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	PATENT NUMBER	GROUP ART UNIT	FILE WRAPPER LOCATION
10/035,911	7154905	2616	9200

Correspondence Address / Fee Address Change

The following fields have been set to Customer Number 63170 on 07/23/2007

- Maintenance Fee Address

The address of record for Customer Number 63170 is:
SILICON IMAGE, INC.
P.O. BOX 2168
MENLO PARK,CA 94026

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	PATENT NUMBER	GROUP ART UNIT	FILE WRAPPER LOCATION
10/035,911	7154905	2616	9200

Correspondence Address / Fee Address Change

The following fields have been set to Customer Number 63170 on 06/12/2007

- Correspondence Address

The address of record for Customer Number 63170 is:
PERKINS COLE LLP
P.O. BOX 2168
MENLO PARK,CA 94026

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	PATENT NUMBER	GROUP ART UNIT	FILE WRAPPER LOCATION
10/035,911	7154905	2616	9200

Correspondence Address / Fee Address Change

The following fields have been set to Customer Number 63170 on 05/17/2007

- Correspondence Address

The address of record for Customer Number 63170 is:
PERKINS COLE LLP
P.O. BOX 2168
MENLO PARK,CA 94026

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	PATENT NUMBER	GROUP ART UNIT	FILE WRAPPER LOCATION
10/035,911	7154905	2616	9200

Correspondence Address / Fee Address Change

The following fields have been set to Customer Number 63170 on 04/11/2007

- Correspondence Address

The address of record for Customer Number 63170 is:
PERKINS COLE LLP
P.O. BOX 2168
MENLO PARK,CA 94026

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	PATENT NUMBER	GROUP ART UNIT	FILE WRAPPER LOCATION
10/035,911	7154905	2616	9200

Correspondence Address / Fee Address Change

The following fields have been set to Customer Number 63170 on 03/16/2007

- Correspondence Address

The address of record for Customer Number 63170 is:
PERKINS COLE LLP
P.O. BOX 2168
MENLO PARK,CA 94026

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.
10/035,911	12/26/2006	7154905	594728119US	2160

25096 7590 12/06/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

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

The Patent Term Adjustment is 953 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 Customer Service Center of the Office of Patent Publication at (571)-272-4200.

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

Yeshik Shin, Sunnyvale, CA;
David D. Lee, Palo Alto, CA;
Deog-Kyoon Jeong, Seoul, KOREA, REPUBLIC OF;
Shing Kong, Brisbane, CA;

OIPE
APR 17 2006
PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2002-0154633-A1	10-24-2002	SHIN ET AL.	
		US-2002-0157054-A1	10-24-2002	SHIN ET AL.	
		US-2002-0159450-A1	10-31-2002	SHIN ET AL.	
		US-2002-0159552-A1	10-31-2002	SHIN ET AL.	
		US-2002-0167945-A1	11-14-2002	SHIN ET AL.	
		US-2002-0172200-A1	11-21-2002	SHIN ET AL.	
		US-2002-0191475-A1	12-19-2002	LEE ET AL.	
		US-2002-0191603-A1	12-19-2002	SHIN ET AL.	
		US-20020035633	03/2002	BOSE ET AL.	
		US-20020041650	04/2002	RICHMOND	
		US-20020049825	04/2002	JEWETT ET AL.	
		US-20020122420	09/2002	YUAN ET AL.	
		US-20020171741	11/2002	TONKIN ET AL.	
		US-2003-0208855-A1	11-06-2003	LEE ET AL.	
		US-2003-0210162-A1	11-13-2003	SHIN ET AL.	
		US-2003-0227947-A1	12-11-2003	SHIN ET AL.	
		US-20030140300	07/2003	CHEN ET AL.	
		US-20030189922	10/2003	HOWE	
		US-2004-0004975-A1	01-08-2004	SHIN ET AL.	
		US-20050018697	01/2005	ENNS ET AL.	
		US-4,425,666	01/1984	GROTH, JR.	
		US-4,658,396	04-14-1987	BARDEN	
		US-4,672,639	06-09-1987	TANABE ET AL.	
		US-4,797,850	01-10-1989	AMITAI	
		US-4,945,548	07/1990	IANNARONE ET AL.	
		US-4,972,314	11-20-1990	GETZINGER ET AL.	
		US-5,307,314	04-26-1994	LEE	
		US-5,343,473	~~07-12-2005~~ 8/1994	CIDON ET AL.	
		US-5,400,323	03-21-1995	FRENZEL, III ET AL.	
		US-5,430,485	07-04-1995	LANKFORD ET AL.	
		US-5,497,371	03/1996	ELLIS ET AL.	
		US-5,539,696	07-23-1996	PATEL	
		US-5,557,608	09/1996	CALVIGNAC ET AL.	
		US-5,561,618	10/1996	DEHESH	
		US-5,574,849	11-12-1996	SONNIER ET AL.	
		US-5,583,872	12-10-1996	ALBRECHT ET AL.	
		US-5,587,709	12-24-1996	JEONG	
		US-5,588,025	12-24-1996	STROLLE ET AL.	
		US-5,592,487	01/1997	KNECHT ET AL.	
		US-5,663,724	09/1997	WESTBY	
		US-5,699,367	12-16-1997	HAARTSEN	
		US-5,703,875	12-30-1997	BURNETT	
		US-5,751,947	05/1998	ARAKAWA	
		US-5,825,824	10-20-1998	LEE ET AL.	
		US-5,838,960	11/1998	HARRIMAN, JR.	

08 11/9/06

Examiner Signature		Date Considered	9/13/06

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.
10/035,911	11/07/2001	Yeshik Shin	594728119US	2160

25096 7590 10/17/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

EXAMINER	
BLOUNT, STEVEN	
ART UNIT	**PAPER NUMBER**
2616	

DATE MAILED: 10/17/2006

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

Supplemental Notice of Allowability	Application No. 10/035,911	Applicant(s) SHIN ET AL.	
	Examiner Steven Blount	Art Unit 2616	

-- 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 _____.

2. ☒ The allowed claim(s) is/are 1 - 2, 4 - 10, 12 - 18, 20 - 25, 27 - 30.

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

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ 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. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 4/17/06
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

DORIS H. TO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600

IPR2024-001428
HP – EX1002, Page 9 of 487

OIPE IAP84
APR 17 2006
PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2002-0154633-A1	10-24-2002	SHIN ET AL.	
		US-2002-0157054-A1	10-24-2002	SHIN ET AL.	
		US-2002-0159450-A1	10-31-2002	SHIN ET AL.	
		US-2002-0159552-A1	10-31-2002	SHIN ET AL.	
		US-2002-0167945-A1	11-14-2002	SHIN ET AL.	
		US-2002-0172200-A1	11-21-2002	SHIN ET AL.	
		US-2002-0191475-A1	12-19-2002	LEE ET AL.	
		US-2002-0191603-A1	12-19-2002	SHIN ET AL.	
		US-20020035633	03/2002	BOSE ET AL.	
		US-20020041650	04/2002	RICHMOND	
		US-20020049825	04/2002	JEWETT ET AL.	
		US-20020122420	09/2002	YUAN ET AL.	
		US-20020171741	11/2002	TONKIN ET AL.	
		US-2003-0208655-A1	11-06-2003	LEE ET AL.	
		US-2003-0210162-A1	11-13-2003	SHIN ET AL.	
		US-2003-0227947-A1	12-11-2003	SHIN ET AL.	
		US-20030140300	07/2003	CHEN ET AL.	
		US-20030189922	10/2003	HOWE	
		US-2004-0004975-A1	01-08-2004	SHIN ET AL.	
		US-20050018697	01/2005	ENNS ET AL.	
		US-4,425,666	01/1984	GROTH, JR.	
		US-4,658,396	04-14-1987	BARDEN	
		US-4,672,639	06-09-1987	TANABE ET AL.	
		US-4,797,850	01-10-1989	AMITAI	
		US-4,945,548	07/1990	IANNARONE ET AL.	
		US-4,972,314	11-20-1990	GETZINGER ET AL.	
		US-5,307,314	04-26-1994	LEE	
		US-5,343,473	07-12-2005	CIDON ET AL.	
		US-5,400,323	03-21-1995	FRENZEL, III ET AL.	
		US-5,430,485	07-04-1995	LANKFORD ET AL.	
		US-5,497,371	03/1996	ELLIS ET AL.	
		US-5,539,696	07-23-1996	PATEL	
		US-5,557,608	09/1996	CALVIGNAC ET AL.	
		US-5,561,618	10/1996	DEHESH	
		US-5,574,849	11-12-1996	SONNIER ET AL.	
		US-5,583,872	12-10-1996	ALBRECHT ET AL.	
		US-5,587,709	12-24-1996	JEONG	
		US-5,588,025	12-24-1996	STROLLE ET AL.	
		US-5,592,487	01/1997	KNECHT ET AL.	
		US-5,663,724	09/1997	WESTBY	
		US-5,699,367	12-16-1997	HAARTSEN	
		US-5,703,875	12-30-1997	BURNETT	
		US-5,751,947	05/1998	ARAKAWA	
		US-5,825,824	10-20-1998	LEE ET AL.	
		US-5,838,960	11/1998	HARRIMAN, JR.	

Examiner Signature		Date Considered	9/13/06

OIPE IAP&4 APR 17 2006 PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 2 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

		US-5,872,777	02/1999	BRAILEAN ET AL.	
		US-5,963,703	10/1999	HIGURASHI ET AL.	
		US-5,974,464	10-26-1999	SHIN ET AL.	
		US-5,983,287	11-09-1999	KUROIWA ET AL.	
		US-5,999,571	12-07-1999	SHIN ET AL.	
		US-6,016,510	01-18-2000	QUATTROMANI ET AL.	
		US-6,026,124	02-15-2000	LEE ET AL.	
		US-6,067,301	05/2000	AATRESH	
		US-6,151,334	11-21-2000	KIM ET AL.	
		US-6,154,464	11/2000	FEUERSTRAETER ET AL.	
		US-6,188,337	02/2001	SOLJANIN	
		US-6,188,670	02/2001	LACKMAN ET AL.	
		US-6,198,413	03/2001	WIDMER	
		US-6,229,859	05-08-2001	JEONG ET AL.	
		US-6,266,335	07/2001	BHASKARAN	
		US-6,400,730	06-04-2002	LATIF ET AL.	
		US-6,415,353	07-02-2002	LEUNG	
		US-6,430,230	08/2002	CUNNINGHAM ET AL.	
		US-6,438,128	08-20-2002	KASHYAP	
		US-6,470,391	10/2002	TAKAMOTO ET AL.	
		US-6,487,171	11/2002	HONIG ET AL.	
		US-6,560,290-A1	05-06-2003	AHN ET AL.	
		US-6,560,728	05-06-2003	MERRITT	
		US-6,587,525-A1	07-01-2003	JEONG ET AL.	
		US-6,594,325	07/2003	LI	
		US-6,594,329	07/2003	SUSNOW	
		US-6,631,132	10/2003	SOURANI	
		US-6,738,821	05/2004	WILSON ET AL.	
		US-6,771,192	08-03-2004	LEE ET AL.	
		US-6,791,952	09/2004	LIN ET AL.	
		US-6,826,199	11-30-2004	DAVIDSSON ET AL.	
		US-6,891,845	05/2005	BOOCK ET AL.	

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3]-Number[4]-Kind Code[5] *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
		WO-97/13347-A2	04-10-1997	Silicon Image, Inc.		
		WO-99/12306-A1	03-11-1999	Silicon Image, Inc.		
		WO-97/36408-A2	10-02-1997	Silicon Image, Inc.		
		WO-02/058318-A1	07-25-2002	Silicon Image, Inc.		
		EP-0582537A2	09-02-1994	International Business Machines Corporation		

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. * CITE NO.: Those application(s) which are marked with an single asterisk (*) next to the Cite No. are not supplied (under 37 CFR 1.98(a)(2)(iii)) because that application was filed after June 30, 2003 or is available in the IFW. [1] Applicant's unique citation designation number (optional). [2] See Kinds 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.

Examiner Signature		Date Considered	9/13/06

OIPE
APR 17 2006
PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet	3	of	3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

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[2]
SB		Richmond, Serial protocol elastic buffer description, 07/11/00, 60/217520, Pages 1-13	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through 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 (optional). [2]Applicant is to place a check mark here if English language Translation is attached.

Examiner Signature	SB	Date Considered	9/13/06

09-18-06

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Express Mail No. EV 813996467 US

or Fax **(571)-273-2885**

PTO/SB/31 (08-06)
Approved for use through 06/30/2006. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
SEP 15 2006

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)

25096 7590 06/22/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

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 facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

Carl-Erich Kruse	(Depositor's name)
	(Signature)
9/15/06	(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/035,911	11/07/2001	Yeshik Shin	~~3917281190S~~ 5947288190S	2160

TITLE OF INVENTION: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$1700	09/22/2006

EXAMINER	ART UNIT	CLASS-SUBCLASS
BLOUNT, STEVEN	2616	370-389000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
- ☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
- ☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 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 Perkins Coie 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 has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE: Silicon Image

(B) RESIDENCE: (CITY and STATE OR COUNTRY): Sunnyvale, CA

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

4a. The following fee(s) are enclosed:
- ☒ Issue Fee
- ☒ Publication Fee (No small entity discount permitted)
- ☒ Advance Order - # of Copies 2

4b. Payment of Fee(s):
- ☒ A check in the amount of the fee(s) is enclosed.
- ☐ Payment by credit card. Form PTO-2038 is attached.
- ☒ The Director is hereby authorized by charge ~~the required fee(s)~~, or credit any overpayment, to Deposit Account Number 50-0665 (enclose an extra copy of this form). any deficiency

5. **Change in Entity Status** (from status indicated above)
- ☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.
- ☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

The Director of the USPTO is requested to apply the Issue Fee and Publication Fee (if any) or to re-apply any previously paid issue fee to the application identified above.
NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature Maurice J. Pirio Date 9-15-06

Typed or printed name Maurice J. Pirio Registration No. 33,273

This collection of information 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 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, 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.

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

25096 7590 06/22/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

EXAMINER	
BLOUNT, STEVEN	
ART UNIT	PAPER NUMBER
2616	

DATE MAILED: 06/22/2006

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/035,911	11/07/2001	Yeshik Shin	594728119US	2160

TITLE OF INVENTION: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$1700	09/22/2006

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 REFLECTS A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE APPLIED IN THIS APPLICATION. THE PTOL-85B (OR AN EQUIVALENT) MUST BE RETURNED WITHIN THIS PERIOD EVEN IF NO FEE IS DUE OR THE APPLICATION WILL BE REGARDED AS ABANDONED.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL should be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). Even if the fee(s) have already been paid, Part B - Fee(s) Transmittal should be completed and returned. If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted.

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: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 01/06) Approved for use through 04/30/2007.

IPR2024-001428
HP – EX1002, Page 14 of 487

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (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)

25096 7590 06/22/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

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 facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/035,911	11/07/2001	Yeshik Shin	594728119US	2160

TITLE OF INVENTION: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$1700	09/22/2006

EXAMINER	ART UNIT	CLASS-SUBCLASS
BLOUNT, STEVEN	2616	370-389000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 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 has been filed for recordation as set forth in 37 CFR 3.11. 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. The following fee(s) are enclosed:
☐ Issue Fee
☐ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s):
☐ A check in the amount of the fee(s) is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☐ The Director is hereby authorized by charge the required fee(s), or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)
☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27. ☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

The Director of the USPTO is requested to apply the Issue Fee and Publication Fee (if any) or to re-apply any previously paid issue fee to the application identified above.
NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information 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 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, 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.

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.
10/035,911	11/07/2001	Yeshik Shin	594728119US	2160

25096 7590 06/22/2006

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

EXAMINER	
BLOUNT, STEVEN	
ART UNIT	PAPER NUMBER
2616	

DATE MAILED: 06/22/2006

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

The Patent Term Adjustment to date is 953 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 953 day(s).

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 Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

PTOL-85 (Rev. 01/06) Approved for use through 04/30/2007.

IPR2024-001428
HP – EX1002, Page 16 of 487

Notice of Allowability	Application No.	Applicant(s)	
	10/035,911	SHIN ET AL.	
	Examiner	Art Unit	
	Steven Blount	2616	

-- 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 *Amendment filed 4/17/06*.

2. ☒ The allowed claim(s) is/are *1 - 2, 4 - 10,12 - 18, 20 - 25, 27 - 30*.

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

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ 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. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO-1449 or PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application (PTO-152)
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

DORIS H. TO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600

Issue Classification	Application/Control No. 10/035,911	Applicant(s)/Patent under Reexamination SHIN ET AL.
	Examiner Steven Blount	Art Unit 2616

ISSUE CLASSIFICATION

ORIGINAL		INTERNATIONAL CLASSIFICATION									
CLASS	SUBCLASS	CLAIMED					NON-CLAIMED				
370	465	H	04	J	15	/00				/	
CROSS REFERENCES						/				/	
CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)										
370	468					/				/	
						/				/	
						/				/	
						/				/	
						/				/	

Steve Blount
(Assistant Examiner) (Date)

John Butcher 6/12/06
(Legal Instruments Examiner) (Date)

DORIS H. TO
SUPERVISORY PATENT EXAMINER
(Primary Examiner) (Date)
TECHNOLOGY CENTER 2600

Total Claims Allowed: 26

O.G. Print Claim(s)	O.G. Print Fig.
1	29

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

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
1	1		31		61		91		121		151		181		
2	2		32		62		92		122		152		182		
	3		33		63		93		123		153		183		
3	4		34		64		94		124		154		184		
4	5		35		65		95		125		155		185		
5	6		36		66		96		126		156		186		
6	7		37		67		97		127		157		187		
7	8		38		68		98		128		158		188		
8	9		39		69		99		129		159		189		
9	10		40		70		100		130		160		190		
	11		41		71		101		131		161		191		
10	12		42		72		102		132		162		192		
11	13		43		73		103		133		163		193		
12	14		44		74		104		134		164		194		
13	15		45		75		105		135		165		195		
14	16		46		76		106		136		166		196		
15	17		47		77		107		137		167		197		
16	18		48		78		108		138		168		198		
	19		49		79		109		139		169		199		
17	20		50		80		110		140		170		200		
18	21		51		81		111		141		171		201		
19	22		52		82		112		142		172		202		
20	23		53		83		113		143		173		203		
21	24		54		84		114		144		174		204		
22	25		55		85		115		145		175		205		
	26		56		86		116		146		176		206		
23	27		57		87		117		147		177		207		
24	28		58		88		118		148		178		208		
25	29		59		89		119		149		179		209		
26	30		60		90		120		150		180		210		

Index of Claims

Application/Control No.	Applicant(s)/Patent under Reexamination
10/035,911	SHIN ET AL.
Examiner	**Art Unit**
Steven Blount	2616

√	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

Claim Final	Claim Original	Date
1	1	
2	2	
	3	
3	4	
4	5	
5	6	
6	7	
7	8	
8	9	
9	10	
	11	
10	12	
11	13	
12	14	
13	15	
14	16	
15	17	
16	18	
17	19	
18	20	
19	21	
20	22	
21	23	
22	24	
	25	
23	26	
24	27	
25	28	
26	29	
	30	
	31	
	32	
	33	
	34	
	35	
	36	
	37	
	38	
	39	
	40	
	41	
	42	
	43	
	44	
	45	
	46	
	47	
	48	
	49	
	50	

Claim Final	Claim Original	Date
	51	
	52	
	53	
	54	
	55	
	56	
	57	
	58	
	59	
	60	
	61	
	62	
	63	
	64	
	65	
	66	
	67	
	68	
	69	
	70	
	71	
	72	
	73	
	74	
	75	
	76	
	77	
	78	
	79	
	80	
	81	
	82	
	83	
	84	
	85	
	86	
	87	
	88	
	89	
	90	
	91	
	92	
	93	
	94	
	95	
	96	
	97	
	98	
	99	
	100	

Claim Final	Claim Original	Date
	101	
	102	
	103	
	104	
	105	
	106	
	107	
	108	
	109	
	110	
	111	
	112	
	113	
	114	
	115	
	116	
	117	
	118	
	119	
	120	
	121	
	122	
	123	
	124	
	125	
	126	
	127	
	128	
	129	
	130	
	131	
	132	
	133	
	134	
	135	
	136	
	137	
	138	
	139	
	140	
	141	
	142	
	143	
	144	
	145	
	146	
	147	
	148	
	149	
	150	

Search Notes

Application/Control No.	Applicant(s)/Patent under Reexamination	
10/035,911	SHIN ET AL.	
Examiner	**Art Unit**	
Steven Blount	2668	

SEARCHED			
Class	Subclass	Date	Examiner
370	465	11/2/05	SB
↓	468	↓	↓
↓	473	↓	↓
370	493	11/4/05	SB
East report		6/7/06	SB
for Mtg. in file			

INTERFERENCE SEARCHED			
Class	Subclass	Date	Examiner
See East		6/7/06	SB
Report			

SEARCH NOTES (INCLUDING SEARCH STRATEGY)		
	DATE	EXMR

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1	(out-of-band and packet and continue).clm.	USPAT	OR	OFF	2006/06/07 22:15

IPR2024-001428
HP – EX1002, Page 21 of 487

OIPE IAP&4 APR 17 2006 PATENT & TRADEMARK OFFICE

04-18-06

2661

AMENDMENT TRANSMITTAL LETTER

Docket No. 594728819US

Application No.	Filing Date	Examiner	Art Unit
10/035,911-Conf. #2160	November 7, 2001	S. Blount	2661

Applicant(s): Shin et al.

Invention: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

TO THE COMMISSIONER FOR PATENTS

Transmitted herewith is an amendment in the above-identified application.

The fee has been calculated and is transmitted as shown below.

CLAIMS AS AMENDED

	Claims Remaining After Amendment	Highest Number Previously Paid	Number Extra Claims Present	Rate	
Total Claims	26	- 30 =		x	
Independent Claims	4	- 4 =		x	
Multiple Dependent Claims (check if applicable)		☐			
Other fee (please specify):	IDS Submission 1.17(p) Petition for Extension of Time 1.136(a)				180.00 450.00
TOTAL ADDITIONAL FEE FOR THIS AMENDMENT:					630.00

☒ Large Entity ☐ Small Entity

☐ No additional fee is required for this amendment.

☐ Please charge Deposit Account No. _____ in the amount of $ _____.
A duplicate copy of this sheet is enclosed.

☒ A check in the amount of $ 630.00 to cover the filing fee is enclosed.

☐ Payment by credit card. Form PTO-2038 is attached.

☒ The Director is hereby authorized to charge and credit Deposit Account No. 50-0665 as described below. A duplicate copy of this sheet is enclosed.

☒ Credit any overpayment.

☒ Charge any additional filing or application processing fees required under 37 CFR 1.16 and 1.17.

Maurice J. Pirio

Dated: April 17, 2006

Maurice J. Pirio
Attorney/Agent Reg. No.: 33,273

PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 359-8000

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV 853118105US, on the date shown below in an envelope addressed to:
MS Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

Dated: 4/17/06 Signature: _____ (Attila Kovacs-Szabo)

OIPE IAP84 APR 17 2006 PATENT & TRADEMARK OFFICE

Express Mail No. EV 853118105US

Docket No.: 594728819US
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Shin et al.

Application No.: 10/035,911 Confirmation No.: 2160

Filed: November 7, 2001 Art Unit: 2668

For: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS Examiner: S. Blount

AMENDMENT IN RESPONSE TO NON-FINAL OFFICE ACTION

MS Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

INTRODUCTORY COMMENTS

In response to the Office Action dated November 15, 2005, please amend the above-identified U.S. patent application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 7 of this paper.

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method for transmitting packets, the method comprising:

receiving a first packet;

transmitting a portion of the received first packet;

receiving a second packet;

upon receiving the second packet,

stopping the transmitting of the first packet so that not all of the first packet has been transmitted;

transmitting a preempt indicator;

transmitting the second packet; and

upon completion of transmitting the second packet,

transmitting a continue indicator; and

transmitting the portion of the first packet that has not yet been transmitted;

wherein the packets include in-band symbols and the indicators include one or more out-of-band symbols.

2. (Original) The method of claim 1 wherein the transmitted indicators are primitives.

3. (Canceled)

4. (Currently Amended) The method of claim 1~~3~~ wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

5. (Original) The method of claim 1 wherein the first packet is a data packet and the second packet is a control packet.

6. (Original) The method of claim 1 wherein the first and second packets are transmitted through the same communications link.

7. (Original) The method of claim 1 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

8. (Original) The method of claim 1 wherein the first packet includes a header that is transmitted only once.

9. (Currently Amended) A method for receiving packets, the method comprising:

receiving a first portion of symbols of a first packet;

receiving a preempt indicator indicating that a second packet of symbols is to be received;

receiving the second packet of symbols;

receiving a continue indicator indicating that a second portion of symbols of the first packet is to be received; and

receiving the second portion of symbols of the first packet;

wherein the symbols of the packets include in-band symbols and the indicators include one or more out-of-band symbols.

10. (Original) The method of claim 9 wherein the received indicators are primitives.

11. (Canceled)

12. (Currently Amended) The method of claim ~~9~~11 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

13. (Original) The method of claim 9 wherein the first packet is a data packet and the second packet is a control packet.

14. (Original) The method of claim 9 wherein the first and second packets are received via the same communications link.

15. (Original) The method of claim 9 wherein the receiving of the second packet is preempted so that a third packet can be received.

16. (Original) The method of claim 9 wherein the first packet includes a header that is received only once.

17. (Currently Amended) A method for transmitting packets via a communications link, the method comprising:

transmitting a first portion of a first packet;

transmitting a preempt indicator indicating that a second packet is to be transmitted;

transmitting the second packet;

transmitting a continue indicator indicating that a second portion of the first packet is to be transmitted; and

transmitting the second portion of the first packet

wherein the first and second packets and the preempt and continue indicators are transmitted via the same communications link and wherein further the packets include in-band symbols and the indicators include one or more out-of-band symbols.

18. (Original) The method of claim 17 wherein the indicators are primitives.

19. (Canceled)

20. (Currently Amended) The method of claim 17~~19~~ wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

21. (Original) The method of claim 17 wherein the first packet is a data packet and the second packet is a control packet.

22. (Original) The method of claim 17 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

23. (Original) The method of claim 17 wherein the first packet includes a header that is transmitted only once.

24. (Currently Amended) A communications device for transmitting packets via a communications link, comprising:

a transmission component that transmits a first packet; and

a preemption component that signals the transmission component to stop transmitting the first packet, transmits a preempt indicator indicating that a second packet is to be transmitted, transmits the second packet, and signals the transmission component to continue transmitting the first packet;

wherein packets include in-band symbols and the indicators include one or more out-of-band symbols.

25. (Original) The communications device of claim 24 wherein the indicators are primitives.

26. (Canceled)

27. (Currently Amended) The communications device of claim 2426 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

28. (Original) The communications device of claim 24 wherein the first packet is a data packet and the second packet is a control packet.

29. (Original) The communications device of claim 24 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

30. (Original) The communications device of claim 24 wherein the first packet includes a header that is transmitted only once.

REMARKS

Claims 1-2, 4-10, 12-18, 20-25, and 27-30 are pending. Claims 1, 9, 17, and 24 have been amended to incorporate subject matter deemed allowable by the Examiner.

In view of the above amendment, applicant believes the pending application is in condition for allowance and respectfully requests its early allowance.

Applicant submits herewith a Petition for extension of time and the requisite fees. Applicant believes no additional fee is due with this response. However, if a fee is due, please charge our Deposit Account No. 50-0665, under Order No. 594728819US from which the undersigned is authorized to draw.

Dated: 4/17/06

Respectfully submitted,

By Maurice J Pirio

Maurice J. Pirio
Registration No.: 33,273
PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 359-8000
(206) 359-7198 (Fax)
Attorney for Applicant

OIPE IAP84 APR 17 2006 PATENT & TRADEMARK OFFICE

PTO/SB/22 (12-04)
Approved for use through 7/31/2006. 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 if displays a valid OMB control number.

PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a) FY 2005 (Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).)	Docket Number (Optional) 594728819US
Application Number 10/035,911-Conf. #2160	Filed November 7, 2001
For METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS	
Art Unit 2661	Examiner S. Blount

This is a request under the provisions of 37 CFR 1.136(a) to extend the period for filing a reply in the above identified application.

The requested extension and fee are as follows (check time period desired and enter the appropriate fee below):

		Fee	Small Entity Fee	
[ ]	One month (37 CFR 1.17(a)(1))	$120	$60	$
[X]	Two months (37 CFR 1.17(a)(2))	$450	$225	$ 450.00
[ ]	Three months (37 CFR 1.17(a)(3))	$1020	$510	$
[ ]	Four months (37 CFR 1.17(a)(4))	$1590	$795	$
[ ]	Five months (37 CFR 1.17(a)(5))	$2160	$1080	$

[ ] Applicant claims small entity status. See 37 CFR 1.27.

[X] A check in the amount of the fee is enclosed.

[ ] Payment by credit card. Form PTO-2038 is attached.

[ ] The Director has already been authorized to charge fees in this application to a Deposit Account.

[X] The Director is hereby authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account Number 50-0665 . I have enclosed a duplicate copy of this sheet.

I am the

[ ] applicant/inventor.

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

[X] attorney or agent of record. Registration Number 33,273

[ ] attorney or agent under 37 CFR 1.34. Registration number if acting under 37 CFR 1.34 _____ .

Maurice J Pirio
Signature

April 17, 2006
Date

Maurice J. Pirio
Typed or printed name

(206) 359-8000
Telephone Number

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

[ ] Total of 1 forms are submitted.

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV 853118105US, on the date shown below in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Dated: 4/17/06 Signature: (Attila Kovacs-Szabo)

04/19/2006 WASFAHI1 00000070 10035911 450.00 OP
01 FC:1252

IPR2024-001428
HP – EX1002, Page 30 of 487

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV 853118105US, on the date shown below in an envelope addressed to: MS Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Dated: 4/17/06 Signature: (Attila Kovacs-Szabo)

Docket No.: 594728819US
(PATENT)

OIPE IAP64 APR 17 2006 PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Shin et al.

Application No.: 10/035,911 Confirmation No.: 2160

Filed: November 7, 2001 Art Unit: 2661

For: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS Examiner: S. Blount

INFORMATION DISCLOSURE STATEMENT (IDS)

MS Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Pursuant to 37 CFR 1.56, 1.97 and 1.98, the attention of the Patent and Trademark Office is hereby directed to the references listed on the attached PTO/SB/08. It is respectfully requested that the information be expressly considered during the prosecution of this application, and that the references be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

This Information Disclosure Statement is filed more than three months after the U.S. filing date, OR more than three months after the date of entry of the national stage of a PCT application, AND after the mailing date of the first Office Action on the merits, whichever occurs first, but before the mailing date of a Final Office Action or Notice of Allowance (37 CFR 1.97(c)).

Applicant has not submitted copies of each cited U.S. patent and U.S. patent application as required by 37 CFR 1.98(a)(2)(i), amended October 2004, as the U.S.

04/19/2006 WASFAW1 00000070 10035911
02 FC:1806 180.00 OP

Patent and Trademark Office has waived this requirement for all U.S. patent applications. Applicant submits herewith copies of foreign and non-patents in accordance with 37 CFR 1.98(a)(2).

Our check in the amount of $180.00 covering the fee set forth in 37 CFR 1.17(p) is enclosed. The Director is hereby authorized to charge any deficiency in the fees filed, asserted to be filed or which should have been filed herewith (or with any paper hereafter filed in this application by this firm) to our Deposit Account No. 50-0665, under Order No. 594728819US.

Dated: April 17, 2006

Respectfully submitted,

By Maurice J Pirio

Maurice J. Pirio
Registration No.: 33,273
PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 359-8000
(206) 359-7198 (Fax)
Attorney for Applicant

IPR2024-001428
HP – EX1002, Page 32 of 487

OIPE IAP84
APR 17 2006
PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (*if known*)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2002-0154633-A1	10-24-2002	SHIN ET AL.	
		US-2002-0157054-A1	10-24-2002	SHIN ET AL.	
		US-2002-0159450-A1	10-31-2002	SHIN ET AL.	
		US-2002-0159552-A1	10-31-2002	SHIN ET AL.	
		US-2002-0167945-A1	11-14-2002	SHIN ET AL.	
		US-2002-0172200-A1	11-21-2002	SHIN ET AL.	
		US-2002-0191475-A1	12-19-2002	LEE ET AL.	
		US-2002-0191603-A1	12-19-2002	SHIN ET AL.	
		US-20020035633	03/2002	BOSE ET AL.	
		US-20020041650	04/2002	RICHMOND	
		US-20020049825	04/2002	JEWETT ET AL.	
		US-20020122420	09/2002	YUAN ET AL.	
		US-20020171741	11/2002	TONKIN ET AL.	
		US-2003-0208655-A1	11-06-2003	LEE ET AL.	
		US-2003-0210162-A1	11-13-2003	SHIN ET AL.	
		US-2003-0227947-A1	12-11-2003	SHIN ET AL.	
		US-20030140300	07/2003	CHEN ET AL.	
		US-20030189922	10/2003	HOWE	
		US-2004-0004975-A1	01-08-2004	SHIN ET AL.	
		US-20050018697	01/2005	ENNS ET AL.	
		US-4,425,666	01/1984	GROTH, JR.	
		US-4,658,396	04-14-1987	BARDEN	
		US-4,672,639	06-09-1987	TANABE ET AL.	
		US-4,797,850	01-10-1989	AMITAI	
		US-4,945,548	07/1990	IANNARONE ET AL.	
		US-4,972,314	11-20-1990	GETZINGER ET AL.	
		US-5,307,314	04-26-1994	LEE	
		US-5,343,473	07-12-2005	CIDON ET AL.	
		US-5,400,323	03-21-1995	FRENZEL, III ET AL.	
		US-5,430,485	07-04-1995	LANKFORD ET AL.	
		US-5,497,371	03/1996	ELLIS ET AL.	
		US-5,539,696	07-23-1996	PATEL	
		US-5,557,608	09/1996	CALVIGNAC ET AL.	
		US-5,561,618	10/1996	DEHESH	
		US-5,574,849	11-12-1996	SONNIER ET AL.	
		US-5,583,872	12-10-1996	ALBRECHT ET AL.	
		US-5,587,709	12-24-1996	JEONG	
		US-5,588,025	12-24-1996	STROLLE ET AL.	
		US-5,592,487	01/1997	KNECHT ET AL.	
		US-5,663,724	09/1997	WESTBY	
		US-5,699,367	12-16-1997	HAARTSEN	
		US-5,703,875	12-30-1997	BURNETT	
		US-5,751,947	05/1998	ARAKAWA	
		US-5,825,824	10-20-1998	LEE ET AL.	
		US-5,838,960	11/1998	HARRIMAN, JR.	

Examiner Signature		Date Considered	

OIPE IAP84 APR 17 2006 PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 2 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

		US-5,872,777	02/1999	BRAILEAN ET AL.	
		US-5,963,703	10/1999	HIGURASHI ET AL.	
		US-5,974,464	10-26-1999	SHIN ET AL.	
		US-5,983,287	11-09-1999	KUROIWA ET AL.	
		US-5,999,571	12-07-1999	SHIN ET AL.	
		US-6,016,510	01-18-2000	QUATTROMANI ET AL.	
		US-6,026,124	02-15-2000	LEE ET AL.	
		US-6,067,301	05/2000	AATRESH	
		US-6,151,334	11-21-2000	KIM ET AL.	
		US-6,154,464	11/2000	FEUERSTRAETER ET AL.	
		US-6,188,337	02/2001	SOLJANIN	
		US-6,188,670	02/2001	LACKMAN ET AL.	
		US-6,198,413	03/2001	WIDMER	
		US-6,229,859	05-08-2001	JEONG ET AL.	
		US-6,266,335	07/2001	BHASKARAN	
		US-6,400,730	06-04-2002	LATIF ET AL.	
		US-6,415,353	07-02-2002	LEUNG	
		US-6,430,230	08/2002	CUNNINGHAM ET AL.	
		US-6,438,128	08-20-2002	KASHYAP	
		US-6,470,391	10/2002	TAKAMOTO ET AL.	
		US-6,487,171	11/2002	HONIG ET AL.	
		US-6,560,290-A1	05-06-2003	AHN ET AL.	
		US-6,560,728	05-06-2003	MERRITT	
		US-6,587,525-A1	07-01-2003	JEONG ET AL.	
		US-6,594,325	07/2003	LI	
		US-6,594,329	07/2003	SUSNOW	
		US-6,631,132	10/2003	SOURANI	
		US-6,738,821	05/2004	WILSON ET AL.	
		US-6,771,192	08-03-2004	LEE ET AL.	
		US-6,791,952	09/2004	LIN ET AL.	
		US-6,826,199	11-30-2004	DAVIDSSON ET AL.	
		US-6,891,845	05/2005	BOOCK ET AL.	

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document: Country Code[3]-Number[4]-Kind Code[5] *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
		WO-97/13347-A2	04-10-1997	Silicon Image, Inc.		
		WO-99/12306-A1	03-11-1999	Silicon Image, Inc.		
		WO-97/36408-A2	10-02-1997	Silicon Image, Inc.		
		WO-02/058318-A1	07-25-2002	Silicon Image, Inc.		
		EP-0582537A2	09-02-1994	International Business Machines Corporation		

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. * CITE NO.: Those application(s) which are marked with an single asterisk (*) next to the Cite No. are not supplied (under 37 CFR 1.98(a)(2)(iii)) because that application was filed after June 30, 2003 or is available in the IFW. [1] Applicant's unique citation designation number (optional). [2] See Kinds 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.

Examiner Signature		Date Considered	

OIPE IAP84 APR 17 2006 PATENT & TRADEMARK OFFICE

PTO/SB/08a/b (07-05)
Approved for use through 07/31/2006. 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.

Substitute for form 1449A/B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 3 of 3

Complete if Known	
Application Number	10/035,911-Conf. #2160
Filing Date	November 7, 2001
First Named Inventor	Yeshik Shin
Art Unit	2661
Examiner Name	S. Blount
Attorney Docket Number	594728819US

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[2]
		Richmond, Serial protocol elastic buffer description, 07/11/00, 60/217520, Pages 1-13	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through 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 (optional). [2]Applicant is to place a check mark here if English language Translation is attached.

Examiner Signature		Date Considered	

BEST AVAILABLE COPY

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 25/49, H03M 5/14	A2	(11) International Publication Number: WO 97/13347
		(43) International Publication Date: 10 April 1997 (10.04.97)

(21) International Application Number: PCT/US96/15661

(22) International Filing Date: 30 September 1996 (30.09.96)

(30) Priority Data:
08/539,816 5 October 1995 (05.10.95) US

(71) Applicant: SILICON IMAGE, INC. [US/US]; Suite 222, 1032 Elwell Court, Palo Alto, CA 94303 (US).

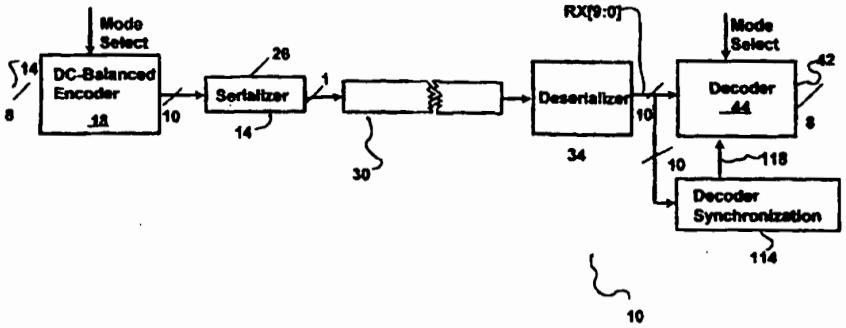
(72) Inventors: SHIN, Yeshik; 244-303, Sangdo 4 dong, 16/3 Dongjak-ku, Seoul 156-034 (KR). LEE, Kyeongho; 93-102, Sangsu-dong Mapo Gu, Seoul (KR). KIM, Sungjoon; 5-1107 Mujigaeo Apartment, Seocho-dong, Seocho-gu, Seoul (KR). LEE, David, D.; 3715 Redwood Circle, Palo Alto, CA 94306 (US).

