi-page read command (but not the other commands); alternatively, the memory controller can be designed to

receive multi-page write command (but not a multi-page read command). Indeed, any combination of commands is contemplated as an optional design feature.

[0210] In still another embodiment, these techniques can be applied to specifically managing multi-plane memory. In a variant, these techniques can be used in lieu of or in addition to techniques for managing multi-die memory.

[0211] This disclosure also contemplates systems, methods and other devices that leverage the principles introduced above. For example, one implementation provides a host that operates in a manner reciprocally to the memory controller just discussed. The host can be in the form of a host device (e.g., an IC, computer, web appliance, storage manager, or another master). In one embodiment, the host device has an interface to transmit outbound transaction requests to a memory controller, a table to identify address space corresponding to memory units that span multiple page (or planes or dies) of memory managed by the memory controller, and logic to generate a multi-page (or multi-plane or multi-die) transaction request directed to the address space. In more specific forms of this implementation, the table can be a logical-to-physical address translation table and the memory and the memory controller can be directed to a form of nonvolatile memory such as NAND flash memory; in such setting, the host, possessing an understanding of physical address assignment, can issue new write requests in a manner that directly specifies the writing of data across planes or dies, and the host can also issue read requests on this same basis. Note that this structure substantially reduces command competition with host needs by functions of the memory controller, e.g., the host can directly manage maintenance operations such as garbage collection, erase operations and so forth in a manner that does not interfere with its own data needs; it also reduces need for the memory controller to load and manage complex tables. If desired, the host can issue multi-page transactions requests to the memory controller which specify a base physical address for each affected plane or die (e.g., such that each plane or die operates on a like-physical address, or on addresses based on an express or implied offset).

[0212] This disclosure also contemplates memory management techniques, devices, and systems which rely on the various structures just discussed, either host side or memory controller side, or aggregations that combine these things. In one embodiment, a host determines the need for maintenance actions in flash memory notwithstanding the presence of a memory controller. For example, a host can optionally handle all management tasks of flash memory (e.g., marking of bad data blocks) such that no address translation at all is necessary at the memory controller, and such that the host schedules and initiates one or more of garbage collection, data relocation, defragmentation, wear

leveling, or any one or of the functions discussed earlier. As mentioned, a model can also be used where the memory controller handles certain select functions such as defect management and temporary physical address reassignment responsive to errors; as noted earlier, serving memory controller-tracked physical memory status information, on an EU-specific basis (or other unit basis) to the host permits eventual identification and isolation of bad blocks in a manner where the host updates its understanding of physical address space assignment.

[0213] 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.

[0214] 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.

[0215] 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.

CLAIMS

We Claim:

1. A solid state storage drive (SSD) comprising:

a host interface configured to communicate with a host using commands from a Non-Volatile Memory Express (NVMe) standard;

non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages;

logic coupled to the host interface and to the flash memory, wherein the logic comprises at least one of hardware logic and software logic, and communicates with the host via the host interface using NVMe commands;

wherein the logic and the flash memory are operable to:

divide the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in at least one zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses;

in response to a write command from the host to write data within the range of logical block addresses of a specified zone, perform a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes;

generate metadata based on accessing and maintenance of erase units in the specified zone; and

based on the metadata, make a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone.

2. The SSD of claim 1 further comprising:

additional non-volatile memory, wherein the logic is further operable to store the generated metadata in the additional non-volatile memory.

3. The SSD of claim 1 wherein the logic is further operable to compare the metadata with at least one threshold, and wherein making the notification available to the host is dependent on the comparison.

4. The SSD of claim 1 wherein the logic is further operable to determine, based on the metadata, that maintenance is recommended for the specified zone and to identify the zone start logical block address associated with the specified zone for which maintenance is recommended; wherein determining that maintenance is recommended for the specified zone, identifying the zone start logical block address associated with the specified zone, and making the notification available to the host are all unsolicited by the host.

5. The SSD of claim 1 wherein the logic is further operable to receive a command from the host; and, in response to said command, transmit to the host (a) information representing a list of host accessible zones including a zone start logical block address for each listed zone, and (b) a host accessible capacity for each listed zone.

6. The SSD of claim 1 wherein for at least one write command addressed within the range of logical block addresses of the specified zone, the specified zone contains an integer number of erase units in the NAND flash memory, and the logic is operable to generate an address offset from the write command which corresponds to a location within a sequential range of locations that corresponds to the integer number of erase units.

7. The SSD of claim 1 wherein:

the logic is further operable to receive read commands and write commands from the host, and to execute the read commands and the write commands by transferring associated data between the host and the NAND flash memory; and

each read command directed to the SSD is accompanied by a logical read address, each logical read address has a first address portion which identifies one of the zones according to a value of the first portion, and a second address portion which identifies a physical memory location in the SSD in a manner that depends on the first address portion.

8. The SSD of claim 1 wherein the logic is further operable to:

store information identifying a number of zones comprising the NAND flash memory and to transmit, to the host, information identifying the number of zones and sizes of the zones; and

receive, from the host, data access commands which are dependent, for the specified zone, on the information transmitted to the host identifying the number and the sizes of the zones.

9. The SSD of claim 8 wherein the logic is further operable to:

in response to a maintenance command from the host directed to the specified zone, issue erase commands to the erase units in the specified zone, wherein the issued erase commands are based on the size of the specified zone in accordance with the information transmitted to the host.

10. The SSD of claim 1 wherein the flash memory comprises multiple dies of flash memory and the erase units on different planes are also in different dies.

11. The SSD of claim 1 wherein the logic is further operable to, upon the performance of the multi-plane write, update the metadata to indicate an amount of unerased host accessible space in the specified zone.

12. The SSD of claim 11 wherein the logic is further operable to: upon the performance of the multi-plane write, detect that the specified zone contains no erased host accessible space; and, in response to said detection, transition the specified zone to a full state.

13. The SSD of claim 12 wherein the logic is further operable to receive a command from the host; and, in response to said command, transmit to the host information representing the full state of the specified zone and an amount of erased host accessible space in the specified zone that is available for writing by the host.

14. The SSD of claim 1 wherein the logic is further operable to receive a maintenance command directed to the specified zone and, in response to the maintenance command, erase host accessible space only within the specified zone and transition the specified zone to an empty state.

15. The SSD of claim 1 wherein the metadata is dependent on an erase count for the specified zone; and the logic is further operable to compare the metadata with at least one threshold for erase count, wherein making the notification available to the host is dependent on the comparison.

16. The SSD of claim 1 wherein the logic is further operable to:

in response to a maintenance command from the host directed to the zone start logical block address of the specified zone, perform an erase of erase units in the specified zone, and transition a state of the specified zone to an empty state whereby all host accessible pages of the specified zone are available for writing by the host.

17. The SSD of claim 16 wherein the maintenance command from the host is in response to the notification that recommends performing maintenance on the specified zone.

18. The SSD of claim 16 wherein the metadata is dependent on erase counts for erase units in the specified zone; and the logic is further operable to update the metadata responsive to performance of the erase.

19. The SSD of claim 16 wherein performing the erase of erase units in the specified zone comprises performing the erase on erase units of different planes in the specified zone.

20. The SSD of claim 1 wherein the logic is further operable to, in response to a read command from the host to read data from sequential addressing of the specified zone, perform a multi-plane read of said data from erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes.

21. The SSD of claim 1 wherein the logic is further operable to, in response to a write command from the host to the zone start logical block address of the specified zone: assign a logical block address within the range of logical block addresses of the specified zone, transfer data from the host to the specified zone at the assigned logical block address, and return the assigned logical block address to the host.

22. The SSD of claim 1 wherein the metadata is dependent on wear of the erase units and the logic is further operable to, in response to a command from the host, transmit to the host information based on such metadata.

23. The SSD of claim 1 wherein the logic is further operable to, in response to a command from the host, transmit to the host information dependent on wear for the specified zone.

24. The SSD of claim 1 wherein the metadata is dependent on a time period since data was written to the specified zone; and the logic is further operable to compare the metadata with at least one threshold, wherein making the notification available to the host is dependent on the comparison.

25. The SSD of claim 24 wherein comparing the metadata with at least one threshold and making the notification available to the host are unsolicited by the host.

26. The SSD of claim 1 wherein logic is further operable to, in response to a command from the host, transmit to the host information dependent on a time period since data was written to the specified zone.

27. The SSD of claim 1 wherein the logic is further operable to detect a failure condition of one of the erase units of the specified zone and transmit to the host the zone start logical block address and a state of the specified zone.

28. The SSD of claim 1 wherein the logic is further operable to detect a failure condition of one of the erase units of the specified zone and, in response to a write command from the host within the range of logical block addresses of the specified zone, transmit to the host a notification of a write failure.

29. The SSD of claim 1 wherein the metadata is dependent on read errors detected for the specified zone.

30. A method for providing a host with access to an SSD that contains non-volatile memory comprising NAND flash memory, the NAND flash memory comprising a plurality of erase units and each erase unit comprising a plurality of pages, the method comprising, at the SSD:

dividing the flash memory into zones; wherein the zones are non-overlapping subdivisions of the flash memory, each zone comprises two or more erase units, the erase units in each zone are on two or more different planes, and each zone is associated with a zone start logical block address and a range of logical block addresses;

in response to a write command from the host to write data to the range of logical block addresses of a specified zone, performing a multi-plane write of said data to erase units on different planes in the specified zone using a common page address for the respective erase units on the different planes;

generating metadata based on accessing and maintenance of erase units in the specified zone; and

based on the metadata, making a notification available to the host that recommends performing maintenance on the specified zone, wherein the recommendation is associated with the zone start logical block address of the specified zone;

wherein the SSD and the host communicate over a host interface, using commands from a Non-Volatile Memory Express (NVMe) standard.

ABSTRACT

This disclosure provides for improvements in managing multi-drive, multi-die or multi-plane NAND flash memory. In one embodiment, the host directly assigns physical addresses and performs logical-to-physical address translation in a manner that reduces or eliminates the need for a memory controller to handle these functions, and initiates functions such as wear leveling in a manner that avoids competition with host data accesses. A memory controller optionally educates the host on array composition, capabilities and addressing restrictions. Host software can therefore interleave write and read requests across dies in a manner unencumbered by memory controller address translation. For multi-plane designs, the host writes related data in a manner consistent with multi-plane device addressing limitations. The host is therefore able to "plan ahead" in a manner supporting host issuance of true multi-plane read commands.

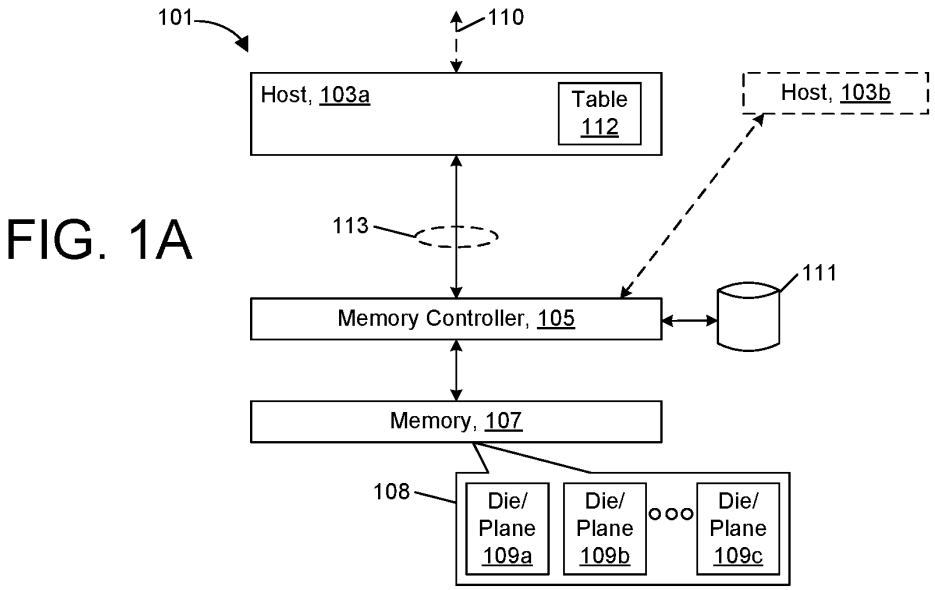

101
110
Host, 103a
Table 112
Host, 103b
113
111
Memory Controller, 105
Memory, 107
108
Die/ Plane 109a
Die/ Plane 109b
Die/ Plane 109c

FIG. 1A

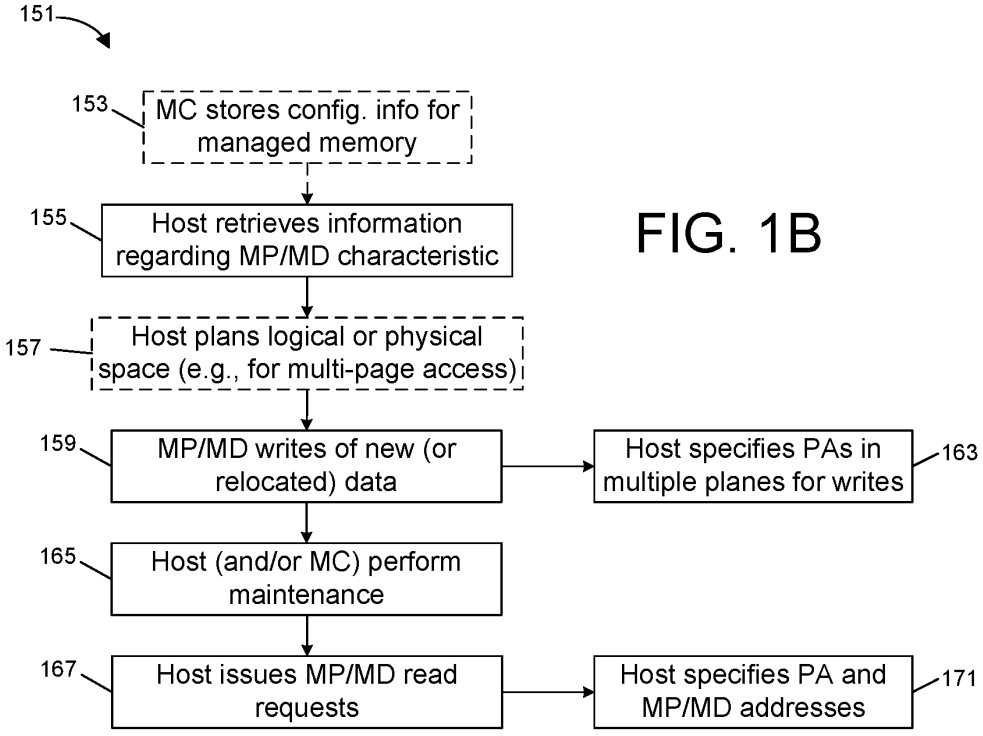

151
153
MC stores config. info for managed memory
155
Host retrieves information regarding MP/MD characteristic
157
Host plans logical or physical space (e.g., for multi-page access)
159
MP/MD writes of new (or relocated) data
Host specifies PAs in multiple planes for writes
163
165
Host (and/or MC) perform maintenance
167
Host issues MP/MD read requests
Host specifies PA and MP/MD addresses
171

FIG. 1B

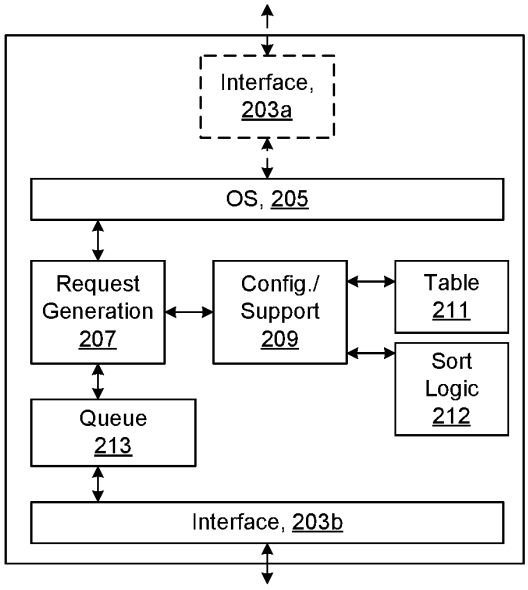

Interface, 203a
OS, 205
201
Request Generation 207
Config./ Support 209
Table 211
Sort Logic 212
Queue 213
Interface, 203b

FIG. 2A

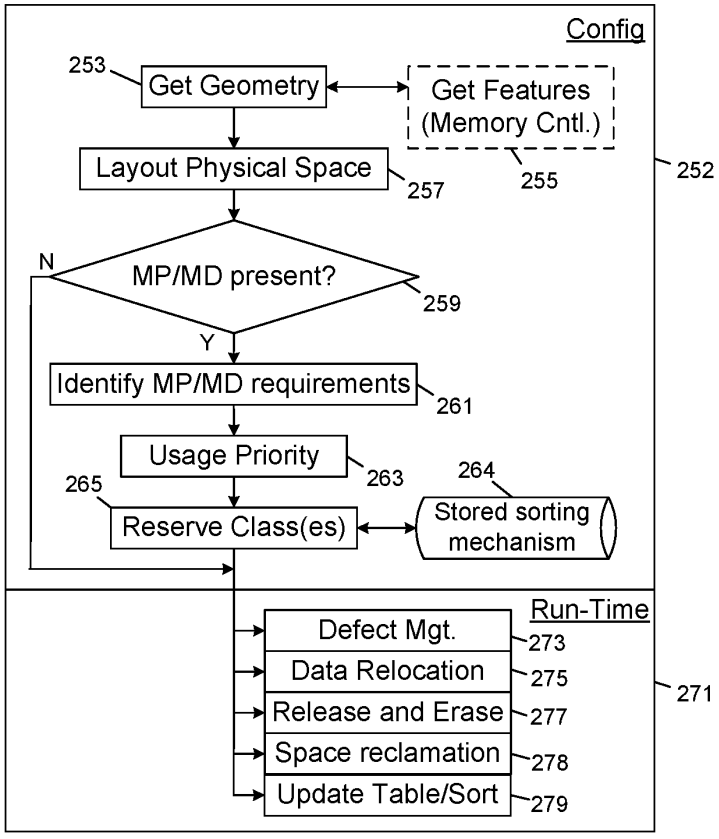

Config
253
Get Geometry
Get Features (Memory Cntl.)
255
Layout Physical Space
257
252
N
MP/MD present?
259
Y
Identify MP/MD requirements
261
Usage Priority
263
264
265
Reserve Class(es)
Stored sorting mechanism
Run-Time
Defect Mgt.
273
Data Relocation
275
Release and Erase
277
Space reclamation
278
Update Table/Sort
279
271

FIG. 2B

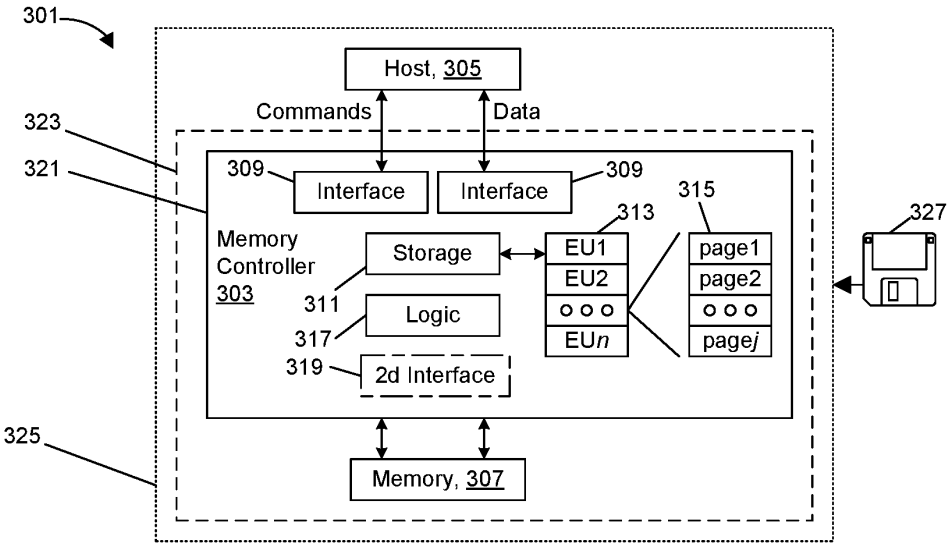

301
Host, 305
Commands
Data
323
321
309
Interface
Interface
309
315
313
327
Memory
Controller
303
Storage
EU1
EU2
EUn
page1
page2
pagej
311
Logic
317
319
2d Interface
325
Memory, 307

FIG. 3A

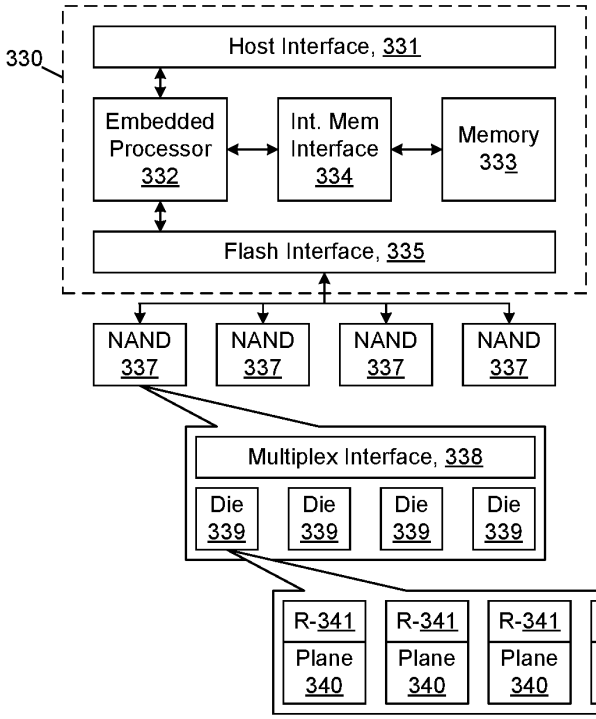

330
Host Interface, 331
Embedded
Processor
332
Int. Mem
Interface
334
Memory
333
Flash Interface, 335
NAND
337
NAND
337
NAND
337
NAND
337
Multiplex Interface, 338
Die
339
Die
339
Die
339
Die
339
R-341
Plane
340
R-341
Plane
340
R-341
Plane
340
R-341
Plane
340