(74) Agents: OKAMOTO, James, K. et al.; Fenwick & West L.L.P., Suite 700, Two Palo Alto Square, Palo Alto, CA 94306 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: TRANSITION-CONTROLLED DIGITAL ENCODING AND SIGNAL TRANSMISSION SYSTEM



(57) Abstract

A method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes is disclosed herein. The bits in each of the data bytes are selectively complemented in accordance with the number of logical transitions in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity. In a high-transition mode of operation, the bits within data blocks including less than a minimum number of logical transitions are selectively complemented so that each such selectively complemented data block includes in excess of the minimum number of logical transitions. In a low-transition mode of operation, the bits within data blocks having more than a predefined number of logical transitions are selectively complemented so that each such selectively complemented data block includes less than the maximum number of logical transitions.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

TRANSITION-CONTROLLED DIGITAL ENCODING AND SIGNAL TRANSMISSION SYSTEM

BACKGROUND OF THE INVENTION

I. Field of the Invention

This invention relates to coding schemes for digital transmission systems. More particularly, the present invention relates to a DC-balanced, transition-controlled coding system in which rapid byte synchronization allows for prompt initiation of decoding.

II. Description of the Related Art

As electronic and computer technology continues to evolve, communication of information among different devices, either situated near by or at a distance becomes increasingly important. For example, it is now more desirable than ever to provide for high speed communications among different chips on a circuit board, different circuit boards in a system, and different systems with each other. It is also increasingly desirable to provide such communications at very high speeds, especially in view of the large amount of data required for data communications in intensive data consuming systems using graphical or video information, multiple input-output channels, local area networks, and the like.

It is particularly desirable to enable individual personal computers, workstations, or other computing devices, within which data is normally internally transferred using parallel data buses, to communicate with each other over relatively simple transmission lines. Such transmission lines typically include only one or two conductors, in contrast with the 64-bit and wider data paths within computing

IPR2024-001428
HP – EX1002, Page 38 of 487

systems now commonly available. In the case of video data transmission to computer displays, as well as in the case of high-speed video input from digital cameras to computer systems, existing interconnection interfaces typically employ such parallel data paths. Recently, the requisite bandwidth of such interconnection systems has increased as a consequence of increased display resolution. This has increased electromagnetic interference (EMI) as well as transmission line noise, thereby raising concerns as to safety and reliability. In addition, the large number of signal and data lines required by advanced liquid crystal display panels has increased the potential for mutual interference.

There have been a number of commercially available products which attempt to provide high speed conversion of parallel data to serial form and transmission over a serial link. The Hewlett-Packard G-link chip set is one such product. That chip set includes a transmitter set and is capable of handling 21-bit wide parallel data. To obtain the necessary speed, however, the chip set is fabricated using a bipolar process, and the receiver and transmitter require separate chips. Such a solution is highly power consumptive and expensive.

Another commercial solution has been provided by Bull of France. The Bull technology employs a frequency multiplier for parallel to serial data conversion. Such devices typically introduce noise into the silicon substrate and interfere with other multipliers on the chip. In addition, the Bull technology uses an exclusive OR tree for parallel to serial conversion. The use of exclusive OR trees is well known, together with the difficulty of equalizing the delay through all paths of such devices. Additionally, the Bull technology uses output signals having full logic swings. This results in slower performance.

Various techniques exist for improving the characteristics of transmission over serial links. For example, transmission codes may be employed to alter the frequency spectrum of the transmitted serial data so as to facilitate clock recovery and enable AC coupling. Each transmission code will also typically provide special characters, not included within the data alphabet, to be used in character

IPR2024-001428
HP – EX1002, Page 39 of 487

synchronization, frame delimiting, as well as perhaps for diagnostic purposes. Coding may also be employed to reduce transmission bandwidth as a means of limiting the signal distortion occurring during propagation through the transmission medium. In the case of wire links, it is desirable to utilize codes with no DC and little low frequency content in order to allow for DC isolation of the driver and receiver circuitry from the transmission line, as well as to reduce signal distortion on the line. An efficient coding system should also be disposed to encode clock information with the encoded data in a manner allowing for extraction of the clock information during decoding. This obviates the need for provision of a separate clock signal over a dedicated clock line, since the clock information recovered during decoding may be instead used by the receiver circuitry.

Within local area networks (LANs), transmission coding schemes exist for converting words of various length to characters of greater length. For example, three-bit words may be converted to four-bit characters (3B/4B), four-bit words may be converted to five-bit characters (4B/5B), and so on. Typically, coding and decoding is achieved using a "key" in which each word is mapped to a corresponding character. Unfortunately, the complexity of this type of mapping scheme generally precludes utilization of random logic, and often requires implementations involving look-up tables or the like. This is disadvantageous given that look-up tables realized using ROM consume significant chip area and tend to slow circuit operation.

A particular 8B/10B coding scheme is described in U.S. Patent No. 4,486,739. In particular, a binary DC balanced code and associated encoder circuit are described as being operative to translate an 8 bit byte of information into 10 binary digits for transmission. The 8B/10B coder is partitioned into a 5B/6B plus a 3B/4B coder. Despite ostensibly facilitating DC-balanced encoding, this system tends to require relatively lengthy encoding and decoding intervals.

Although progress has been made in the development of coding techniques disposed to facilitate serial data transmission, there remains a need for a coding

IPR2024-001428
HP – EX1002, Page 40 of 487

scheme capable of efficiently supporting very high speed serial data transmission. Such a coding scheme should also be DC-balanced in order to facilitate AC coupling and clock recovery. In addition, it would be desirable to provide a coding scheme capable of facilitating real-time data transfer by allowing for rapid synchronization during decoding.

IPR2024-001428
HP – EX1002, Page 41 of 487

SUMMARY OF THE INVENTION

In one aspect, the present invention is directed to a method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes. The bits in each of the data bytes are selectively complemented in accordance with the number of logical transitions in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity.

In a high-transition mode of operation, the bits within those of the data blocks including less than a minimum number of logical transitions are selectively complemented in order that each such selectively complemented data block include in excess of the minimum number of logical transitions. In a low-transition mode of operation, the bits within those of the data blocks having more than a predefined number of logical transitions are selectively complemented in order that each such selectively complemented data block includes less than the maximum number of logical transitions. During both high-transition mode and low-transition mode encoding of each such selectively complemented data block into an encoded character, a predefined bit is set indicating that selective complementing has been effected.

In another aspect, the present invention comprises a high speed digital signal transmission system. The system includes an encoder for encoding digital input data

IPR2024-001428
HP – EX1002, Page 42 of 487

into data characters and for encoding control data into control characters. Each of the data characters have a first plurality of logical transitions within a first range, and each of the control characters have a second plurality of logical transitions in a second range different from the first range. The data and control characters are converted into a serial data stream and transmitted over a communication link.

The high speed digital signal transmission system further includes a deserializer for receiving the serial data stream from the communication link and for recovering received data and control characters therefrom. The received control characters are distinguished from the received data characters on the basis of numbers of logical transitions detected therein. A decoder, coupled to the deserializer, is operative to decode the received data and received control characters.

BRIEF DESCRIPTION OF THE DRAWINGS

Additional objects and features of the invention will be more readily apparent from the following detailed description and appended claims when taken in conjunction with the drawings, in which:

FIG. 1 is a block diagram depicting a DC-balanced encoding system of the present invention implemented within a high-speed digital transmission system.

FIG. 2 shows the overall functional organization of a DC-balanced encoder of the present invention in the form of a data flow chart.

FIG. 3 provides a block diagrammatic representation of a decoder included within the transmission system of FIG. 1.

FIG. 4 shows a random logic implementation of an alternate byte inversion decoder module of the decoder of FIG. 3.

FIG. 5 is a flow chart depicting the synchronization process effected by a decoder synchronization module operative in conjunction with the decoder of FIG. 3.

FIG. 6 is a flow chart depicting the synchronization process effected by the decoder synchronization module during high-transition mode operation.

IPR2024-001428
HP – EX1002, Page 43 of 487

FIGS. 7A and 7B provide a schematic representation of an exemplary implementation of the encoder.

FIGS. 8A and 8B provide a schematic representation of a preferred implementation of the decoder synchronization module.

DESCRIPTION OF THE PREFERRED EMBODIMENT

I. System Overview

FIG. 1 is a block diagram depicting a transition-controlled, DC-balanced encoding system of the present invention implemented within a high-speed digital transmission system 10. In the system 10 of FIG. 1, 8-bit bytes of parallel data 14 are provided to a DC-balanced encoder 18 operative to effect transition-controlled, DC-balanced 8B/10B coding in accordance with the present invention. The resultant 10B encoded characters 22 are provided to a serializer 26 disposed to convert the 10-bit characters into a serial data stream for transmission over a serial data link 30 (e.g., an optical fiber cable, or twisted-pair copper wire). As is described herein, the relatively straightforward mathematical characteristics of the encoding algorithm performed by the encoder 18 allow for economical, high-speed implementations in random logic.

The serial data stream is received from the serial data link 30 by a deserializer 34 and converted into 10-bit character data 38. The 10-bit character data 38 is then decoded into 8-bit data bytes 42 by a decoder 44. As is described hereinafter, both the encoder 18 and decoder 44 may be economically realized using simple digital logic circuits capable of real-time data processing.

II. DC-Balanced Transition-Controlled Encoding and Decoding System

The following provides a detailed description of the 8B/10B transition-controlled coding scheme carried out within the encoder 18. The transition-controlled code disclosed herein contemplates the use of either a high-transition or low-transition set of "in-band" code characters. Each high-transition in-band code character is derived from an input data byte in which four or more logical transitions

IPR2024-001428
HP – EX1002, Page 44 of 487

exist between the eight bits thereof. Similarly, each low-transition in-band code character is derived from an input data byte in which fewer than four logical transitions exist between its eight bits. This scheme takes advantage of the fact that 128 of the 258 eight-bit ASCII codes include four or more logical transitions, and the remaining 128 ASCII codes include fewer than four logical transitions.

It has been found that each of the 128 eight-bit codes including fewer than four logical transitions may be mapped to a corresponding eight-bit code having four or more logical transitions, and vice-versa. As is described herein, this mapping may be achieved by complementing predefined bits in each eight-bit code to be mapped. During high-transition mode encoding, the bits within those input bytes having fewer than four logical transitions are selectively complemented and thereby mapped to bytes having four or more logical transitions. Alternately, during low-transition mode encoding the bits within those input bytes having four or more logical transitions are also selectively complemented and thereby mapped to bytes having fewer than four logical transitions. During both encoding modes, a bit of predefined value is appended to the selectively complemented byte in order to produce an intermediate 9-bit encoded symbol prior to creation of a corresponding 10-bit encoded character. When the input byte includes the number of logical transitions mandated by the current encoding mode (i.e, high-transition mode or low-transition mode), the appended bit is set to the complement of the predefined value in order to identify which of the 9-bit intermediate symbols include selectively complemented bytes. This results in the entire set of 256 eight-bit codes being made available for encoding into 10-bit characters during both low-transition mode and high-transition mode operation.

It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as

IPR2024-001428
HP – EX1002, Page 45 of 487

a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream.

Given the relatively high number of transitions within each of the in-band characters of the high-transition set, the high-transition set of characters may advantageously be employed to facilitate timing recovery. On the other hand, the low number of transitions within the code characters of the low-transition set makes this set of characters ideal for use in applications in which it is desired to minimize power consumption and/or electromagnetic interference (EMI).

In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the

IPR2024-001428
HP – EX1002, Page 46 of 487

low-transition set. The following constitutes an exemplary set of out-of-band synchronization characters for use with the low-transition set of code characters:

1100101010
1101001010
1101010010
1101010100

It is a feature of the invention that if one of the above out-of-band synchronization characters is transmitted three or more consecutive times within any preamble period, the synchronization character is ensured of being detected during the associated data recovery process. In this regard a "preamble" sequence is sent during a preamble period preceding each transmission of encoded characters. The transmission of preamble sequences may occur not only as part of system initialization, but also at various other times to ensure that synchronization is maintained between encoding and decoding processes.

The coding scheme of the present invention is predicated on particular characteristics of the 256 different 8-bit binary code values. Referring to TABLE I, the 256 different 8-bit binary codes may be divided into eight groups G0-G7, where the binary codes within each group G0-G7 include the same number of transitions. It is observed that each binary code within group G0 can be converted to a corresponding binary code within group G7 by inverting alternate bits within the group G0 code. In the same way, each of the binary codes within groups G1, G2 and G3 may be converted to one of the binary codes within groups G6, G5 and G4, respectively, through inversion of alternate bits. As is described herein, the high-transition set of 10-bit characters is obtained by encoding of the 8-bit binary codes in groups G0-G3, and the low-transition set obtained by encoding groups G4-G7.

IPR2024-001428
HP – EX1002, Page 47 of 487

TABLE I

GROUP	NUMBER OF TRANSITIONS	NUMBER OF BYTES IN GROUP	EXAMPLE BYTES IN GROUP
G0	0	2	00000000, 11111111
G1	1	14	00000001, 11111110
G2	2	42	00000010, 11111101
G3	3	70	00000101, 11111010
G4	4	70	00001010, 11110101
G5	5	42	00010101, 11101010
G6	6	14	00101010, 11010101
G7	7	2	01010101, 10101010

During operation of the encoder 18 in a high-transition encoding mode, each 8-bit binary code within byte groups G0-G3 provided thereto is converted to a corresponding binary code within byte groups G4-G7 through inversion of alternate bits. Conversely, during operation in the low-transition encoding mode, each 8-bit binary code within groups G4-G7 provided to encoder 18 is mapped to a corresponding binary code within groups G0-G3. In the exemplary embodiment, the inversion of alternate bits is effected through the inversion of the even bits of the 8-bit binary codes. When the alternate bits of a given 8-bit code have been so inverted, a predefined bit within the resulting 10-bit encoded character derived from the given eight-bit code is set so as to indicate that mapping has occurred between byte groups.

IPR2024-001428
HP – EX1002, Page 48 of 487

Referring now to FIG. 2, the overall functional organization of the transition-controlled DC-balanced encoder 18 of the present invention is shown in the form of a data flow chart. In FIG. 2, the 8-bit parallel data 14 to be encoded is latched within an input latch 70 comprised of, for example, eight D-type flip-flops. A transition counter 74 is operative to count the number of transitions (T) in logical value between adjacent bits of each byte of parallel data 14 within latch 70. If D_7, D_6, ... D_0, (i.e., D[7:0] comprise the eight bits of data latched within the input latch 70, then the transition counter 74 may determine T as follows:

$$T := (D_7 \text{ xor } D_6) + (D_6 \text{ xor } D_5) + (D_5 \text{ xor } D_4) + (D_4 \text{ xor } D_3) + (D_3 \text{ xor } D_2) + (D_2 \text{ xor } D_1) + (D_1 \text{ xor } D_0)$$

A COUNT line 78 is set to a predefined logical value by counter 74 if more four or more logical transitions are counted between bits of the latched byte ($T > 3$), and is set to the complement of the predefined logical value otherwise ($T \leq 3$). In what follows it will be assumed that COUNT = 0 if four or more logical transitions are counted by counter 78 ($T > 3$), and that COUNT = 1 otherwise ($T \leq 3$).

As is indicated by FIG. 2, a transition controller 82 is responsive to COUNT line 78 and to a mode select line 86. Mode select line 86 determines whether encoding is to be performed using the high-transition set or the low-transition set of code characters. When mode select line 86 indicates high-transition encoding is in effect, and COUNT line 78 registers that less than four logical transitions exist within the byte stored within latch 70, the transition controller 82 instructs conditional alternate bit inversion (CABI) logic 90 to invert the even bits the byte stored within latch 70. The resultant conditionally inverted byte, which will have four or more logical transitions, is stored within intermediate latch 94. Conversely, if high-transition encoding is in effect and four or more logical transitions are counted within the byte stored in the input latch 70, the transition controller 82 causes CABI logic 90 to simply transfer the byte from latch 70 (without bit

IPR2024-001428
HP – EX1002, Page 49 of 487

inversion) to the intermediate latch 94. Accordingly, during high-transition encoding mode,

if $T < 4$,

then $E[8:0] = '1'\ D_7\bar{D}_6D_5\bar{D}_4D_3\bar{D}_2D_1\bar{D}_0$,

else if $T \geq 4$,

then $E[8:0] = '0'\ D_7D_6D_5D_4D_3D_2D_1D_0$

where E[7:0] comprise the eight bits stored within the intermediate latch 94, and E[8] comprises the value of COUNT stored within COUNT latch 95.

When mode select line 86 indicates that low-transition encoding has been selected, and COUNT line 78 registers that four or more logical transitions are present within the byte stored within latch 70, the transition controller 82 instructs conditional alternate bit inversion (CABI) logic 90 to invert the even bits the byte stored within latch 70. Otherwise, if low-transition encoding being performed and four or more logical transitions are counted within the byte stored in the input latch 70, the stored byte is simply transferred without bit inversion to the intermediate latch 94. Accordingly, during low-transition encoding mode,

if $T > 3$, then $E[8:0] = '1'\ D_7\bar{D}_6D_5\bar{D}_4D_3\bar{D}_2D_1\bar{D}_0$,

else $E[8:0] = '0'\ D_7D_6D_5D_4D_3D_2D_1D_0$

After CABI logic 90 has provided a byte having a number of logical transitions within the appropriate range to latch 94, a DC-balancing process is performed in which the cumulative disparity between complementary logical values within the stream of 10-bit encoded characters produced by the encoder 18. As used herein, the term "cumulative disparity" (D_{cum}) denotes the excess of one bits relative to zero bits produced by the encoder 18 subsequent to synchronization being achieved with the decoder 44 in the manner described below. The term "current disparity" (D_{cur}) refers to the excess of one bits to zero bits within the byte currently

IPR2024-001428
HP – EX1002, Page 50 of 487

stored within the latch 94, and is determined by disparity checker 96. A DC-balancing module 98 serves to compare the current disparity to the cumulative disparity stored within latch 99. The result of the comparison is then used to determine whether the byte stored within latch 94 is inverted by conditional byte inversion (CTBI) logic 100 during the course of transfer thereof to output register 104. In this way CTBI logic 100 serves to minimize the cumulative disparity associated with the serial stream produced by the encoder 18. The following provides a logical description of the manner in which each of the 10-bit characters T[9:0] in the DC-balanced character stream produced by the encoder are derived from the byte E[7:0] stored within the intermediate latch 94 and the bit E[8] within COUNT latch 95.

The current disparity (D_{cur}) is computed by the disparity checker 96 as follows:

$$D_{cur} := \{(E_7 \text{ and } E_6) + (E_5 \text{ and } E_4) + (E_3 \text{ and } E_2) + (E_1 \text{ and } E_0)\} - \{(E_7 \text{ nor } E_6) + (E_5 \text{ nor } E_4) + (E_3 \text{ nor } E_2) + (E_1 \text{ nor } E_0)\}$$

It is noted that during operation in the high-transition mode, $-2 \leq D_{cum} \leq 2$, while during operation in the low-transition mode $-4 \leq DP \leq 4$. Within the DC-balancing module 98, if it is determined that $D_{cur} = 0$ or $D_{cum} = 0$, then

if $E_8 = '0'$
then $T[9:0] = '1'E_8\bar{E}_7\bar{E}_6\bar{E}_5\bar{E}_4\bar{E}_3\bar{E}_2\bar{E}_1\bar{E}_0$, and

$$D'_{cum} = D_{cum} - D_{cur}$$

else if E_8 is not equal to zero,
then $T[9:0] = '0'E_8E_7E_6E_5E_4E_3E_2E_1E_0$, and
$D'_{cum} = D_{cum} + D_{cur}$

where D'_{cum} is the updated cumulative disparity computed by disparity updater 108 and stored thereby within latch 99.

IPR2024-001428
HP – EX1002, Page 51 of 487

Alternately, if the DC-balancing module determines that the most significant bit (MSB) of D_{cur} and the MSB of D_{cum} are not equivalent, then

$$T[9:0] = '0'E_8E_7E_6E_5E_4E_3E_2E_1E_0, \text{ and}$$
$$D'_{cum} = D_{cum} + D_{cur} - E_8$$

Finally, in all other cases if the MSB of D_{cur} and the MSB of D_{cum} are equivalent, then, then

$$T[9:0] = '1'E_8\bar{E}_7\bar{E}_6\bar{E}_5\bar{E}_4\bar{E}_3\bar{E}_2\bar{E}_1\bar{E}_0, \text{ and}$$
$$D'_{cum} = D_{cum} - D_{cur} + E_8$$

In this way the cumulative disparity is reduced, and DC balance achieved, through selective inversion of the byte E[7:0] by CBI logic 100 during the course of filling the output latch 104 as T[7:0]. It is observed that the logical value of T[8] is indicative of whether the even bits of byte D[7:0] received at input latch 70 were complemented during generation of the byte E[0:8]. Similarly, the logical value of T[9] indicates whether byte E[7:0] was inverted during transfer to latch 104.

III. Decoding

Referring to FIG. 1, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX_9, RX_8, ..., RX_0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9:0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding

IPR2024-001428
HP – EX1002, Page 52 of 487

to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner.

FIG. 3 provides a block diagrammatic representation of the decoder 44. The 10-bit parallel data produced by the deserializer is seen to be received over bit lines RX[9:0] by a decoder switch 150. The decoder switch 150 serves to switch the 10-bit data received over bit lines RX[9:0] to switched bit lines S[9:0] in accordance with frame boundary pointer value 118 provided by the synchronization module 114. Specifically, the one of the received bits RX[9:0] corresponding to the first transmitted bit T[0] is switched to bit line S[0], the one of the received bits RX[9:0] corresponding to the second transmitted bit T[1] is switch to bit line S[1], and so on. The switched data impressed upon bit lines S[7:0], which corresponds to the transmitted data byte T[7:0], is stored within 8-bit latch 154. Similarly, the switched data bit S[8], which corresponds to the transmitted bit T[8], is provided to 1-bit latch 158. Since the logical value of bit line S[8] tracks the logical value of T[8], the bit line S[8] informs the Alternate Bit Inversion (ABI) decoder 160 as to whether the even bits of the input data D[7:0] were complemented by CABI logic 90 (FIG. 2) during the encoding process. Likewise, the bit line S[9] (which tracks the logical value of T[9]) informs the ABI decoder 160 as to whether CTBI logic 100 complemented the byte stored within latch 104 during the DC-balancing portion of the encoding process. In this way the decoder 160 is informed of the logical operations performed upon the byte S[7:0] stored within 8-bit latch 154 during the encoding process, thereby facilitating straightforward decoding using random logic.

Turning now to FIG. 4, there is shown a random logic implementation of the ABI decoder 160. The ABI decoder includes a set of nine exclusive-or (XOR) gates N1-N9 for decoding the 10-bit frame-aligned data S[9:0] in order to produce the 8-bit decoded byte DE[7:0]. In the implementation of FIG. 4, the XOR gates N1-N9 produce the decoded byte DE[7:0] as follows:

DE[7] := S[7] xor S[9]

IPR2024-001428
HP – EX1002, Page 53 of 487

DE[6] := S[6] xor β

DE[5] := S[5] xor S[9]

DE[4] := S[4] xor β

DE[3] := S[3] xor S[9]

DE[2] := S[2] xor β

DE[1] := S[1] xor S[9]

DE[0] := S[0] xor β

where β := S[9] xor S[8].

IV. Synchronization

As mentioned above, the decoder synchronization module 114 provides an indication to the decoder 44 of the frame boundary of each transmitted 10-bit character T[9:0]. The decoder module 114, together with the decoder switch 150 (FIG. 3), effectively function as a barrel-shifter to shuffle the parallel data RX[9:0] from the deserializer into the frame-aligned data S[9:0]. In accordance with the invention, a preamble sequence is produced by the encoder 18 at various times (e.g., at system power-up) in order to facilitate frame boundary detection by the synchronization module 114. In the exemplary embodiment this preamble sequence includes several repetitions of selected out-of-band characters easily distinguishable from in-band characters. Again, during high-transition mode operation each out-of-band character will include less than four logical transitions, and during low-transition mode operation each out-of-band character will includes four or more logical transitions. As is discussed below, during operation in each mode several repetitions of specially selected out-of-band characters are produced by the encoder 18 during the preambling period as a means of ensuring rapid frame boundary identification within the decoder synchronization module 114. At the conclusion of the preambling period, the module 114 will "know" which of the bit lines RX[9:0] corresponds to the first bit T[0] of the 10-bit transmitted character, and will inform decoder via frame boundary pointer 118.

IPR2024-001428
HP – EX1002, Page 54 of 487

By selecting an appropriate subset of out-of-band characters for transmission during the preambling period, the worst-case time required for synchronization to be achieved may be reduced relative to that required by conventional synchronization schemes. In particular, during low-transition mode operation the following out-of-band characters are used as "synchronization characters".

1100101010
1101001010
1101010010
1101010100

During high-transition mode operation, the following out-of-band characters are used as synchronization characters:

1000001111
1000011111
1000111111
1001111111
1011111111

During each preambling period, three repetitions of the same synchronization character are produced by the encoder 18. As is described herein, by processing the 21 bits most recently produced by the encoder 14, synchronization module 114 is able to detect at least one of the three repetitions of the synchronization character transmitted during a given preambling period. This advantageously allows synchronization to be achieved within a relatively short preambling period.

Turning now to FIG. 5, a flow chart is provided of the synchronization process effected by the decoder synchronization module 114 during low-transition mode operation. During each clock cycle of the module 114, a block of ten bits is loaded from the deserializer 34 into a first 10-bit latch 150. Also during each clock cycle, a 10-bit block is transferred from the first 10-bit latch 150 to a second 10-bit

IPR2024-001428
HP – EX1002, Page 55 of 487

latch 154. Similarly, the 10-bit block currently stored within the second 10-bit latch 154 is transferred during each clock cycle to a third 10-bit latch 158.

As is indicated by FIG. 5, an exclusive-NOR (XNOR) operation (step 162) is performed between adjacent bits included within a 21-bit "window" of data held by the latches 150,154,158. Specifically, this 21-bit window includes a 10-bit block L3[9:0] from the third latch 158, a 10-bit block L2[9:0] from the second latch 154, and the bit L1[9] from the first latch 150. In this regard the bit L1[9] is that bit which becomes bit L2[9] upon being transferred to the second latch 154. As an example of operation during the low-transition mode, consider a 21-bit window (i.e., L3[9:0], L2[9:0], L1[9]) comprised of the following parallel bit sequence:

101101001010110100101

If an XNOR operation is performed between each pair of adjacent bits, the following result is obtained:

001000100000100010000

As is indicated by FIG. 5, this 20-bit result of the XNOR operation (step 160) is divided into four 5-bit groups (i.e., Group A, Group B, Group C and Group D). In the current example, these four 5-bit groups are defined as follows:

```
00100010000010001000    {Result of XNOR operation}

00100                   {Group A}
     01000              {Group B}
          00100         {Group C}
               01000    {Group D}
```

The synchronization characters for both the high-transition mode and the low-transition modes enumerated above have been selected such that particular relationships arise between Groups A, B, C and D during the preambling period. That is, when three consecutive occurrences of the same synchronization character - produced by the encoder 18 during preambling - are received by the deserializer 34 and provide as 10-bit parallel data to the synchronization module 114.

IPR2024-001428
HP – EX1002, Page 56 of 487

In an exemplary implementation, the following two relationships (Condition I and Condition II) arise between Groups A, B, C and D during the preambling period:

Condition I. The number of logical "1's" collectively present in Groups A, B, C, and D is exactly four, and corresponds to one of the following three cases:

	Number of Logical 1's			
Case	Group A	Group B	Group C	Group D
#1	1	1	1	1
#2	2	0	2	0
#3	0	2	0	2

As is indicated by FIG. 5, the number of "1's" in each of the Groups A, B, C and D are determined by "1" counter modules 172, 174, 176 and 178, respectively. Since the number of "1's" within each Group A, B, C and D is determined by the results of the XNOR operation between adjacent bits in the 21-bit window (step 160), the number of "1's" within each group is indicative of the number of "non-transitions" in logical value between adjacent bits in each of four segments of the 21-bit window associated with Groups A, B, C and D. In the current example, it is seen that each of Groups A, B, C and D each include a single "1". Accordingly, the current example corresponds to Case *#1*.

Condition II. The sequence of bits comprising Group A is equivalent to the bit sequence of Group C, and the sequence of bits comprising Group B is equivalent to the bit sequence of Group D. That is, Group A = Group C, and Group B = Group D.

In accordance with the invention, both Condition I AND Condition II are satisfied if and only if the same synchronization character is stored within the first, second and third 10-bit latches 150, 154 and 158. That is, both Condition I and Condition II are satisfied only during the preambling period, when three repetitions

IPR2024-001428
HP – EX1002, Page 57 of 487

of the same synchronization character are produced by the encoder 18. This aspect of the invention is explained immediately below with reference to low-transition mode operation.

As was described above, the adjacent bits within a 21-bit window provided by latches 150, 154 and 158 are XNOR'ed during step 160 (FIG. 5). Because each in-band or out-of-band character produced by the encoder 18 is exactly ten bits in length, the 21-bit window will include all or part of 1st, 2nd and 3rd 10-bit characters. The various ways in which the 21-bit window may include bits from these 1st, 2nd and 3rd 10-bit characters is set forth below:

	# Of Bits	
1st Character	2nd Character	3rd Character
1	10	10
2	10	9
3	10	8
4	10	7
5	10	6
6	10	5
7	10	4
8	10	3
9	10	2
10	10	1

Since each character is either an in-band (e.g., DATA) character or an out-of-band command or synchronization (i.e., SYNC) character, the following sets forth the possible combinations of DATA and SYNC characters among the 1st, 2nd and 3rd 10-bit characters contributing to the 21-bit window:

IPR2024-001428
HP – EX1002, Page 58 of 487

Combination	1st Character	2nd Character	3rd Character
A	SYNC	SYNC	SYNC
B	SYNC	SYNC	DATA
C	SYNC	DATA	DATA
D	DATA	DATA	DATA
E	DATA	DATA	SYNC
F	DATA	SYNC	SYNC

For example, the 21-bit window could be comprised of two bits of a 1st SYNC Character, ten bits of a 2nd DATA Character, and nine bits of a 3rd DATA Character (i.e., Combination C).

During low-transition mode operation, all in-band (e.g., DATA) characters include a maximum of three logical transitions or, equivalently, more than four "non-transitions" in logical value between the ten adjacent bits of the in-band character. Accordingly, during low-transition mode operation if the 2nd character is a DATA character, it will include more than four logical non-transitions. Since Condition I indicates that the number of logical non-transitions within the entire 21-bit window will be exactly four when three identical SYNC characters are present therein, Condition I will not be satisfied when the 2nd character is a DATA character since it would include more than four logical non-transitions. Hence, if Condition I is to be satisfied then the 21-bit window cannot be comprised of the character sets specified by Combinations C, D and E (i.e., 2nd character is a DATA character).

In accordance with the invention, the synchronization characters listed above have been chosen such that Condition II will be satisfied if the 1st and 3rd characters transmitted during any preamble period are identical. Hence, Combination B and F do not satisfy Condition II. It follows that only Combination A (i.e., three consecutive SYNC characters) satisfies both conditions I and II.

Referring to FIG. 5, if both Condition I and Condition II are satisfied (step 190) then selected adjacent bits within Groups A and B are AND'ed (step 196) as described below in order to identify the frame boundaries of the SYNC characters

IPR2024-001428
HP – EX1002, Page 59 of 487

detected within the 21-bit window. Since each SYNC character within the 21-bit window is loaded by the deserializer 34 into latch 150, the frame boundary of each SYNC character may be identified in terms of the one of the bit lines R[9:0] from the deserializer 34 upon which the first bit of each such SYNC character is impressed. Once this identification is achieved, the decoder __ is informed of the identity of this bit line R[9:0] by way of frame boundary pointer 118.

The AND operation of step 196 is performed between all of the adjacent bits in the third latch 158 (i.e., L3[9:0]), as well as between L3[0] and L2[9]. When the result of step 190 has indicated that both Condition I and Condition II are satisfied, the result of the AND operation of step 196 will produce only a single logical one indicative of the value of the frame boundary pointer 118. In the current example, L3[9:0] := {1011010010} and L2[9] :=[1], and thus the result of the AND operation of step 196 is {0010000000}. That is, the third position in the 21-bit window corresponds to the first bit of a synchronization character. Accordingly, in the current example the frame boundary pointer 118 would be set so as to identify the third (RX[7]) of the ten bit lines RX[9:0] as carrying the first bit of each 10-bit character produced by the deserializer 34.

In the exemplary embodiment a preambling sequence (i.e., three repetitions of the same out-of-band SYNC character) is sent upon system power-up as well as during lapses in data transmission over the serial link 30. This allows timing synchronization to be maintained between the encoder 18 and decoder 44 even in the absence of data transmission for extended periods.

FIG. 6 is a flow chart depicting the synchronization process effected by the decoder synchronization module 114 during high-transition mode operation. As is indicated by FIG. 6, the high-transition mode synchronization process is substantially similar to that performed during low-transition mode operation (FIG. 5). In particular, the high-transition mode synchronization process differs from the low-transition mode synchronization process primarily in that:

IPR2024-001428
HP – EX1002, Page 60 of 487

(i) In step 160', an exclusive-OR (XOR) rather than an exclusive-NOR (XNOR) operation is performed upon adjacent bits within the latches 150', 154' and 158'.

(ii) In step 196', an AND operation is performed between each bit and the complement of the bit immediately to the right (e.g., L3[9] AND $\overline{L3[8]}$, L3[8] AND $\overline{L3[7]}$, L3[7] AND $\overline{L3[6]}$, and so on. In this way the AND gate producing a logical "1" identifies a "10" or "falling edge" sequence corresponding to the frame boundary.

V. Hardware Implementations of Encoder and Decoder Synchronization Module

In this section a description is provided of a specific hardware implementation of the encoder 18, and of an implementation of the decoder synchronization module 114 suitable for use during low-transition mode operation. A description of an exemplary hardware realization of the decoder 44 in random logic was provided above in section III.

FIGS. 7A and 7B provide a schematic representation of an exemplary implementation of the encoder 18. The 8-bit parallel data D[7:0] from latch 70 to be encoded is seen to be provided to seven exclusive-OR gates 240 of the transition counter 74. The outputs of the exclusive-OR gates 240 are provided to a set of full-adders 242, 244, 246 and 248. The carry output (C) of the full-adder 248 corresponds to the COUNT line 78, and indicates whether less than four logical transitions exist between the bits in the data D[7:0]. When an out-of-band command is being received from latch 70, a command line (TX_CMD) provided to NOR gate 260 is raised so as to prevent the COUNT line 78 from causing inversion of the even bits of D[7:0] within CABI logic 90. Otherwise, when the data D[7:0] from latch is being encoded in accordance with the invention, the output 78' of NOR gate 260 tracks the logical value of COUNT line 78.

As is indicated by FIG. 7A, in the exemplary embodiment CABI logic 90 is comprised of a plurality of NOR gates 270. Each NOR gate 270 includes one input coupled to COUNT line 78', and another input connected to one of the even bits of

IPR2024-001428
HP – EX1002, Page 61 of 487

D[7:0]. The output of CABI logic 90 is provided to intermediate latch 94, which is coupled to the input of disparity checker 96 (FIG. 7B).

Turning to FIG. 7B, the disparity checker 96 includes four AND gates 290-293 for determining occurrences of "11" within the conditionally bit-inverted byte E[7:0]. Similarly, four NOR gates 296-299 are provided for detecting occurrences of "00" within E[7:0]. Since pattern of "01" and "01" within E[7:0] are already "DC-balanced" in the sense of including equal numbers of ones and zeroes, there exists no need to detect such patterns during the DC-balancing process effected by the circuitry of FIG. 7B. A first full-adder 302 and first half-adder 306 are disposed to count the occurrences of "11" detected by AND gates 290-293. In like manner a second full-adder 308 and second half-adder 312 are provided for counting the occurrences of "00" detected by the NOR gates 296-299. A first pair of full-adders 316 and 318 determine the difference in the counted occurrences of "11" and "00".

The DC-balancing module 98 includes a three-input NOR gate 330, a first exclusive-OR gate 332, a latch 336 and a second exclusive-OR gate 338. When the occurrences of "11" and "00" are determined to be equivalent by full-adders 316 and 318, the complement of E[8] determines the value of T[9], and hence whether the byte E[7:0] is inverted by CTBI logic 100. When the counted occurrences of "00" and "11" are not equivalent, the value of T[9] comprises the output of XOR gate 332. In this regard a first input 342 to XOR gate 332 comprises the most significant bit (MSB) produced by the pair of full-adders 316 and 318, which is equivalent to the MSB of the current disparity D_{cur} (i.e., the difference in "1's and "0's" in E[7:0]). A second input 344 to XOR gate 332 corresponds to the MSB of the cumulative disparity D_{cum}. As is indicated by FIG. 7B, the latch 99 for storing the cumulative disparity is comprises of three registers 350-352. The cumulative disparity is updated by a disparity updater 108 comprised of a backward chain of full adders 356 and 358, a set of three exclusive-OR gates 360-362, and a corresponding set of three full-adders 365-367. Finally, CTBI logic 100 includes a set of eight exclusive-OR gates 374.

IPR2024-001428
HP – EX1002, Page 62 of 487

FIGS. 8A and 8B provide a schematic representation of a preferred implementation of the decoder synchronization module 114. In FIG. 8A, the 10-bit latches 150, 154 and 158, for storing L3[9:0], L2[9:0] and L1[9:0], respectively, may each be realized using an array of ten D-type flip-flops. A plurality of XNOR gates 402 are provided for XNOR'ing adjacent bits included within the 21-bit window corresponding to L3[9:0], L2[9:0] and l1[9]. The Group A, Group B, Group C, and Group D outputs of the XNOR gates 402 are then respectively provided to the "1" counters 172, 174, 176 and 178. As is indicated by FIG. 8A, the existence of Condition I is detected by random logic 190a, which is comprised of four NAND gates 410-413.

Turning now to FIG. 8B, the existence of Condition II is identified by an arrangement of random logic identified by reference numeral 190b. Random logic 190b includes a set of ten XOR gates 422, the outputs of which are provided as shown to NAND gates 426 and 428. The outputs of NAND gates 426 and 428 are coupled to the inputs of a NOR gate 430, the output of which is driven to a logical "1" when Condition II is satisfied. Finally, the logical state of the output (Sync_Sig) of an AND gate 440 is indicative as to whether Condition I and Condition II have been satisfied (i.e., as to whether synchronization has been achieved). If so, adjacent bits within L3[9:0] and L2[9] are AND'ed (step 196 of FIG. 5) by a set of ten AND gates 450. The outputs PTR[9:0] comprise the frame boundary pointer 118, which informs decoder 44 as to which of the bit lines RX[9:0] correspond to the first bit T[0] of each transmitted 10-bit character T[9:0].

The previous description of the preferred embodiments has been provided to enable any person skilled in the art to make or use the present invention. Various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments without the use of inventive faculty. Thus, the present invention is not intended to be limited to the embodiments shown herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

IPR2024-001428
HP – EX1002, Page 63 of 487

WHAT IS CLAIMED IS:

1. A method for producing a DC balanced sequence of characters from an inputsequence of 8 bit data blocks, said method including the steps of:

selectively complementing bits in said 8 bit data blocks in accordance with the number of logical transitions in each said 8 bit data block in order to produce selectively complemented data blocks;

determining a cumulative disparity in the number of logical values of different type included within ones of said selectively complemented data blocks previously encoded into ones of said characters;

determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded; and

assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity.

2. The method of claim 1 wherein said step of selectively complementing includes the step of selectively complementing predetermined ones of said bits in said 8 bit data blocks having less than a predetermined number of said logical transitions.

3. The method of claim 1 wherein said step of selectively complementing includes the step of selectively complementing predetermined ones of said bits in said 8 bit data blocks having more than a predetermined number of said logical transitions.