FIG. 3B

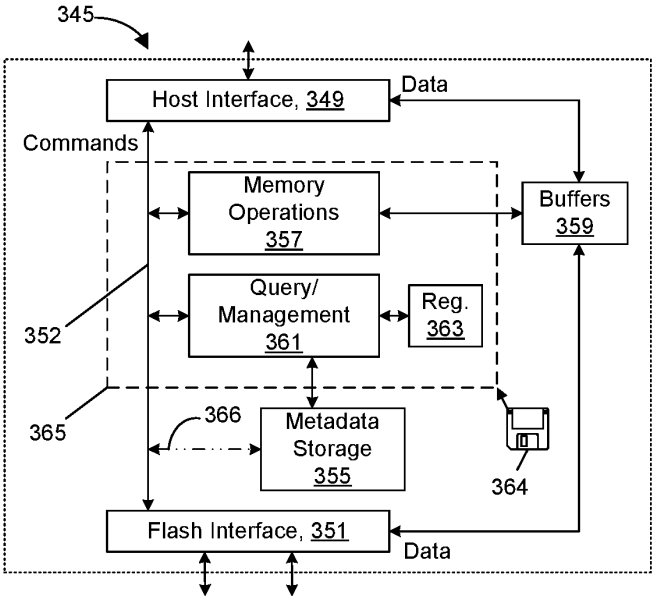

345
Host Interface, 349
Data
Commands
Memory Operations 357
Buffers 359
Query/ Management 361
Reg. 363
352
365
366
Metadata Storage 355
364
Flash Interface, 351
Data

FIG. 3C

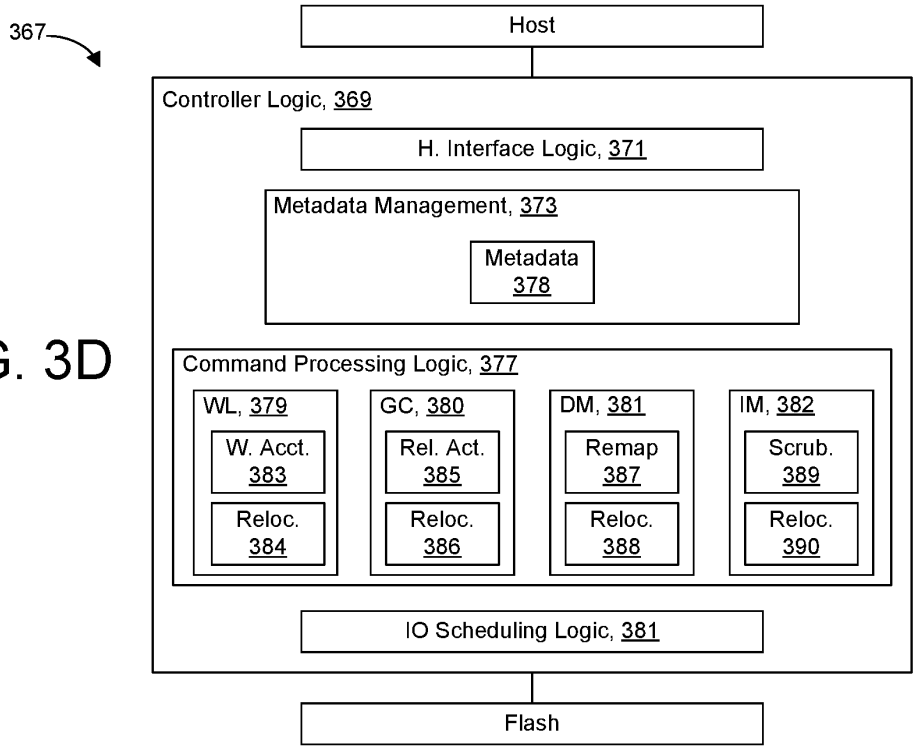

367
Host
Controller Logic, 369
H. Interface Logic, 371
Metadata Management, 373
Metadata 378
Command Processing Logic, 377
WL, 379
W. Acct. 383
Reloc. 384
GC, 380
Rel. Act. 385
Reloc. 386
DM, 381
Remap 387
Reloc. 388
IM, 382
Scrub. 389
Reloc. 390
IO Scheduling Logic, 381
Flash

FIG. 3D

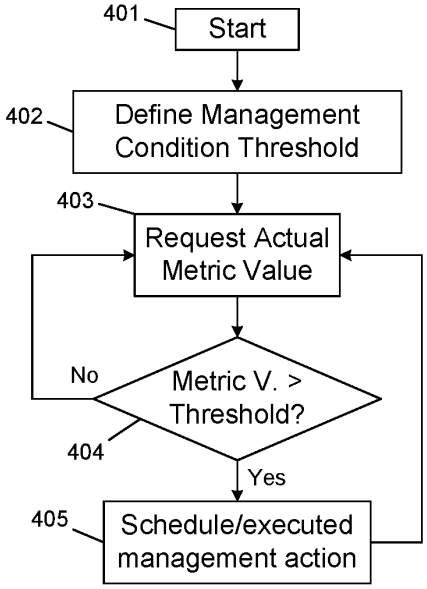

401
Start
402
Define Management Condition Threshold
403
Request Actual Metric Value
No
Metric V. > Threshold?
404
Yes
405
Schedule/executed management action

FIG. 4

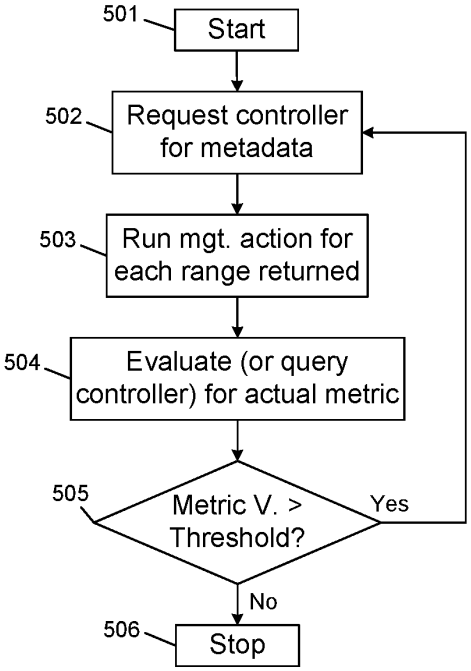

501
Start
502
Request controller for metadata
503
Run mgt. action for each range returned
504
Evaluate (or query controller) for actual metric
505
Metric V. > Threshold?
Yes
No
506
Stop

FIG. 5

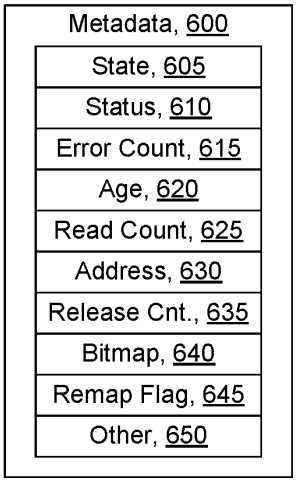

Metadata, 600
State, 605
Status, 610
Error Count, 615
Age, 620
Read Count, 625
Address, 630
Release Cnt., 635
Bitmap, 640
Remap Flag, 645
Other, 650

FIG. 6

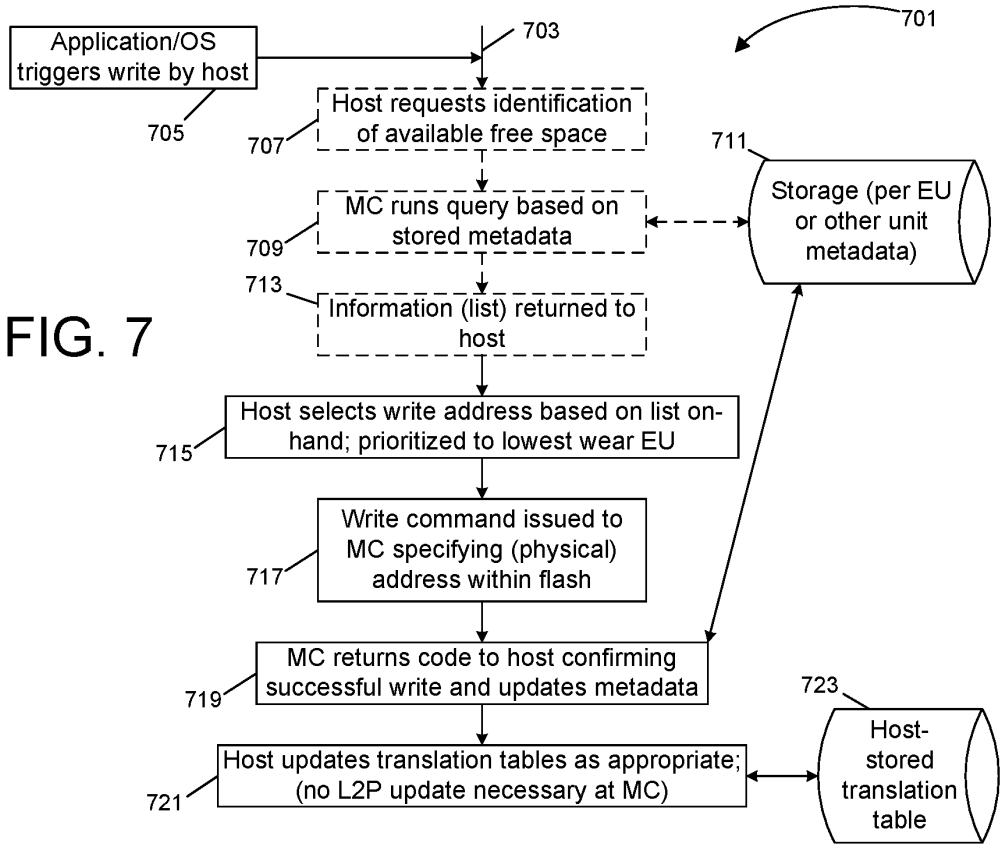

Application/OS triggers write by host
703
701
705
Host requests identification of available free space
707
711
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
709
713
Information (list) returned to host
Host selects write address based on list on-hand; prioritized to lowest 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

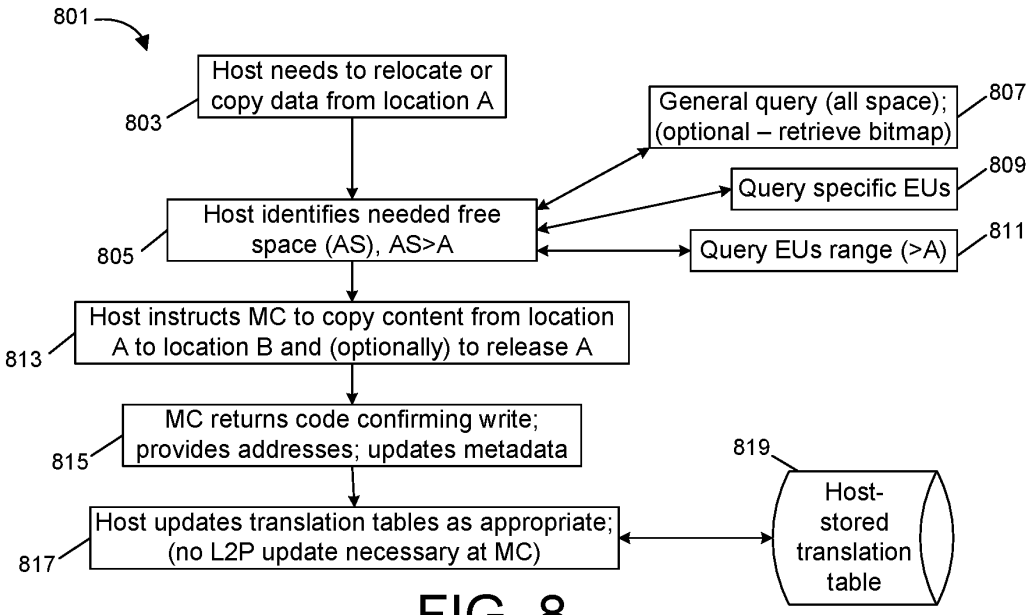

801
Host needs to relocate or copy data from location A
803
General query (all space); (optional – retrieve bitmap)
807
Query specific EUs
809
Host identifies needed free space (AS), AS>A
805
Query EUs range (>A)
811
Host instructs MC to copy content from location A to location B and (optionally) to 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

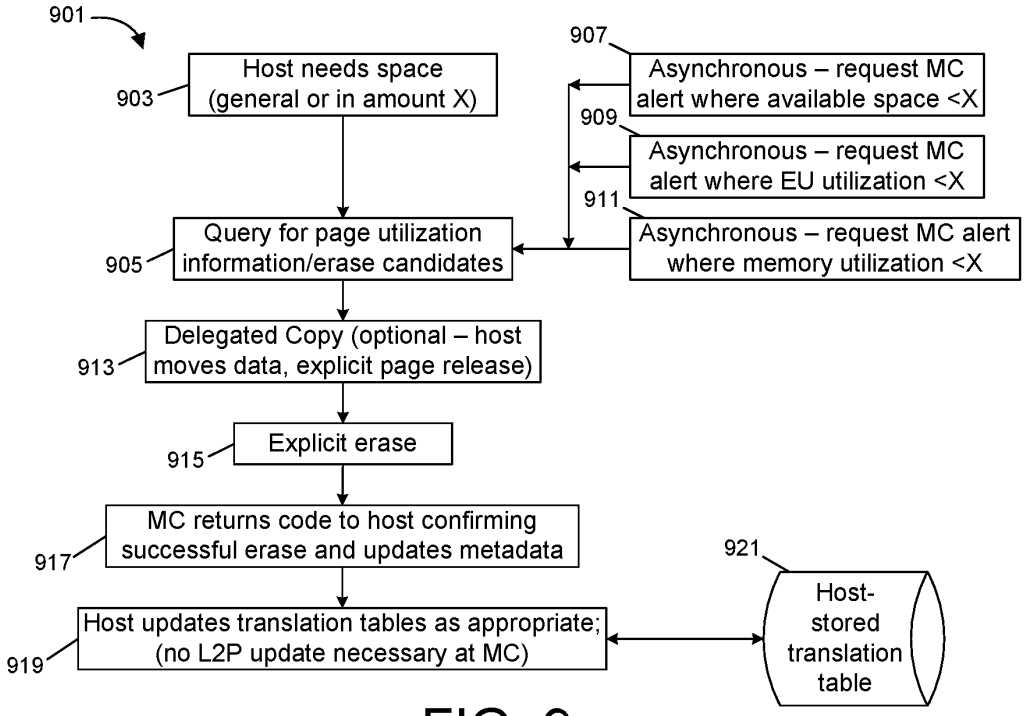

901
Host needs space (general or in amount X)
903
907
Asynchronous – request MC alert where available space <X
909
Asynchronous – request MC alert where EU utilization <X
911
Asynchronous – request MC alert where memory utilization <X
Query for page utilization information/erase candidates
905
Delegated 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