IPR2024-001428
HP – EX1002, Page 64 of 487

4. The method of claim 1 further including the step of setting a predetermined bit in said candidate character to a first logical value when a current 8 bit data block associated therewith has less than a predetermined number of transitions, and setting said predetermined bit to the complement of said first logical value otherwise.

5. The method of claim 1 further including the step of setting a predetermined bit said candidate character to a first logical value when a current 8 bit data block associated therewith has more than a predetermined number of transitions, and setting said predetermined bit to the complement of said first logical value otherwise.

6. The method of claim 1 wherein the logical value of a predetermined bit in each character assigned to one of said selectively complemented data blocks indicates whether a candidate character or complement thereof was assigned to said selectively complemented data block.

7. The method of claim 1 wherein said step of determining a current disparity in a candidate character includes the step of performing a logic analysis upon bits of said candidate character.

8. The method of claim 1 further including the step of assigning specific ones of said 8 bit data blocks having less than a predetermined number logical transitions to be representative of special characters, said special characters being encoded distinctly from ones of said 8 bit data blocks not corresponding to ones of said special characters.

9. A binary data encoding apparatus for producing a DC balanced sequence of characters from an input sequence of 8 bit data blocks, said binary data encoding apparatus comprising:

means for selectively complementing bits in said 8 bit data blocks in accordance with the number of logical transitions in each

IPR2024-001428
HP – EX1002, Page 65 of 487

said 8 bit data block in order to produce selectively complemented data blocks;

means for determining a cumulative disparity in the number of logical values of different type included within ones of said selectively complemented data blocks previously encoded into ones of said characters;

means for determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded; and

means for assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity.

10. The binary data encoding apparatus of claim 9 wherein said means for selectively complementing includes means for selectively complementing predetermined ones of said bits in said 8 bit data blocks having less than a predetermined number of said logical transitions.

11. The binary data encoding apparatus of claim 9 wherein said means for selectively complementing includes means for selectively complementing predetermined ones of said bits in said 8 bit data blocks having more than a predetermined number of said logical transitions.

12. The binary data encoding apparatus of claim 9 further including means for setting a predetermined bit in said candidate character a first logical value when said current 8 bit data block has less than a predetermined number of transitions, and means for setting said predetermined bit to the complement of said first logical value otherwise.

IPR2024-001428
HP – EX1002, Page 66 of 487

13. The binary data encoding apparatus of claim 9 further including means for setting a predetermined bit in said candidate character to a first logical value when said current 8 bit data block has more than a predetermined number of transitions, and means for setting said predetermined bit to the complement of said first logical value otherwise.

14. The binary data encoding apparatus of claim 9 wherein the logical value of a predetermined bit in each character assigned to one of said selectively complemented data blocks indicates whether a candidate character or complement thereof was assigned to said selectively complemented data block.

15. The binary data encoding apparatus of claim 9 wherein said means for determining a current disparity in a candidate character includes means for performing a logic analysis upon bits of said candidate character.

16. The binary data encoding apparatus of claim 9 further including means for assigning specific ones of said 8 bit data blocks having less than a predetermined number logical transitions to be representative of special characters and for encoding said special characters distinctly from ones of said 8 bit data blocks not corresponding to ones of said special characters.

17. A high speed digital signal transmission system comprising:

means for encoding digital input data into data characters and for encoding control data into control characters, each of said data characters having a first plurality of logical transitions within a first range and each of said control characters having a second plurality of logical transitions in a second range different from said first range;

means for converting said data and control characters into a serial data stream and for transmitting said serial data stream over a communication link;

IPR2024-001428
HP – EX1002, Page 67 of 487

means for receiving said serial data stream from said communication link and for recovering received data and control characters therefrom; and

means for separating said received control characters from said received data characters on the basis of numbers of logical transitions detected in said received control characters and said received data characters, and for decoding said received data and received control characters.

18. The high speed digital signal transmission system of claim 17 wherein said means for encoding further includes means for selectively complementing bits in 8 bit data blocks within said input data on the basis of the number of logical transitions in each said 8 bit data block, thereby producing selectively complemented data blocks.

19. The high speed digital signal transmission system of claim 17 wherein said means for encoding further includes:

means for determining a cumulative disparity in the number of logical values of different type included within ones of said selectively complemented data blocks previously encoded into ones of said characters;

means for determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded; and

means for assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity.

IPR2024-001428
HP – EX1002, Page 68 of 487

20. The high speed digital signal transmission system of claim 17 wherein said means for transmitting includes means for transmitting at least three repetitions of a synchronization character, said synchronization character having a selected number of logical transitions within said second range wherein said means for separating includes means for detecting a boundary of one of said at least three repetitions of said synchronization character received from said communication link.

21. The high speed digital transmission system of claim 20 wherein said means for detecting a boundary includes:

means for coupling an input set of bits from said communication linkinto a register,

means for performing a first set of logical operations at least upon adjacent ones of a first subset of said input set of bits and for grouping the results of said predefined logical operations into a set of groups, and

means for comparing said groups in order to determine if said input set of bits corresponds to said at least three repetitions of said synchronization character.

22. The high speed digital transmission system of claim 21 wherein said means for detecting a boundary further includes detection logic means for identifying said boundary of said synchronization character on the basis of results of performance of a second set of logical operations upon said input set of bits.

23. The system of claim 21 wherein said means for comparing includes means for comparing first and third of said groups and for comparing second and fourth of said groups, said input set of bits corresponding to said at least three repetitions of said synchronization character if said first group is equivalent to said third group and if said second group is equivalent to said fourth group.

IPR2024-001428
HP – EX1002, Page 69 of 487

24. The system of claim 21 wherein said second set of logical operations correspond to logical AND operations performed on adjacent ones of a second subset of said input set of bits.

25. The system of claim 21 wherein each of said repetitions of said synchronization character includes 10 bits, said means for grouping including means for forming four of said groups from said first subset of said input set of bits wherein said first subset is comprised of less than 22 bits.

26. A high speed digital video signal transmission system comprising:

first encoder means for encoding a first sequence of 8 bit data words into data characters and for encoding control data into control characters, each of said data characters having a first plurality of logical transitions within a first range and each of said control characters having a second plurality of logical transitions in a second range different from said first range;

a serializer responsive to said data and control characters;

a link transmitter coupled to an output of said serializer for transmitting said data and control characters over a communication link; and

means for separating received ones of said data and control characters received from said communication link on the basis of numbers of logical transitions in said received ones of said data and control characters.

27. A system for detecting a boundary of a synchronization character within an encoded serial data stream including at least three repetitions of said synchronization character, said synchronization character having a number of logical transitions within a predetermined range, said system comprising:

means for coupling an input set of bits from said encoded serial data stream into a register;

IPR2024-001428
HP – EX1002, Page 70 of 487

means for performing a first set of logical operations at least upon adjacent ones of a first subset of said input set of bits and for grouping the results of said predefined logical operations into a set of groups;

means for comparing said groups in order to determine if said input set of bits corresponds to said at least three repetitions of said synchronization character; and

detection logic means for identifying said boundary of said synchronization character on the basis of results of performance of a second set of logical operations upon said input set of bits.

28. The system of claim 27 wherein said means for comparing includes means for comparing first and third of said groups and for comparing second and fourth of said groups, said input set of bits corresponding to said at least three repetitions of said synchronization character if said first group is equivalent to said third group and if said second group is equivalent to said fourth group.

29. The system of claim 27 wherein said second set of logical operations correspond to logical AND operations performed on adjacent ones of a second subset of said input set of bits.

30. The system of claim 27 wherein each of said repetitions of said synchronization character includes 10 bits, said means for grouping including means for forming four of said groups from said first subset of said input set of bits wherein said first subset is comprised of less than 22 bits.

31. A method for detecting a boundary of a synchronization character within an encoded serial data stream including at least three repetitions of said synchronization character, said synchronization character having a number of logical transitions within a predetermined range, said method comprising the steps of:

IPR2024-001428
HP – EX1002, Page 71 of 487

coupling an input set of bits from said encoded serial data stream into a register;

performing a first set of logical operations at least upon adjacent ones of a first subset of said input set of bits and for grouping the results of said predefined logical operations into a set of groups;

comparing said groups in order to determine if said input set of bits corresponds to said at least three repetitions of said synchronization character; and

identifying said boundary of said synchronization character on the basis of results of performance of a second set of logical operations upon said input set of bits.

32. The method of claim 31 wherein said step of comparing includes the step of comparing first and third of said groups and for comparing second and fourth of said groups, said input set of bits corresponding to said at least three repetitions of said synchronization character if said first group is equivalent to said third group and if said second group is equivalent to said fourth group.

33. The method of claim 31 wherein said second set of logical operations correspond to logical AND operations performed on adjacent ones of a second subset of said input set of bits.

34. The method of claim 31 wherein each of said repetitions of said synchronization character includes 10 bits, said step of grouping including the step of forming four of said groups from said first subset of said input set of bits wherein said first subset is comprised of less than 22 bits.

35. A method for high speed transmission of digital data comprising the steps of:

IPR2024-001428
HP – EX1002, Page 72 of 487

encoding an input sequence of 8 bit data blocks into a sequence of characters by: (i) selectively complementing bits in an input sequence of 8 bit data blocks in accordance with the number of logical transitions in each said 8 bit data block in order to produce selectively complemented data blocks, (ii) determining a cumulative disparity in logical values associated with ones of said selectively complemented data blocks previously encoded into ones of said characters, (iv) determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded, and (iv) assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity;

converting each character into a serial data string; and

transmitting each said serial data string over a serial communication link.

IPR2024-001428
HP – EX1002, Page 73 of 487

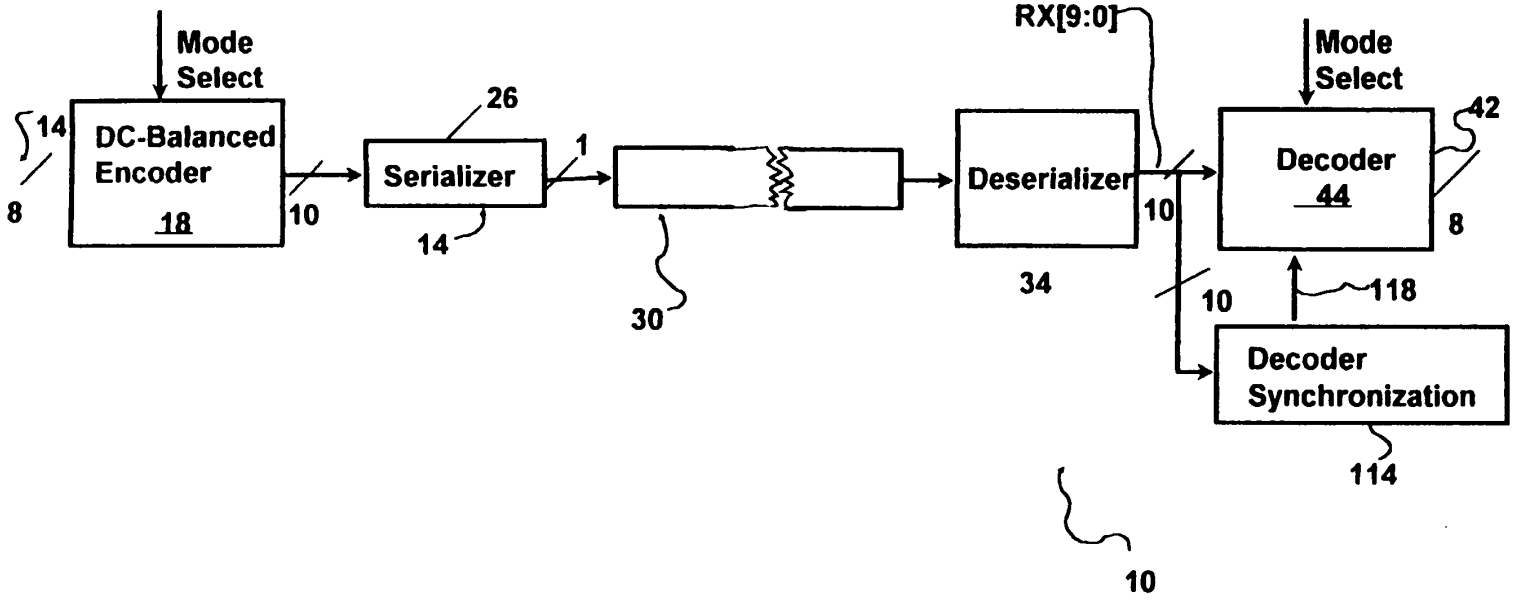

Mode Select
14
DC-Balanced Encoder
18
8
10
26
Serializer
14
1
30
RX[9:0]
Deserializer
34
10
Mode Select
42
Decoder
44
8
10
118
Decoder Synchronization
114
10

FIG. 1

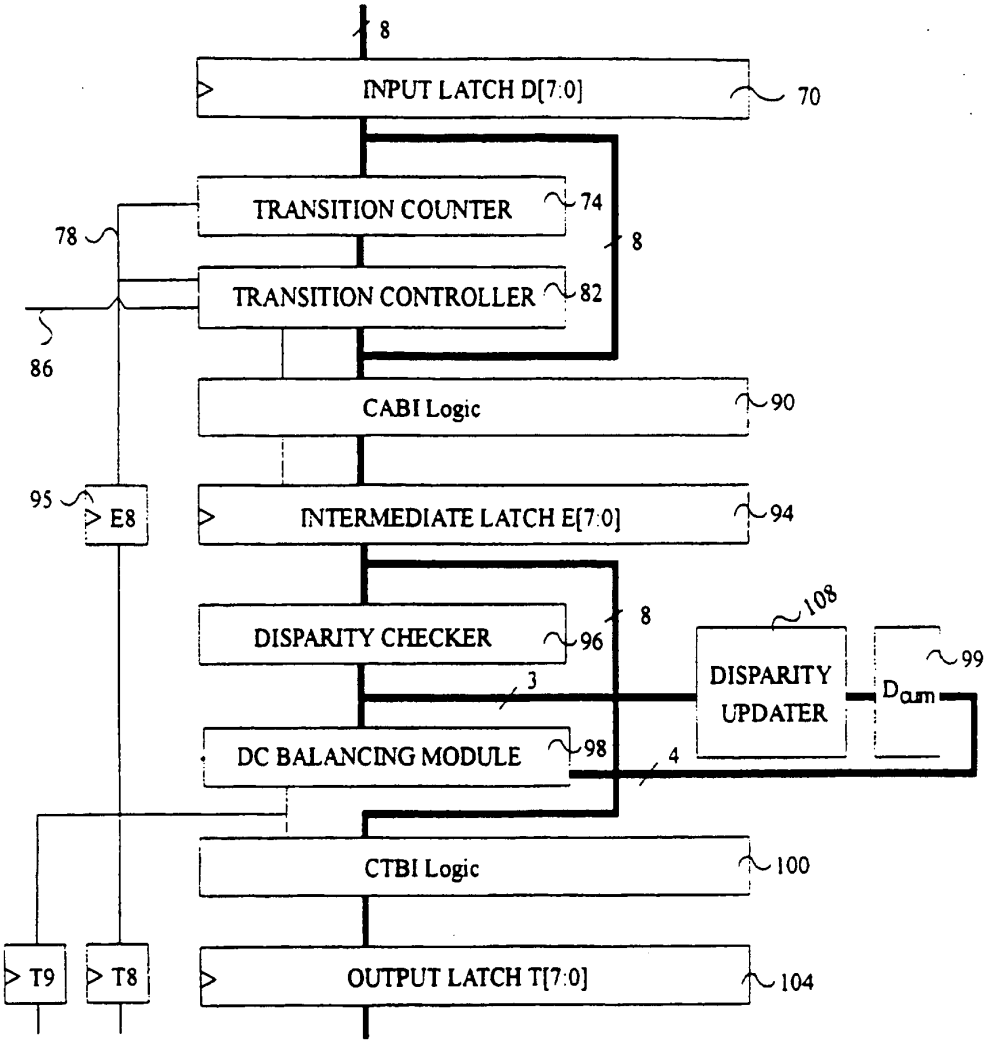

8
INPUT LATCH D[7:0]
70
TRANSITION COUNTER
74
78
8
TRANSITION CONTROLLER
82
86
CABI Logic
90
95
E8
INTERMEDIATE LATCH E[7:0]
94
DISPARITY CHECKER
96
8
108
DISPARITY UPDATER
99
D_{adm}
3
DC BALANCING MODULE
98
4
CTBI Logic
100
T9
T8
OUTPUT LATCH T[7:0]
104

FIG. 2

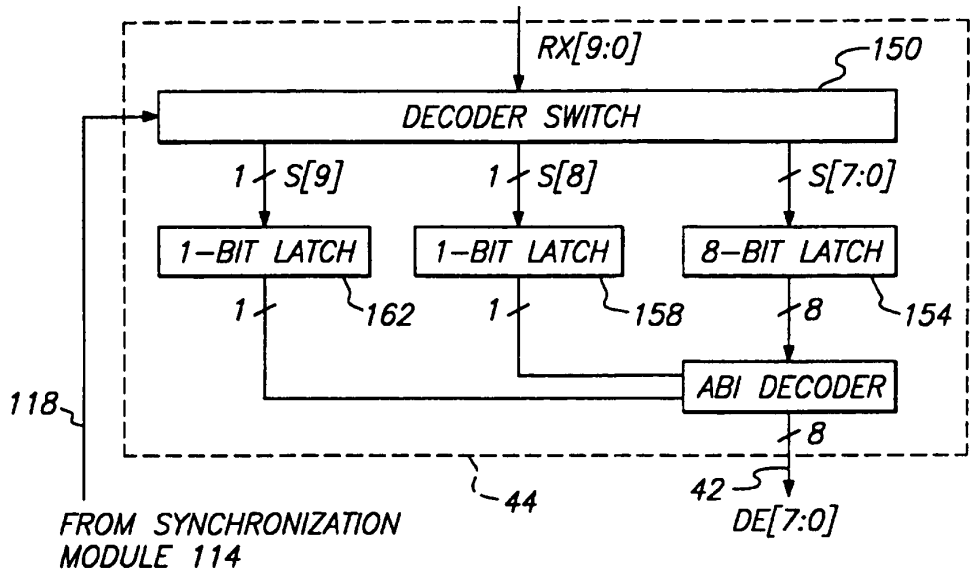


FIG. 3

IPR2024-001428
HP – EX1002, Page 76 of 487

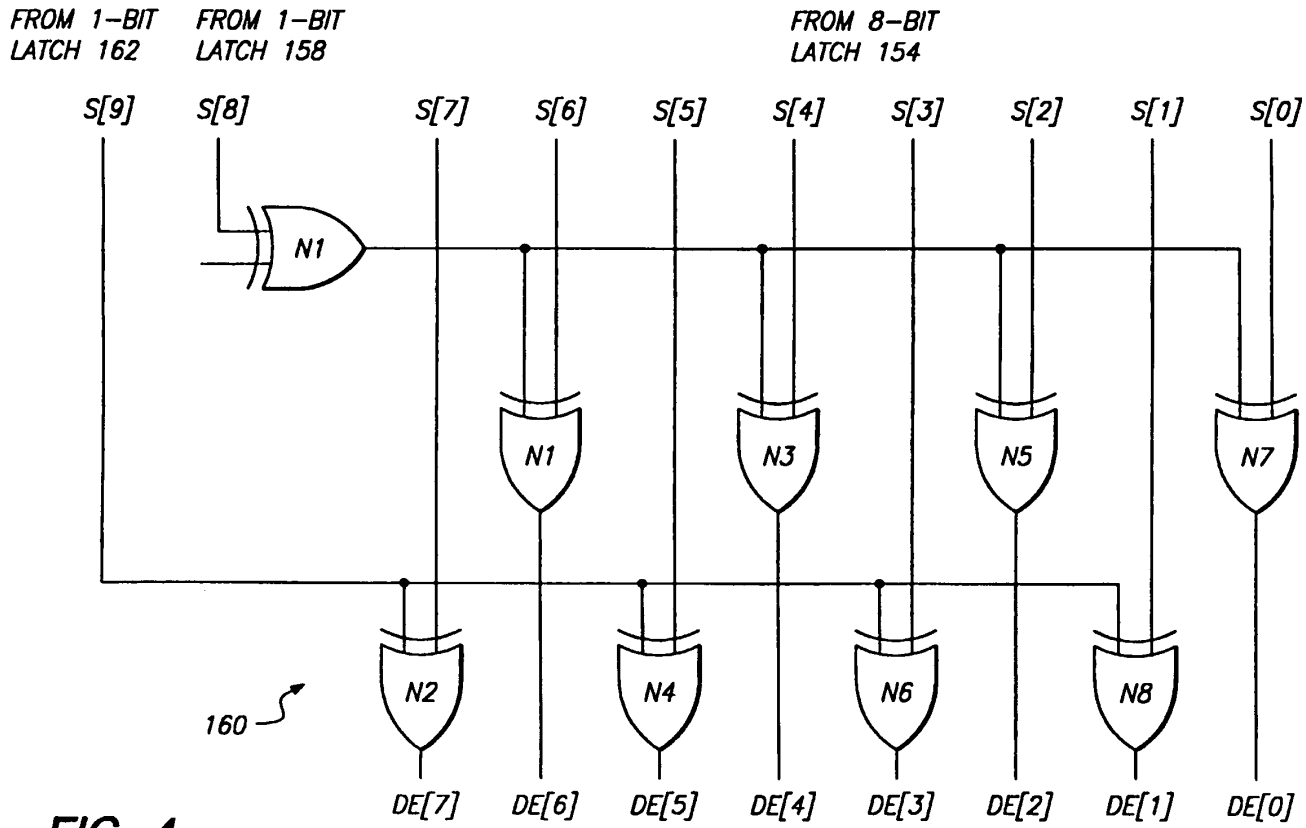


FIG. 4

SUBSTITUTE SHEET (RULE 26)

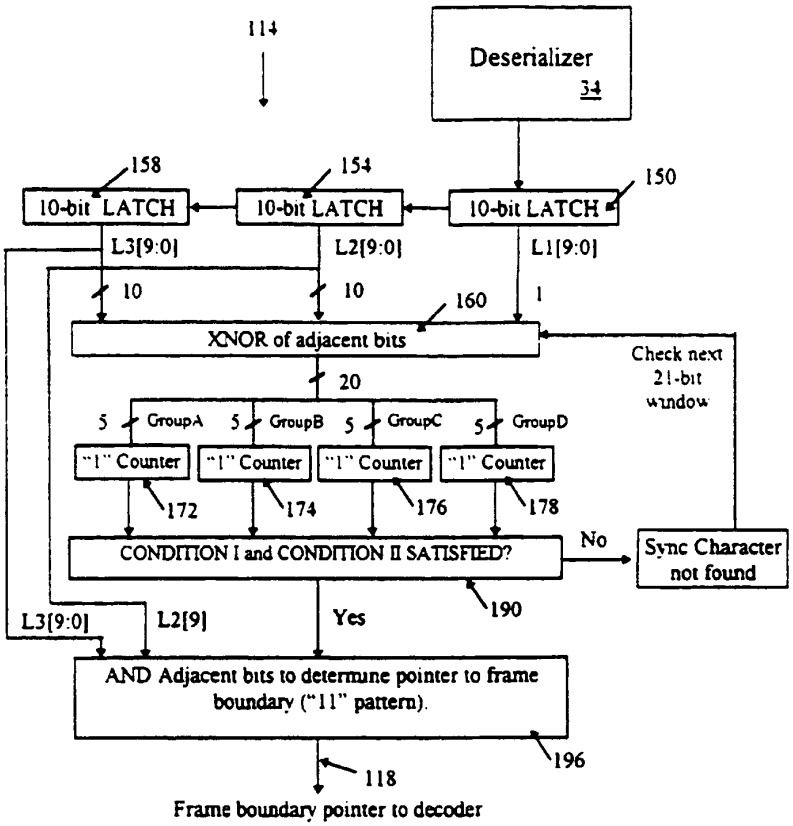

114
Deserializer
34
158
154
150
10-bit LATCH
10-bit LATCH
10-bit LATCH
L3[9:0]
L2[9:0]
L1[9:0]
10
10
1
160
XNOR of adjacent bits
Check next
21-bit
window
20
5
GroupA
5
GroupB
5
GroupC
5
GroupD
"1" Counter
"1" Counter
"1" Counter
"1" Counter
172
174
176
178
CONDITION I and CONDITION II SATISFIED?
No
Sync Character
not found
190
L3[9:0]
L2[9]
Yes
AND Adjacent bits to determine pointer to frame
boundary ("11" pattern).
196
118
Frame boundary pointer to decoder

FIG. 5

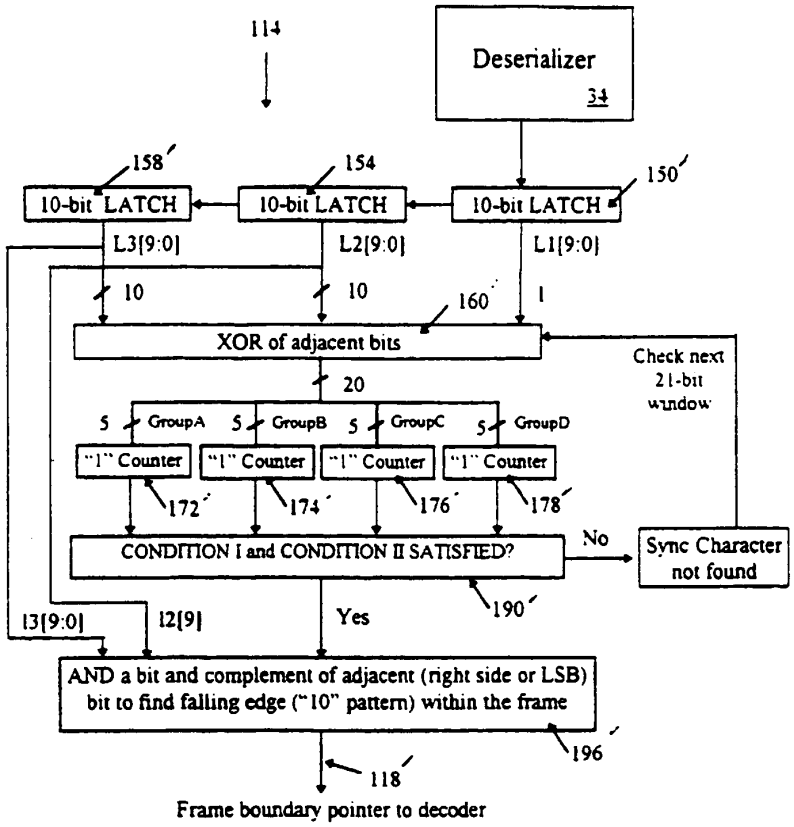

114
Deserializer
34
158
154
150
10-bit LATCH
10-bit LATCH
10-bit LATCH
L3[9:0]
L2[9:0]
L1[9:0]
10
10
160
1
XOR of adjacent bits
Check next 21-bit window
20
5 GroupA
5 GroupB
5 GroupC
5 GroupD
"1" Counter
"1" Counter
"1" Counter
"1" Counter
172
174
176
178
CONDITION I and CONDITION II SATISFIED?
No
Sync Character not found
190
I3[9:0]
I2[9]
Yes
AND a bit and complement of adjacent (right side or LSB) bit to find falling edge ("10" pattern) within the frame
196
118
Frame boundary pointer to decoder

Fig. 6

SUBSTITUTE SHEET (RULE 26)

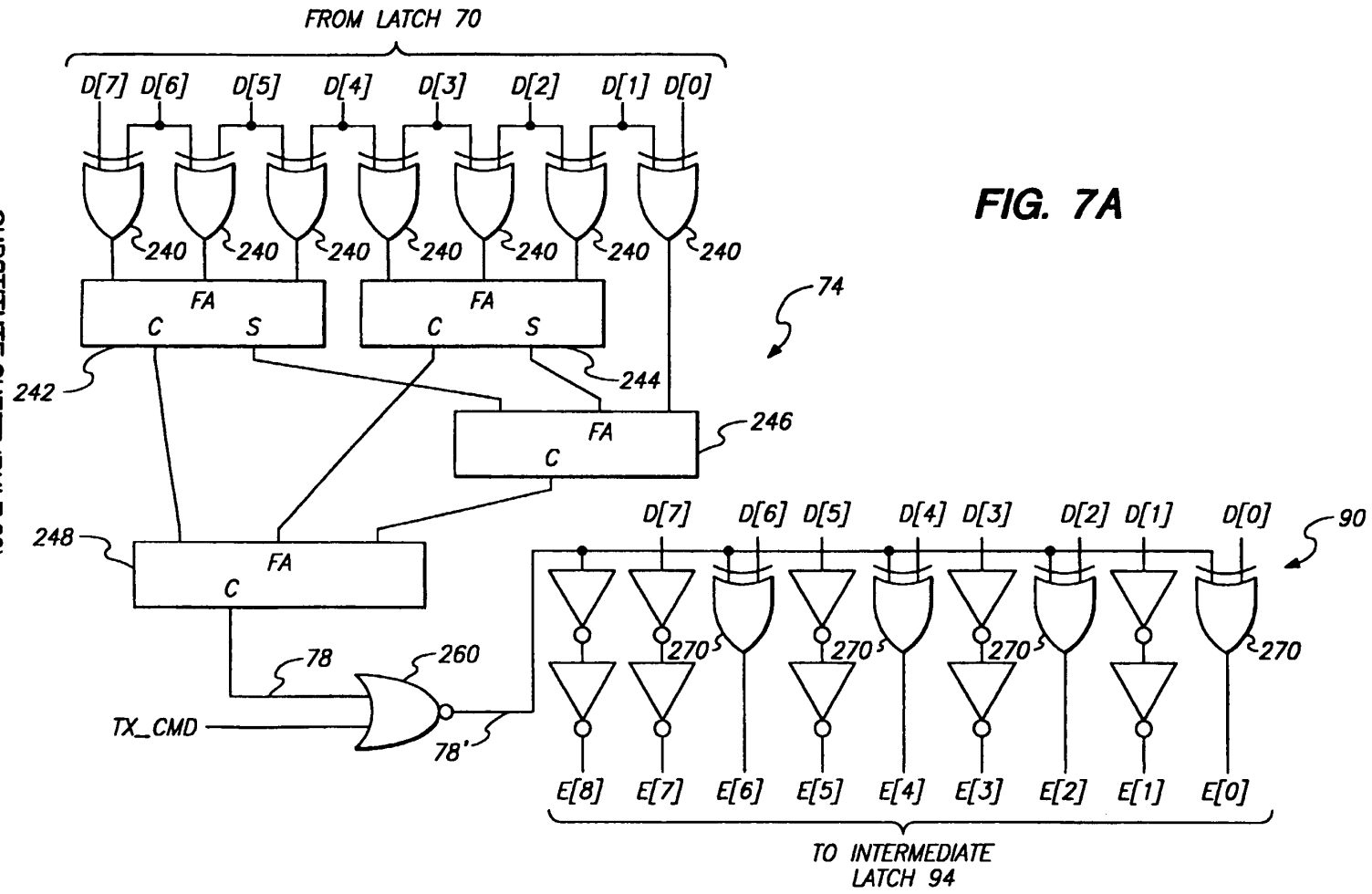
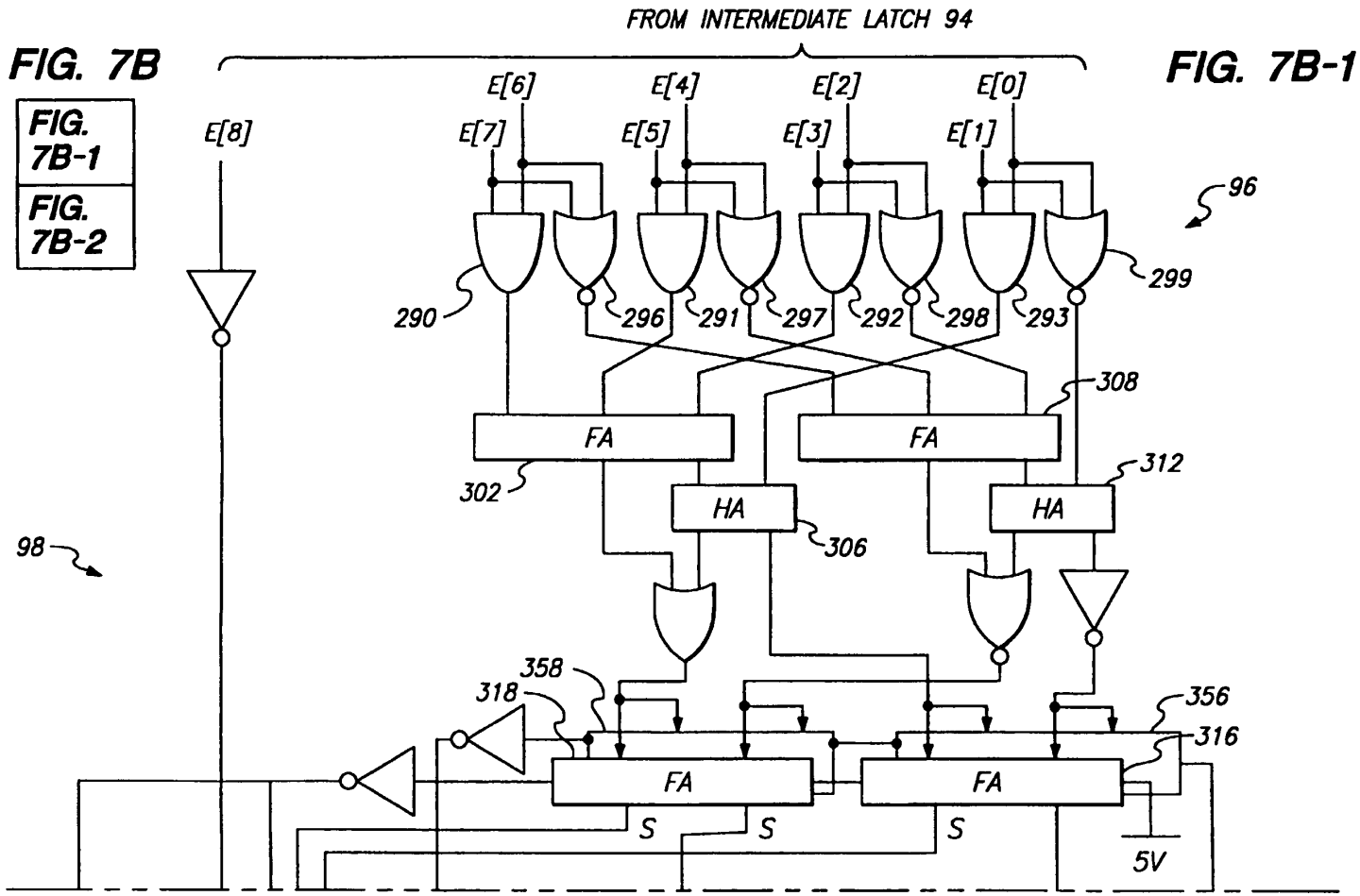


FIG. 7A

FIG. 7B

FIG. 7B-1
FIG. 7B-2

FIG. 7B-1



SUBSTITUTE SHEET (RULE 26)

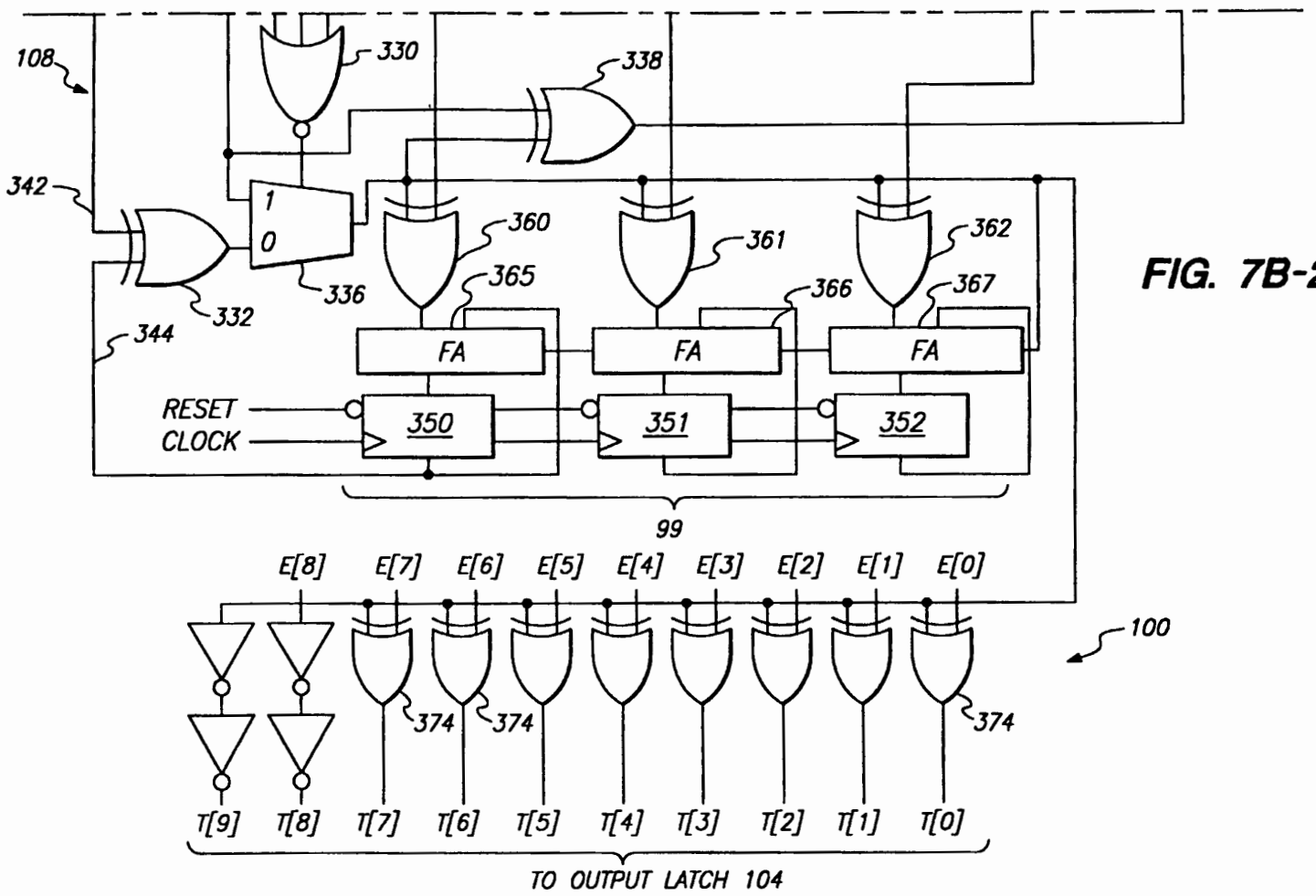


FIG. 7B-2

SUBSTITUTE SHEET (RULE 26)