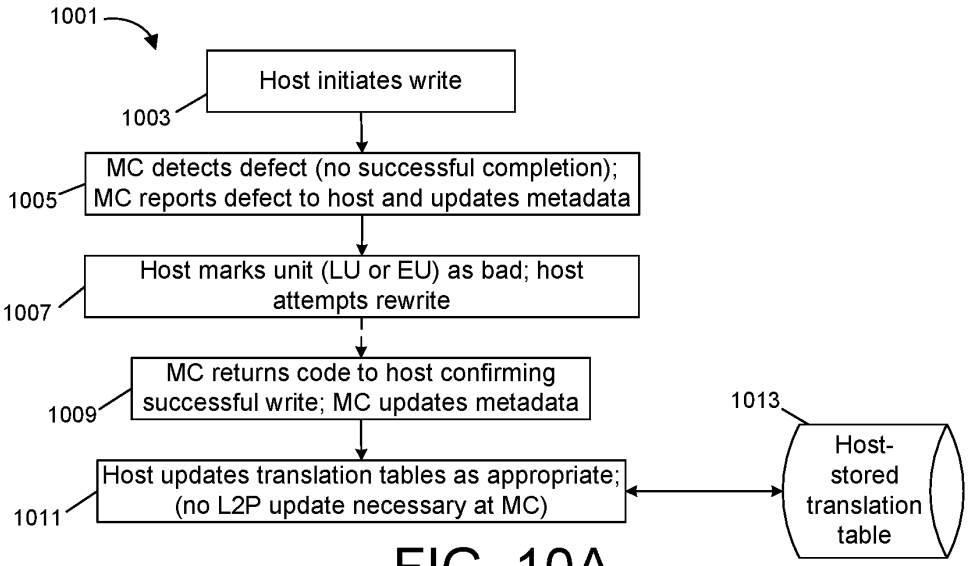


FIG. 10A

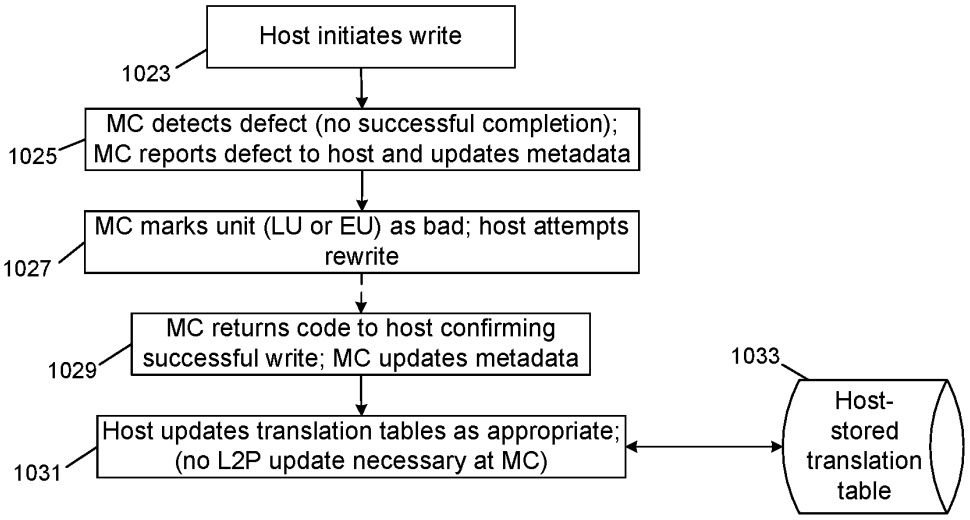


FIG. 10B

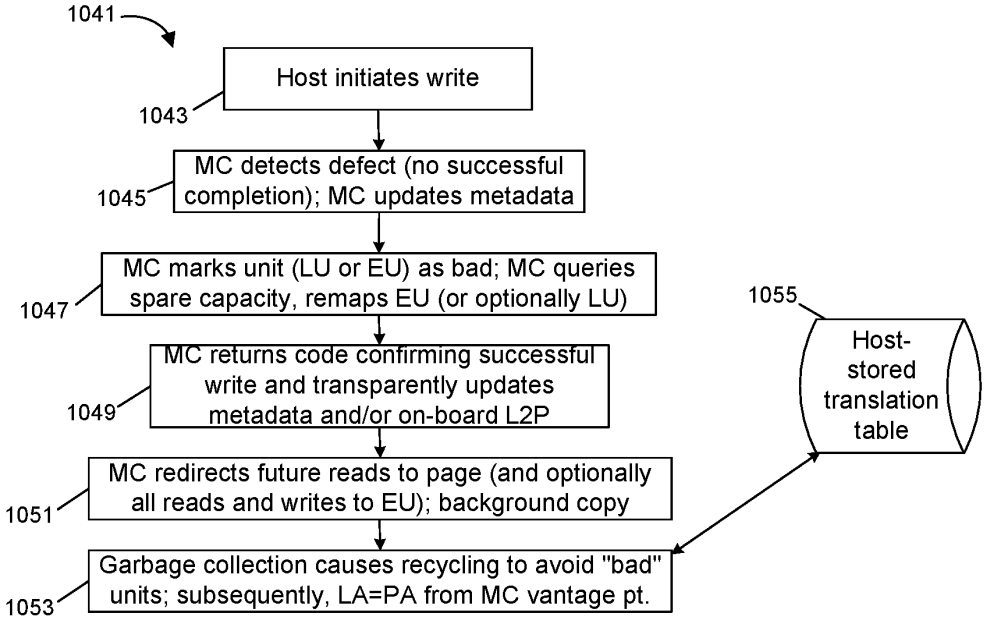


FIG. 10C

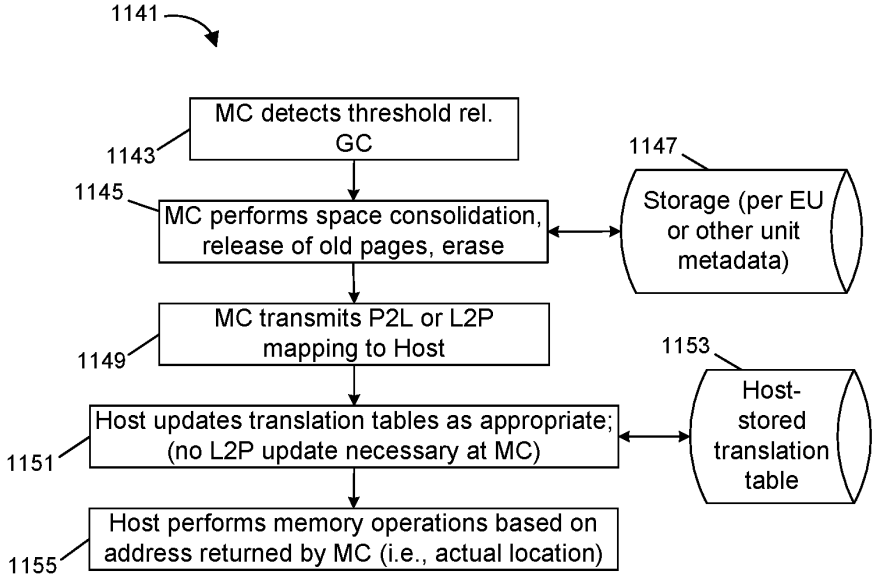


FIG. 11B

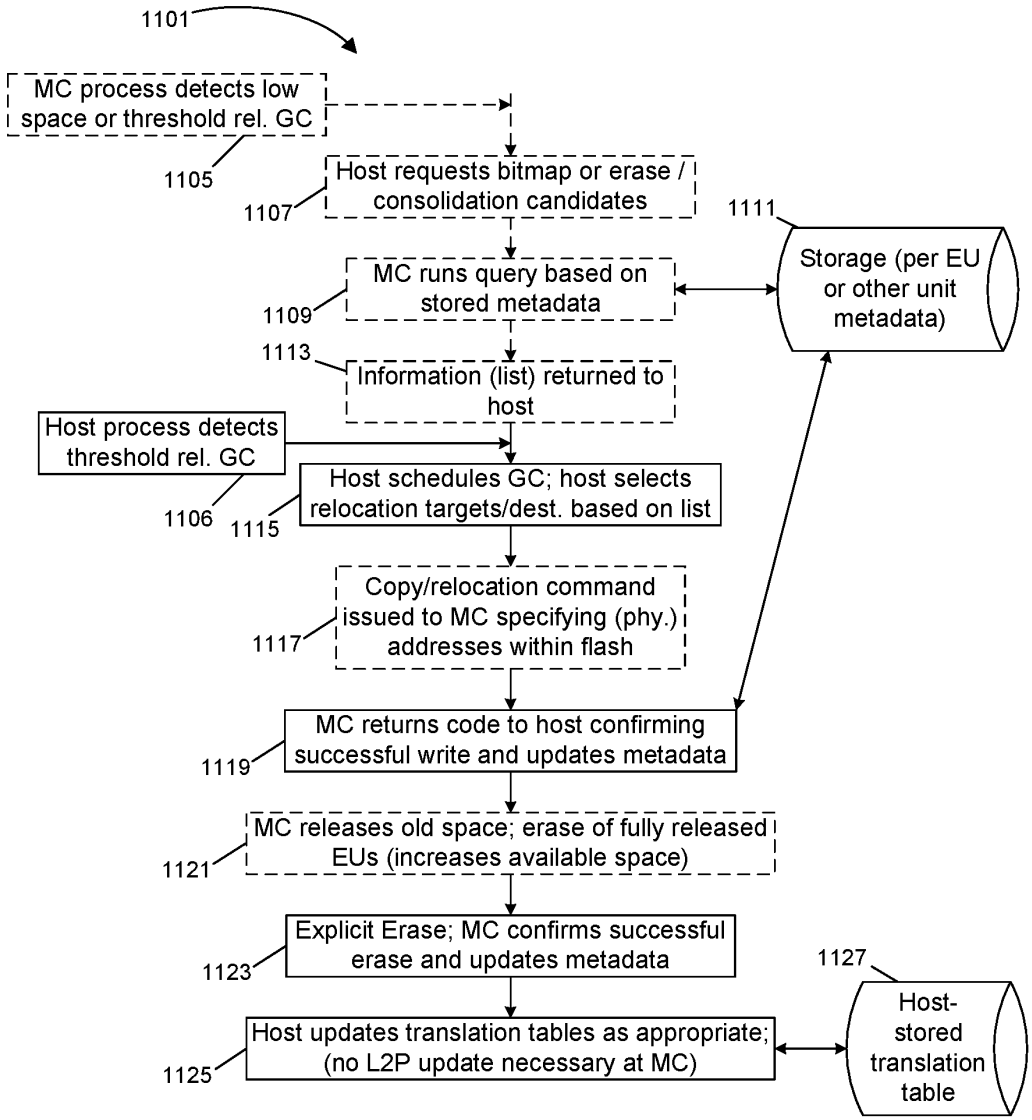

1101
MC process detects low space or threshold rel. GC
1105
Host requests bitmap or erase / consolidation candidates
1107
1111
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
1109
1113
Information (list) returned to host
Host process detects threshold rel. GC
1106
Host 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

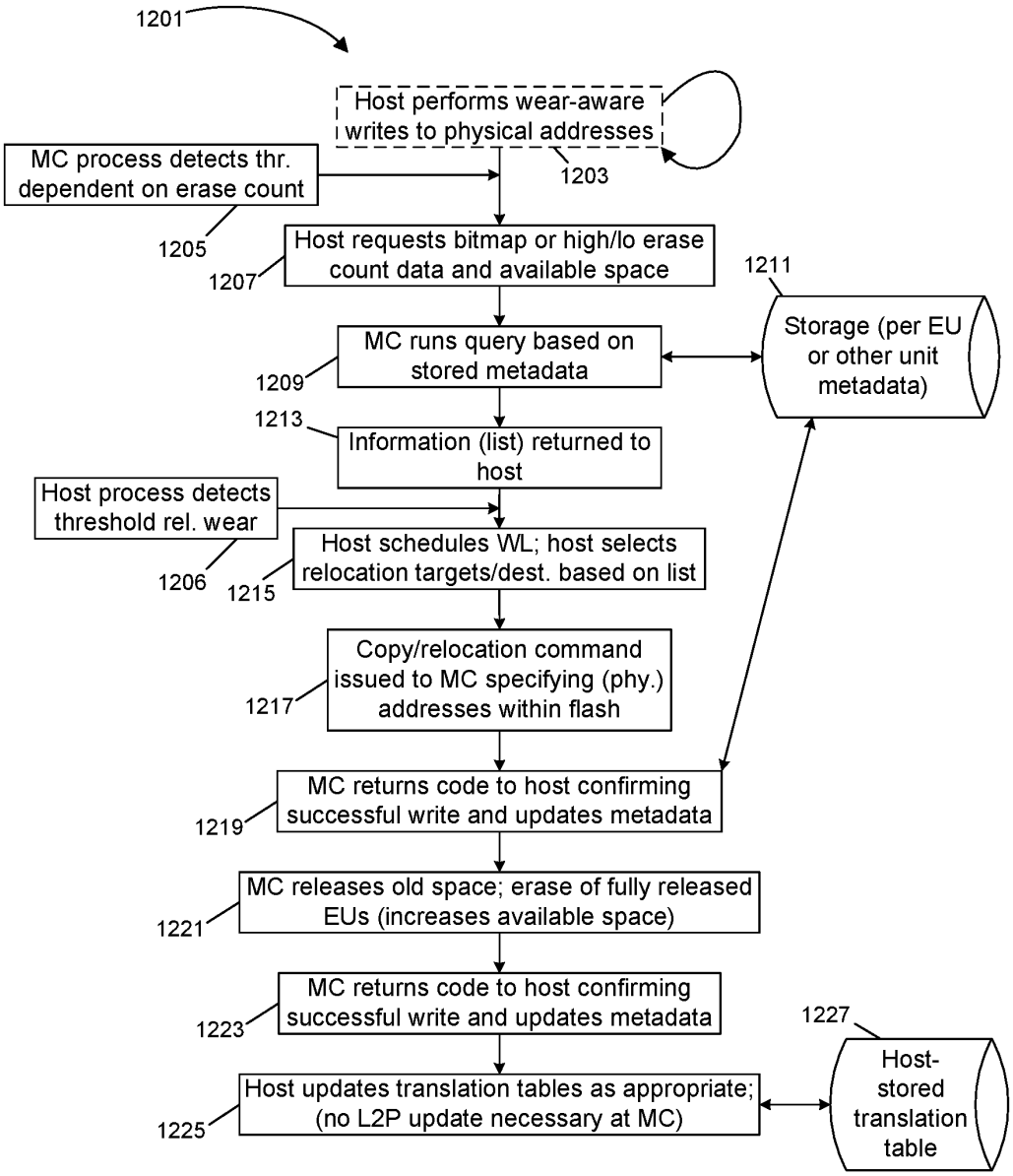

1201
Host performs wear-aware writes to physical addresses
1203
MC process detects thr. dependent on erase count
1205
Host requests bitmap or high/lo erase count data and available space
1207
1211
Storage (per EU or other unit metadata)
MC runs query based on stored metadata
1209
1213
Information (list) 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