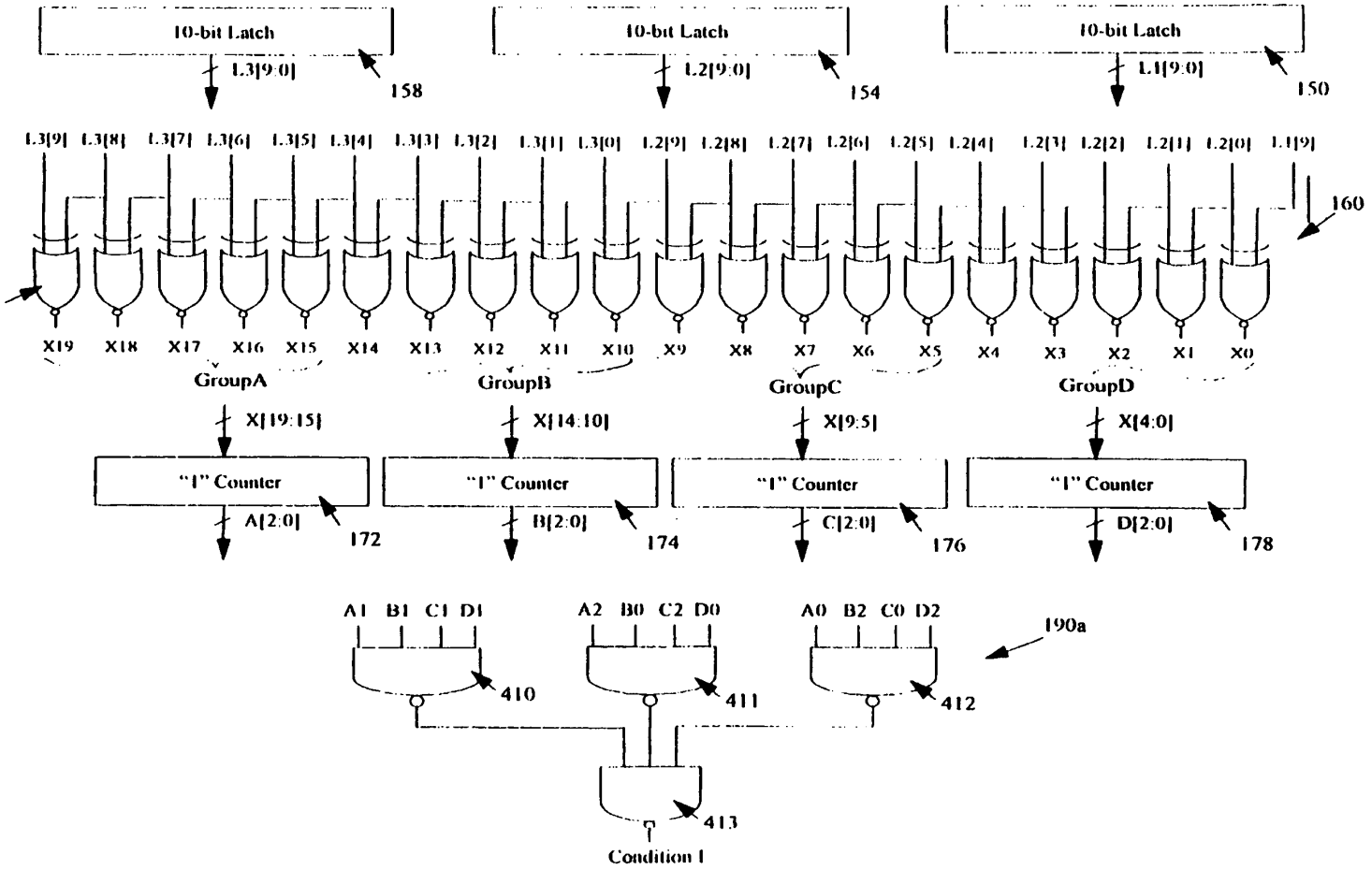

10-bit Latch
L3[9:0]
158
10-bit Latch
L2[9:0]
154
10-bit Latch
L1[9:0]
150
L3[9] L3[8] L3[7] L3[6] L3[5] L3[4] L3[3] L3[2] L3[1] L3[0] L2[9] L2[8] L2[7] L2[6] L2[5] L2[4] L2[3] L2[2] L2[1] L2[0] L1[9]
160
X19 X18 X17 X16 X15 X14 X13 X12 X11 X10 X9 X8 X7 X6 X5 X4 X3 X2 X1 X0
GroupA
GroupB
GroupC
GroupD
X[19:15]
X[14:10]
X[9:5]
X[4:0]
"1" Counter
"1" Counter
"1" Counter
"1" Counter
A[2:0]
172
B[2:0]
174
C[2:0]
176
D[2:0]
178
A1 B1 C1 D1
A2 B0 C2 D0
A0 B2 C0 D2
190a
410
411
412
413
Condition 1

FIG. 8A

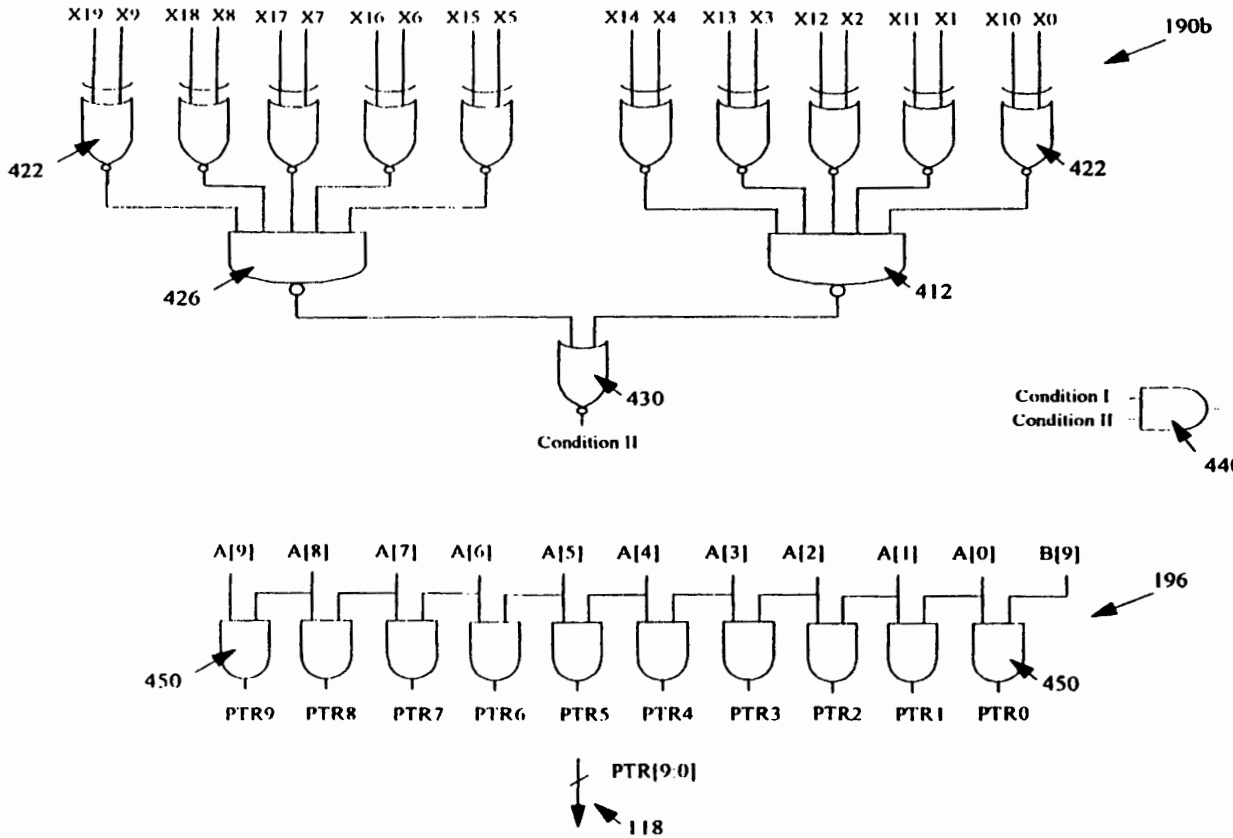

X19 X9
X18 X8
X17 X7
X16 X6
X15 X5
X14 X4
X13 X3
X12 X2
X11 X1
X10 X0
190b
422
422
426
412
430
Condition II
Condition I
Condition II
Sync_Sig
440
A[9]
A[8]
A[7]
A[6]
A[5]
A[4]
A[3]
A[2]
A[1]
A[0]
B[9]
196
450
450
PTR9
PTR8
PTR7
PTR6
PTR5
PTR4
PTR3
PTR2
PTR1
PTR0
PTR[9:0]
118
Frame Boundary Pointer to Decoder 44

FIG. 8B

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : **H04L 25/49, 25/08** | **A3**

(11) International Publication Number: **WO 97/13347**

(43) International Publication Date: 10 April 1997 (10.04.97)

(21) International Application Number: PCT/US96/15661

(22) International Filing Date: 30 September 1996 (30.09.96)

(30) Priority Data:
08/539,816 5 October 1995 (05.10.95) US

(71) Applicant: SILICON IMAGE, INC. [US/US]; Suite 222, 1032 Elwell Court, Palo Alto, CA 94303 (US).

(72) Inventors: SHIN, Yeshik; 244-303, Sangdo 4 dong, 16/3 Dongjak-ku, Seoul 156-034 (KR). LEE, Kyeongho; 93-102, Sangsu-dong Mapo Gu, Seoul (KR). KIM, Sungjoon; 5-1107 Mujigaeo Apartment, Seocho-dong, Seocho-gu, Seoul (KR). LEE, David, D.; 3715 Redwood Circle, Palo Alto, CA 94306 (US).

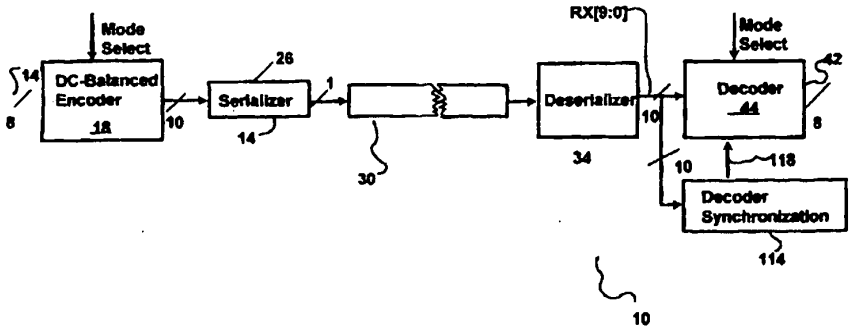
(74) Agents: OKAMOTO, James, K. et al.; Fenwick & West L.L.P., Suite 700, Two Palo Alto Square, Palo Alto, CA 94306 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report:
31 July 1997 (31.07.97)

(54) Title: TRANSITION-CONTROLLED DIGITAL ENCODING AND SIGNAL TRANSMISSION SYSTEM



(57) Abstract

A method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes is disclosed herein. The bits in each of the data bytes are selectively complemented in accordance with the number of logical transitions in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity. In a high-transition mode of operation, the bits within data blocks including less than a minimum number of logical transitions are selectively complemented so that each such selectively complemented data block includes in excess of the minimum number of logical transitions. In a low-transition mode of operation, the bits within data blocks having more than a predefined number of logical transitions are selectively complemented so that each such selectively complemented data block includes less than the maximum number of logical transitions.

IPR2024-001428
HP – EX1002, Page 85 of 487

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	**GB**	United Kingdom	**MW**	Malawi
AT	Austria	**GE**	Georgia	**MX**	Mexico
AU	Australia	**GN**	Guinea	**NE**	Niger
BB	Barbados	**GR**	Greece	**NL**	Netherlands
BE	Belgium	**HU**	Hungary	**NO**	Norway
BF	Burkina Faso	**IE**	Ireland	**NZ**	New Zealand
BG	Bulgaria	**IT**	Italy	**PL**	Poland
BJ	Benin	**JP**	Japan	**PT**	Portugal
BR	Brazil	**KE**	Kenya	**RO**	Romania
BY	Belarus	**KG**	Kyrgystan	**RU**	Russian Federation
CA	Canada	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CF	Central African Republic			**SE**	Sweden
CG	Congo	**KR**	Republic of Korea	**SG**	Singapore
CH	Switzerland	**KZ**	Kazakhstan	**SI**	Slovenia
CI	Côte d'Ivoire	**LI**	Liechtenstein	**SK**	Slovakia
CM	Cameroon	**LK**	Sri Lanka	**SN**	Senegal
CN	China	**LR**	Liberia	**SZ**	Swaziland
CS	Czechoslovakia	**LT**	Lithuania	**TD**	Chad
CZ	Czech Republic	**LU**	Luxembourg	**TG**	Togo
DE	Germany	**LV**	Latvia	**TJ**	Tajikistan
DK	Denmark	**MC**	Monaco	**TT**	Trinidad and Tobago
EE	Estonia	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**UG**	Uganda
FI	Finland	**ML**	Mali	**US**	United States of America
FR	France	**MN**	Mongolia	**UZ**	Uzbekistan
GA	Gabon	**MR**	Mauritania	**VN**	Viet Nam

INTERNATIONAL SEARCH REPORT

Interna l Application No
PCT/US 96/15661

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04L25/49 H04L25/08

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)
IPC 6 H04L H03M

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)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 92 22160 A (NORTHERN TELECOM) 10 December 1992 see abstract; figures 3,4 see page 1, line 26 - page 2, line 2 see page 2, line 8 - line 11 see page 2, line 17 - line 20 see page 4, line 11 - line 26 see page 8, line 5 - line 12	1,2, 4-10, 12-16,35
A	---	3,11
A	EP 0 556 981 A (ADVANCED MICRO DEVICES) 25 August 1993 see page 2, line 3 - line 7 see page 2, line 19 - line 21 see page 2, line 47 - line 53; table 1	1-16
	--- -/--	

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

[X] Patent family members are listed in 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

2

Date of the actual completion of the international search	Date of mailing of the international search report
8 April 1997	10.06.97
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer SCRIVEN P.

Form PCT/ISA/210 (second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 87 of 487

INTERNATIONAL SEARCH REPORT

Interna d Application No
PCT/US 96/15661

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	ELECTRONICS LETTERS, vol. 19, no. 6, March 1983, STEVENAGE GB, pages 202-203, XP000670400 WIDMER & FRANASZEK: "Transmission code for high-speed fibre-optic data networks" see abstract; figure 1 see page 202, right-hand column, paragraph 3 see page 202, right-hand column, paragraph 5 ---	1-16
A	WO 88 06384 A (UNIVERSITY OF ILLINOIS) 25 August 1988 see abstract; figure 1 see page 1, line 14 - line 20 see page 2, line 11 - line 25 see page 8, line 30 - page 9, line 6 see page 9, line 26 - line 29 -----	1-16

2

IPR2024-001428
HP – EX1002, Page 88 of 487

INTERNATIONAL SEARCH REPORT

Int uonal application No.
PCT/US 96/ 15661

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see extra sheet

1. ☐ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1-16,35

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest.

☐ No protest accompanied the payment of additional search fees.

IPR2024-001428
HP – EX1002, Page 89 of 487

INTERNATIONAL SEARCH REPORT

International Application No. PCT/US 96/ 15661

FURTHER INFORMATION CONTINUED FROM PCT/ISA/210

1. claims 1-16,35:	DC balanced encoding, in particular, for serial transmission, of eight bit words, allowing control of the number of transitions in the code words
2. claims 17-26:	transmission of data and control signals, using a code which distinguished data and control code words on the basis of the number of transitions
3. claims 27-34:	detection of the boundary of a synchronisation character

INTERNATIONAL SEARCH REPORT

Intern. al Application No
PCT/US 96/15661

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9222160 A	10-12-92	US 5200979 A	06-04-93
		CA 2109428 A	07-12-92
		DE 69216111 D	30-01-97
		DE 69216111 T	17-04-97
		EP 0610204 A	17-08-94
		JP 6503217 T	07-04-94
EP 0556981 A	25-08-93	US 5387911 A	07-02-95
		JP 5284180 A	29-10-93
WO 8806384 A	25-08-88	US 4864303 A	05-09-89
		EP 0417072 A	20-03-91
		JP 2502148 T	12-07-90

Form PCT/ISA/210 (patent family annex) (July 1992)

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☑ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 7/033 | A1

(11) International Publication Number: WO 99/12306

(43) International Publication Date: 11 March 1999 (11.03.99)

(21) International Application Number: PCT/US98/18886

(22) International Filing Date: 4 September 1998 (04.09.98)

(30) Priority Data:
60/058,040 4 September 1997 (04.09.97) US

(71) Applicant: SILICON IMAGE, INC. [US/US]; 10131 Bubb Road, Cupertino, CA 95014 (US).

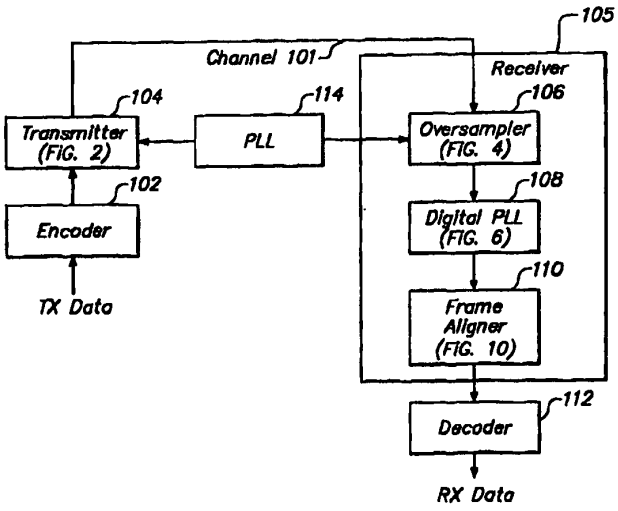
(72) Inventors: JEONG, Deog-Kyoon; 106-706, Daewoo-HyoRyong Apt., 1038, Bangbae-3-dongu, Seocho-ku, Seoul 137-063 (KR). AHN, Gijung; 523 Walker Drive #4, Mountain View, CA 94043 (US).

(74) Agents: OKAMOTO, James, K. et al.; Fenwick & West LLP, Two Palo Alto Square, Palo Alto, CA 94306 (US).

(81) Designated States: CA, JP, KR, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: SYSTEM AND METHOD FOR HIGH-SPEED, SYNCHRONIZED DATA COMMUNICATION



(57) Abstract

A system for transmission and recovery of original digital data includes an encoder, a transmitter, a receiver, a decoder, and an analog phase locked loop. The analog phase locked loop supplies a sender's clock to the transmitter and a receiver's clock to the receiver, where the sender's clock frequency is a first integer multiple of the system clock frequency, and the receiver's clock frequency is a second integer multiple of the sender's clock frequency within 0.1 % tolerance. In a normal flow situation, data frames are output by the receiver in alternate cycles of the system clock. In an overflow situation, data frames are output by the receiver in consecutive cycles of the system clock. In an underflow situation, data frames are not output by the receiver in consecutive cycles of the system clock.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

SYSTEM AND METHOD FOR HIGH-SPEED, SYNCHRONIZED DATA COMMUNICATION

BACKGROUND OF THE INVENTION

Technical Field

This invention relates to data communication systems. More particularly, this invention relates to the recovery using oversampling in a receiving system of digital data transmitted via a high-speed link.

Description of Related Art

As electronic and computer technology continues to evolve, communication of information among different devices, either situated near by or at a distance becomes increasingly important. For example, it is now more desirable than ever to provide for high speed communications among different chips on a circuit board, different circuit boards in a system, and different systems with each other. It is also increasingly desirable to provide such communications at very high speeds, especially in view of the large amount of data required for data communications in intensive data consuming systems using graphical or video information, multiple input-output channels, local area networks, and the like.

It is particularly desirable to enable individual personal computers, workstations, or other computing devices, within which data is normally internally transferred using parallel data buses, to communicate with each other over relatively simple transmission lines. Such transmission lines typically include only one or two conductors, in contrast with the 64-bit and wider data paths within computing systems now commonly available.

A communication system that includes oversampling is often utilized to recover data transmitted data. Such a system includes a receiver which samples the incoming serial data stream at a rate greater than the rate at which symbols (bits) are being transmitted. For example, in a three-times (3x) oversampling receiver, the incoming data stream is sampled at a rate

order to effectively implement such a receiver when the rate of data transmission is very high.

One problem is the difficulty in handling overflow and/or underflow of incoming data at the receiving end. For example, usually the output of a 3x oversampler contains sets of three consecutive 1s or 0s, but occasionally 4 or 2 consecutive 1s or 0s may occur. Such overflow and/or underflow will occasionally occur, for example, when the frequency of the local clock of the transmission system is not in precise synchronization with the local clock of the receiving system, when there is jitter in the sampling clock, and so on.

Other problems include difficulties in determining the correct symbol (bit) boundaries in the oversampled data stream and difficulties in determining the correct values of the symbols from the multiple samplings of each symbol.

SUMMARY OF THE INVENTION

The above described needs are met and problems are solved by the present invention. A system for transmission and recovery of original digital data includes an encoder, a transmitter, a receiver, a decoder, and an analog phase locked loop. The analog phase locked loop supplies a sender's clock to the transmitter and a receiver's clock to the receiver, where the sender's clock frequency is a first integer multiple of the system clock frequency, and the receiver's clock frequency is a second integer multiple of the sender's clock frequency within 0.1% tolerance. In a normal flow situation, data frames are output by the receiver in alternate cycles of the system clock. In an overflow situation, data frames are output by the receiver in consecutive cycles of the system clock. In an underflow situation, data frames are not output by the receiver in consecutive cycles of the system clock.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a communication system for transmission and recovery of digital data via a communication channel according to a preferred embodiment of the present invention.

Fig. 2 is a block diagram of a transmitter within the communication system according to a preferred embodiment of the present invention.

IPR2024-001428
HP – EX1002, Page 96 of 487

present invention.

Fig. 4 is a block diagram of an oversampler located in the receiving side of the communication system according to a preferred embodiment of the present invention.

Fig. 5 is a timing diagram for the oversampler according to a preferred embodiment of the present invention.

Fig. 6 is a block diagram of a digital phase locked loop (digital PLL) located in the receiving side of the communication system according to a preferred embodiment of the present invention.

Fig. 7 is a diagram illustrating the mapping of the oversampled data to groups and phases within the digital PLL according to a preferred embodiment of the present invention.

Fig. 8 is a diagram illustrating an example of data selection within the digital PLL according to a preferred embodiment of the present invention.

Fig. 9 is a diagram illustrating the operation of data alignment within the digital PLL according to a preferred embodiment of the present invention.

Fig. 10 is a block diagram of a frame aligner located in the receiving side of the communication system according to a preferred embodiment of the present invention.

Fig. 11 is a diagram illustrating the operation of an aligned FIFO within the frame aligner according to a preferred embodiment of the present invention.

Fig. 12 is a timing diagram illustrating the output from the frame aligner according to a preferred embodiment of the present invention.

Fig. 13 is a diagram illustrating the usage of the various bits and ranges of bits in the aligned FIFO within the frame aligner according to a preferred embodiment of the present invention.

Fig. 14 is a state diagram of the operation of the frame aligner 110. Four states are shown:

IPR2024-001428
HP – EX1002, Page 97 of 487

the graph arrows A-H in the state diagram of Fig. 14.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Overall System

Fig. 1 is a block diagram of a communication system for transmission and recovery of digital data via a communication channel 101 according to a preferred embodiment of the present invention. The communication system includes the channel 101, an encoder 102, a transmitter 104, a receiver 105, a decoder 112, and a phase locked loop (PLL) 114. The receiver 105 includes an oversampler 106, a digital PLL 108, and a frame aligner 110.

The channel 101 comprises a high-speed point-to-point link in a preferred embodiment of the present invention. Such a link includes a transmission line made from twisted pair, coaxial cable, or some other implementation of the physical layer.

The encoder 102 receives original data to be transmitted (TX data) from a source (not shown). For example, the original data may be graphics data from a computer for display on a graphics monitor, and the encoder 102 may be embodied within a chip on a graphics card in the computer. In a preferred embodiment, the original data is received in parallel form from the source. The source may, for example, comprise of a graphics processor chip, or a bus of a workstation, server, or personal computer. The encoder 102 encodes the original data in order to make it more suitable for transmission over the channel 101. Such encoding may, for example, transform the data to incorporate error-correction capabilities, minimize transitions in the data to reduce electromagnetic interference, incorporate special signaling (such as synchronization characters), and so on. The encoded data may include more bits than the data before encoding. For example, the data before encoding may be organized in 8-bit frames (bytes). The encoding process may, for example, convert each of those 8-bit frames to 10-bit frames. In other embodiments, different encoding schemes may be employed, such as 8-to-9 bit encoding schemes, or encoding schemes from m bits to n bits. After the data is encoded, the encoder 102 outputs the encoded data to the transmitter 104 for transmission over the channel 101. In a preferred embodiment, the transmitter 104 receives the encoded data in parallel form from the encoder 102 and converts the encoded data from parallel form to serial form for output as an encoded data waveform to the channel 101. In addition, in a preferred embodiment, the transmitter 104 receives a sender's clock signal from the PLL 114 in order to synchronize its

IPR2024-001428
HP – EX1002, Page 98 of 487

encoded data waveform to the channel 101 in synchronization with the sender's clock signal.

The receiver 105 receives the encoded data waveform from the channel 101. Within the receiver 105, the encoded data waveform is received by the oversampler 106. In a preferred embodiment, the oversampler 106 also receives a receiver's clock signal from the PLL 114 in order to synchronize its operation with that of the transmitter 104. The oversampler 106 obtains an integer multiple number of samples per symbol from the encoded data waveform to generate an oversampled data stream, and so the receiver's clock signal which has a frequency that is an integer multiple of the frequency of the sender's clock signal. In a preferred embodiment, the receiver's clock frequency is three times that of the sender's clock frequency to within a tolerance of 0.1%. For example, if the sender's clock is 100 MHz, then the receiver's clock would be 300 MHz. In other words, on average three samples are obtained per symbol from the encoded data stream. The oversampler 106 outputs the oversampled data stream to the digital PLL 108. The digital PLL 108 receives the oversampled data stream and extracts clock information from edges (data transitions from 1 to 0, or 0 to 1) in the oversampled data stream. In a preferred embodiment, the digital PLL 108 also generates left-aligned data frames and related flag information for output to the frame aligner 110. The frame aligner 110 receives the left-aligned data frames and the related flag information and outputs aligned data frames and output valid signal to the decoder 112. Finally, the decoder 112 receives the aligned data frames and output valid signal and decodes the aligned data frames to regenerate the original data and output the regenerated data as the received data (RX data). In a preferred embodiment, a liquid crystal display (LCD) controller chip receives the RX data from the decoder 112 and has the RX data displayed graphically on the display.

Transmitter

Fig. 2 is a block diagram of a transmitter 104 within the communication system according to a preferred embodiment of the present invention. The transmitter 104 includes a timing aligner 202, a data selector 204, a channel driver 206, and a signal level controller 208.

Fig. 3 is a timing diagram for the transmitter 104 according to a preferred embodiment of the present invention. The signals shown in Fig. 3 include the system clock signal 302, the encoded parallel data signal 304, a first half clock delayed data signal 306, and data selector output signal 308.

IPR2024-001428
HP – EX1002, Page 99 of 487

receives the encoded parallel data signal 304 from the encoder 102. For purposes of illustration, the encoded parallel data signal 304 in particular includes 10-bit data frames with bits labeled D[0:9]. In this embodiment, the timing aligner 202 delays the last 5 bits of each data frame by half of a system clock cycle to generate the first half clock delayed data signal 306.

The data selector 204 converts the encoded data from parallel to serial form. In this embodiment, both the parallel data signal 304 and the delayed data signal 306 are transferred to the data selector 204. The data selector 204 also receives the sender's clock signal from PLL 114. The frequency of the sender's clock signal in this particular case is ten (10) times the frequency of the system clock signal 302. The data selector output signal 308 is synchronized to the sender's clock signal.

The channel driver 206 receives the data selector output signal 308 and drives that signal onto the channel 101 in the form of an encoded data waveform. Signal level controller 208 controls the strength of the encoded data waveform driven by the channel driver 206.

Oversampler

Fig. 4 is a block diagram of an oversampler 106 located in the receiving side of the communication system according to a preferred embodiment of the present invention. The oversampler 106 includes a channel impedance controller 402, a channel amplifier 404, a sampler 406, and a timing aligner 408.

Fig. 5 is a timing diagram for the oversampler 106 according to a preferred embodiment of the present invention. The signals shown in Fig. 5 include the system clock signal 302, sampling timing signal 502, second half clock delayed data signal 504, a series of sample data signals 506-0, 506-1, 506-2, ..., 506-(xn-2), and 506-(xn-1) (where n = the number of bits in a data frame of the encoded data, and x = a multiplier due to the oversampling), and timing aligner output signal 508. In the particular example shown in Fig. 5, n = 10, x = 3 (for 3x oversampling), so xn = 30.

The channel impedance controller 402 allows adjustments to be made to the input impedance of the receiver 105 so that the input impedance matches the channel impedance. The channel amplifier 404 receives the encoded data waveform and amplifies and/or shifts it so that the sampler 406 can better sense the features in the waveform.

IPR2024-001428
HP – EX1002, Page 100 of 487

amplifier 404 and the receiver's clock signal from the PLL 114. In the particular example illustrated in Fig. 5, the receiver's clock signal would have a frequency 30 times that of the system clock signal. The sampler 406 takes samples of the amplified waveform at a sampling frequency corresponding to the receiver's clock frequency. Thus, in a single system clock cycle, thirty (30) samples are taken in this particular case. These 30 samples are shown as S0 through S29 in the sampling timing signal 502, and the 30 samples correspond to a single oversampled data frame.

For the sampler 406 whose operation is illustrated in Fig. 5, the sampler 406 holds the sampled data for a duration of a half cycle of the system clock 302 and precharges circuits for a duration of the other half cycle. This is illustrated by the series of sample data signals 506-0, 506-1, 506-2, ..., 506-28, and 506-29 in Fig.. 5. Therefore, for this sampler 406, it is needed for a device to hold the sampled data for a duration of the other half cycle. Such a device is the timing aligner 408. The timing aligner 408 holds the first half of the oversampled data frame (i.e. the first 15 of the 30 samples in this particular case) for the other half cycle. The timing aligner 408 also synchronizes the entire oversampled data frame and also outputs 508 the entire oversampled data frame beginning at the next cycle of the system clock signal 302.

Digital PLL

Fig. 6 is a block diagram of a digital phase locked loop (digital PLL) 108 located in the receiving side of the communication system according to a preferred embodiment of the present invention. One function of the digital PLL 108 is to extract clock information, such as phase information, from the oversampled data stream 702. The digital PLL 108 includes an edge extractor 602, an edge counter 604, a data select controller 606, a data delay device 608, a data selector 610, a last 1-bit delay 612, and a data left-side aligner 614. Fig. 7 is a diagram illustrating the mapping of an oversampled data stream 702 to groups 706 and phases 704 within the digital PLL 108 according to a preferred embodiment of the present invention.

A current oversampled data stream 702 is received by the data delay device 608. The data delay device 608 holds the current oversampled data stream 702 and makes the current oversampled data stream 702 available to the data selector 610 until the data select controller 606 has determined the data selection position. The last 1-bit delay device 612 holds the last bit sample S29 from the previous oversampled data stream 702 and makes the last bit sample S29 available to the data selector 610 until the data select controller 606 has determined the data

IPR2024-001428
HP – EX1002, Page 101 of 487

device 612 replaces the last bit sample S29 from the previous oversampled data stream 702 with the last bit sample S29 from the current oversampled data stream 702.

Fig. 8 is a diagram illustrating an example of data selection within the digital PLL 108 according to a preferred embodiment of the present invention. A current oversampled data stream 802 is received by the edge extractor 602 from the oversampler 106. The edge extractor 602 determines the locations of data transitions (edges) within the current oversampled data stream 802 and outputs an edge extractor signal 804. The edge extractor signal 804 has a value of one at locations where edges are found and has a value of zero elsewhere (where no edges are found). The edge counter 604 receives the edge extractor signal 804 and determines the phase 806 of any edge in each group of the current oversampled data stream 802. The edge counter 604 counts the number of edges with each phase. For a 3x oversampling system, the edge counter 604 counts the number of edges in phase0 (edge0), phase1 (edge1), and phase2 (edge2). The data select controller 606 uses phase information about the edges from the edge counter 604 to determine a data selection position. For the example shown in Fig. 8, the data selection position is phase2. The data selector 610 uses the data selection position from the data select controller 606 to select a received data stream 812 from the current oversampled data stream 802.

Fig. 9 is a diagram illustrating the operation of data alignment within the digital PLL 108 according to a preferred embodiment of the present invention. The data left-side aligner 614 receives as input 902 the last sample bit 803 [S29] from the last 1-bit delay device 612 and receives the received data stream 812 [D0:D9] from the data select controller 606. In this embodiment, the received data stream 812 comprises 10 bits [D0:D9]. In this embodiment, the data left-side aligner 614 generates as output 904 eleven data bits [Y0:Y10] and 2 flag bits [V9 and V10]. In this embodiment, the eleven data bits and the two flag bits are determined as follows.

If the data selection position 810 for the previous oversampled data stream was phase0 [P0] and the data selection position 810 for the current oversampled data stream 802 is phase2 [P2], then, as shown by line 906, Y0 = S29, Y1 = D0, Y2 = D1, Y3 = D2, Y4 = D3, ..., Y9 = D8, and Y10 = D9, and V9 = V10 = 1. V9 and V10 represent valid bits for data bits Y9 and Y10, respectively. In this case, V9 and V10 both being one indicates that Y9 and Y10 both hold

IPR2024-001428
HP – EX1002, Page 102 of 487

bit in the received data stream 812.

On the other hand, if the data selection position 810 for the previous oversampled data stream was phase2 [P2] and the data selection position 810 for the current oversampled data stream 802 is phase0 [P0], then, as shown by line 908, Y0 = D1, Y1 = D2, ..., Y7 = D8, and Y8 = D9, and V9 = V10 = 0. In this case, V9 and V10 both being zero indicates that neither Y9 nor Y10 hold a valid data bit. Furthermore, in this case, D0 is discarded in order to remove a duplicated bit in the received data stream 812.

Finally, in other cases that do not fall into the two above-described situations, then, as shown by line 910, Y0 = D0, Y1 = D1, ..., Y8 = D8, and Y9 = D9, and V9 = 1, and V10 = 0. For these cases, V9 being one indicates that Y9 holds a valid bit, but V10 being zero indicates that Y10 does not hold a valid bit.

Frame Aligner

Fig. 10 is a block diagram of a frame aligner 110 located in the receiving side of the communication system according to a preferred embodiment of the present invention. The frame aligner 110 includes an aligned FIFO (first in first out) 1002, a data delay 1004, a frame selector 1006, a sync pattern finder 1008, and a frame controller 1010.

Fig. 11 is a diagram illustrating the operation of an aligned FIFO 1002 within the frame aligner 110 according to a preferred embodiment of the present invention. The aligned FIFO 1002 receives the output 904 (including 11 data bits [Y0:Y10] and 2 flag bits [V9 and V10]) from the data left-side aligner 614. In this embodiment, the aligned FIFO 1002 also contains 21 bits of data previously received from the data left-side aligner 614.

In the particular instance illustrated in Fig. 11, the "previous" data in aligned FIFO 1002 is shown in line 1102 and includes the following 32 bits: A9; B0-B9; C0-C9; and D0-D10. The "next" data in aligned FIFO 1002 are shown by line 1104 and are designated as the 32 bits X0-X31. The contents of the 32 bits X0-X31 depend upon the values of the two flag bits [V9 and V10] accompanying the 11 previous data bits D0-D10. If V9 and V10 were both zero, then D9 and D10 are not valid, so the next data in aligned FIFO 1002 are shown by line 1106. In this first case, the previous data in the aligned FIFO 1002 are shifted to the left by 9 bit places, and then the last 11 bit places X21-X31 are overwritten with newly received 11 data bits E0-E10. On the other hand, if V9 = 1 and V10 = 0, then D9 is valid, but D10 is not, so the next data in

IPR2024-001428
HP – EX1002, Page 103 of 487

FIFO 1002 are shifted to the left by 10 bit places, and then the last 11 bit places X21-X31 are overwritten with newly received 11 data bits E0-E10. Finally, if V9 and V10 are both one, then D9 and D10 are both valid, so the next data in aligned FIFO 1002 are shown by line 1110. In this third case, the previous data in the aligned FIFO 1002 are shifted to the left by 11 bit places, and then the last 11 bit places X21-X31 are overwritten with newly received 11 data bits E0-E10.

Fig. 12 is a timing diagram illustrating the output from the frame aligner 110 according to a preferred embodiment of the present invention. Fig. 13 is a diagram illustrating the usage of the various bits and ranges of bits in the aligned FIFO 1002 within the frame aligner 110 according to a preferred embodiment of the present invention.

The contents of the aligned FIFO 1002 are received by both the sync pattern finder 1008 and the data delay device 1004.

The data delay device 1004 selectively delays the contents of the aligned FIFO 1002 depending on the possible overflow or underflow situations. The data delay device 1004 outputs (possibly delayed) X0-X31 to the frame selector 1006. The frame selector 1006 selects a frame for output by the frame aligner 110. The frame is 20 bits wide and begins at a bit pointed to by a frame pointer. The frame pointer must lie within the frame pointer range 1304 of X0 to X10. Since the frame pointer range 1304 is from X0 to X10, and the frame is 20 bits wide, the window 1308 from which a frame may be possibly selected is from X0 to X29.

The sync pattern finder 1008 searches for a synchronization (sync) pattern within the contents of the aligned FIFO 1002. In this embodiment, a frame may be aligned with a single sync pattern which is 10 bits wide. Such an alignment would be desirable if misalignment occurs during communication. Since the frame pointer range 1304 is from X0 to X10, and the sync pattern is 10 bits wide, the window 1306 over which a sync pattern must be search is from X0 to X19.

If a sync pattern is found by the sync pattern finder 1008, then the frame is realigned by the frame controller 1010 sending a signal to have the frame selector 1006 change the frame pointer to a position which is dependent upon the start position of the sync pattern. In addition, the frame controller 1010 sends an output invalid signal to the decoder 112. On the other hand, if a sync pattern is not found, then the frame controller 1010 causes the frame selector 1006 to

IPR2024-001428
HP – EX1002, Page 104 of 487

addition, the frame controller 1010 sends an output valid signal to the decoder 112 when 20 bits of valid data are ready to be sent to the decoder 112..

The output from the frame aligner 110 is synchronized with the system clock signal 1202. In a normal case 1204, 20 bits of valid data are generated in every other cycle of the system clock. Fig. 12 shows the output valid signal 1204A and the output data signal 1204B by the frame aligner 110 for this normal situation. However, if the clock frequency of the transmitter is higher than that of the receiver, then once in a while an overflow case 1206 occurs. In an overflow case 1206, 20 bits of valid data are generated in two consecutive cycles of the system clock. Fig. 12 shows the output valid signal 1206A and the output data signal 1206B for this overflow situation. Furthermore, if the clock frequency of the receiver is higher than that of the transmitter, then once in a while an underflow case 1208 occurs. In an underflow case 1208, a delay of two consecutive cycles of the system clock is introduced so that the transmitter may "catch up" to the receiver. Fig. 12 shows the output valid signal 1208A and the output data signal 1208B for this underflow situation.

Fig. 14 is a state diagram of the operation of the frame aligner 110. Four states are shown: S0 for normal flow and invalid output; S1 for normal flow and valid output; S2 for overflow and valid output; and S3 for underflow and invalid output. Example transitions between states are also shown by the various arrows A to H.

Fig. 15 is a table including information regarding the example transitions represented by the graph arrows A-H in the state diagram of Fig. 14. The table includes values for the current flags V9 and V10, the current frame pointer, the next frame pointer, and the next valid bit.

The above description is included to illustrate the operation of the preferred embodiments and is not meant to limit the scope of the invention. The scope of the invention is to be limited only by the following claims. From the above discussion, many variations will be apparent to one skilled in the art that would yet be encompassed by the spirit and scope of the invention. For example, it would be relatively straightforward to extend the technique from 3x oversampling to arbitrary times oversampling.

IPR2024-001428
HP – EX1002, Page 105 of 487

CLAIMS

What is claimed is:

1. A system for transmission and recovery of original digital data, the system including:

(a) an encoder for receiving the original digital data and encoding the original digital data to form encoded data frames, and for outputting the encoded data frames;

(b) a transmitter for receiving the encoded data frames, forming an encoded data stream, and transmitting the encoded data stream to a channel;

(c) a receiver for receiving the encoded data stream from the channel, the receiver including

(1) an oversampler for oversampling the encoded data stream to form an oversampled data stream,

(2) a digital phase locked loop for receiving the oversampled data stream, extracting clock information from the oversampled data stream, and generating selected data from the oversampled data stream using the clock information, and

(3) a frame aligner for generating encoded data frames using the selected data;

(d) an analog phase locked loop for receiving a system clock signal having a system clock frequency, supplying a sender's clock signal having a sender's clock frequency to the transmitter, and supplying a receiver's clock signal having a receiver's clock frequency to the receiver, where the sender's clock frequency is approximately a first integer multiple of the system clock frequency, and where the receiver's clock frequency is approximately a second integer multiple of the sender's clock frequency; and

(e) a decoder for receiving the encoded data frames from the frame aligner and for decoding the data frames to recover the original digital data.

2. The system of claim 1, wherein the receiver's clock frequency is approximately the second integer multiple of the sender's clock frequency with less than 0.1% tolerance.

IPR2024-001428
HP – EX1002, Page 106 of 487

integer multiple is at least three.

4. The system of claim 1, where the original digital data comprises digital data for display as images on a screen.

5. The system of claim 1, wherein (b) the transmitter includes:

(1) a timing aligner for receiving the encoded data frames in parallel from the encoder and delaying a portion of each of the encoded data frames by a fraction of a cycle of the system clock signal;

(2) a data selector for serializing the encoded data frames to form the encoded data stream which is synchronized with the sender's clock signal; and

(3) a channel driver for driving the encoded data stream to the channel.

6. The system of claim 5, where the fraction of the cycle of the system clock signal is one half of the cycle of the system clock signal.

7. The system of claim 5, further including:

a signal level controller coupled to the channel driver for determining a signal strength at which the channel driver drives the encoded data stream to the channel.

8. The system of claim 5, where the portion each of the encoded data frames is a latter half of each of the encoded data frames.

9. The system of claim 5, where the fraction of the cycle of the system clock signal is one half of the system clock signal.

10. The system of claim 1, wherein (c)(1) the oversampler includes:

(i) a channel amplifier for receiving the encoded data stream from the channel and for amplifying a signal strength of the encoded data stream;

(ii) a sampler for receiving the receiver's clock signal and for oversampling the encoded data stream at the receiver's clock frequency to form samplings in a first number of detect cells, where the first number equals a product of the first integer multiple and the second integer multiple; and

IPR2024-001428
HP – EX1002, Page 107 of 487

holding the first portion of the samplings while corresponding detect cells are precharged, and for outputting the samplings in parallel to the digital phase locked loop.

11. The system of claim 10, wherein (c)(1) the oversampler further includes:

(iv) a channel impedance controller for adjusting an input impedence of the receiver at the sender's clock frequency.

12. The system of claim 10, where the first portion of the samplings comprises a first half of the samplings.

13. The system of claim 10, where the first number is at least thirty.

14. The system of claim 1, wherein (c)(2) the digital PLL includes:

(i) an edge extractor for generating a signal indicating positions of data transitions in the oversampled data stream; and

(ii) a data select controller for determining a phase at which to select data from the oversampled data stream.

15. The system of claim 1 wherein (c)(3) the frame aligner includes:

(i) a sync pattern finder for determining if the oversampled data stream contains a predetermined sync pattern.

16. The system of claim 1, wherein the receiver normally outputs encoded data frames during alternate cycles of the system clock signal.

17. The system of claim 16, wherein in an overflow situation, the receiver outputs encoded data frames during two consecutive cycles of the system clock signal.

18. The system of claim 16, wherein in an underflow situation, the receiver refrains from outputting encoded data frames during two consecutive cycles of the system clock signal.

19. A method for high-speed communication of digital data including:

IPR2024-001428
HP – EX1002, Page 108 of 487

signal, where the sender's clock signal is a first integer multiple of the system clock signal, and the receiver's clock signal is a second integer multiple of the sender's clock signal;

receiving the digital data;

encoding the digital data to generate encoded data frames;

forming an encoded data stream from the encoded data frames;

transmitting the encoded data stream to a channel in synchronization with the sender's clock signal;

receiving the encoded data stream from the channel in synchronization with the receiver's clock signal;

generating encoded data frames from the encoded data stream; and

decoding the encoded data frames to regenerate the digital data.

20. A method for high-speed communication of digital data including:

inputting the digital data to a transmitter, where during each cycle of a system clock signal one predetermined portion of the digital data is input;

transmitting the digital data as a waveform from the transmitter to a channel, where during each cycle of the system clock signal one predetermined portion of the digital data is transmitted.

receiving the waveform from the channel to a receiver;

oversampling the waveform to generate an oversampled data stream;

determining the digital data from the oversampled data stream; and

outputting the digital data from the receiver, where normally two predetermined portions of the digital data are output during each alternate cycle of the system clock signal, where for overflow situations the digital data is output during consecutive cycles of the system clock, and where for underflow situations no digital data is output during consecutive cycles of the system clock.

IPR2024-001428
HP – EX1002, Page 109 of 487

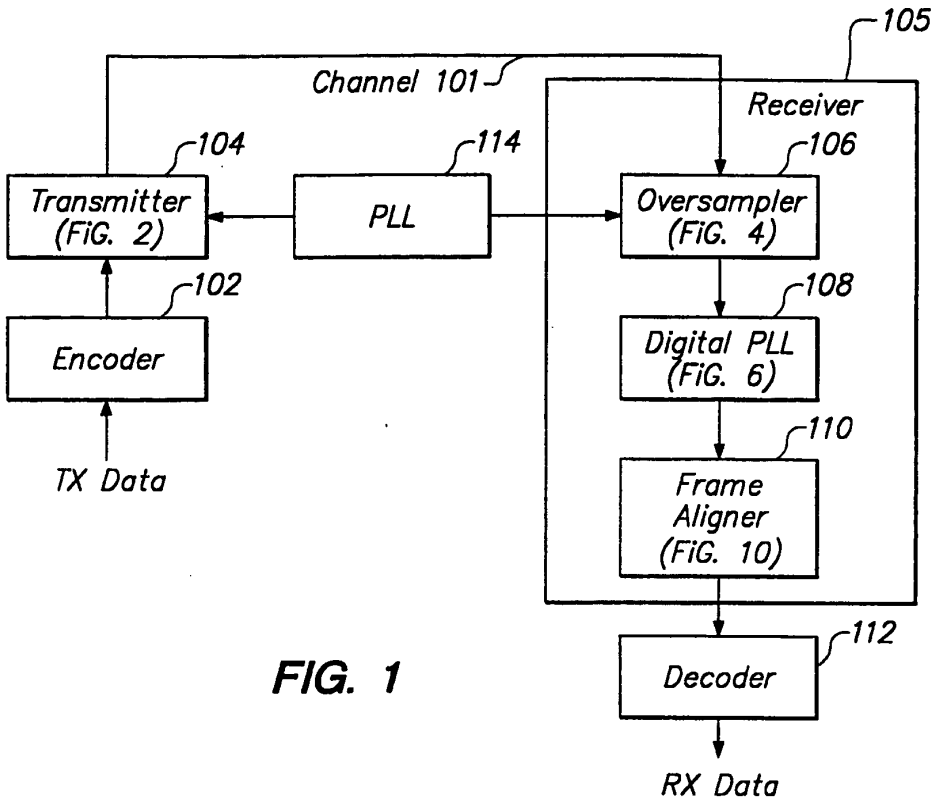


FIG. 1

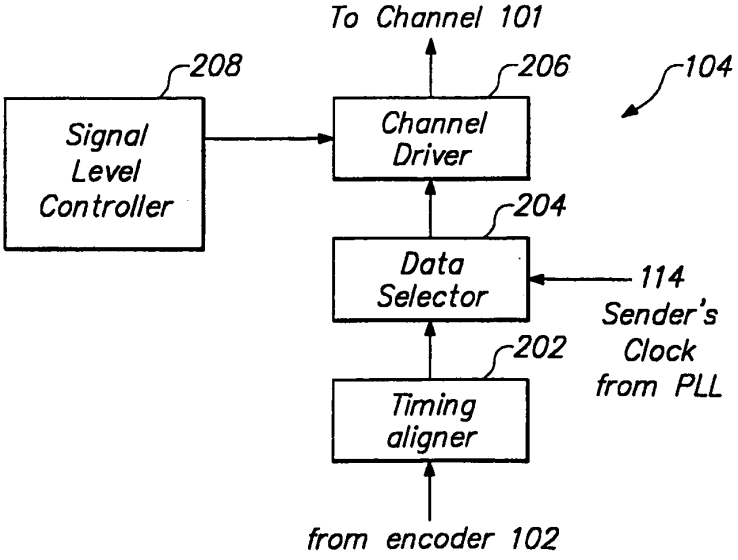


FIG. 2

IPR2024-001428
HP – EX1002, Page 110 of 487

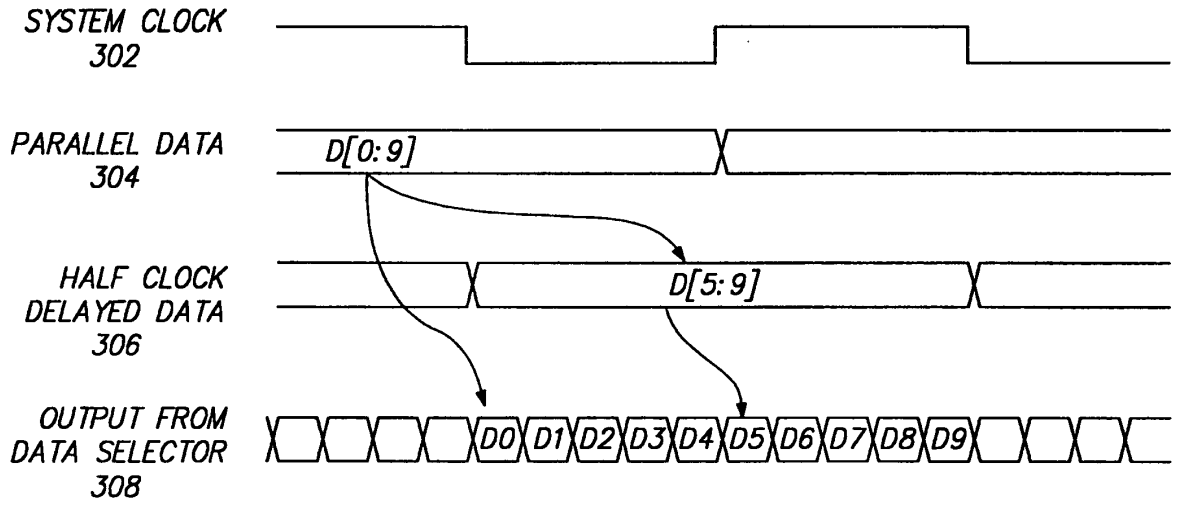


FIG. 3

SUBSTITUTE SHEET (RULE 26)

SUBSTITUTE SHEET (RULE 26)

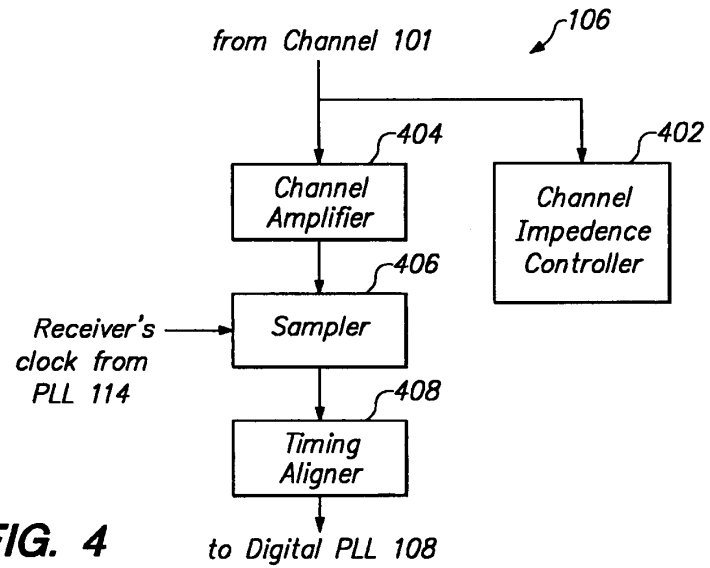


FIG. 4

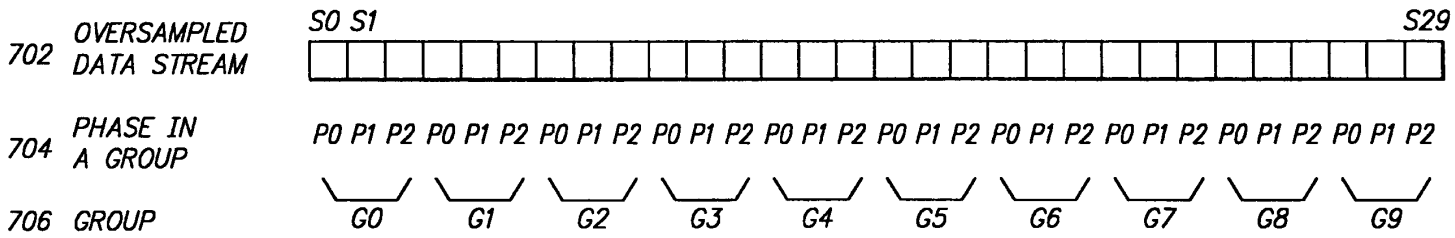


FIG. 7

SUBSTITUTE SHEET (RULE 26)

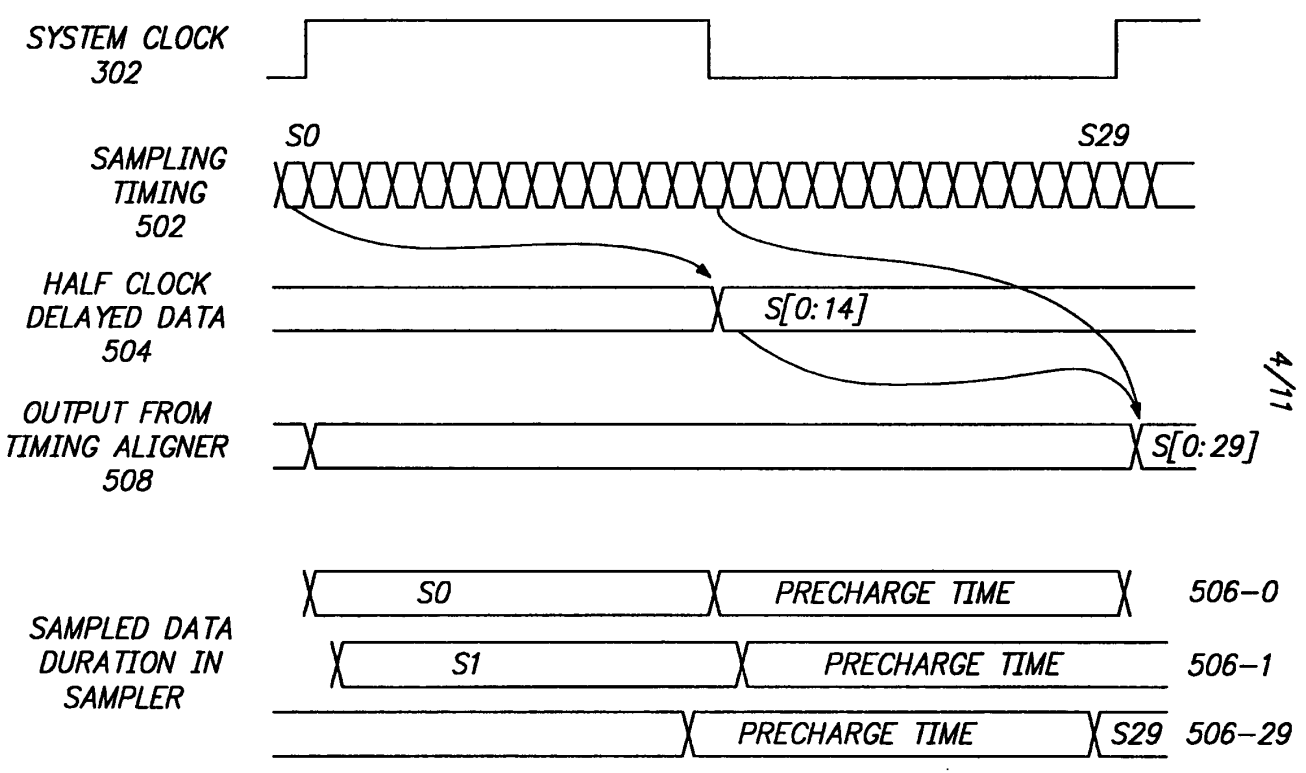


FIG. 5

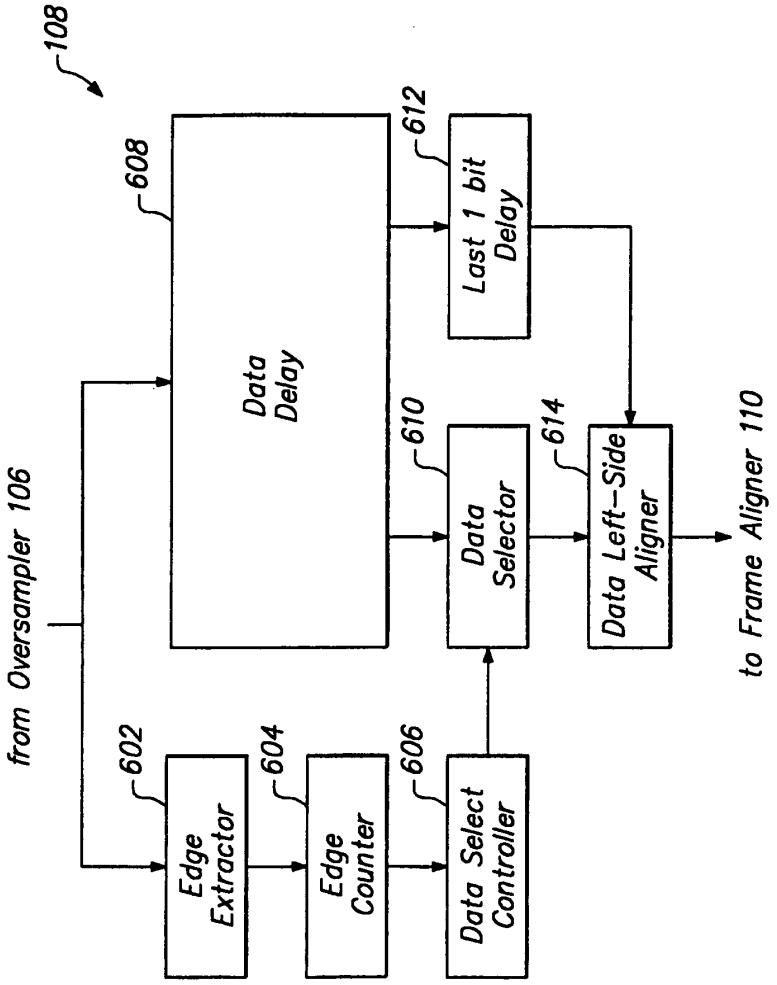


FIG. 6

IPR2024-001428
HP – EX1002, Page 114 of 487

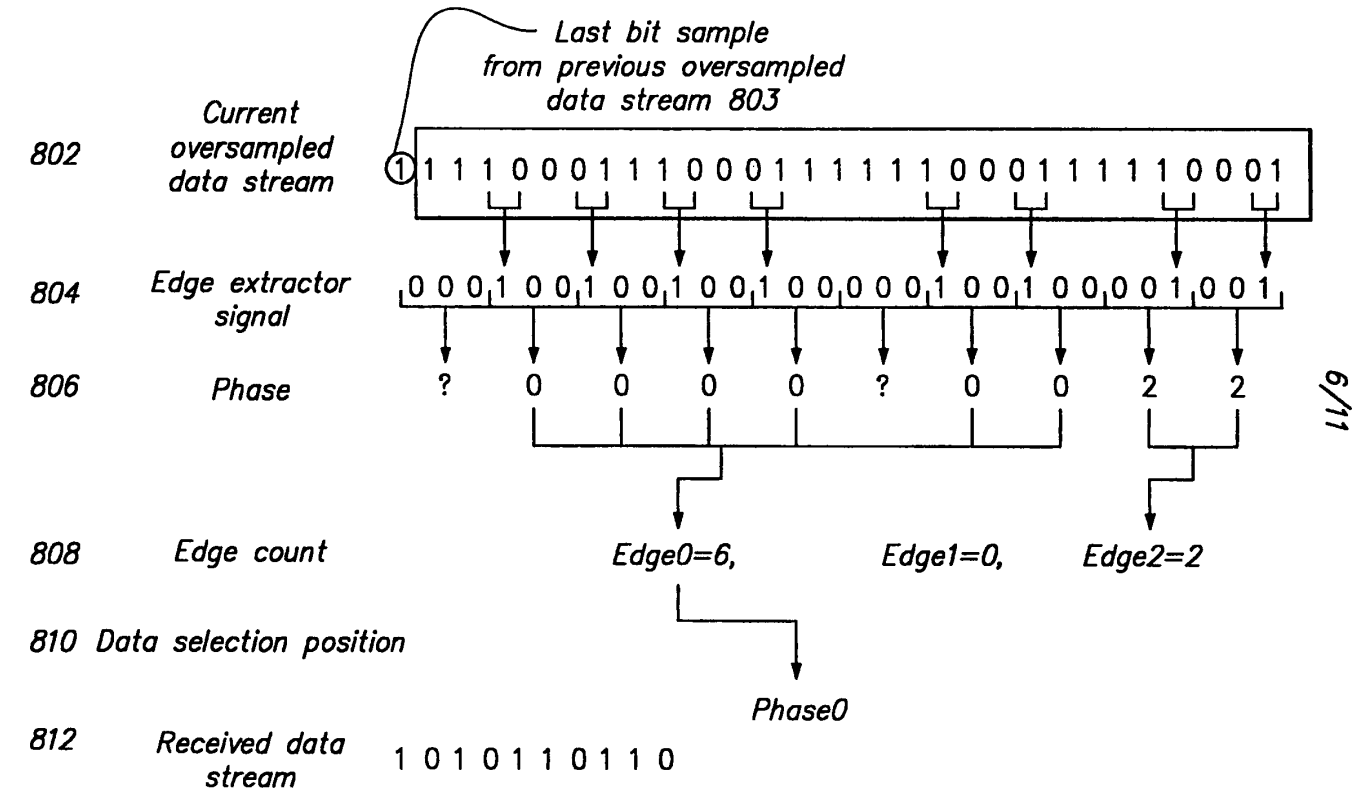


FIG. 8

IPR2024-001428
HP – EX1002, Page 115 of 487

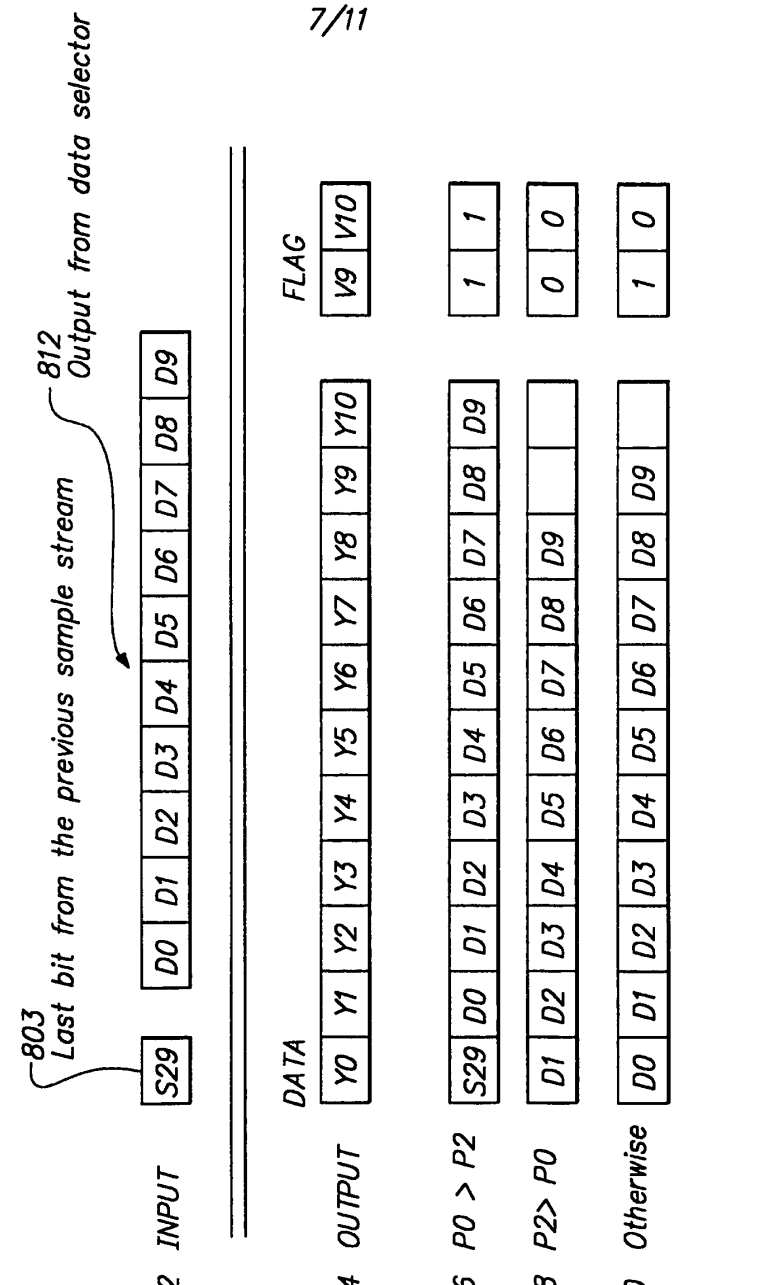


FIG. 9

IPR2024-001428
HP – EX1002, Page 116 of 487

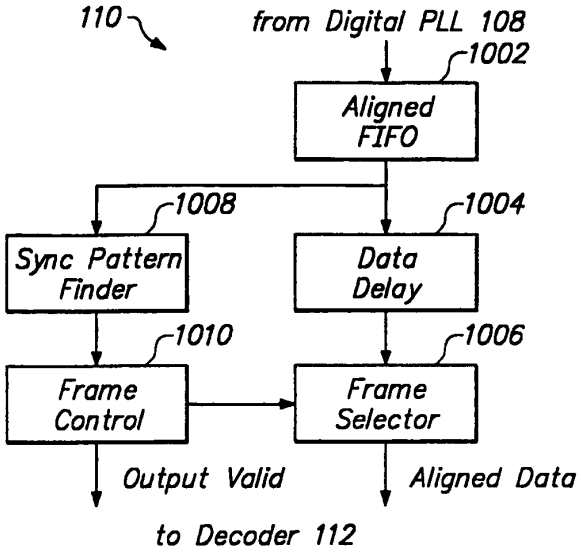


FIG. 10

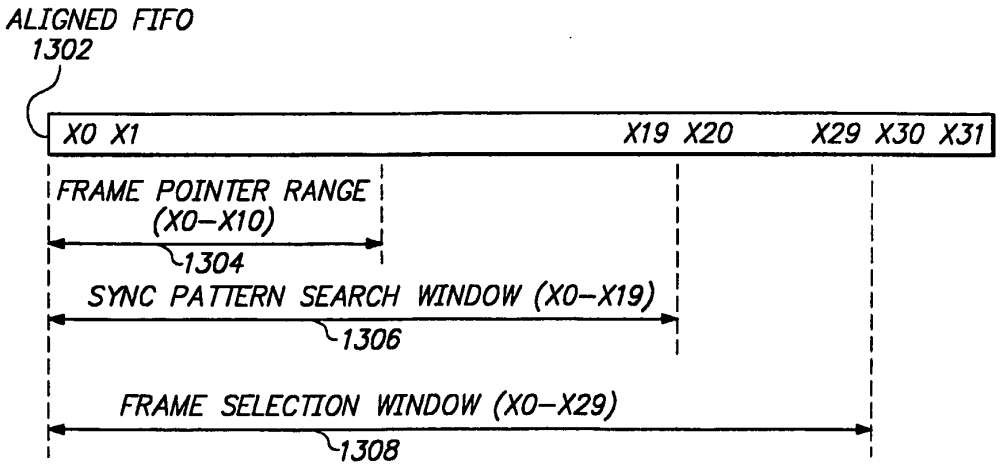


FIG. 13

IPR2024-001428
HP – EX1002, Page 117 of 487

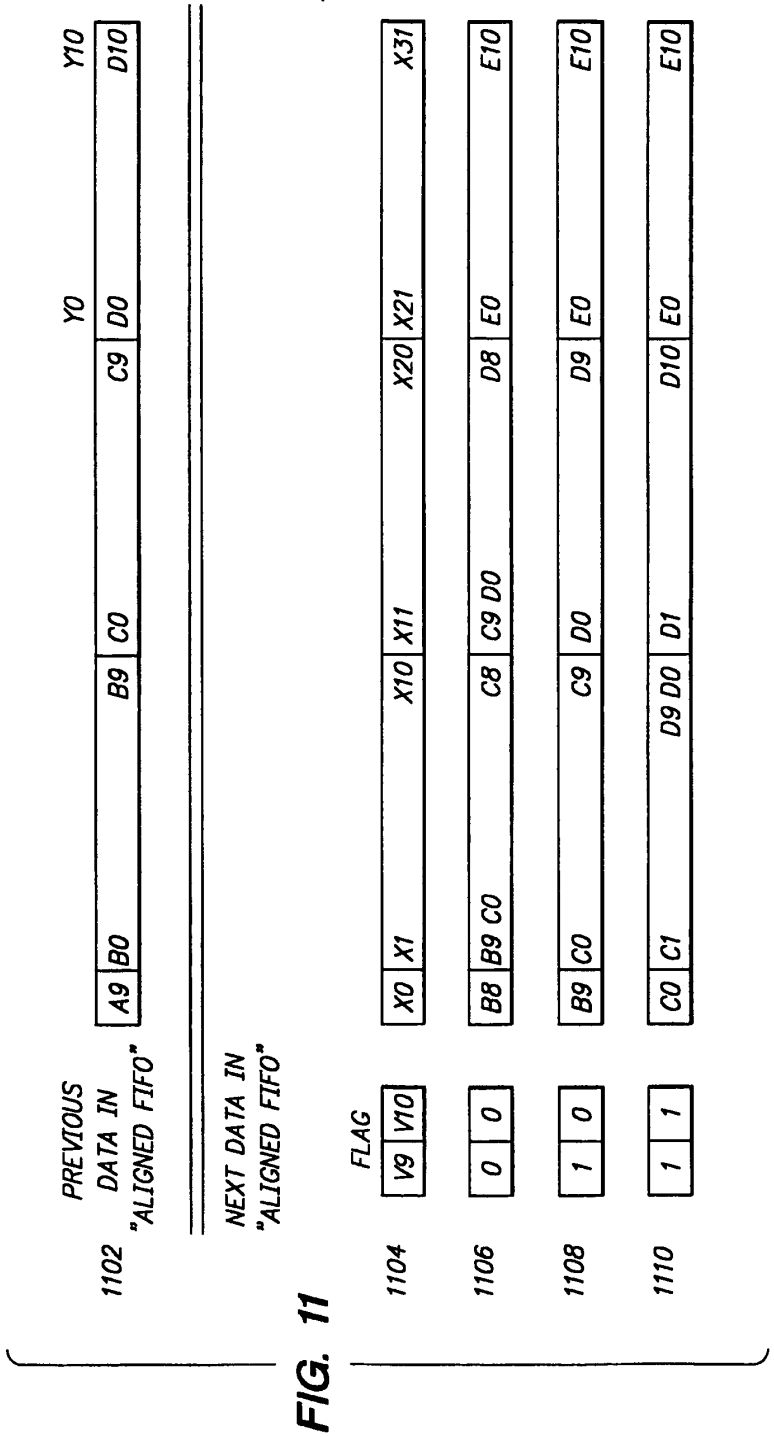


FIG. 11

IPR2024-001428
HP – EX1002, Page 118 of 487

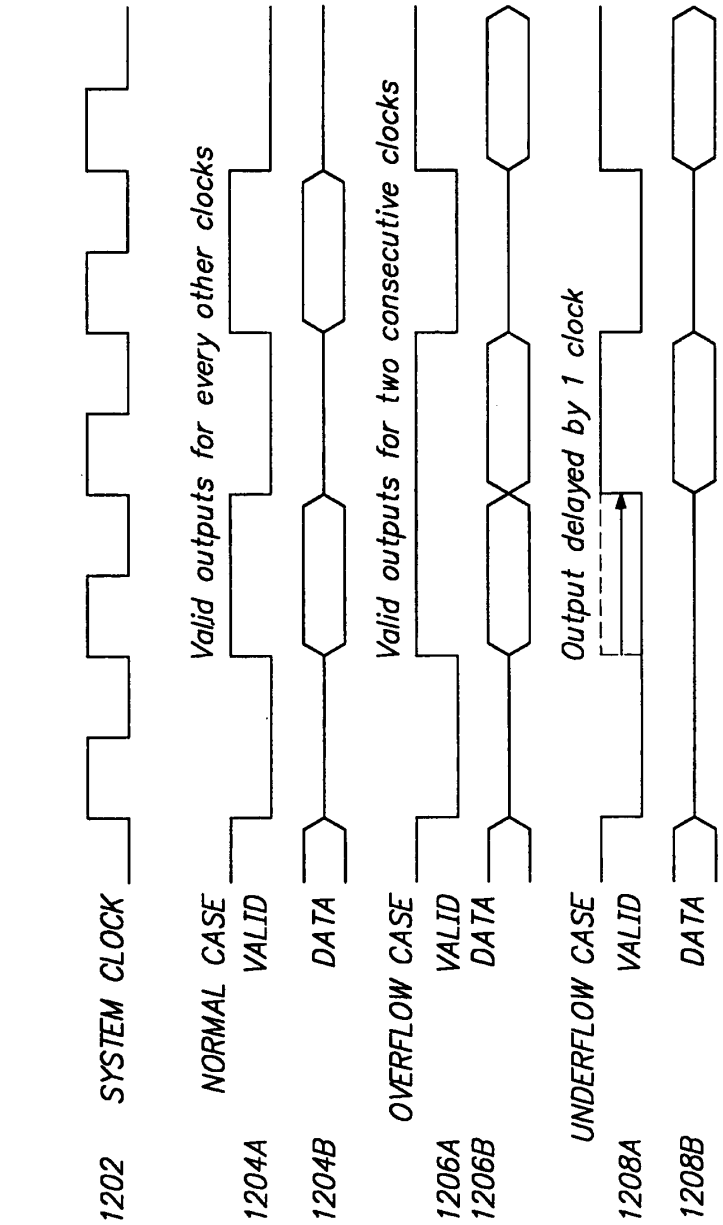


FIG. 12

IPR2024-001428
HP – EX1002, Page 119 of 487

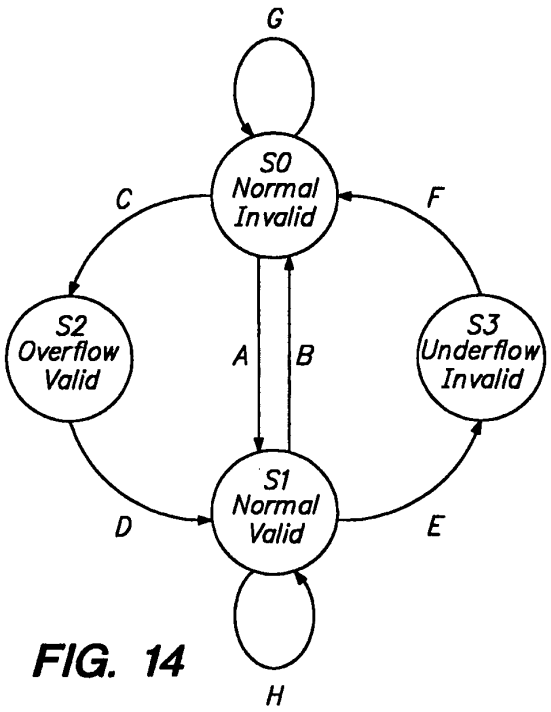


FIG. 14

Graph Arrow	Current Flags V9	V10	Current Frame Pointer	Next Frame Pointer	Next Valid Bit
A	0	0	X[n]	X[n+1]	1
	1	0		X[n]	1
	1	1		X[n−1]	1
B	0	0	X[n]	X[n+1]	0
	1	0		X[n]	0
	1	1		X[n−1]	0
C	1	1	X[1]	X[0]	1
D	1	0	X[10]	X[1]	1
E	0	0	X[1]	X[1]	0
F	1	0	x[10]	X[1]	0
G	0	0	X[10]	X[1]	0
H	1	1	X[0]	X[10]	1

FIG. 15

IPR2024-001428
HP – EX1002, Page 120 of 487

INTERNATIONAL SEARCH REPORT

Int. ..al Application No
PCT/US 98/18886

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04L7/033

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)
IPC 6 H04L

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)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LEE K ET AL: "A CMOS SERIAL LINK FOR FULLY DUPLEXED DATA COMMUNICATION" IEICE TRANSACTIONS ON ELECTRONICS, vol. E78-C, no. 6, June 1995, pages 601-611, XP000524416 see page 601, column 2, line 4 - line 19 see page 601, column 2, line 27 - page 602, column 1, line 10 see page 605, column 1, line 18 - line 20 see page 605, column 2, line 18 - line 19 see figure 14 ---	1-3,19
P,X	WO 97 42731 A (SILICON IMAGE INC) 13 November 1997 *abstract* ---	1-3,19
A	EP 0 498 064 A (IBM) 12 August 1992 see page 1, line 4 - page 2, line 37 ---	1-20
	-/--	

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

[X] Patent family members are listed in 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
21 December 1998	05/01/1999

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, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Van den Berg, J.G.J.

Form PCT/ISA/210 (second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 121 of 487

INTERNATIONAL SEARCH REPORT

Int nal Application No
PCT/US 98/18886

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	US 4 672 639 A (TANABE TOSHIYUKI ET AL) 9 June 1987 * abstract *	1-20

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 122 of 487

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int nal Application No
PCT/US 98/18886

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9742731 A	13-11-1997	AU 2823697 A	26-11-1997
		US 5835498 A	10-11-1998
EP 0498064 A	12-08-1992	US 5212716 A	18-05-1993
		JP 4320109 A	10-11-1992
US 4672639 A	09-06-1987	JP 1764351 C	28-05-1993
		JP 4053152 B	25-08-1992
		JP 60248086 A	07-12-1985
		JP 1764352 C	28-05-1993
		JP 4053153 B	25-08-1992
		JP 60248087 A	07-12-1985
		JP 1738528 C	26-02-1993
		JP 4023877 B	23-04-1992
		JP 60248088 A	07-12-1985
		CA 1252156 A	04-04-1989
		EP 0165498 A	27-12-1985

IPR2024-001428
HP – EX1002, Page 123 of 487

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 25/49 — A2

(11) International Publication Number: **WO 97/36408**

(43) International Publication Date: 2 October 1997 (02.10.97)

(21) International Application Number: PCT/US97/05220

(22) International Filing Date: 25 March 1997 (25.03.97)

(30) Priority Data:
08/622,810 27 March 1996 (27.03.96) US

(71) Applicant: SILICON IMAGE, INC. [US/US]; Suite 222, 1032 Elwell Court, Palo Alto, CA 94303 (US).

(72) Inventors: LEE, Kyeongho; 523 Walker Drive #9, Mountain View, CA 94043 (US). JEONG, Deog-Kyoon; 3351 Alma Street #307, Palo Alto, CA 94306 (US).

(74) Agents: OKAMOTO, James, K. et al.; Fenwick & West L.L.P., Suite 700, Two Palo Alto Square, Palo Alto, CA 94306 (US).