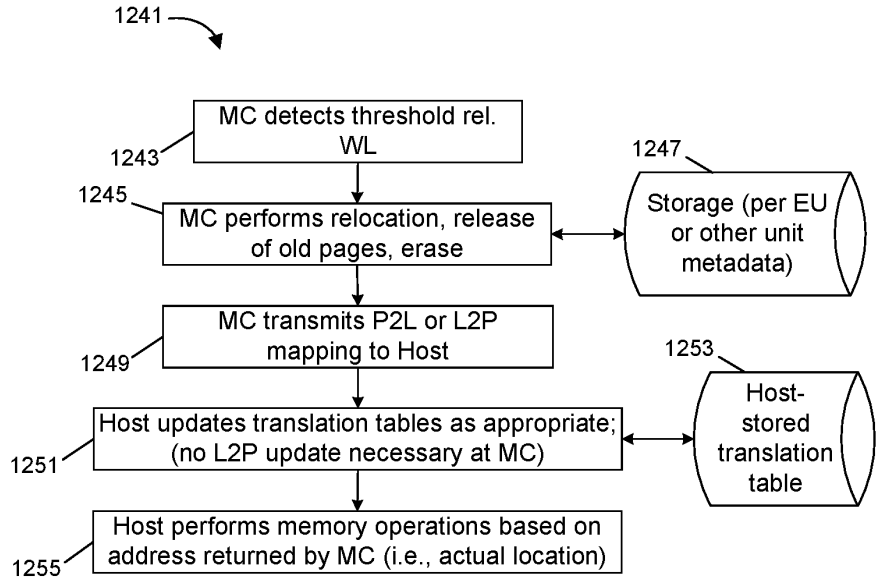


FIG. 12B

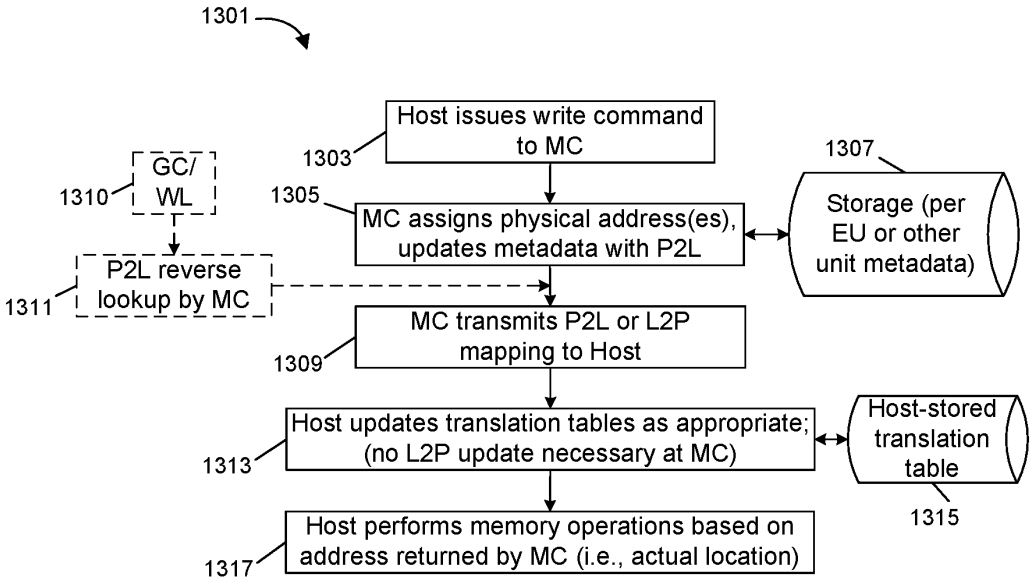


FIG. 13

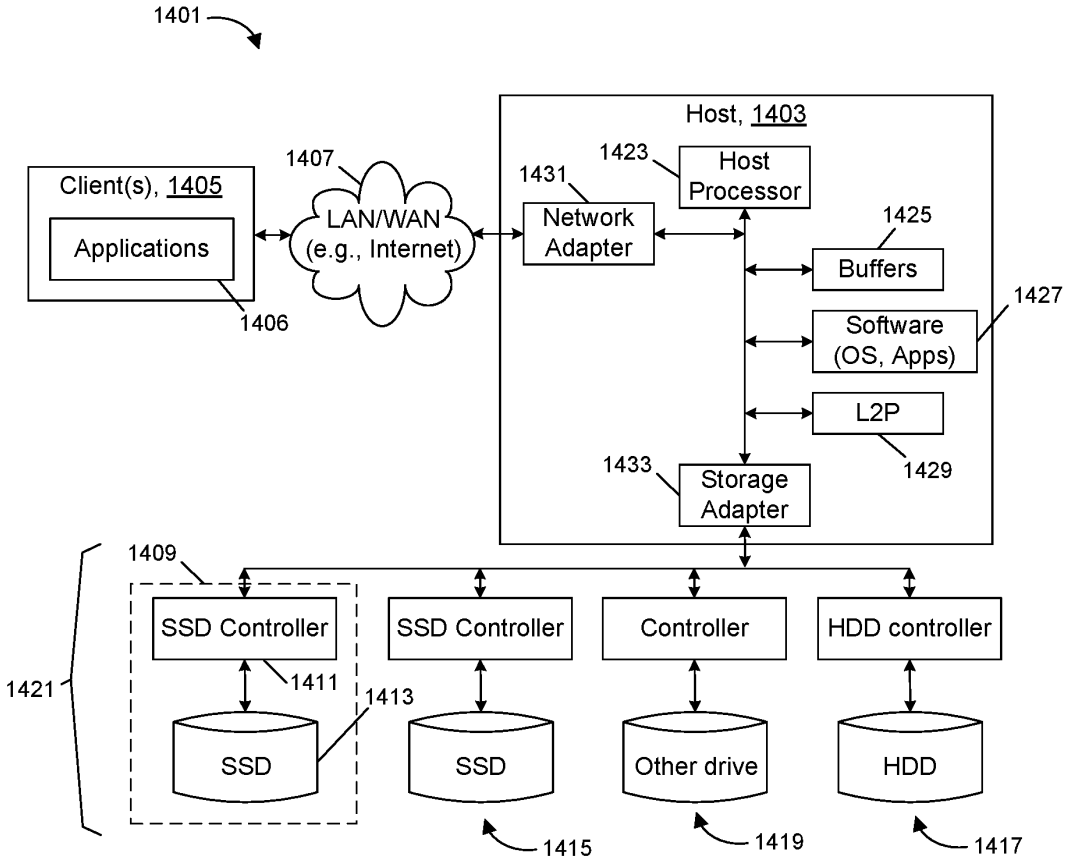

1401
Host, 1403
1407
Client(s), 1405
Applications
1406
LAN/WAN
(e.g., Internet)
1423
Host
Processor
1431
Network
Adapter
1425
Buffers
1427
Software
(OS, Apps)
L2P
1429
1433
Storage
Adapter
1409
SSD Controller
SSD Controller
Controller
HDD controller
1421
1411
1413
SSD
SSD
Other drive
HDD
1415
1419
1417

FIG. 14

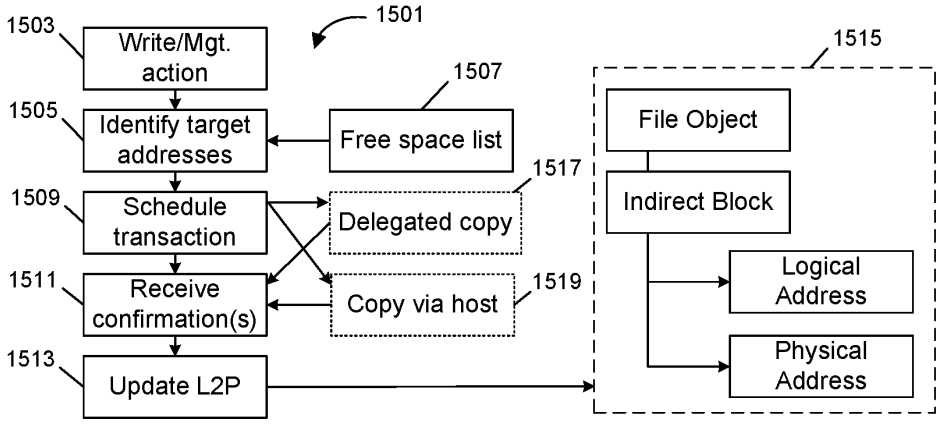

1501
1503
Write/Mgt.
action
1507
1505
Identify target
addresses
Free space list
1517
1509
Schedule
transaction
Delegated copy
1511
Receive
confirmation(s)
1519
Copy via host
1513
Update L2P
1515
File Object
Indirect Block
Logical
Address
Physical
Address

FIG. 15

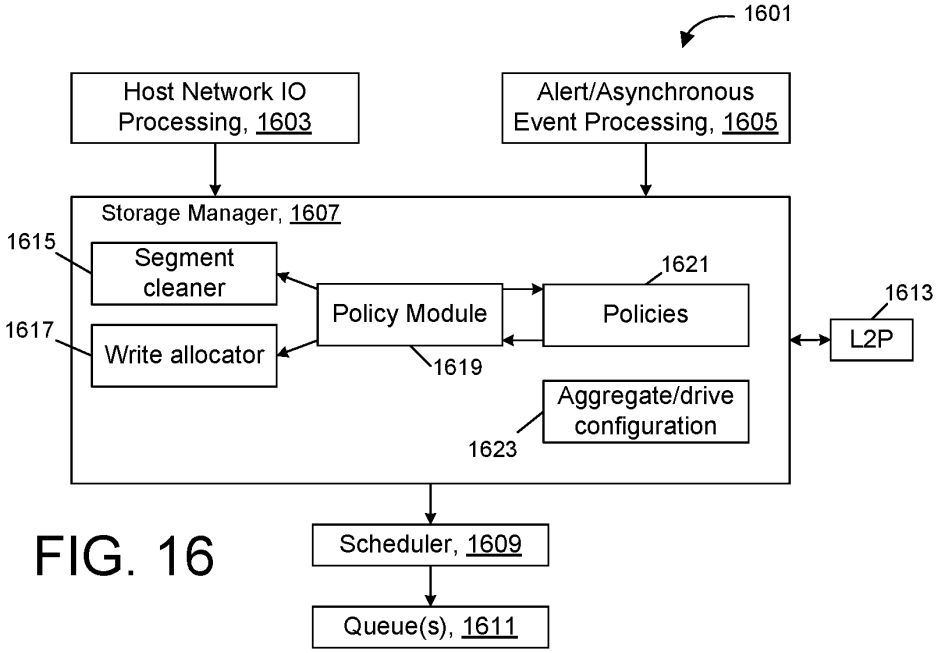

1601
Host Network IO Processing, 1603
Alert/Asynchronous Event Processing, 1605
Storage Manager, 1607
1615
Segment cleaner
1617
Write allocator
Policy Module
1619
1621
Policies
1613
L2P
Aggregate/drive configuration
1623
Scheduler, 1609
Queue(s), 1611

FIG. 16

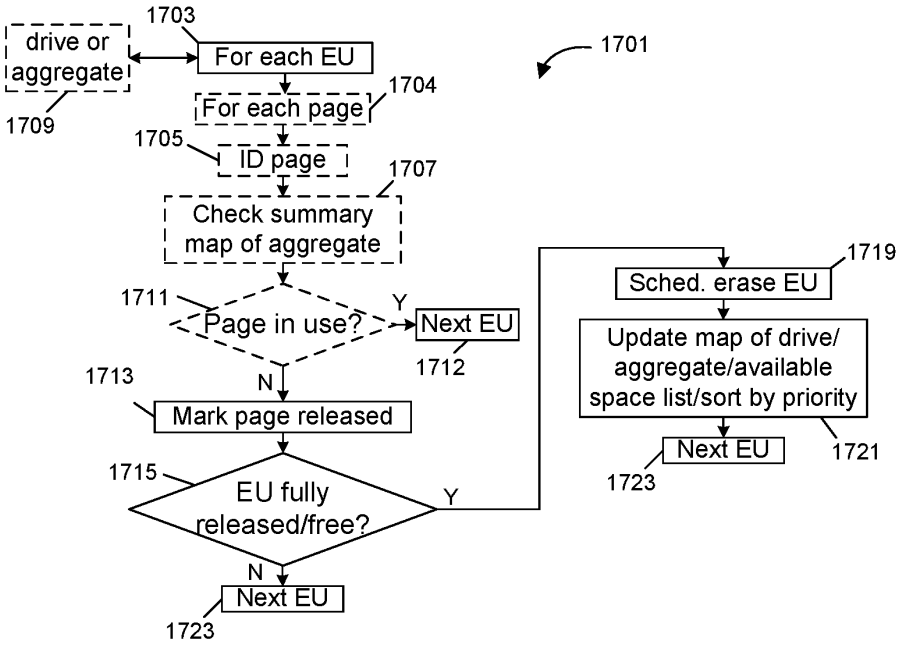

1701
drive or aggregate
1709
1703
For each EU
1704
For each page
1705
ID page
1707
Check summary map of aggregate
1711
Page in use?
Y
Next EU
1712
N
1713
Mark page released
1715
EU fully released/free?
Y
N
Next EU
1723
1719
Sched. erase EU
Update map of drive/ aggregate/available space list/sort by priority
Next EU
1721
1723

FIG. 17

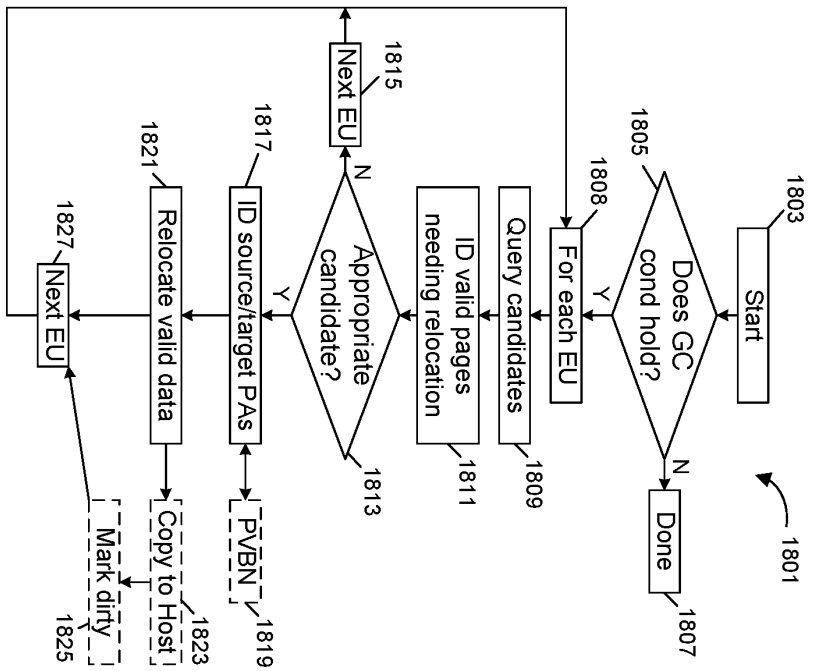

Start
1803
Does GC cond hold?
1805
N
Done
1807
Y
For each EU
1808
Query candidates
1809
ID valid pages needing relocation
1811
Appropriate candidate?
1813
N
Next EU
1815
Y
ID source/target PAs
1817
PVBN
1819
Relocate valid data
1821
Copy to Host
1823
Next EU
1827
Mark dirty
1825
1801