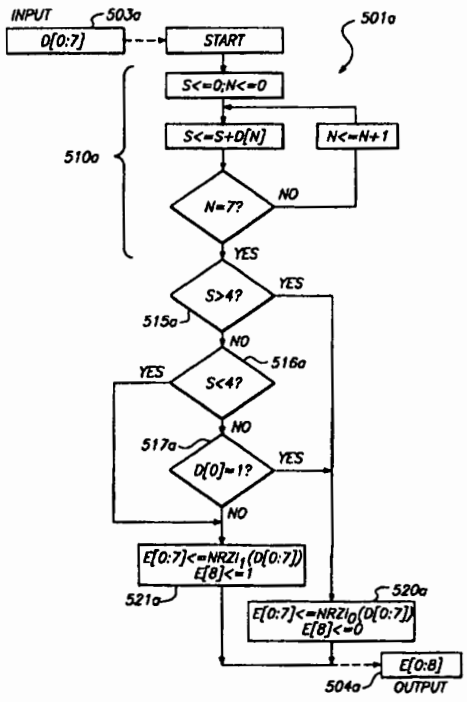
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ARIPO patent (GH, KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: TRANSITION CONTROLLED BALANCED ENCODING SCHEME

(57) Abstract

A method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes is disclosed herein. The bits in each of the data bytes are selectively complemented in accordance with the number of logical '1' signals in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity. In a high-transition mode of operation, the bits within data blocks including fewer than a minimum number of logical '1' signals are selectively complemented so that each such selectively complemented data block includes in excess of the minimum number of logical transitions. In a low-transition mode of operation, the bits within data blocks having more than a predefined number of logical '1' signals are selectively complemented so that each such selectively complemented data block includes less than the maximum number of logical transitions.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

TRANSITION CONTROLLED BALANCED ENCODING SCHEME

CROSS-REFERENCE TO RELATED APPLICATION

This application is a continuation-in-part of copending application Serial No. 08/539,816, filed October 5, 1995, the disclosure of which is herein incorporated by reference.

BACKGROUND OF THE INVENTION

I. Field of the Invention

This invention relates to coding schemes for digital transmission systems. More particularly, the present invention relates to a DC-balanced, transition-controlled coding system in which an unbalanced datastream is converted to a DC-balanced stream to facilitate transmission.

II. Description of the Related Art

As electronic and computer technology continues to evolve, communication of information among different devices, either situated near by or at a distance becomes increasingly important. For example, it is now more desirable than ever to provide for high speed communications among different chips on a circuit board, different circuit boards in a system, and different systems with each other. It is also increasingly desirable to provide such communications at very high speeds, especially in view of the large amount of data required for data communications in intensive data consuming systems using graphical or video information, multiple input-output channels, local area networks, and the like.

It is particularly desirable to enable individual personal computers, workstations, or other computing devices, within which data is normally internally transferred using parallel data buses, to communicate with each other over relatively simple transmission lines. Such transmission lines typically include only one or two conductors, in contrast with the 64-bit and wider data paths within computing systems now commonly available. In the case of video data transmission to computer displays, as well as in the case of high-speed video input from digital cameras to computer systems, existing interconnection interfaces typically employ such parallel data paths. Recently, the requisite bandwidth of such interconnection systems has increased as a consequence of increased display resolution. This has increased electromagnetic interference (EMI) as well as transmission line noise, thereby raising concerns as to safety and reliability. In addition, the large number of signal and data lines required by advanced liquid crystal display panels has increased the potential for mutual interference.

There have been a number of commercially available products which attempt to provide high speed conversion of parallel data to serial form and transmission over a serial link. The Hewlett-Packard G-link chip set is one such product. That chip set includes a transmitter set and is capable of handling 21-bit wide parallel data. To obtain the necessary speed, however, the chip set is fabricated using a bipolar process, and the receiver and transmitter require separate chips. Such a solution is highly power consumptive and expensive.

Another commercial solution has been provided by Bull of France. The Bull technology employs a frequency multiplier for parallel to serial data conversion. Such devices typically introduce noise into the silicon substrate and interfere with other multipliers on the chip. In addition, the Bull technology uses an exclusive OR tree for parallel to serial conversion. The use of exclusive OR trees is well known, together with the difficulty of equalizing the delay through all paths of such devices. Additionally, the Bull technology uses output signals having full logic swings. This results in slower performance.

Various techniques exist for improving the characteristics of transmission over serial links. For example, transmission codes may be employed to alter the frequency spectrum of the transmitted serial data so as to facilitate clock recovery and enable AC coupling. Each transmission code will also typically provide special characters, not included within the data alphabet, to be used in character synchronization, frame delimiting, as well as perhaps for diagnostic purposes. Coding may also be employed to reduce transmission bandwidth as a means of limiting the signal distortion occurring during propagation through the transmission medium. In the case of wire links, it is desirable to utilize codes with no DC and little low frequency content in order to allow for DC isolation of the driver and receiver circuitry from the transmission line, as well as to reduce signal distortion on the line. An efficient coding system should also be disposed to encode clock information with the encoded data in a manner allowing for extraction of the clock information during decoding. This obviates the need for provision of a separate clock signal over a dedicated clock line, since the clock information recovered during decoding may be instead used by the receiver circuitry.

Within local area networks (LANs), transmission coding schemes exist for converting words of various length to characters of greater length. For example, three-bit words may be converted to four-bit characters (3B/4B), four-bit words may be converted to five-bit characters (4B/5B), and so on. Typically, coding and decoding is achieved using a "key" in which each word is mapped to a corresponding character. Unfortunately, the complexity of this type of mapping scheme generally precludes utilization of random logic, and often requires implementations involving look-up tables or the like. This is disadvantageous given that look-up tables realized using ROM consume significant chip area and tend to slow circuit operation.

A particular 8B/10B coding scheme is described in U.S. Patent No. 4,486,739. In particular, a binary DC balanced code and associated encoder circuit are described as being operative to translate an 8 bit byte of information into 10

binary digits for transmission. The 8B/10B coder is partitioned into a 5B/6B plus a 3B/4B coder. Despite ostensibly facilitating DC-balanced encoding, this system tends to require relatively lengthy encoding and decoding intervals.

Although progress has been made in the development of coding techniques disposed to facilitate serial data transmission, there remains a need for a coding scheme capable of efficiently supporting very high speed serial data transmission. Such a coding scheme should also be DC-balanced in order to facilitate AC coupling and clock recovery. In addition, it would be desirable to provide a coding scheme capable of facilitating real-time data transfer by allowing for rapid synchronization during decoding. In addition, it would be desirable to provide a coding scheme capable of producing a DC-balanced datastream without the necessity of providing additional bits to indicate whether a particular block was modified by the balance-encoding scheme.

SUMMARY OF THE INVENTION

In one aspect, the present invention is directed to a method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes. The bits in each of the data bytes are selectively complemented in accordance with the number of logical '1' signals in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity.

In a high-transition mode of operation, the bits within those of the data blocks including fewer than a minimum number of logical '1' signals are selectively complemented in order that each such selectively complemented data block include in excess of the minimum number of logical '1' signals. In a low-transition mode of operation, the bits within those of the data blocks having more than a predefined number of logical transitions are selectively complemented in order that each such selectively complemented data block includes less than the maximum number of logical transitions. During both high-transition mode and low-transition mode encoding of each such selectively complemented data block into an encoded character, a predefined bit is set indicating that selective complementing has been effected.

In another aspect, in one embodiment, an input sequence of 9-bit data blocks is balanced to produce a DC-balanced sequence of characters. A shift register generator generates a pseudo-random binary sequence. Two bits of the pseudo-random binary sequence are logically combined to determine whether to invert another bit in the pseudo-random binary sequence, thereby cycling the pseudo-random binary sequence. A bit of the pseudo-random binary sequence is used as a criterion to selectively invert all the bits in the incoming 9-bit data block thereby producing a 9-bit data block, in an output stream that, over time, tends to be DC-balanced.

BRIEF DESCRIPTION OF THE DRAWINGS

Additional objects and features of the invention will be more readily apparent from the following detailed description and appended claims when taken in conjunction with the drawings, in which:

FIG. 1 is a block diagram depicting a DC-balanced encoding system of the present invention implemented within a high-speed digital transmission system.

FIG. 2 shows the overall functional organization of a DC-balanced encoder of the present invention in the form of a data flow chart.

FIG. 3 provides a block diagrammatic representation of a decoder included within the transmission system of FIG. 1.

FIG. 4 shows a random logic implementation of an alternate byte inversion decoder module of the decoder of FIG. 3.

FIG. 5 is a flow chart depicting the synchronization process effected by a decoder synchronization module operative in conjunction with the decoder of FIG. 3.

FIG. 6 is a flow chart depicting the synchronization process effected by the decoder synchronization module during high-transition mode operation.

FIGS. 7A and 7B provide a schematic representation of an exemplary implementation of the encoder.

IPR2024-001428
HP – EX1002, Page 131 of 487

FIGS. 8A and 8B provide a schematic representation of a preferred implementation of the decoder synchronization module.

FIGS. 9A and 9B provide a flowchart representation of an alternative embodiment of a transition optimizer.

FIG. 10 depicts a logic circuit that implements the transition optimizer of FIGS. 9A and 9B.

FIG. 11 depicts a balance-encoding circuit for DC-balancing a transition-optimized frame using frame-synchronous scrambling.

FIG. 12 depicts a balance-decoding circuit for decoding a DC-balanced frame encoded with the balance-encoding circuit of FIG. 11.

FIG. 13 depicts a balance-encoding circuit for DC-balancing a transition-optimized frame using self-synchronous scrambling.

FIG. 14 depicts a balance-decoding circuit for decoding a DC-balanced frame encoded with the balance-encoding circuit of FIG. 13.

DESCRIPTION OF THE PREFERRED EMBODIMENT

I. System Overview

FIG. 1 is a block diagram depicting a transition-controlled, DC-balanced encoding system of the present invention implemented within a high-speed digital transmission system 10. In the system 10 of FIG. 1, 8-bit bytes of parallel data 14 are provided to a DC-balanced encoder 18 operative to effect transition-controlled, DC-balanced 8B/10B coding in accordance with the present invention. The resultant 10B encoded characters 22 are provided to a serializer 26 disposed to convert the 10-bit characters into a serial data stream for transmission over a serial data link 30 (e.g., an optical fiber cable, or twisted-pair copper wire). As is described herein, the relatively straightforward mathematical characteristics of the encoding algorithm performed by the encoder 18 allow for economical, high-speed implementations in random logic.

The serial data stream is received from the serial data link 30 by a deserializer 34 and converted into 10-bit character data 38. The 10-bit character data 38 is then decoded into 8-bit data bytes 42 by a decoder 44. As is described hereinafter, both the encoder 18 and decoder 44 may be economically realized using simple digital logic circuits capable of real-time data processing.

II. DC-Balanced Transition-Controlled Encoding and Decoding System

The following provides a detailed description of the 8B/10B transition-controlled coding scheme carried out within the encoder 18. The transition-controlled code disclosed herein contemplates the use of either a high-transition or low-transition set of "in-band" code characters. Each high-transition in-band code character is derived from an input data byte in which four or more logical transitions exist between the eight bits thereof. Similarly, each low-transition in-band code character is derived from an input data byte in which fewer than four logical transitions exist between its eight bits. This scheme takes advantage of the fact that 128 of the 258 eight-bit ASCII codes include four or more logical transitions, and the remaining 128 ASCII codes include fewer than four logical transitions.

It has been found that each of the 128 eight-bit codes including fewer than four logical transitions may be mapped to a corresponding eight-bit code having four or more logical transitions, and vice-versa. As is described herein, this mapping may be achieved by complementing predefined bits in each eight-bit code to be mapped. During high-transition mode encoding, the bits within those input bytes having fewer than four logical transitions are selectively complemented and thereby mapped to bytes having four or more logical transitions. Alternately, during low-transition mode encoding the bits within those input bytes having four or more logical transitions are also selectively complemented and thereby mapped to bytes having fewer than four logical transitions. During both encoding modes, a bit of predefined value is appended to the selectively complemented byte in order to produce an intermediate 9-bit encoded symbol prior to creation of a corresponding 10-bit encoded character. When the input byte includes the number of logical

transitions mandated by the current encoding mode (i.e, high-transition mode or low-transition mode), the appended bit is set to the complement of the predefined value in order to identify which of the 9-bit intermediate symbols include selectively complemented bytes. This results in the entire set of 256 eight-bit codes being made available for encoding into 10-bit characters during both low-transition mode and high-transition mode operation.

It may thus be appreciated that each eight-bit code converted into a 10-bit encoded character during high-transition mode operation includes four or more logical transitions. Similarly, each eight-bit code converted into a 10-bit encoded character during low-transition mode operation includes less than four logical transitions. These sets of 10-bit encoded characters capable of being produced during high-transition and low-transition modes of operation may be characterized as a high-transition set of "in-band" encoded characters, and a low-transition set of in-band encoded characters, respectively. Beyond the 256 in-band characters within the high-transition set and the 256 in-band characters of the low-transition set, there exists a high-transition set of 256 out-of-band 10-bit characters and a low-transition set of 10-bit characters. In accordance with another aspect of the invention, various synchronization and other special characters are defined using the high-transition and low-transition sets of out-of-band characters. Each character corresponding to one of these "out-of-band" characters associated with the high-transition set includes less than four logical transitions, and each of the out-of-band characters associated with the low-transition set of code characters includes more than four logical transitions. The difference in the number of transitions between in-band and out-of-band characters allows selected out-of-band characters to serve as control characters, and to be readily distinguished from in-band characters within the transmitted data stream.

Given the relatively high number of transitions within each of the in-band characters of the high-transition set, the high-transition set of characters may advantageously be employed to facilitate timing recovery. On the other hand, the

low number of transitions within the code characters of the low-transition set makes this set of characters ideal for use in applications in which it is desired to minimize power consumption and/or electromagnetic interference (EMI).

In accordance with one aspect of the invention, the synchronization characters associated with both the low-transition and high-transition sets of code characters are selected so as to facilitate rapid synchronization during data recovery. When the low-transition set of code characters is being employed, a special group of out-of-band characters is used during synchronization. Each special synchronization character includes a predefined number larger than four (e.g., 7) of logical transitions between binary character values, and a predefined number (e.g., two) of "non-transitions" between character values. As will be discussed below, the special synchronization characters are selected such that random logic may be used to distinguish each special synchronization character from the in-band characters of the low-transition set. The following constitutes an exemplary set of out-of-band synchronization characters for use with the low-transition set of code characters:

1100101010
1101001010
1101010010
1101010100

It is a feature of the invention that if one of the above out-of-band synchronization characters is transmitted three or more consecutive times within any preamble period, the synchronization character is ensured of being detected during the associated data recovery process. In this regard a "preamble" sequence is sent during a preamble period preceding each transmission of encoded characters. The transmission of preamble sequences may occur not only as part of system initialization, but also at various other times to ensure that synchronization is maintained between encoding and decoding processes.

The coding scheme of the present invention is predicated on particular characteristics of the 256 different 8-bit binary code values. Referring to TABLE I. the 256 different 8-bit binary codes may be divided into eight groups G0-G7. where the binary codes within each group G0-G7 include the same number of transitions. It is observed that each binary code within group G0 can be converted to a corresponding binary code within group G7 by inverting alternate bits within the group G0 code. In the same way, each of the binary codes within groups G1, G2 and G3 may be converted to one of the binary codes within groups G6, G5 and G4, respectively, through inversion of alternate bits. As is described herein, the high-transition set of 10-bit characters is obtained by encoding of the 8-bit binary codes in groups G0-G3, and the low-transition set obtained by encoding groups G4-G7.

TABLE I

GROUP	NUMBER OF TRANSITIONS	NUMBER OF BYTES IN GROUP	EXAMPLE BYTES IN GROUP
G0	0	2	00000000, 11111111
G1	1	14	00000001, 11111110
G2	2	42	00000010, 11111101
G3	3	70	00000101, 11111010
G4	4	70	00001010, 11110101
G5	5	42	00010101, 11101010
G6	6	14	00101010, 11010101
G7	7	2	01010101, 10101010

During operation of the encoder 18 in a high-transition encoding mode, each 8-bit binary code within byte groups G0-G3 provided thereto is converted to a corresponding binary code within byte groups G4-G7 through inversion of alternate bits. Conversely, during operation in the low-transition encoding mode, each 8-bit binary code within groups G4-G7 provided to encoder 18 is mapped to a corresponding binary code within groups G0-G3. In the exemplary embodiment, the inversion of alternate bits is effected through the inversion of the even bits of the 8-bit binary codes. When the alternate bits of a given 8-bit code have been so inverted, a predefined bit within the resulting 10-bit encoded character derived from

the given eight-bit code is set so as to indicate that mapping has occurred between byte groups.

Referring now to FIG. 2, the overall functional organization of the transition-controlled DC-balanced encoder 18 of the present invention is shown in the form of a data flow chart. In FIG. 2, the 8-bit parallel data 14 to be encoded is latched within an input latch 70 comprised of, for example, eight D-type flip-flops. A transition counter 74 is operative to count the number of transitions (T) in logical value between adjacent bits of each byte of parallel data 14 within latch 70. If D_7, D_6, ... D_0, (i.e., D[7:0] comprise the eight bits of data latched within the input latch 70, then the transition counter 74 may determine T as follows:

$$T := (D_7 \text{ xor } D_6) + (D_6 \text{ xor } D_5) + (D_5 \text{ xor } D_4) + (D_4 \text{ xor } D_3)$$
$$+ (D_3 \text{ xor } D_2) + (D_2 \text{ xor } D_1) + (D_1 \text{ xor } D_0)$$

A COUNT line 78 is set to a predefined logical value by counter 74 if more four or more logical transitions are counted between bits of the latched byte ($T > 3$), and is set to the complement of the predefined logical value otherwise ($T \leq 3$). In what follows it will be assumed that COUNT = 0 if four or more logical transitions are counted by counter 78 ($T > 3$), and that COUNT = 1 otherwise ($T \leq 3$).

As is indicated by FIG. 2, a transition controller 82 is responsive to COUNT line 78 and to a mode select line 86. Mode select line 86 determines whether encoding is to be performed using the high-transition set or the low-transition set of code characters. When mode select line 86 indicates high-transition encoding is in effect, and COUNT line 78 registers that less than four logical transitions exist within the byte stored within latch 70, the transition controller 82 instructs conditional alternate bit inversion (CABI) logic 90 to invert the even bits the byte stored within latch 70. The resultant conditionally inverted byte, which will have four or more logical transitions, is stored within intermediate latch 94. Conversely,

if high-transition encoding is in effect and four or more logical transitions are counted within the byte stored in the input latch 70, the transition controller 82 causes CABI logic 90 to simply transfer the byte from latch 70 (without bit inversion) to the intermediate latch 94. Accordingly, during high-transition encoding mode,

if $T < 4$,

then $E[8:0] = '1'\ D_7\bar{D}_6D_5\bar{D}_4D_3\bar{D}_2D_1\bar{D}_0$,

else if $T \geq 4$,

then $E[8:0] = '0'\ D_7D_6D_5D_4D_3D_2D_1D_0$

where E[7:0] comprise the eight bits stored within the intermediate latch 94, and E[8] comprises the value of COUNT stored within COUNT latch 95.

When mode select line 86 indicates that low-transition encoding has been selected, and COUNT line 78 registers that four or more logical transitions are present within the byte stored within latch 70, the transition controller 82 instructs conditional alternate bit inversion (CABI) logic 90 to invert the even bits the byte stored within latch 70. Otherwise, if low-transition encoding being performed and four or more logical transitions are counted within the byte stored in the input latch 70, the stored byte is simply transferred without bit inversion to the intermediate latch 94. Accordingly, during low-transition encoding mode,

if $T > 3$, then $E[8:0] = '1'\ D_7\bar{D}_6D_5\bar{D}_4D_3\bar{D}_2D_1\bar{D}_0$,

else $E[8:0] = '0'\ D_7D_6D_5D_4D_3D_2D_1D_0$

After CABI logic 90 has provided a byte having a number of logical transitions within the appropriate range to latch 94, a DC-balancing process is performed in which the cumulative disparity between complementary logical values within the stream of 10-bit encoded characters produced by the encoder 18. As used

herein, the term "cumulative disparity" (D_{cum}) denotes the excess of one bits relative to zero bits produced by the encoder 18 subsequent to synchronization being achieved with the decoder 44 in the manner described below. The term "current disparity" (D_{cur}) refers to the excess of one bits to zero bits within the byte currently stored within the latch 94, and is determined by disparity checker 96. A DC-balancing module 98 serves to compare the current disparity to the cumulative disparity stored within latch 99. The result of the comparison is then used to determine whether the byte stored within latch 94 is inverted by conditional byte inversion (CTBI) logic 100 during the course of transfer thereof to output register 104. In this way CTBI logic 100 serves to minimize the cumulative disparity associated with the serial stream produced by the encoder 18. The following provides a logical description of the manner in which each of the 10-bit characters T[9:0] in the DC-balanced character stream produced by the encoder are derived from the byte E[7:0] stored within the intermediate latch 94 and the bit E[8] within COUNT latch 95.

The current disparity (D_{cur}) is computed by the disparity checker 96 as follows:

$$D_{cur} := \{(E_7 \text{ and } E_6) + (E_5 \text{ and } E_4) + (E_3 \text{ and } E_2) + (E_1 \text{ and } E_0)\} - \{(E_7 \text{ nor } E_6) + (E_5 \text{ nor } E_4) + (E_3 \text{ nor } E_2) + (E_1 \text{ nor } E_0)\}$$

It is noted that during operation in the high-transition mode, $-2 \leq D_{cum} \leq 2$, while during operation in the low-transition mode $-4 \leq DP \leq 4$. Within the DC-balancing module 98, if it is determined that $D_{cur} = 0$ or $D_{cum} = 0$, then

if $E_8 = '0'$
then $T[9:0] = '1'E_8\bar{E}_7\bar{E}_6\bar{E}_5\bar{E}_4\bar{E}_3\bar{E}_2\bar{E}_1\bar{E}_0$, and

$$D'_{cum} = D_{cum} - D_{cur}$$

else if E_8 is not equal to zero,
then $T[9:0] = '0'E_8E_7E_6E_5E_4E_3E_2E_1E_0$, and

$$D'_{cum} = D_{cum} + D_{cur}$$

where D'_{cum} is the updated cumulative disparity computed by disparity updater 108 and stored thereby within latch 99.

Alternately, if the DC-balancing module determines that the most significant bit (MSB) of D_{cur} and the MSB of D_{cum} are not equivalent, then

$$T[9:0] = '0'E_8E_7E_6E_5E_4E_3E_2E_1E_0, \text{ and}$$
$$D'_{cum} = D_{cum} + D_{cur} - E_8$$

Finally, in all other cases if the MSB of D_{cur} and the MSB of D_{cum} are equivalent, then, then

$$T[9:0] = '1'E_8\bar{E}_7\bar{E}_6\bar{E}_5\bar{E}_4\bar{E}_3\bar{E}_2\bar{E}_1\bar{E}_0, \text{ and}$$
$$D'_{cum} = D_{cum} - D_{cur} + E_8$$

In this way the cumulative disparity is reduced, and DC balance achieved, through selective inversion of the byte E[7:0] by CBI logic 100 during the course of filling the output latch 104 as T[7:0]. It is observed that the logical value of T[8] is indicative of whether the even bits of byte D[7:0] received at input latch 70 were complemented during generation of the byte E[0:8]. Similarly, the logical value of T[9] indicates whether byte E[7:0] was inverted during transfer to latch 104.

III. Decoding

Referring to FIG. 1, the deserializer 34 receives the sequence of 10-bit characters T[9:0] produced by the encoder and generates 10-bit parallel received data upon bit lines RX_9, RX_8, ..., RX_0 (i.e., RX[9:0]). This 10-bit parallel received data is provided over bit lines RX[9:0] to the decoder 44, as well as to a decoder synchronization module 114. As is described below in section IV, the synchronization module 114 is operative to ascertain boundaries within the 10-bit

parallel received data corresponding to the frame boundaries of the transmitted data (i.e., to T[9:0]). Specifically, synchronization module 114 determines upon which of the bit lines RX[9:0] the deserializer 34 is providing the received bits corresponding to the first bit T[0] of each transmitted byte T[9:0]. Upon making this determination, the synchronization module 114 provides a frame boundary pointer 118 to decoder 44 identifying the one of the bit lines RX[9:0] corresponding to the first bit T[0] of each transmitted 10-bit character T[9:0]. Upon receiving this synchronization information, the decoder 44 is disposed to decode the received data RX[9:0] in the following manner.

FIG. 3 provides a block diagrammatic representation of the decoder 44. The 10-bit parallel data produced by the deserializer is seen to be received over bit lines RX[9:0] by a decoder switch 150. The decoder switch 150 serves to switch the 10-bit data received over bit lines RX[9:0] to switched bit lines S[9:0] in accordance with frame boundary pointer value 118 provided by the synchronization module 114. Specifically, the one of the received bits RX[9:0] corresponding to the first transmitted bit T[0] is switched to bit line S[0], the one of the received bits RX[9:0] corresponding to the second transmitted bit T[1] is switch to bit line S[1], and so on. The switched data impressed upon bit lines S[7:0], which corresponds to the transmitted data byte T[7:0], is stored within 8-bit latch 154. Similarly, the switched data bit S[8], which corresponds to the transmitted bit T[8], is provided to 1-bit latch 158. Since the logical value of bit line S[8] tracks the logical value of T[8], the bit line S[8] informs the Alternate Bit Inversion (ABI) decoder 160 as to whether the even bits of the input data D[7:0] were complemented by CABI logic 90 (FIG. 2) during the encoding process. Likewise, the bit line S[9] (which tracks the logical value of T[9]) informs the ABI decoder 160 as to whether CTBI logic 100 complemented the byte stored within latch 104 during the DC-balancing portion of the encoding process. In this way the decoder 160 is informed of the logical operations performed upon the byte S[7:0] stored within 8-bit latch 154 during the encoding process, thereby facilitating straightforward decoding using random logic.

Turning now to FIG. 4, there is shown a random logic implementation of the ABI decoder 160. The ABI decoder includes a set of nine exclusive-or (XOR) gates N1-N9 for decoding the 10-bit frame-aligned data S[9:0] in order to produce the 8-bit decoded byte DE[7:0]. In the implementation of FIG. 4, the XOR gates N1-N9 produce the decoded byte DE[7:0] as follows:

DE[7] := S[7] xor S[9]

DE[6] := S[6] xor β

DE[5] := S[5] xor S[9]

DE[4] := S[4] xor β

DE[3] := S[3] xor S[9]

DE[2] := S[2] xor β

DE[1] := S[1] xor S[9]

DE[0] := S[0] xor β

where β := S[9] xor S[8].

IV. Synchronization

As mentioned above, the decoder synchronization module 114 provides an indication to the decoder 44 of the frame boundary of each transmitted 10-bit character T[9:0]. The decoder module 114, together with the decoder switch 150 (FIG. 3), effectively function as a barrel-shifter to shuffle the parallel data RX[9:0] from the deserializer into the frame-aligned data S[9:0]. In accordance with the invention, a preamble sequence is produced by the encoder 18 at various times (e.g., at system power-up) in order to facilitate frame boundary detection by the synchronization module 114. In the exemplary embodiment this preamble sequence includes several repetitions of selected out-of-band characters easily distinguishable from in-band characters. Again, during high-transition mode operation each out-of-band character will include less than four logical transitions, and during low-

transition mode operation each out-of-band character will includes four or more logical transitions. As is discussed below, during operation in each mode several repetitions of specially selected out-of-band characters are produced by the encoder 18 during the preambling period as a means of ensuring rapid frame boundary identification within the decoder synchronization module 114. At the conclusion of the preambling period, the module 114 will "know" which of the bit lines RX[9:0] corresponds to the first bit T[0] of the 10-bit transmitted character, and will inform decoder via frame boundary pointer 118.

By selecting an appropriate subset of out-of-band characters for transmission during the preambling period, the worst-case time required for synchronization to be achieved may be reduced relative to that required by conventional synchronization schemes. In particular, during low-transition mode operation the following out-of-band characters are used as "synchronization characters".

1100101010
1101001010
1101010010
1101010100

During high-transition mode operation, the following out-of-band characters are used as synchronization characters:

1000001111
1000011111
1000111111
1001111111
1011111111

During each preambling period, three repetitions of the same synchronization character are produced by the encoder 18. As is described herein, by processing the 21 bits most recently produced by the encoder 14, synchronization module 114 is

able to detect at least one of the three repetitions of the synchronization character transmitted during a given preambling period. This advantageously allows synchronization to be achieved within a relatively short preambling period.

Turning now to FIG. 5, a flow chart is provided of the synchronization process effected by the decoder synchronization module 114 during low-transition mode operation. During each clock cycle of the module 114, a block of ten bits is loaded from the deserializer 34 into a first 10-bit latch 150. Also during each clock cycle, a 10-bit block is transferred from the first 10-bit latch 150 to a second 10-bit latch 154. Similarly, the 10-bit block currently stored within the second 10-bit latch 154 is transferred during each clock cycle to a third 10-bit latch 158.

As is indicated by FIG. 5, an exclusive-NOR (XNOR) operation (step 162) is performed between adjacent bits included within a 21-bit "window" of data held by the latches 150,154,158. Specifically, this 21-bit window includes a 10-bit b k L3[9:0] from the third latch 158, a 10-bit block L2[9:0] from the second latc 54, and the bit L1[9] from the first latch 150. In this regard the bit L1[9] is that bit which becomes bit L2[9] upon being transferred to the second latch 154. As an example of operation during the low-transition mode, consider a 21-bit window (i.e., L3[9:0], L2[9:0], L1[9]) comprised of the following parallel bit sequence:

101101001010110100101

If an XNOR operation is performed between each pair of adjacent bits, the following result is obtained:

001000100000010001000

As is indicated by FIG. 5, this 20-bit result of the XNOR operation (step 160) is divided into four 5-bit groups (i.e., Group A, Group B, Group C and Group D). In the current example, these four 5-bit groups are defined as follows:

001000100000010001000 {Result of XNOR operation}

00100 {Group A}
01000 {Group B}
00100 {Group C}
01000 {Group D}

The synchronization characters for both the high-transition mode and the low-transition modes enumerated above have been selected such that particular relationships arise between Groups A, B, C and D during the preambling period. That is, when three consecutive occurrences of the same synchronization character - produced by the encoder 18 during preambling - are received by the deserializer 34 and provide as 10-bit parallel data to the synchronization module 114.

In an exemplary implementation, the following two relationships (Condition I and Condition II) arise between Groups A, B, C and D during the preambling period:

Condition I. The number of logical "1's" collectively present in Groups A, B, C, and D is exactly four, and corresponds to one of the following three cases:

	Number of Logical 1's			
Case	Group A	Group B	Group C	Group D
#1	1	1	1	1
#2	2	0	2	0
#3	0	2	0	2

As is indicated by FIG. 5, the number of "1's" in each of the Groups A, B, C and D are determined by "1" counter modules 172, 174, 176 and 178, respectively. Since the number of "1's" within each Group A, B, C and D is determined by the results of the XNOR operation between adjacent bits in the 21-bit window (step 160), the number of "1's" within each group is indicative of the number of "non-transitions" in logical value between adjacent bits in each of four segments of the 21-bit window associated with Groups A, B, C and D. In the current example, it is

seen that each of Groups A, B, C and D each include a single "1". Accordingly, the current example corresponds to Case *#1*.

Condition II. The sequence of bits comprising Group A is equivalent to the bit sequence of Group C, and the sequence of bits comprising Group B is equivalent to the bit sequence of Group D. That is, Group A = Group C, and Group B = Group D.

In accordance with the invention, both Condition I AND Condition II are satisfied if and only if the same synchronization character is stored within the first, second and third 10-bit latches 150, 154 and 158. That is, both Condition I and Condition II are satisfied only during the preambling period, when three repetitions of the same synchronization character are produced by the encoder 18. This aspect of the invention is explained immediately below with reference to low-transition mode operation.

As was described above, the adjacent bits within a 21-bit window provided by latches 150, 154 and 158 are XNOR'ed during step 160 (FIG. 5). Because each in-band or out-of-band character produced by the encoder 18 is exactly ten bits in length, the 21-bit window will include all or part of 1st, 2nd and 3rd 10-bit characters. The various ways in which the 21-bit window may include bits from these 1st, 2nd and 3rd 10-bit characters is set forth below:

Of Bits

IPR2024-001428
HP – EX1002, Page 147 of 487

1st Character	2nd Character	3rd Character
1	10	10
2	10	9
3	10	8
4	10	7
5	10	6
6	10	5
7	10	4
8	10	3
9	10	2
10	10	1

Since each character is either an in-band (e.g., DATA) character or an out-of-band command or synchronization (i.e., SYNC) character, the following sets forth the possible combinations of DATA and SYNC characters among the 1st, 2nd and 3rd 10-bit characters contributing to the 21-bit window:

Combination	1st Character	2nd Character	3rd Character
A	SYNC	SYNC	SYNC
B	SYNC	SYNC	DATA
C	SYNC	DATA	DATA
D	DATA	DATA	DATA
E	DATA	DATA	SYNC
F	DATA	SYNC	SYNC

For example, the 21-bit window could be comprised of two bits of a 1st SYNC Character, ten bits of a 2nd DATA Character, and nine bits of a 3rd DATA Character (i.e., Combination C).

During low-transition mode operation, all in-band (e.g., DATA) characters include a maximum of three logical transitions or, equivalently, more than four "non-transitions" in logical value between the ten adjacent bits of the in-band character. Accordingly, during low-transition mode operation if the 2nd character is a DATA character, it will include more than four logical non-transitions. Since

Condition I indicates that the number of logical non-transitions within the entire 21-bit window will be exactly four when three identical SYNC characters are present therein, Condition I will not be satisfied when the 2nd character is a DATA character since it would include more than four logical non-transitions. Hence, if Condition I is to be satisfied then the 21-bit window cannot be comprised of the character sets specified by Combinations C, D and E (i.e., 2nd character is a DATA character).

In accordance with the invention, the synchronization characters listed above have been chosen such that Condition II will be satisfied if the 1st and 3rd characters transmitted during any preamble period are identical. Hence, Combination B and F do not satisfy Condition II. It follows that only Combination A (i.e., three consecutive SYNC characters) satisfies both conditions I and II.

Referring to FIG. 5, if both Condition I and Condition II are satisfied (step 190) then selected adjacent bits within Groups A and B are AND'ed (step 196) as described below in order to identify the frame boundaries of the SYNC characters detected within the 21-bit window. Since each SYNC character within the 21-bit window is loaded by the deserializer 34 into latch 150, the frame boundary of each SYNC character may be identified in terms of the one of the bit lines R[9:0] from the deserializer 34 upon which the first bit of each such SYNC character is impressed. Once this identification is achieved, the decoder __ is informed of the identity of this bit line R[9:0] by way of frame boundary pointer 118.

The AND operation of step 196 is performed between all of the adjacent bits in the third latch 158 (i.e., L3[9:0]), as well as between L3[0] and L2[9]. When the result of step 190 has indicated that both Condition I and Condition II are satisfied, the result of the AND operation of step 196 will produce only a single logical one indicative of the value of the frame boundary pointer 118. In the current example, L3[9:0] := {1011010010} and L2[9] :=[1], and thus the result of the AND operation of step 196 is {0010000000}. That is, the third position in the 21-bit window corresponds to the first bit of a synchronization character. Accordingly, in

the current example the frame boundary pointer 118 would be set so as to identify the third (RX[7]) of the ten bit lines RX[9:0] as carrying the first bit of each 10-bit character produced by the deserializer 34.

In the exemplary embodiment a preambling sequence (i.e., three repetitions of the same out-of-band SYNC character) is sent upon system power-up as well as during lapses in data transmission over the serial link 30. This allows timing synchronization to be maintained between the encoder 18 and decoder 44 even in the absence of data transmission for extended periods.

FIG. 6 is a flow chart depicting the synchronization process effected by the decoder synchronization module 114 during high-transition mode operation. As is indicated by FIG. 6, the high-transition mode synchronization process is substantially similar to that performed during low-transition mode operation (FIG. 5). In particular, the high-transition mode synchronization process differs from the low-transition mode synchronization process primarily in that:

(i) In step 160', an exclusive-OR (XOR) rather than an exclusive-NOR (XNOR) operation is performed upon adjacent bits within the latches 150', 154' and 158'.

(ii) In step 196', an AND operation is performed between each bit and the complement of the bit immediately to the right (e.g., L3[9] AND L3[8], L3[8] AND L3[7], L3[7] AND L3[6], and so on. In this way the AND gate producing a logical "1" identifies a "10" or "falling edge" sequence corresponding to the frame boundary.

V. Hardware Implementations of Encoder and Decoder Synchronization Module

In this section a description is provided of a specific hardware implementation of the encoder 18, and of an implementation of the decoder synchronization module 114 suitable for use during low-transition mode operation.

IPR2024-001428
HP – EX1002, Page 150 of 487

A description of an exemplary hardware realization of the decoder 44 in random logic was provided above in section III.

FIGS. 7A and 7B provide a schematic representation of an exemplary implementation of the encoder 18. The 8-bit parallel data D[7:0] from latch 70 to be encoded is seen to be provided to seven exclusive-OR gates 240 of the transition counter 74. The outputs of the exclusive-OR gates 240 are provided to a set of full-adders 242, 244, 246 and 248. The carry output (C) of the full-adder 248 corresponds to the COUNT line 78, and indicates whether less than four logical transitions exist between the bits in the data D[7:0]. When an out-of-band command is being received from latch 70, a command line (TX_CMD) provided to NOR gate 260 is raised so as to prevent the COUNT line 78 from causing inversion of the even bits of D[7:0] within CABI logic 90. Otherwise, when the data D[7:0] from latch is being encoded in accordance with the invention, the output 78' of NOR gate 260 tracks the logical value of COUNT line 78.

As is indicated by FIG. 7A, in the exemplary embodiment CABI logic 90 is comprised of a plurality of NOR gates 270. Each NOR gate 270 includes one input coupled to COUNT line 78', and another input connected to one of the even bits of D[7:0]. The output of CABI logic 90 is provided to intermediate latch 94, which is coupled to the input of disparity checker 96 (FIG. 7B).

Turning to FIG. 7B, the disparity checker 96 includes four AND gates 290-293 for determining occurrences of "11" within the conditionally bit-inverted byte E[7:0]. Similarly, four NOR gates 296-299 are provided for detecting occurrences of "00" within E[7:0]. Since pattern of "01" and "01" within E[7:0] are already "DC-balanced" in the sense of including equal numbers of ones and zeroes, there exists no need to detect such patterns during the DC-balancing process effected by the circuitry of FIG. 7B. A first full-adder 302 and first half-adder 306 are disposed to count the occurrences of "11" detected by AND gates 290-293. In like manner a second full-adder 308 and second half-adder 312 are provided for counting the

occurrences of "00" detected by the NOR gates 296-299. A first pair of full-adders 316 and 318 determine the difference in the counted occurrences of "11" and "00".

The DC-balancing module 98 includes a three-input NOR gate 330, a first exclusive-OR gate 332, a latch 336 and a second exclusive-OR gate 338. When the occurrences of "11" and "00" are determined to be equivalent by full-adders 316 and 318, the complement of E[8] determines the value of T[9], and hence whether the byte E[7:0] is inverted by CTBI logic 100. When the counted occurrences of "00" and "11" are not equivalent, the value of T[9] comprises the output of XOR gate 332. In this regard a first input 342 to XOR gate 332 comprises the most significant bit (MSB) produced by the pair of full-adders 316 and 318, which is equivalent to the MSB of the current disparity D_{cur} (i.e., the difference in "1's and "0's" in E[7:0]). A second input 344 to XOR gate 332 corresponds to the MSB of the cumulative disparity D_{cum}. As is indicated by FIG. 7B, the latch 99 for storing the cumulative disparity is comprises of three registers 350-352. The cumulative disparity is updated by a disparity updater 108 comprised of a backward chain of full adders 356 and 358, a set of three exclusive-OR gates 360-362, and a corresponding set of three full-adders 365-367. Finally, CTBI logic 100 includes a set of eight exclusive-OR gates 374.

FIGS. 8A and 8B provide a schematic representation of a preferred implementation of the decoder synchronization module 114. In FIG. 8A, the 10-bit latches 150, 154 and 158, for storing L3[9:0], L2[9:0] and L1[9:0], respectively, may each be realized using an array of ten D-type flip-flops. A plurality of XNOR gates 402 are provided for XNOR'ing adjacent bits included within the 21-bit window corresponding to L3[9:0], L2[9:0] and l1[9]. The Group A, Group B, Group C, and Group D outputs of the XNOR gates 402 are then respectively provided to the "1" counters 172, 174, 176 and 178. As is indicated by FIG. 8A, the existence of Condition I is detected by random logic 190a, which is comprised of four NAND gates 410-413.

IPR2024-001428
HP – EX1002, Page 152 of 487

Turning now to FIG. 8B, the existence of Condition II is identified by an arrangement of random logic identified by reference numeral 190b. Random logic 190b includes a set of ten XOR gates 422, the outputs of which are provided as shown to NAND gates 426 and 428. The outputs of NAND gates 426 and 428 are coupled to the inputs of a NOR gate 430, the output of which is driven to a logical "1" when Condition II is satisfied. Finally, the logical state of the output (Sync_Sig) of an AND gate 440 is indicative as to whether Condition I and Condition II have been satisfied (i.e., as to whether synchronization has been achieved). If so, adjacent bits within L3[9:0] and L2[9] are AND'ed (step 196 of FIG. 5) by a set of ten AND gates 450. The outputs PTR[9:0] comprise the frame boundary pointer 118, which informs decoder 44 as to which of the bit lines RX[9:0] correspond to the first bit T[0] of each transmitted 10-bit character T[9:0].

VI. Alternate Embodiments

Transition optimization based on 1-bit count

Figures 9A and 9B are flowcharts depicting an alternative encoding technique for producing a transition-optimized signal stream.

Referring to the flowchart of Figure 9A, during a transition minimization step 501a an input frame of data, such as the 8-bit frame D[0:7] 503a, is processed as described below. Specifically, in step 510a, the number of bits set to in "1" in frame 503a are counted. The count value is depicted in the flowchart as "S". In step 515a, the count is compared to a predetermined threshold value that represents one-half of the frame size. Thus, for an 8-bit frame, the count is compared to the value 4.

If the count value exceeds 4, in Step 520a a 9-bit transition-minimized frame is formed by performing a $NRZI_0$-encoding operation over 8-bit frame D[0:7] and prepending a '0' bit in order to form transition-minimized frame E[0:8] 504a. The

$NRZI_0$-encoding operation consists in part of setting the low-order bit E[0] of frame 504a to the value of the high-order D[0] bit of frame 503a. In addition, each bit E[1] through E[7] in frame 504a is obtained by performing an XNOR operation using the corresponding bit in D[1] through D[7] in frame 503a and the next-higher-order bit, and finally by prepending a binary '0'. That is, to produce an $NRZI_0$-encoded 9-bit frame from an unencoded 8-bit frame:

```
E[0] <= D[0]
E[1] <= D[0] XNOR D[1]
E[2] <= D[1] XNOR D[2]
        . . .
E[6] <= D[5] XNOR D[6]
E[7] <= D[6] XNOR D[7]
E[8] <= '0'
```

If the count value derived in Step 510a is less than 4, in Step 521 a 9-bit transition-minimized frame is formed by performing a $NRZI_1$-encoding operation over 8-bit frame D[0:7] 503a and prepending a '1' bit to form transition-minimized frame E[0:8] 504a. The $NRZI_1$-encoding consists of setting the low-order bit E[0] of frame 504a to the value of the high-order D[0] bit of frame 503a. Moreover, each bit E[1] through E[7] is set to a value determined by performing an XOR operation using the corresponding bit D[1] through D[7] in frame 503a and the next-higher-order bit, and finally by prepending a binary '1'. That is, to produce an $NRZI_1$-encoded 9-bit frame from an unencoded 8-bit frame:

```
E[0] <= D[0]
E[1] <= D[0] XOR D[1]
E[2] <= D[1] XOR D[2]
        . . .
E[6] <= D[5] XOR D[6]
E[7] <= D[6] XOR D[7]
```

E[8] < = '1'

If the count value derived in Step 510a is equal to 4, then in Step 517a the value of D[0] is used to select whether transition-minimized frame 504a is formed by $NRZI_0$-encoding or $NRZI_1$-encoding. If D[0] is equal to '1', transition-minimized frame 504a is formed by $NRZI_0$-encoding frame 503a in Step 520a. If D[0] is equal to '0', transition-minimized frame 504a is formed by $NRZI_1$-encoding frame 503a in Step 521.

Figure 9B depicts a flowchart of an encoding process or producing a transition-maximized signal stream. The transition-maximization process 501b of Figure 9B is similar to the transition-minimization process of Figure 9A, except that the positions of the $NRZI_0$-encoding and $NRZI_1$-encoding blocks are swapped.

In step 510b, the number of bits in 8-bit frame D[0:7] which are set to "1" are counted. The count value is depicted in the flowchart of Figure 9B as "S". In step 515b, the count is compared to a predetermined threshold value that represents one-half of the frame size. Thus, for an 8-bit frame, the count is compared to the value 4.

If the count value exceeds 4, In Step 521b a 9-bit transition-maximized frame is created by performing a $NRZI_1$-encoding operation over 8-bit frame D[0:7] 503a and prepending a '1' bit, thereby forming transition-maximized frame E[0:8] 504b. The $NRZI_1$-encoding consists of setting the low-order bit E[0] of frame 504b to the value of the high-order D[0] bit of frame 503b, and setting each bit E[1] through E[7] to a value determined by XOR'ing the corresponding bit D[0:7] in frame 501b and the next-higher-order bit. A binary '1' is then prependended to the resulting set of bits.

If the count value derived in Step 510b is less than 4, in Step 520b a 9-bit transition-maximized frame is formed by performing a $NRZI_0$-encoding operation

over 8-bit frame D[0:7] and prepending a '0' bit to form transition-maximized frame E[0:8] 504b. The $NRZI_0$-encoding consists of setting the low-order bit E[0] of frame 504b to the value of the high-order D[0] bit of frame 503b, and setting each bit E[1] through E[7] to a value determined by XOR'ing the corresponding bit D[0:7] in frame 501b and the next-higher-order bit. A binary '0' is then prepended to the resulting set of bits.

If the count value derived in Step 510b is equal to 4, Step 517b selects whether transition-maximized frame 504b is formed by $NRZI_0$-encoding or $NRZI_1$-encoding depending on the value of D[0]. If D[0] is equal to '1', transition-maximized frame 504b is formed by $NRZI_1$-encoding frame 503b in Step 521b. If D[0] is equal to '0', transition-maximized frame 504b is formed by $NRZI_0$-encoding frame 503b in Step 520b.

Fig. 10 depicts a logic circuit 530 that implements the transition optimization operations of Figs. 9A and 9B. As shown in Fig. 10, 8-bit frame 503 is presented to counter 532. Counter 532 calculates the number (N_1) of '1' bits and the number (N_0) of '0' bits (not shown) in frame 503, and presents these counts to mode selector 534. Mode selector 534 calculates two logical values. The first logical value represents the conditions 515b, 516b and 517b used in selection of the $NRZI_0$-encoding step 520b of transition-maximizer 501b, and is represented by the logical value

$$[(N_1 .GE. 5) + ((N_1 .EQ. N_0 .EQ. 4) \times (D[0] .EQ. 0))].$$

The second logical value represents the conditions 515a, 516a and 517a used in selection of the $NRZI_0$-encoding step 520a of transition-maximizer 501a, and is represented by the logical value

$$[(N_0 .GE. 5) + ((N_1 .EQ. N_0 .EQ. 4) \times (D[0] .EQ. 1))].$$

Mode-selection signal 536 (labeled "MAX") is selectively set to '0' to select transition-minimized optimization, or is set to '1' to select transition-maximized optimization. Mode selector 534 uses MAX to select one of the two calculated logic values, which results in control signal 535 (labeled "CONT") selecting between $NRZI_0$ and $NRZI_1$ encoding.

8-bit frame 503 is presented as input to an array of XNOR gates 538-1 through 538-7, and also to an array of XOR gates 539-1 through 539-7. XNOR gates 538-1 through 538-7 perform $NRZI_0$ encoding over frame 503 and present the resulting signals, along with '1' bit 541, as input to selector 542. XOR gates 539-1 through 539-7 perform $NRZI_1$ encoding over frame 503 and present the resulting signals, along with '0' bit 540, as input to selector 542. Selector 542 selects between the $NRZI_0$-encoded signal and $NRZI_1$-encoded signal in accordance with control signal 535, and produces as output the 9-bit transition-optimized signal 504 E[0:8]. The transition-optimized signal thus formed may be decoded using the same general method and apparatus discussed in connection with Figs. 3 and 4, above.

Frame-wise DC-balancing using a Frame Synchronous Scrambler for 8B/9B coding.

Figure 11 depicts a balance-encoding circuit 550 for DC-balancing a 9-bit transition-optimized frame using frame-synchronous scrambling. Balance-encoding circuit 550 takes as input a 9-bit transition-optimized frame 504 and produces as output a 9-bit DC-balance-encoded frame 505. Although Figure 11 depicts an exemplary embodiment showing DC-balancing of a 9-bit transition-optimized frame, the invention disclosed requires neither that an incoming frame be 9 bits in length, nor that an incoming frame be transition-optimized.

Balance-encoding circuit 550 employs a shift register generator (SRG) 552, which includes a plurality of one-bit shift registers 554-1 through 554-7 and XOR gate 565. Shift Registers 554-1 through 554-7 are initialized with an arbitrary predetermined non-zero value under control of SYNC signal 560. The SRG is organized in such a

manner that the sequence of shift registers 554-1 through 554-7 make up a pseudo-random binary sequence (PRBS) used for scrambling. In operation, two predetermined bits of the PRBS are XORed, and the resulting value is used to DC-balance an unbalanced 9-bit frame.

The PRBS is disposed to repeatedly cycle through a series of values. The period of the cycle is dependent on the number of shift registers employed in the SRG. The period is equal to 2^N-1, where N is the number of shift registers employed. In the exemplary embodiment shown in figure 11, 7 shift registers are employed, resulting in a period of 2^7-1=127.

As the SRG is clocked, the contents of each shift register (with the exception of high-order shift register 554-7) are loaded into the next-higher-order shift register. That is, the contents of low-order shift register 554-1 are loaded into next-higher-order shift register 554-2; the contents of shift register 554-2 are loaded into shift register 554-3, and so on, until the contents of shift register 554-6 have been loaded into shift register 554-7. Concurrent with these shifts, the contents of the two highest order shift registers 554-6 and 554-7 are presented as input to XOR gate 556. The resulting value is passed from XOR gate 565 to low-order shift register 554-1. In addition, the contents of high-order shift register 554-7 are presented as control signal 562, which is used to control the selective inversion of balance-unencoded frame 504 to produce balance-encoded frame 505.

Balance-unencoded frame 504 is presented to inverter 566, and the resulting 9-bit inverted frame is presented as an input to selector 568. In addition, balance-unencoded frame 504 is presented without inversion as a second input to selector 568. Selector 568 operates under the control of control signal 562 from SRG 552. If control signal 562 is '1', the inverted frame is selected and presented as balance-encoded frame 505; otherwise, the uninverted frame is selected and presented as balance-encoded frame 505.

Figure 12 depicts a balance-decoding circuit 570 for decoding a 9-bit DC-balance-encoded frame using frame-synchronous scrambling. Balance-decoding circuit 570 takes as input a 9-bit balance-encoded frame 505 and produces as output a balance-decoded 9-bit transition-optimized frame 506 ('T').

Balance-decoding circuit 570 employs a shift register generator 572. SRG 572 is of identical order to SRG 552, and is set to the same initial state. That is, SRG 572 has the same number of shift registers as SRG 552, and is initialized to the same arbitrary predetermined non-zero value in response to SYNC signal 580. SRG 572 operates identically to SRG 552, and generates an identical PRBS in operation. Accordingly, for each DC-balanced frame 505 received, the value of control signal 582 is assured of being identical to the value of control signal 562 used to selectively invert the balance-unencoded frame 504. Accordingly, control signal 582 may be employed to selectively invert balance-encoded frame 505 in order to produce balance-decoded frame 506, the value of which is assured of being equal to the balance-unencoded frame 504 provided to Balance-encoding circuit 550.

Balance-encoded frame 505 is seen to be presented to inverter 584, and the resulting 9-bit inverted frame is provided as one input to selector 586. In addition, balance-encoded frame 505 is presented uninverted as a second input to selector 586. Selector 586 operates under the control of control signal 582 from SRG 572. If control signal 582 is '1', the inverted frame is selected and presented as balance-decoded frame 506; otherwise, the uninverted frame is selected and presented as balance-decoded frame 505.

Frame-wise DC-balancing using a Self-Synchronous Scrambler for 8B/9B coding.

Figure 13 depicts a balance-encoding circuit 600 for DC-balancing a 9-bit transition-optimized frame using self-synchronous scrambling (SSS). Balance-encoding circuit 600 takes as input a 9-bit unbalanced transition-optimized frame 504 and produces as output a 9-bit DC-balance-encoded frame 505. Although Figure 13

depicts an exemplary embodiment showing DC-balancing of a 9-bit transition-optimized frame, the invention disclosed requires neither that an incoming frame be 9 bits in length, nor that an incoming frame be transition-optimized.

Balance-encoding circuit 600 includes a shift register generator (SRG) 602 comprised of a plurality of one-bit shift registers 604-1 through 604-7 and XOR gate 605. Shift Registers 604-1 through 604-7 are initialized with an arbitrary predetermined value. The SRG is organized in such a manner that the sequence of shift registers 604-1 through 604-7 make up a pseudo-random binary sequence (PRBS) that will be used for scrambling. In operation, a predetermined bit (e.g., bit 604-7) of the PRBS is XORed with a signal supplied as input to the SRG, and the resulting value is used in DC-balancing an unbalanced 9-bit frame.

As the SRG is clocked during operation, the contents of each shift register (with the exception of high-order shift register 604-7) are loaded into the next-higher-order shift register. That is, the contents of low-order shift register 604-1 are loaded into next-higher-order shift register 604-2; the contents of shift register 604-2 are loaded into shift register 604-3, and so on, until the contents of shift register 604-6 are loaded into shift register 604-7. Concurrent with these shifts, the contents of highest order shift register 604-7 and an input signal 611 are presented as input to XOR gate 605. The resulting value is passed from XOR gate 605 to low-order shift register 604-1. In addition, the value from XOR gate 605 is presented as output signal 612, which is used in derivation of a control signal 626. The control signal 626 controls the selective inversion of balance-unencoded frame 504 to produce balance-encoded frame 505.

Balance-unencoded frame 504 is seen to be presented to counter 622, which is disposed to count the number of 1-bits in balance-unencoded frame 504 and pass the count (N) to logic 624. Logic 624 sets signal Nu to logic '1' if N is greater than or equal to 5, and sets Nu to logic '0' otherwise. As is indicated by Fig. 13,

the signal Nu is presented as input signal 611 to SRG 602 and is also provided to XOR gate 620.

XOR gate 620 performs an exclusive-or operation on the value passed from logic 624 and SRG output signal 612 to produce control signal 626. Control signal 626 is used to control the selective inversion of balance-unencoded frame 504 to produce balance-encoded frame 505.

Balance-unencoded frame 504 is additionally presented to inverter 614. The resulting 9-bit inverted frame is presented as one input to selector 618. In addition, balance-unencoded frame 504 is presented uninverted as a second input to selector 618. Selector 618 operates under the control of control signal 626 from XOR gate 620. If control signal 626 is '1', the inverted frame is selected and presented as balance-encoded frame 505; otherwise, the uninverted frame is selected and presented as balance-encoded frame 505.

Figure 14 depicts a balance-decoding circuit 630 for decoding a 9-bit DC-balance-encoded frame using self-synchronous scrambling. Balance-decoding circuit 630 takes as input a 9-bit balance-encoded frame 505 and produces as output a balance-decoded 9-bit transition-optimized frame 506, labeled 'T'.

Balance-decoding circuit 630 employs a shift register generator 632. SRG 632 is identical in order to SRG 602 and is set to the same initial state. That is, SRG 632 has the same number of shift registers as SRG 602, and is initialized to the same arbitrary predetermined value. In operation, a predetermined bit (e.g., bit 634-7) of the PRBS is XORed with a signal supplied as input to the SRG, and the resulting value is used in DC-balancing an unbalanced 9-bit frame.

As the SRG is clocked during operation, the contents of each shift register (with the exception of high-order shift register 634-7) are loaded into the next-higher-order shift register. That is, the contents of low-order shift register 634-1 are

loaded into next-higher-order shift register 634-2; the contents of shift register 634-2 are loaded into shift register 634-3, and so on, until the contents of shift register 634-6 are loaded into shift register 634-7. Concurrent with these shifts, an input signal 641 is passed to low-order shift register 634-1. In addition, the contents of highest order shift register 634-7 and input signal 641 are presented as input to XOR gate 635. The resulting value is presented as output signal 642, which is used in derivation of a control signal 636.

Accordingly, for each DC-balanced frame 505 received, the value of control signal 636 is assured to be identical to the value of control signal 626 that was used to selectively invert the balance-unencoded frame 504. Accordingly, control signal 636 is employed to selectively invert balance-encoded frame 505 to produce balance-decoded frame 506, which is of a value assured to be equal to the balance-unencoded frame 504 provided to balance-encoding circuit 600.

Balance-encoded frame 505 is seen to be presented to inverter 644, and the resulting 9-bit inverted frame presented as one input to selector 646. In addition, balance-encoded frame 505 is presented uninverted as a second input to selector 646. Selector 646 operates under the control of control signal 636, which is derived by performing an XOR operation with SRG output signal 642 and the output of logic 654. If control signal 636 is '1', the inverted frame is selected and presented as balance-decoded frame 506; otherwise, the uninverted frame is selected and presented as balance-decoded frame 505.

The previous description of the preferred embodiments has been provided to enable any person skilled in the art to make or use the present invention. Various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments without the use of inventive faculty. Thus, the present invention is not intended to

be limited to the embodiments shown herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

WHAT IS CLAIMED IS:

1. A method for producing a DC-balanced sequence of characters from an input sequence of data blocks, said method including the steps of:

receiving one data block from said input sequence of data blocks;

generating a pseudo-random binary sequence comprising a series of pseudo-random binary signals;

determining an inversion condition value from said pseudo-random binary sequence;

selectively inverting said one data block in accordance with said inversion condition value in order to produce said DC-balanced sequence of characters.

2. The method of claim 1 further comprising the steps of:

determining a value representative of logical content of said one data block;

generating a control signal based on said value;

selectively inverting one of said pseudo-random binary signals in accordance with said value.

3. The method of claim 2 wherein said control signal indicates inversion if a number of logic values of predefined type in said data block exceeds a threshold value.

4. An apparatus for producing a DC-balanced sequence of characters from an input sequence of data blocks, said apparatus comprising:

means for receiving one data block from said input sequence of data blocks;

means for generating a pseudo-random binary sequence comprising a series of pseudo-random binary signals;

IPR2024-001428
HP – EX1002, Page 164 of 487

means for determining an inversion condition value from the pseudo-random binary sequence;

means for selectively inverting said one data block in accordance with said inversion condition value in order to produce said DC-balanced sequence of characters.

5. The apparatus of claim 4 further comprising:

means for determining a value representative of logical content of said one data block;

means for generating a control signal based on said value;

means for selectively inverting one of said pseudo-random binary signals in accordance with said value.

6. The apparatus of claim 5 wherein said control signal indicates inversion if a number of said logic values of predefined type in said data book exceeds a threshold value.

7. An apparatus for producing a DC-balanced sequence of characters from an input sequence of data blocks, said apparatus comprising:

a buffer for receiving one data block from said input sequence of data blocks;

a shift register generator for generating a pseudo-random binary sequence comprising a series of pseudo-random binary signals;

an XOR gate for determining an inversion condition value from the pseudo-random binary sequence;

an inverter for selectively inverting said one data block in accordance with said inversion condition value in order to produce said DC-balanced sequence of characters.

IPR2024-001428
HP – EX1002, Page 165 of 487

8. The apparatus of claim 7 further comprising:

a counter for determining a value representative of logical content of said one data block;

an XOR gate for generating a control signal based on said value;

an inventor for selectively inverting one of said pseudo-random binary signals in accordance with said value.

9. The apparatus of claim 8 wherein said control signal indicates inversion if a number of logic "1" values in said one data block exceeds a threshold value.

10. A method for producing a DC-balanced sequence of characters from an input sequence of 8 bit data blocks, said method including the steps of:

selectively complementing bits in said 8 bit data blocks in accordance with a number of logical signals of predefined type in each said 8 bit data block in order to produce selectively complemented data blocks;

determining a cumulative disparity in the number of logical values of different type included within ones of said selectively complemented data blocks previously encoded into ones of said characters;

determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded; and

assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity.

IPR2024-001428
HP – EX1002, Page 166 of 487

11. The method of claim 10 wherein said step of selectively complementing includes the step of selectively complementing predetermined ones of said bits in said 8 bit data blocks having fewer than a predetermined number of said logical signals of predefined type.

12. The method of claim 10 wherein said step of selectively complementing includes the step of selectively complementing predetermined ones of said bits in said 8 bit data blocks having more than a predetermined number of said logical signals of predefined type.

13. The method of claim 10 further including the step of setting a predetermined bit in said candidate character to a first logical value when a current 8 bit data block associated therewith has fewer than a predetermined number of logical signals of predefined type, and setting said predetermined bit to the complement of said first logical value otherwise.

14. The method of claim 10 further including the step of setting a predetermined bit said candidate character to a first logical value when a current 8 bit data block associated therewith has more than a predetermined number of logical signals of predefined type, and setting said predefined bit to the complement of said first logical value otherwise.

15. The method of claim 10 wherein the logical value of a predetermined bit in each character assigned to one of said selectively complemented data blocks indicates whether a candidate character or complement thereof was assigned to said selectively complemented data block.

16. The method of claim 10 wherein said step of determining a current disparity in a candidate character includes the step of performing a logic analysis upon bits of said candidate character.

17. The method of claim 10 further including the step of assigning specific ones of said 8 bit data blocks having fewer than a predetermined number of logical signals of predefined type to be representative of special characters, said special characters being encoded distinctly from ones of said 8 bit data blocks not corresponding to ones of said special characters.

18. A binary data encoding apparatus for producing a DC balanced sequence of characters from an input sequence of 8 bit data blocks, said binary data encoding apparatus comprising:

means for selectively complementing bits in said 8 bit data blocks in accordance with a number of logical signals of predefined type in each said 8 bit data block in order to produce selectively complemented data blocks;

means for determining a cumulative disparity in the number of logical values of different type included within ones of said selectively complemented data blocks previously encoded into ones of said characters;

means for determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded; and

means for assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity.

19. The binary data encoding apparatus of claim 18 wherein said means for selectively complementing includes means for selectively complementing predetermined ones of said bits in said 8 bit data blocks having fewer than a predetermined number of said logical signals of predefined type.

20. The binary data encoding apparatus of claim 18 wherein said means for selectively complementing includes means for selectively complementing predetermined ones of said bits in said 8 bit data blocks having more than a predetermined number of said logical signals of predefined type.

21. The binary data encoding apparatus of claim 18 further including means for setting a predetermined bit in said candidate character a first logical value when said current 8 bit data block has fewer than a predetermined number of logical signals of predefined type, and means for setting said predetermined bit to the complement of said first logical value otherwise.

22. The binary data encoding apparatus of claim 18 further including means for setting a predetermined bit in said candidate character to a first logical value when said current 8 bit data block has more than a predetermined number of logical signals of predefined type, and means for setting said predetermined bit to the complement of said first logical value otherwise.

23. The binary data encoding apparatus of claim 18 wherein the logical value of a predetermined bit in each character assigned to one of said selectively complemented data blocks indicates whether a candidate character or complement thereof was assigned to said selectively complemented data block.

24. The binary data encoding apparatus of claim 18 wherein said means for determining a current disparity in a candidate character includes means for performing a logic analysis upon bits of said candidate character.

25. The binary data encoding apparatus of claim 18 further including means for assigning specific ones of said 8 bit data blocks having fewer than

a predefined number of logical signals of predefined type to be representative of special characters and for encoding said special characters distinctly from ones of said 8 bit data blocks not corresponding to ones of said special characters.

26. A method for high speed transmission of digital data comprising the steps of:

encoding an input sequence of 8 bit data blocks into a sequence of characters by: (i) selectively complementing bits in an input sequence of 8 bit data blocks in accordance with the number of logical signals of predefined type in each said 8 bit data block in order to produce selectively complemented data blocks, (ii) determining a cumulative disparity in logical values associated with ones of said selectively complemented data blocks previously encoded into ones of said characters, (iv) determining a current disparity in a candidate character associated with a current one of said selectively complemented data blocks being encoded, and (iv) assigning said candidate character to said current one of said selectively complemented data blocks if said current disparity is of a polarity opposite to a first polarity of said cumulative disparity, and assigning the complement of said candidate character to said current one of said selectively complemented data blocks if said current disparity is of said first polarity;

converting each character into a serial data string; and

transmitting each said serial data string over a serial communication link.

27. A method of producing a transition-optimized DC-balanced sequence of characters from an input sequence of data blocks, said method including the steps of:

receiving one input data block from said input sequence of data blocks, said input data block consisting of N bits;

generating a transition-optimized data block from said input data block;

generating a transition-optimized DC-balanced data block from said transition-optimized data block in order to produce said DC-balanced sequence of characters, said transition-optimized DC-balanced data block consisting of fewer than N+2 bits.

28. The method of claim 27, wherein N is equal to 8.

29. A method of producing a DC-balanced sequence of characters from an input sequence of data blocks, said method including the steps of:

receiving one input data block from said input sequence of data blocks, said input data block consisting of N bits;

generating a DC-balanced data block from said input data block in order to produce said DC-balanced sequence of characters, said DC-balanced data block consisting of at most N bits.

30. The method of claim 29, wherein N is equal to 8.

31. The method of claim 29, wherein N is equal to 9.

32. An apparatus for producing a transition-optimized DC-balanced sequence of characters from an input sequence of data blocks, said apparatus comprising:

means for receiving one input data block from said input sequence of data blocks, said input data block consisting of N bits;

means for generating a transition-optimized data block from said input data block;

IPR2024-001428
HP – EX1002, Page 171 of 487

means for generating a transition-optimized DC-balanced data block from said transition-optimized data block in order to produce said DC-balanced sequence of characters, said transition-optimized DC-balanced data block consisting of fewer than N+2 bits.

33. The apparatus of claim 32, wherein N is equal to 8.

34. An apparatus for producing a DC-balanced sequence of characters from an input sequence of data blocks, said apparatus comprising:

means for receiving one input data block from said input sequence of data blocks, said input data block consisting of N bits;

means for generating a DC-balanced data block from said input data block in order to produce said DC-balanced sequence of characters, said DC-balanced data block consisting of at most N bits.

35. The apparatus of claim 34, wherein N is equal to 8.

36. The apparatus of claim 35, wherein N is equal to 9.

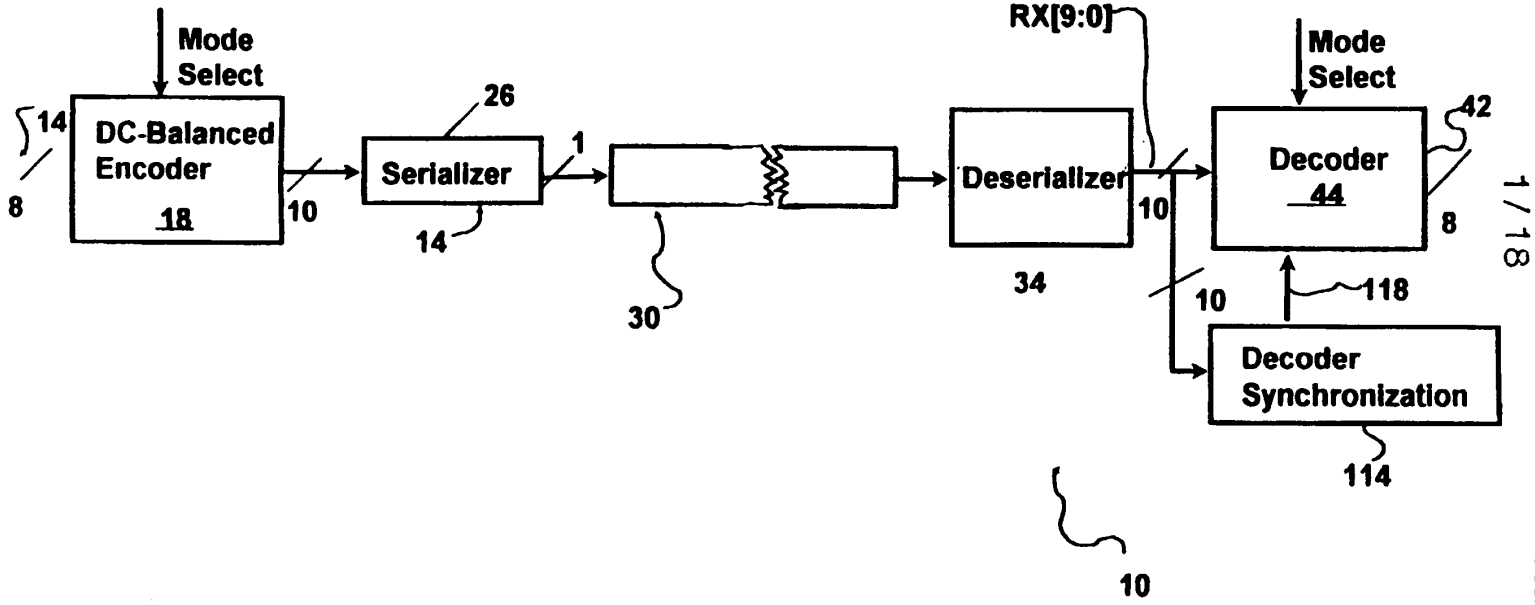


FIG. 1

IPR2024-001428
HP – EX1002, Page 173 of 487

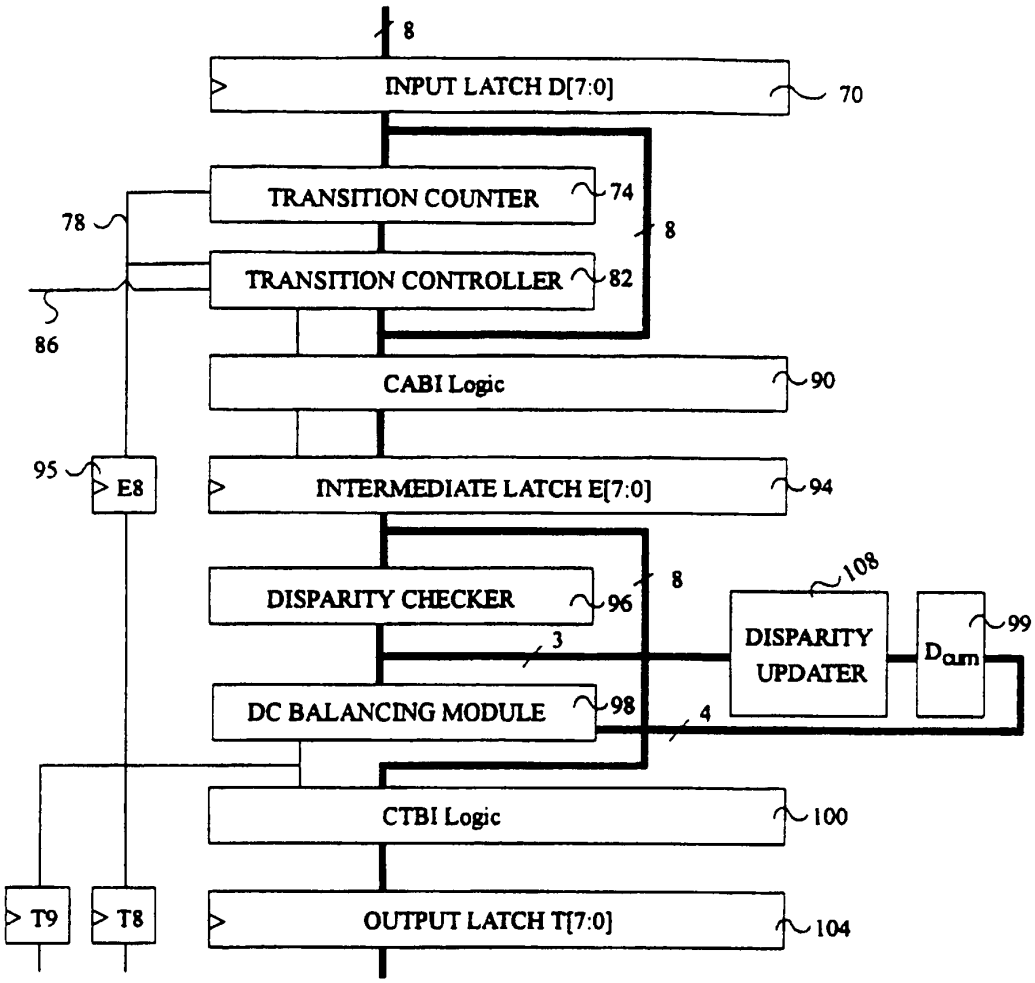

8
INPUT LATCH D[7:0]
70
TRANSITION COUNTER
74
78
8
TRANSITION CONTROLLER
82
86
CABI Logic
90
95
E8
INTERMEDIATE LATCH E[7:0]
94
DISPARITY CHECKER
96
8
108
DISPARITY UPDATER
Dcum
99
3
DC BALANCING MODULE
98
4
CTBI Logic
100
T9
T8
OUTPUT LATCH T[7:0]
104

FIG. 2

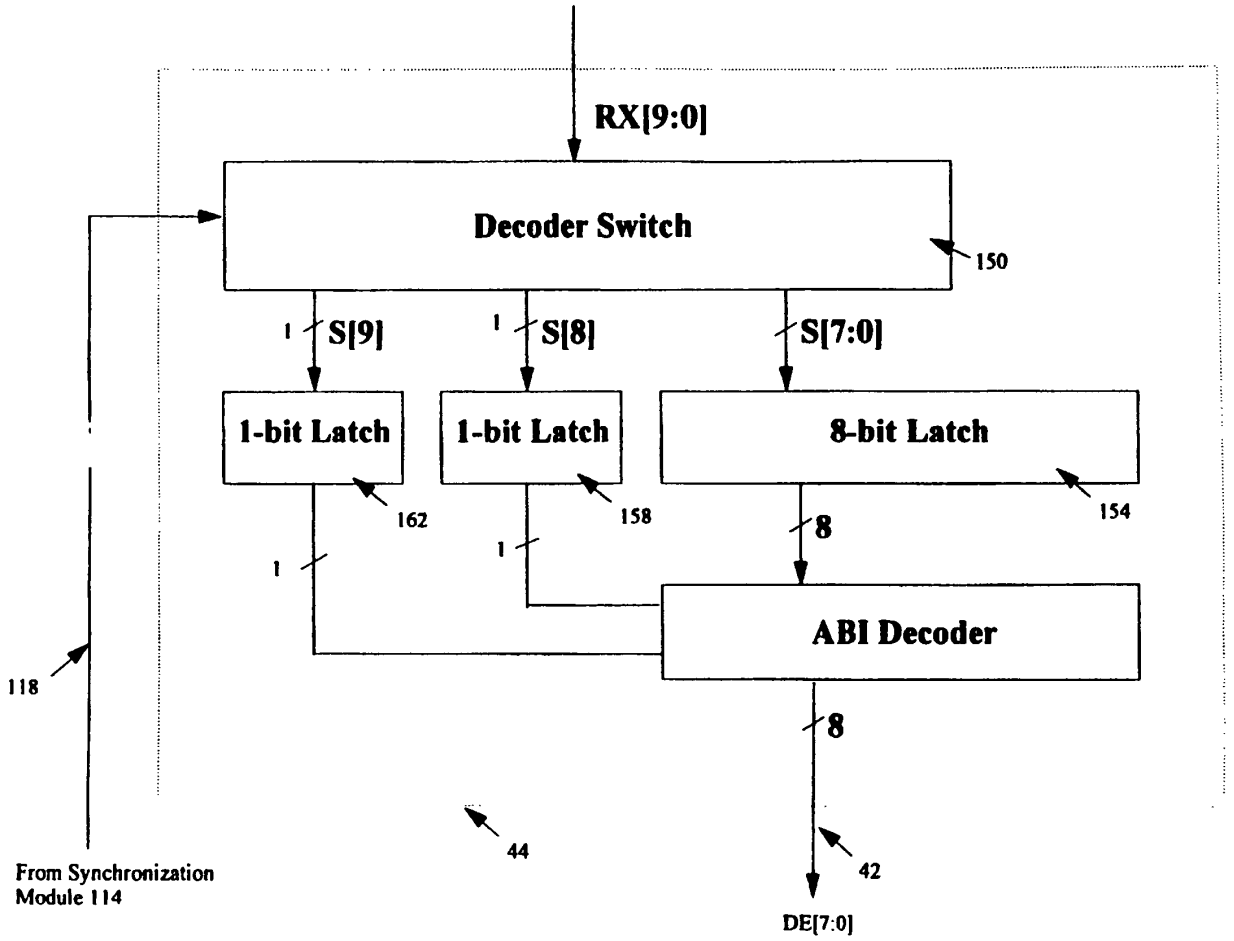

RX[9:0]
Decoder Switch
150
1
S[9]
1
S[8]
S[7:0]
1-bit Latch
1-bit Latch
8-bit Latch
162
158
154
1
1
8
ABI Decoder
118
8
44
42
From Synchronization Module 114
DE[7:0]

FIG. 3

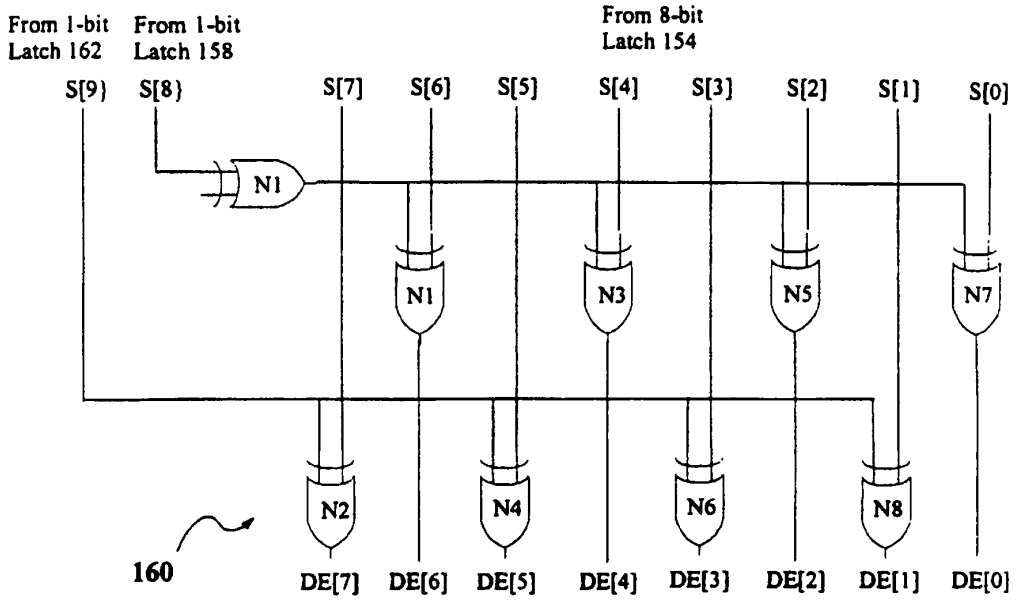

From 1-bit Latch 162
From 1-bit Latch 158
From 8-bit Latch 154
S[9]
S[8]
S[7]
S[6]
S[5]
S[4]
S[3]
S[2]
S[1]
S[0]
N1
N1
N3
N5
N7
N2
N4
N6
N8
160
DE[7]
DE[6]
DE[5]
DE[4]
DE[3]
DE[2]
DE[1]
DE[0]