FIG. 18

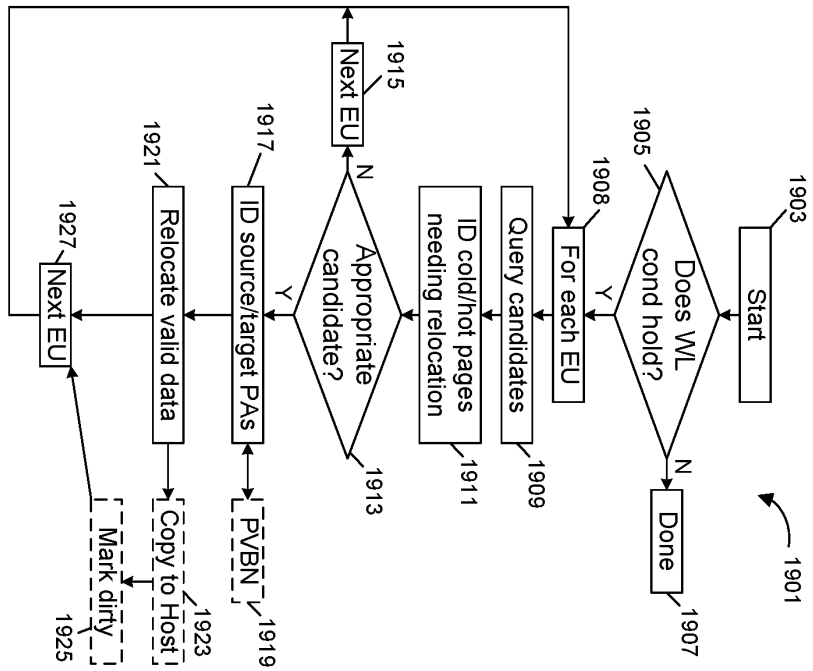

Start
1903
Does WL cond hold?
1905
N
Done
1907
Y
For each EU
1908
Query candidates
1909
ID cold/hot pages needing relocation
1911
Appropriate candidate?
1913
N
Next EU
1915
Y
ID source/target PAs
1917
PVBN
1919
Relocate valid data
1921
Copy to Host
1923
Next EU
1927
Mark dirty
1925
1901

FIG. 19

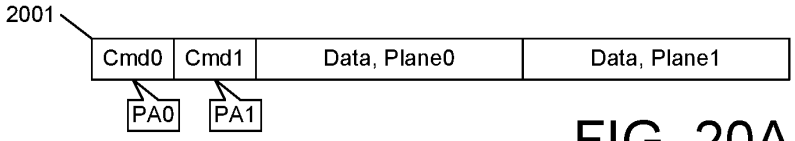


FIG. 20A

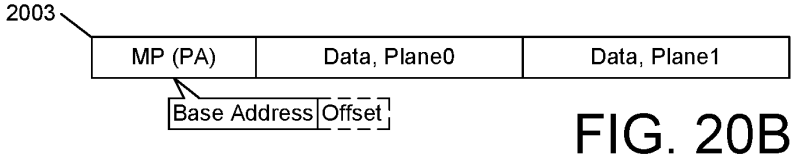


FIG. 20B

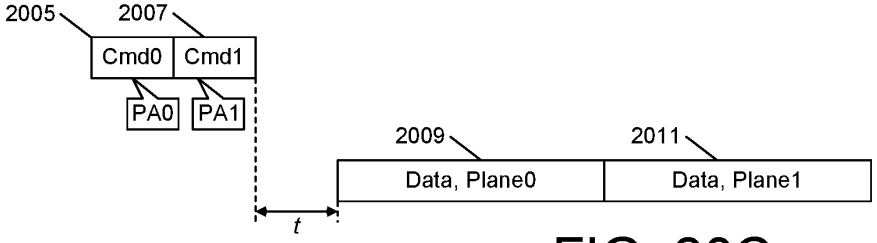


FIG. 20C

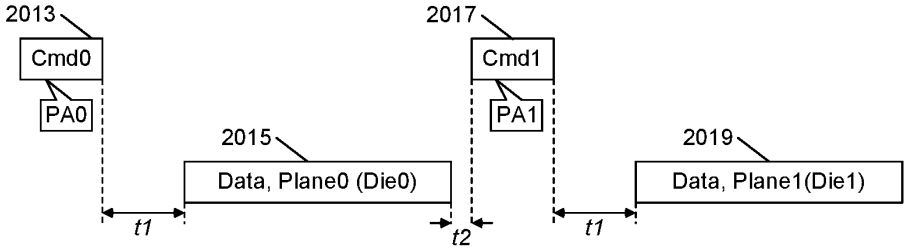


FIG. 20D

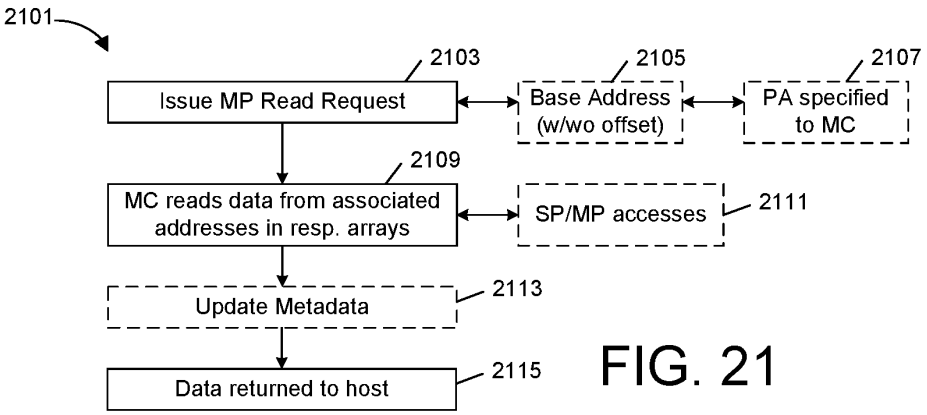


FIG. 21

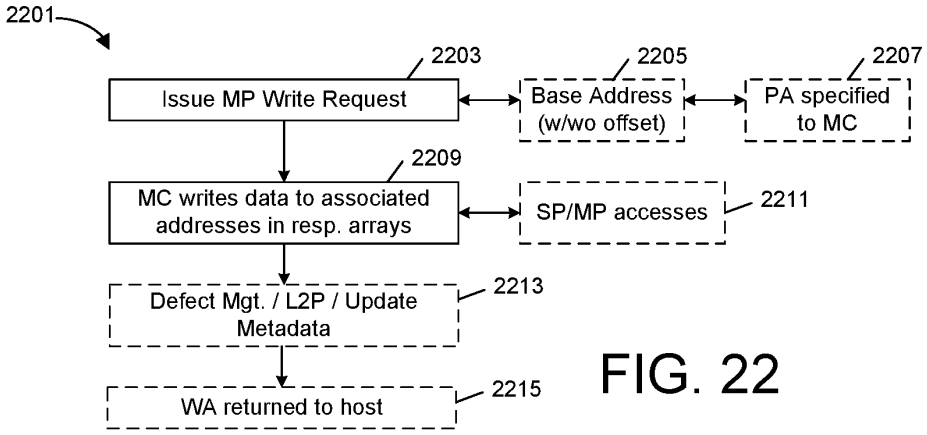


FIG. 22

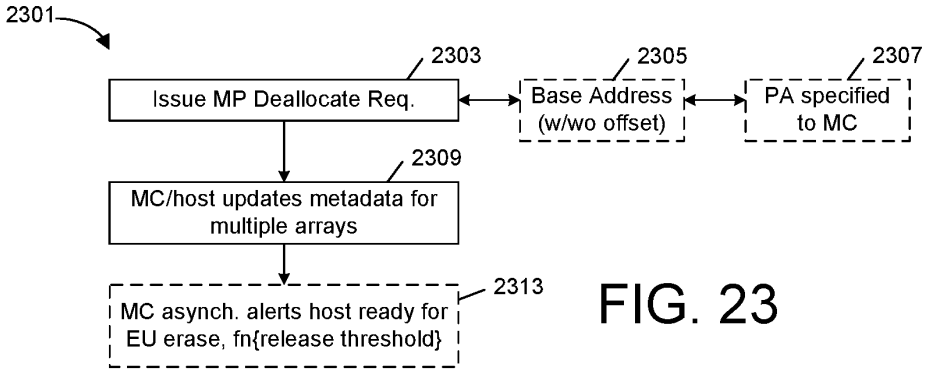


FIG. 23

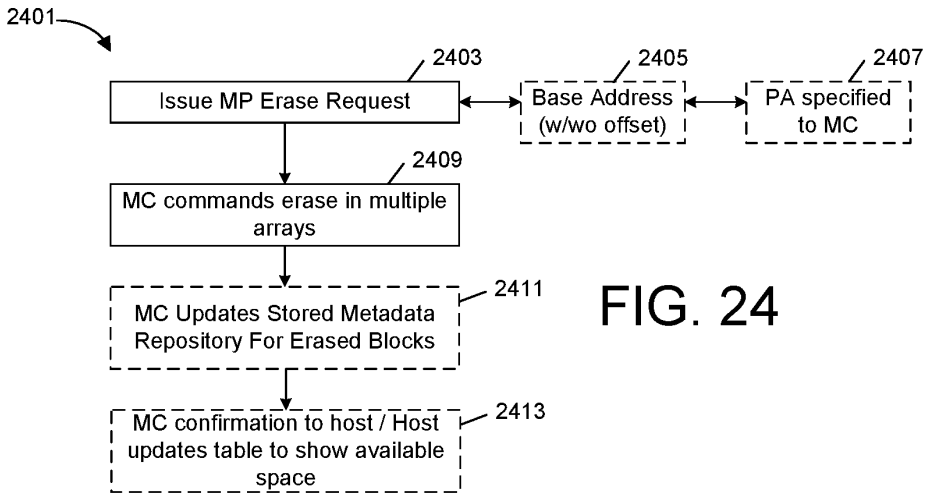

2401
2403
Issue MP Erase Request
2405
Base Address (w/wo offset)
2407
PA specified to MC
2409
MC commands erase in multiple arrays
2411
MC Updates Stored Metadata Repository For Erased Blocks
2413
MC confirmation to host / Host updates table to show available space

FIG. 24

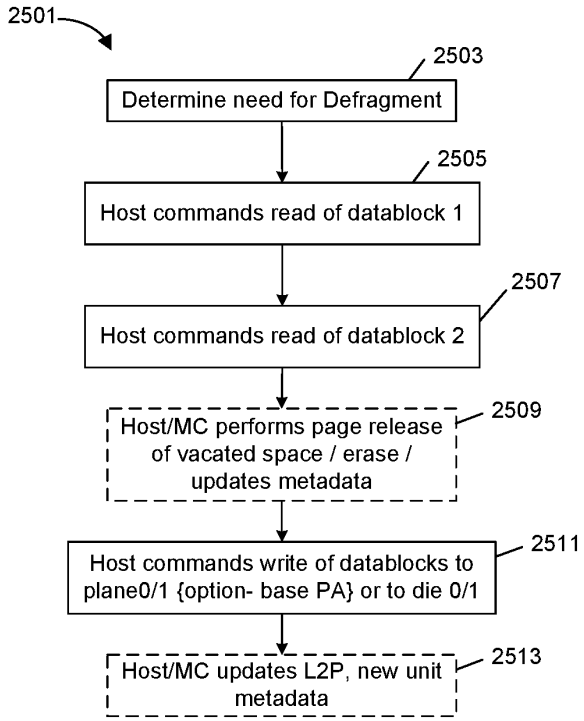

2501
2503
Determine need for Defragment
2505
Host commands read of datablock 1
2507
Host commands read of datablock 2
2509
Host/MC performs page release of vacated space / erase / updates metadata
2511
Host commands write of datablocks to plane0/1 {option- base PA} or to die 0/1
2513
Host/MC updates L2P, new unit metadata

FIG. 25

Electronic Patent Application Fee Transmittal

Application Number:	
Filing Date:	
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Filer:	Michael Wayne Farn/Nicole Walsh
Attorney Docket Number:	39471-53126/US

Filed as Small Entity

Filing Fees for Track I Prioritized Examination - Nonprovisional Application under 35 USC 111(a)

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
UTILITY FILING FEE (ELECTRONIC FILING)	4011	1	80	80
UTILITY SEARCH FEE	2111	1	350	350
UTILITY EXAMINATION FEE	2311	1	400	400
REQUEST FOR PRIORITIZED EXAMINATION	2817	1	2100	2100
Pages:				
Claims:				
CLAIMS IN EXCESS OF 20	2202	10	50	500
Miscellaneous-Filing:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
PUBL. FEE- EARLY, VOLUNTARY, OR NORMAL	1504	1	0	0
PROCESSING FEE, EXCEPT PROV. APPLS.	2830	1	70	70
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**3500**

Electronic Acknowledgement Receipt	
EFS ID:	46308419
Application Number:	17877881
International Application Number:	
Confirmation Number:	3676
Title of Invention:	Storage System with Multiplane Segments and Cooperative Flash Management
First Named Inventor/Applicant Name:	Andrey V. Kuzmin
Customer Number:	758
Filer:	Michael Wayne Farn
Filer Authorized By:	
Attorney Docket Number:	39471-53126/US
Receipt Date:	29-JUL-2022
Filing Date:	
Time Stamp:	23:57:22
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	DA
Payment was successfully received in RAM	$3500
RAM confirmation Number	E20227SN58082709
Deposit Account	192555
Authorized User	Michael Farn
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: 37 CFR 1.17 (Patent application and reexamination processing fees) 37 CFR 1.21 (Miscellaneous fees and charges)	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Application Data Sheet	53126_US_ADS.pdf	1256970 5f01e4102797571 8f12e43124f618fa44fd7eb65	no	9
Warnings:					
Information:					
2	Internet Communications Authorization	53126_US_AIC.pdf	89674 7ab281609ac6366e318c99879cf4193c1fb4c26c	no	2
Warnings:					
Information:					
3	Oath or Declaration filed	53126_US_DECLARATIONS.pdf	2565240 11f91b16c931567204ea150baf3afe0c0950b0e8	no	2
Warnings:					
Information:					
4	Track One Request	53126_US_Track_1_Request.pdf	144240 f32508d225e99962080e3a1f0c5cbd7349f0aaf9	no	2
Warnings:					
Information:					
5		53126_US_SPECIFICATION.pdf	760815 fbe0e302629cb1e9a067d9caad39fc0627f9754d	yes	88

Multipart Description/PDF files in .zip description

Document Description	Start	End
Specification	1	82
Claims	83	87
Abstract	88	88

Warnings:					
Information:					
6	Drawings-other than black and white line drawings	53126_US-Drawings.pdf	314465 88f6f400b150448bd219e511fa9d035b44288dfc	no	18
Warnings:					
Information:					
7	Fee Worksheet (SB06)	fee-info.pdf	53615 dc26211bb3d27639483b56e2ae44b0abbc8da861	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	5185019		

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 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

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 [i]	RU

Mailing Address of Inventor:

Address 1	19 Avantgarde St. #35
Address 2	

City	Moscow	State/Province	
Postal Code	125493	Country [i]	RU

Inventor 2 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	James	G.	Wayda	

Residence Information (Select One) ● US Residency ○ Non US Residency Active US Military Service

City	Laguna Niguel	State/Province	CA	Country of Residence [i]	US

Mailing Address of Inventor:

Address 1	1590 Rosecrans D417
Address 2	

City	Manhattan Beach	State/Province	CA
Postal Code	92066	Country [i]	US

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

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	00758		
Email Address	ptoc@fenwick.com	Add Email	Remove Email

Application Information:

Title of the Invention	Storage System with Multiplane Segments and Cooperative Flash Management		
Attorney Docket Number	39471-53126/US	**Small Entity Status Claimed**	☒
Application Type	Nonprovisional		
Subject Matter	Utility		
Total Number of Drawing Sheets (if any)	18	**Suggested Figure for Publication (if any)**	

Filing By Reference:

Only complete this section when filing an application by reference under 35 U.S.C. 111(c) and 37 CFR 1.57(a). Do not complete this section if application papers including a specification and any drawings are being filed. Any domestic benefit or foreign priority information must be provided in the appropriate section(s) below (i.e., "Domestic Benefit/National Stage Information" and "Foreign Priority Information").

For the purposes of a filing date under 37 CFR 1.53(b), the description and any drawings of the present application are replaced by this reference to the previously filed application, subject to conditions and requirements of 37 CFR 1.57(a).

Application number of the previously filed application	Filing date (YYYY-MM-DD)	Intellectual Property Authority or Country

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:

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	00758		

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

Domestic Benefit/National Stage Information:

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 benefit claim 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.
When referring to the current application, please leave the "Application Number" field blank.

Prior Application Status	Pending			Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)	
	Continuation of	17/563481	2021-12-28	

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
17/563481	Continuation of	16/751925	2020-01-24	11314636	2022-04-26

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
16/751925	Continuation of	16/570922	2019-09-13	11188457	2021-11-30

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
16/570922	Continuation of	15/621888	2017-06-13	10445229	2019-10-15

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
15/621888	Continuation of	15/346641	2016-11-08	9710377	2017-07-18

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
15/346641	Continuation of	14/951708	2015-11-25	9519578	2016-12-13

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
14/951708	Continuation of	14/047193	2013-10-07	9229854	2016-01-21

Prior Application Status	Patented			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
14/047193	Continuation in part of	13/767723	2013-02-14	9652376	2017-05-16

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

Prior Application Status	Expired		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
13/767723	Claims benefit of provisional	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 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)[i] 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).

			Remove
Application Number	Country[i]	Filing Date (YYYY-MM-DD)	Access Code[i] (if applicable)
Additional Foreign Priority Data may be generated within this form by selecting the **Add** button.			Add

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications

☒ This application (1) claims priority to or the benefit of an application filed before March 16, 2013 and (2) also contains, or contained at any time, a claim to a claimed invention that has an effective filing date on or after March 16, 2013.
NOTE: By providing this statement under 37 CFR 1.55 or 1.78, this application, with a filing date on or after March 16, 2013, will be examined under the first inventor to file provisions of the AIA.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

Authorization or Opt-Out of Authorization to Permit Access:

When this Application Data Sheet is properly signed and filed with the application, applicant has provided written authority to permit a participating foreign intellectual property (IP) office access to the instant application-as-filed (see paragraph A in subsection 1 below) and the European Patent Office (EPO) access to any search results from the instant application (see paragraph B in subsection 1 below).

Should applicant choose not to provide an authorization identified in subsection 1 below, applicant **must opt-out** of the authorization by checking the corresponding box A or B or both in subsection 2 below.

NOTE: This section of the Application Data Sheet is **ONLY** reviewed and processed with the **INITIAL** filing of an application. After the initial filing of an application, an Application Data Sheet cannot be used to provide or rescind authorization for access by a foreign IP office(s). Instead, Form PTO/SB/39 or PTO/SB/69 must be used as appropriate.

1. Authorization to Permit Access by a Foreign Intellectual Property Office(s)

A. Priority Document Exchange (PDX) - Unless box A in subsection 2 (opt-out of authorization) is 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 State Intellectual Property Office of the People's Republic of China (SIPO), the World Intellectual Property Organization (WIPO), and any other foreign intellectual property office participating with the USPTO in a bilateral or multilateral priority document exchange agreement in which a foreign application claiming priority to the instant patent application is filed, access to: (1) the instant patent application-as-filed and its related bibliographic data, (2) any foreign or domestic application to which priority or benefit is claimed by the instant application and its related bibliographic data, and (3) the date of filing of this Authorization. See 37 CFR 1.14(h)(1).

B. Search Results from U.S. Application to EPO - Unless box B in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the EPO access to the bibliographic data and search results from the instant patent application when a European patent application claiming priority to the instant patent application is filed. See 37 CFR 1.14(h)(2).

The applicant is reminded that the EPO's Rule 141(1) EPC (European Patent Convention) requires applicants to submit a copy of search results from the instant application without delay in a European patent application that claims priority to the instant application.

2. Opt-Out of Authorizations to Permit Access by a Foreign Intellectual Property Office(s)

☐ A. Applicant **DOES NOT** authorize the USPTO to permit a participating foreign IP office access to the instant application-as-filed. If this box is checked, the USPTO will not be providing a participating foreign IP office with any documents and information identified in subsection 1A above.

☐ B. Applicant **DOES NOT** authorize the USPTO to transmit to the EPO any search results from the instant patent application. If this box is checked, the USPTO will not be providing the EPO with search results from the instant application.

NOTE: Once the application has published or is otherwise publicly available, the USPTO may provide access to the application in accordance with 37 CFR 1.14.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

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 Remove

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.

Clear

● Assignee	Legal Representative under 35 U.S.C. 117	Joint Inventor
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 Applicant is an Organization check here. ☒

Organization Name	Radian Memory Systems, Inc.

Mailing Address Information For Applicant:

Address 1	1590 Rosecrans D417		
Address 2			
City	Manhattan Beach	**State/Province**	CA
Country	US	Postal Code	92066
Phone Number		Fax Number	
Email Address			

Additional Applicant Data may be generated within this form by selecting the Add button. Add

Assignee Information including Non-Applicant Assignee 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.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

Assignee 1

Complete this section if assignee information, including non-applicant assignee information, is desired to be included on the patent application publication. An assignee-applicant identified in the "Applicant Information" section will appear on the patent application publication as an applicant. For an assignee-applicant, complete this section only if identification as an assignee is also desired on the patent application publication.

Remove

If the Assignee or Non-Applicant Assignee is an Organization check here. ☐

Prefix	**Given Name**	Middle Name	**Family Name**	Suffix

Mailing Address Information For Assignee including Non-Applicant Assignee:

Address 1			
Address 2			
City		**State/Province**	
Country i		Postal Code	
Phone Number		Fax Number	
Email Address			

Additional Assignee or Non-Applicant Assignee Data may be generated within this form by selecting the Add button. Add

Signature:

Remove

NOTE: This Application Data Sheet must be signed in accordance with 37 CFR 1.33(b). **However, if this Application Data Sheet is submitted with the INITIAL filing of the application and either box A or B is not checked in subsection 2 of the "Authorization or Opt-Out of Authorization to Permit Access" section, then this form must also be signed in accordance with 37 CFR 1.14(c).**

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	/Michael W. Farn/			Date (YYYY-MM-DD)	2022-07-29
First Name	Michael W.	Last Name	Farn	Registration Number	41,015

Additional Signature may be generated within this form by selecting the Add button. Add

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. 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	39471-53126/US
	Application Number	
Title of Invention	Storage System with Multiplane Segments and Cooperative Flash Management	

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 CooperationTreaty.

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.

DocCode - SCORE

SCORE Placeholder Sheet for IFW Content

Application Number: 17877881 Document Date: 07/29/2022

The presence of this form in the IFW record indicates that the following document type was received in electronic format on the date identified above. This content is stored in the SCORE database.

Since this was an electronic submission, there is no physical artifact folder, no artifact folder is recorded in PALM, and no paper documents or physical media exist. The TIFF images in the IFW record were created from the original documents that are stored in SCORE.

- Drawing

At the time of document entry (noted above):

- USPTO employees may access SCORE content via DAV or via the SCORE web page.
- External customers may access SCORE content via PAIR using the Supplemental Content tab.

Form Revision Date: March 1, 2019