FIG. 4

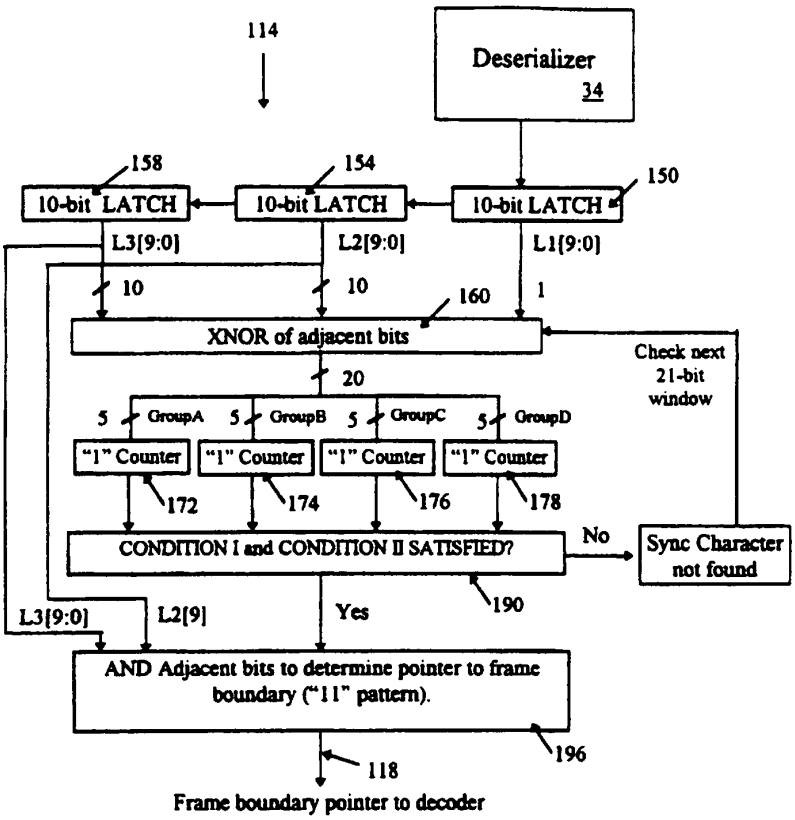

114
Deserializer
34
158
154
150
10-bit LATCH
10-bit LATCH
10-bit LATCH
L3[9:0]
L2[9:0]
L1[9:0]
10
10
160
1
XNOR of adjacent bits
Check next 21-bit window
20
5
GroupA
5
GroupB
5
GroupC
5
GroupD
"1" Counter
"1" Counter
"1" Counter
"1" Counter
172
174
176
178
CONDITION I and CONDITION II SATISFIED?
No
Sync Character not found
190
L3[9:0]
L2[9]
Yes
AND Adjacent bits to determine pointer to frame boundary ("11" pattern).
196
118
Frame boundary pointer to decoder

FIG. 5

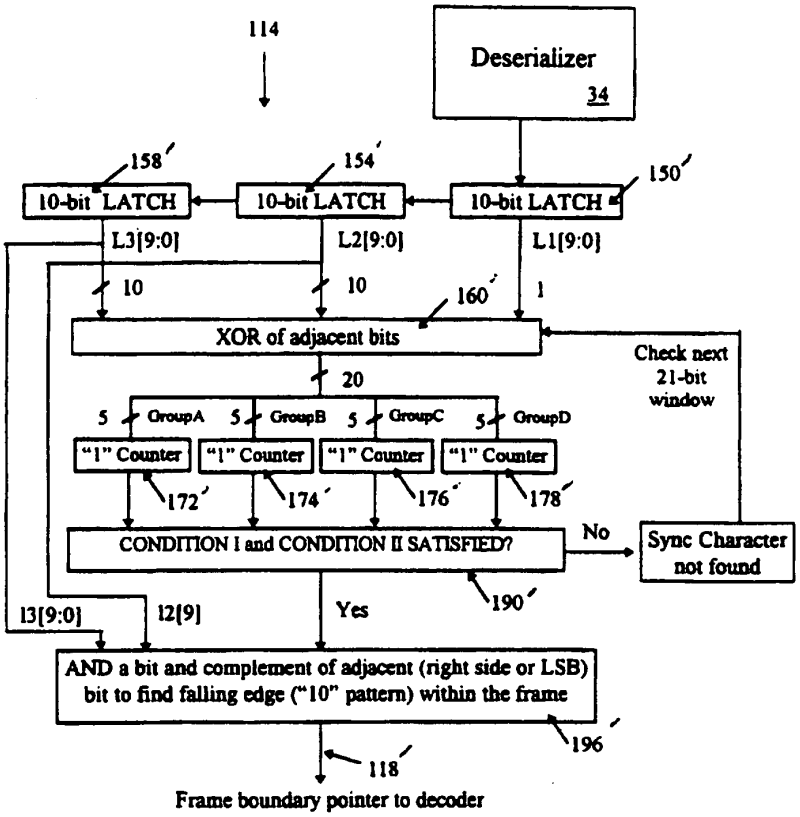

114
Deserializer
34
158
154
150
10-bit LATCH
10-bit LATCH
10-bit LATCH
L3[9:0]
L2[9:0]
L1[9:0]
10
10
160
1
XOR of adjacent bits
Check next 21-bit window
20
5
GroupA
5
GroupB
5
GroupC
5
GroupD
"1" Counter
"1" Counter
"1" Counter
"1" Counter
172
174
176
178
CONDITION I and CONDITION II SATISFIED?
No
Sync Character not found
190
I3[9:0]
I2[9]
Yes
AND a bit and complement of adjacent (right side or LSB) bit to find falling edge ("10" pattern) within the frame
196
118
Frame boundary pointer to decoder

Fig. 6

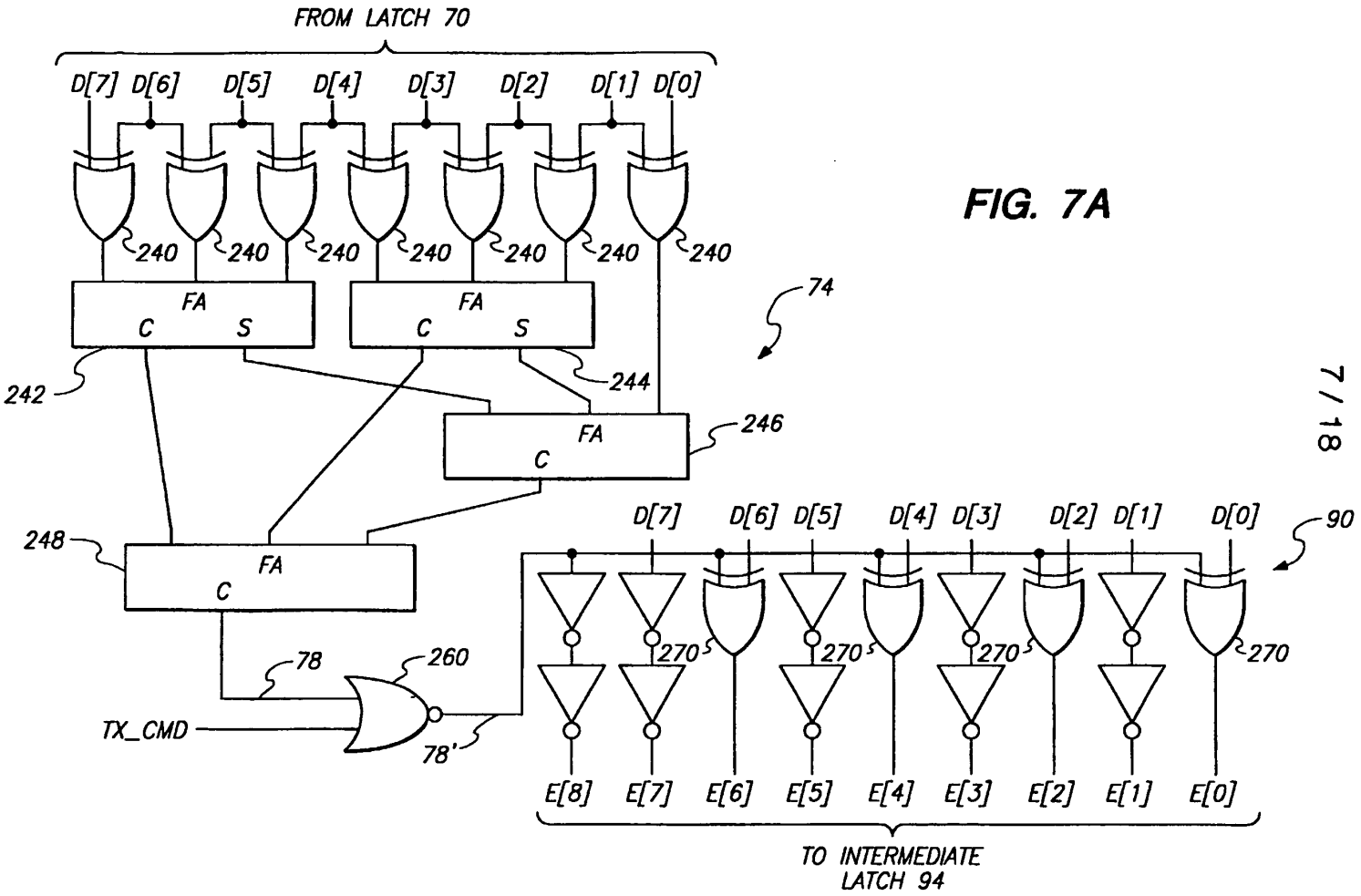
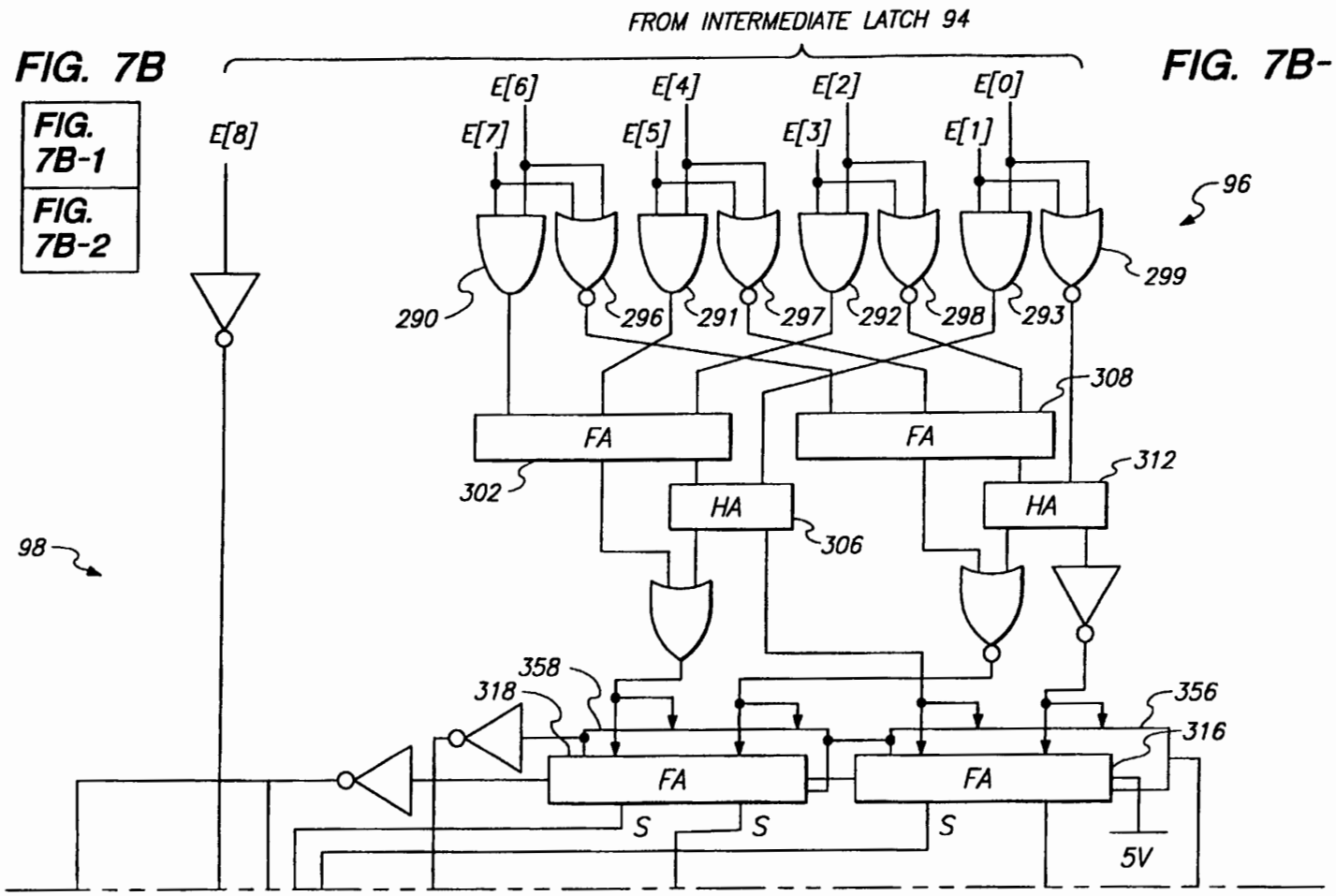

FROM LATCH 70
D[7] D[6] D[5] D[4] D[3] D[2] D[1] D[0]
240
FA
C S
74
242
244
246
248
78
260
TX_CMD
78'
270
90
E[8] E[7] E[6] E[5] E[4] E[3] E[2] E[1] E[0]
TO INTERMEDIATE LATCH 94

FIG. 7A

FIG. 7B

FIG. 7B-1

FIG. 7B-2



SUBSTITUTE SHEET (RULE 26)

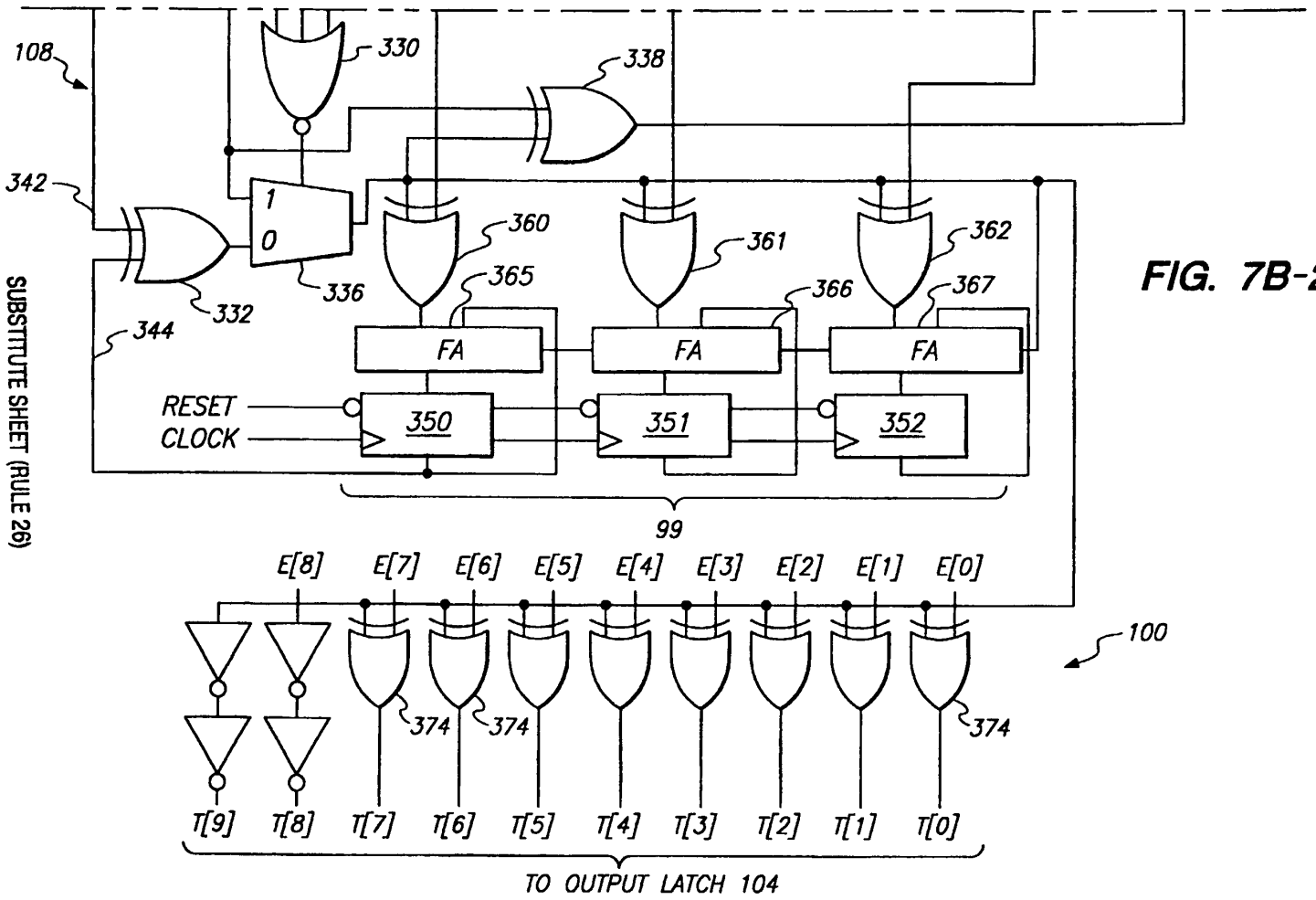


FIG. 7B-2

SUBSTITUTE SHEET (RULE 26)

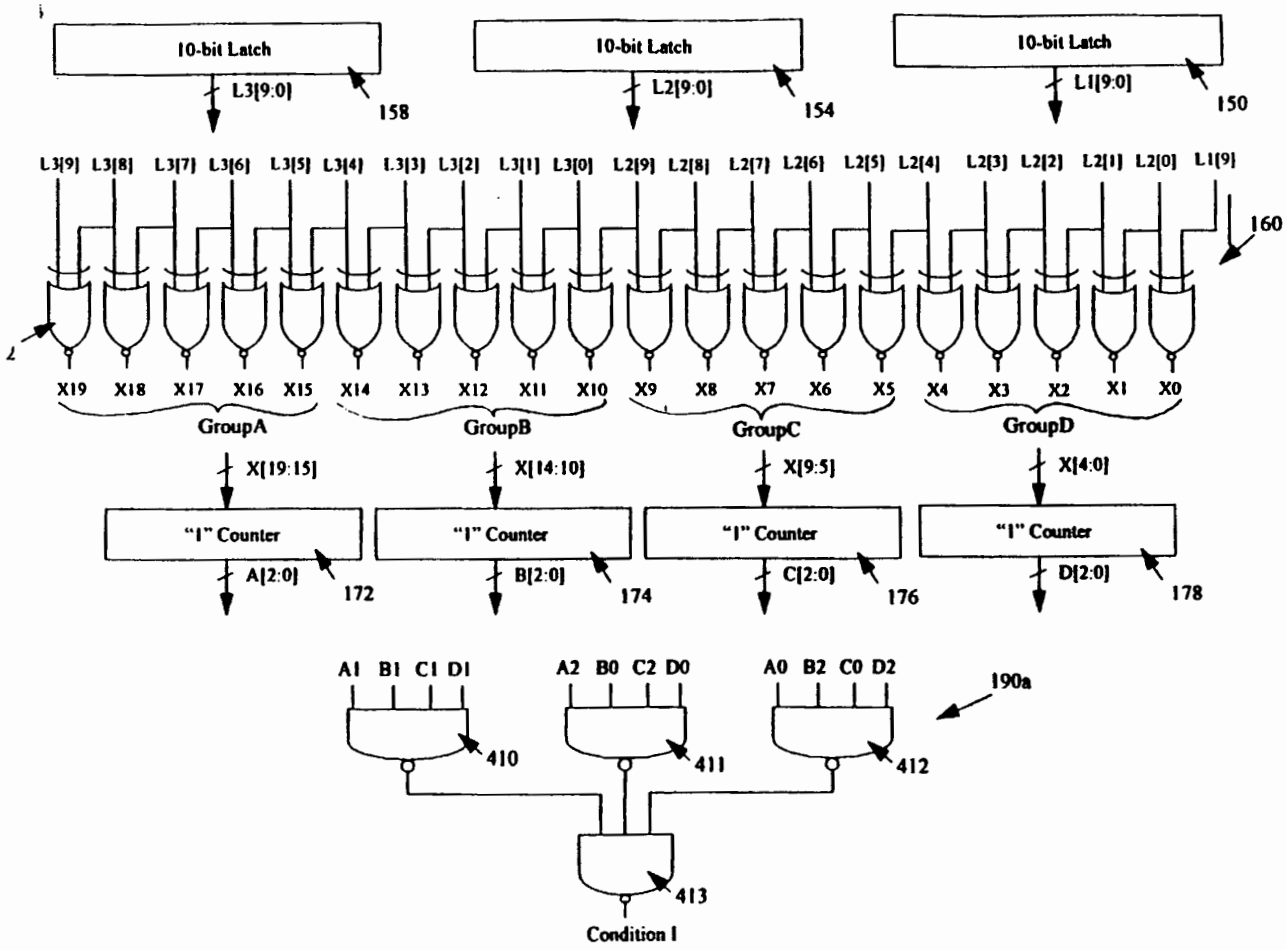


FIG. 8A

IPR2024-001428
HP – EX1002, Page 182 of 487

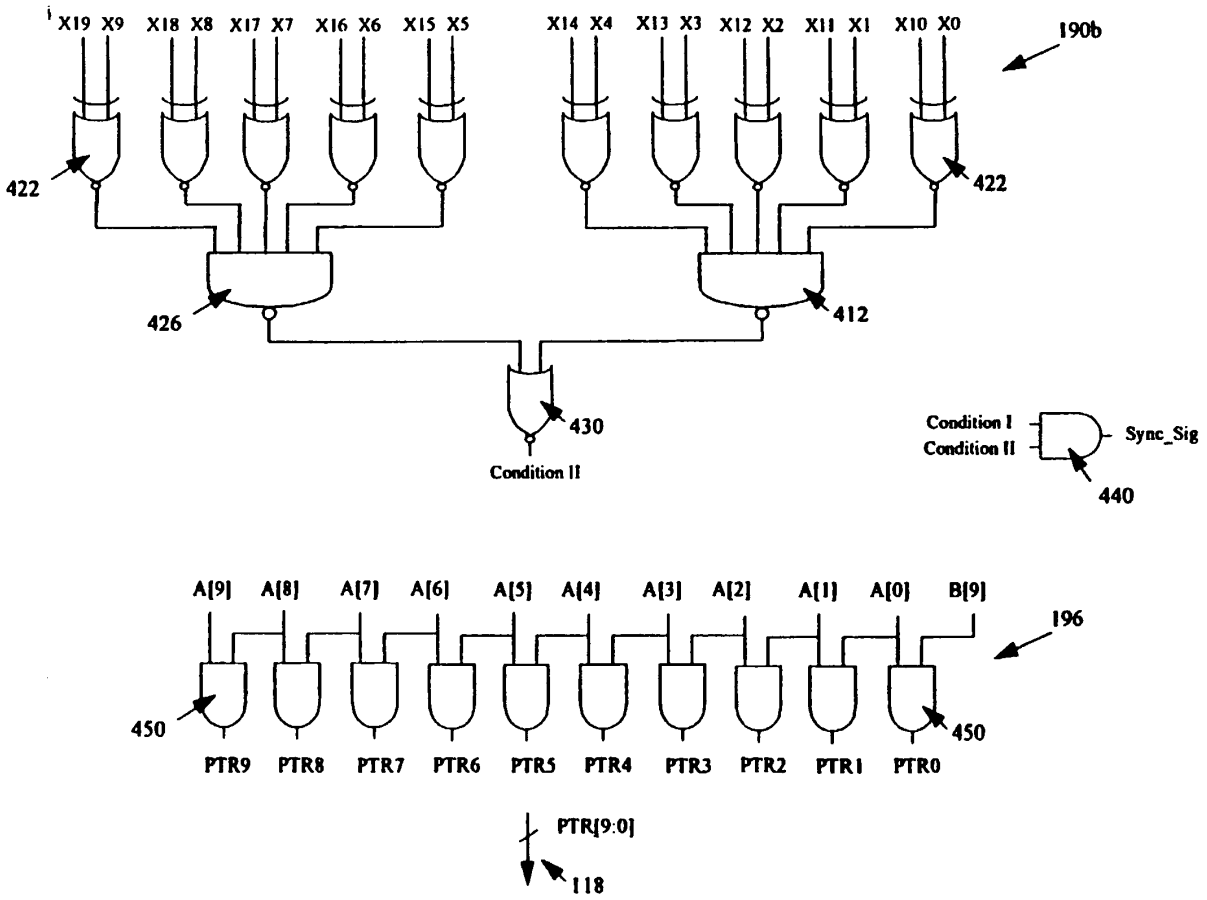

X19 X9
X18 X8
X17 X7
X16 X6
X15 X5
X14 X4
X13 X3
X12 X2
X11 X1
X10 X0
190b
422
422
426
412
430
Condition II
Condition I
Condition II
Sync_Sig
440
A[9] A[8] A[7] A[6] A[5] A[4] A[3] A[2] A[1] A[0] B[9]
196
450
450
PTR9 PTR8 PTR7 PTR6 PTR5 PTR4 PTR3 PTR2 PTR1 PTR0
PTR[9:0]
118
Frame Boundary Pointer to Decoder 44

FIG. 8B

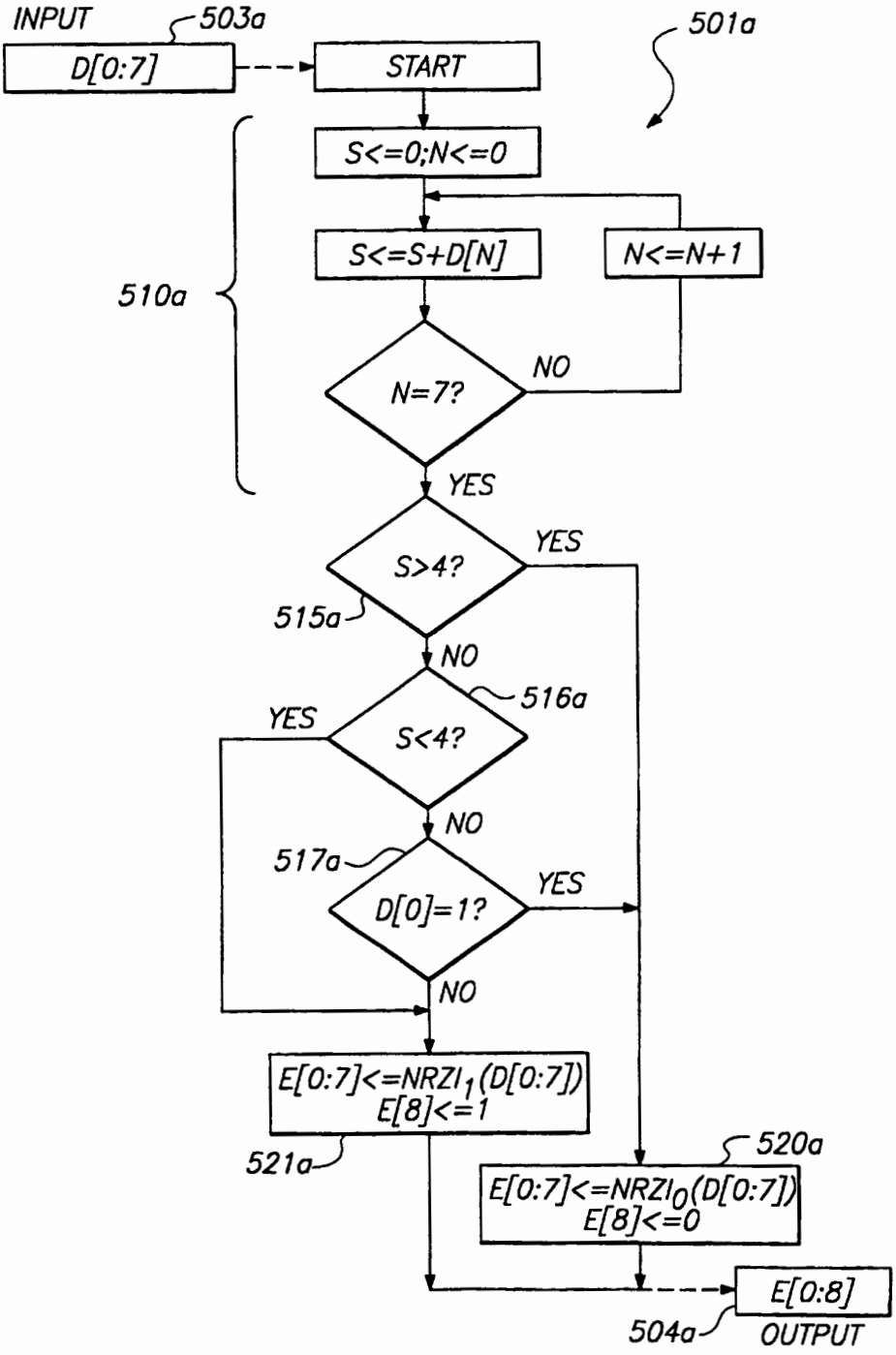


FIG. 9A

IPR2024-001428
HP – EX1002, Page 184 of 487

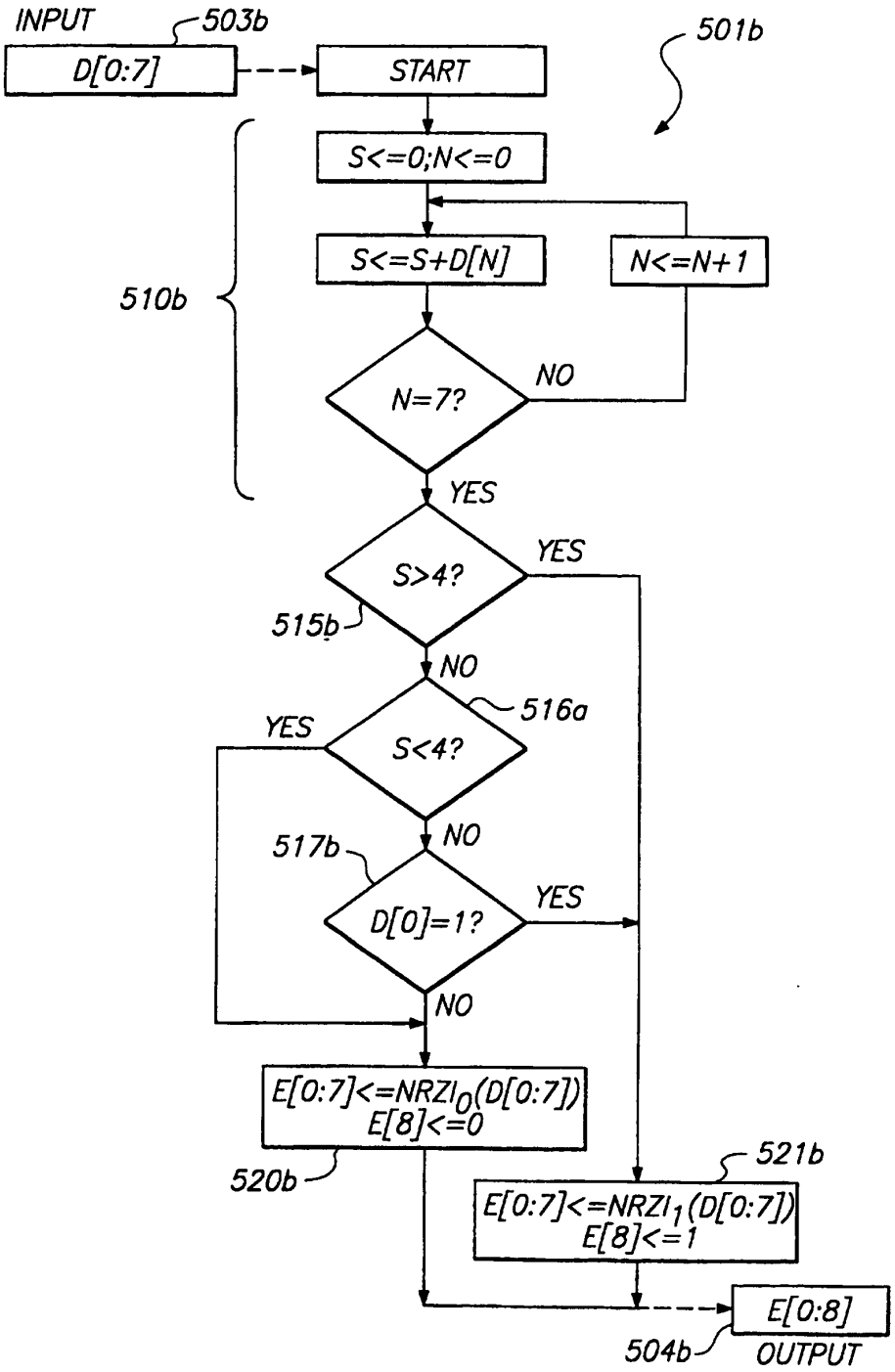


FIG. 9B

IPR2024-001428
HP – EX1002, Page 185 of 487

SUBSTITUTE SHEET (RULE 26)

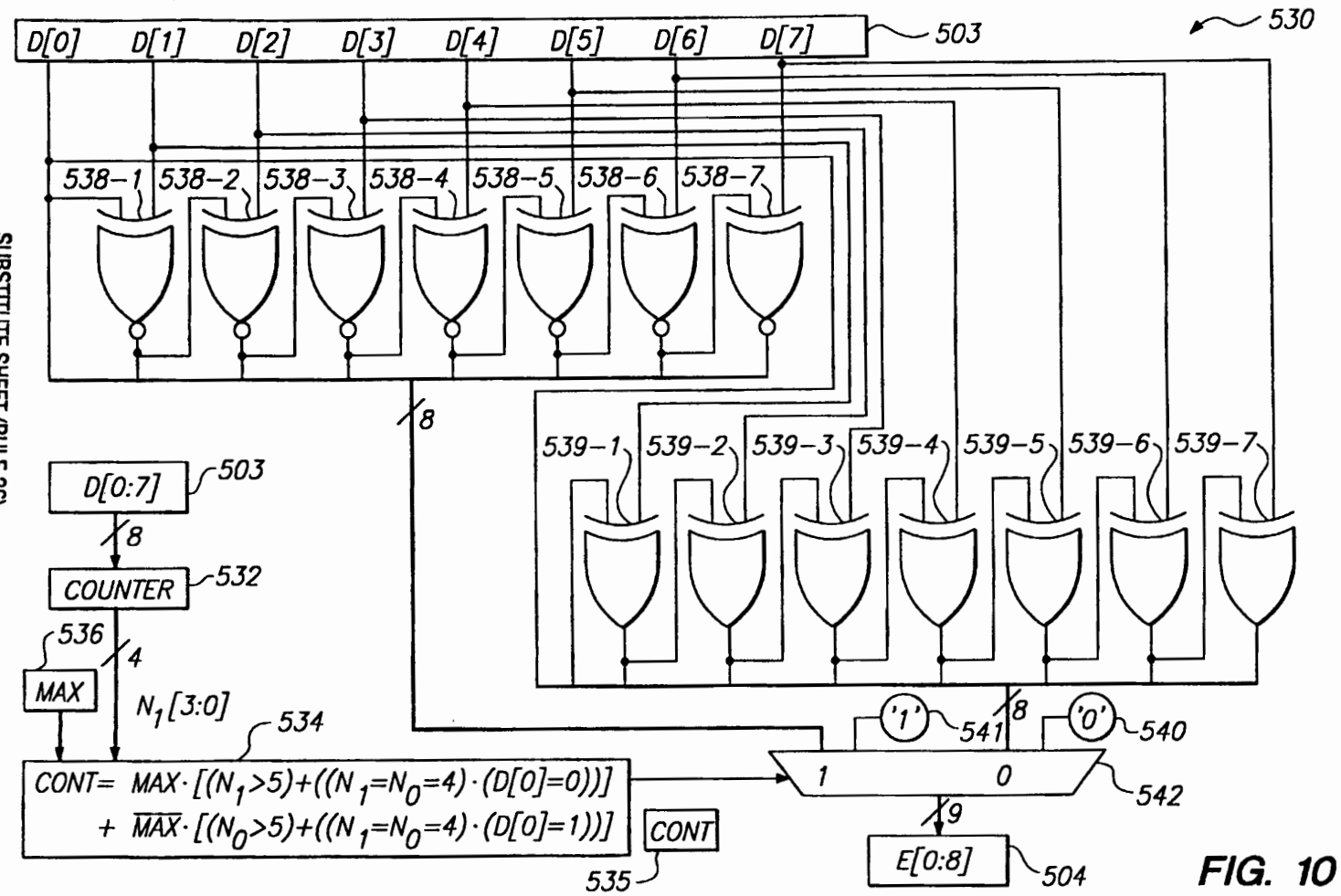


FIG. 10

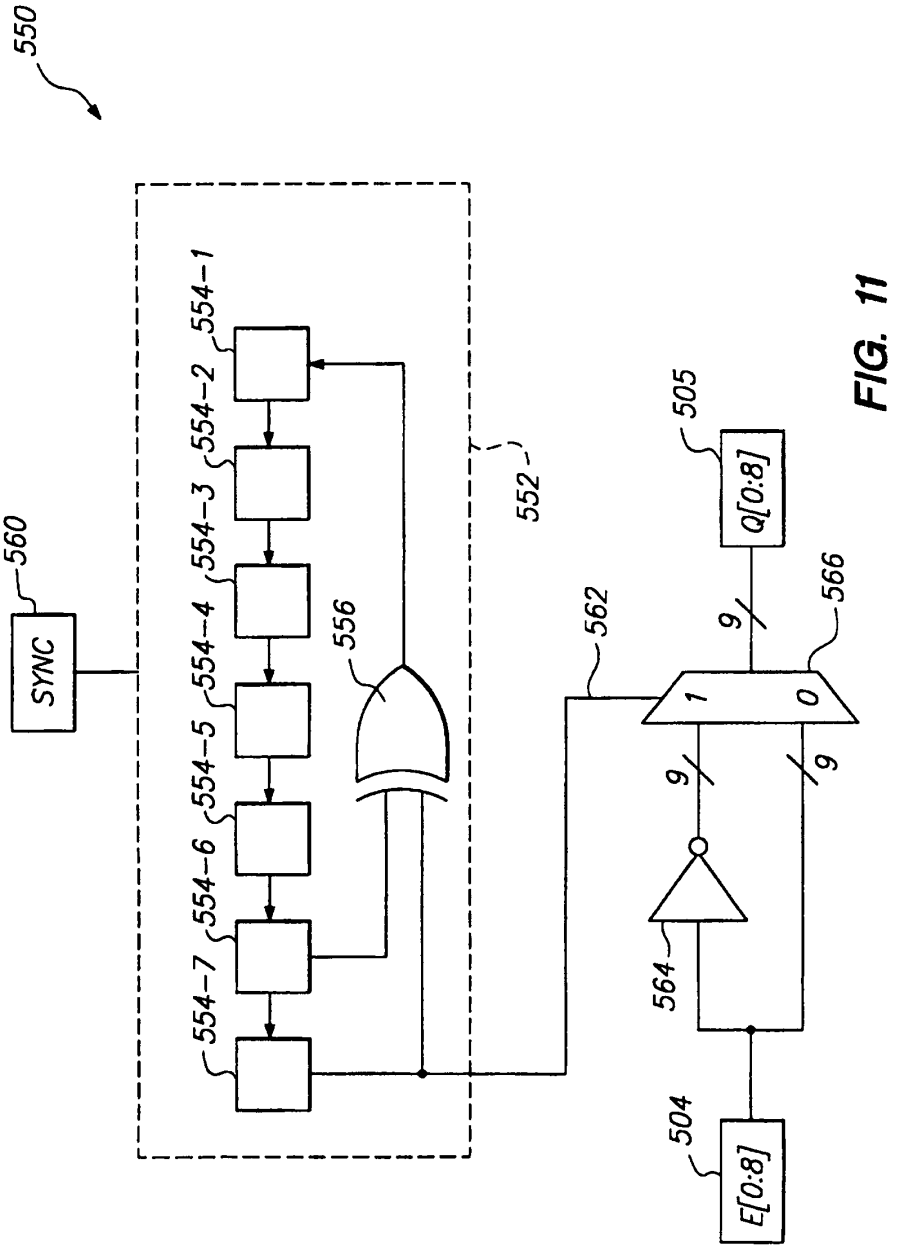


FIG. 11

IPR2024-001428
HP – EX1002, Page 187 of 487

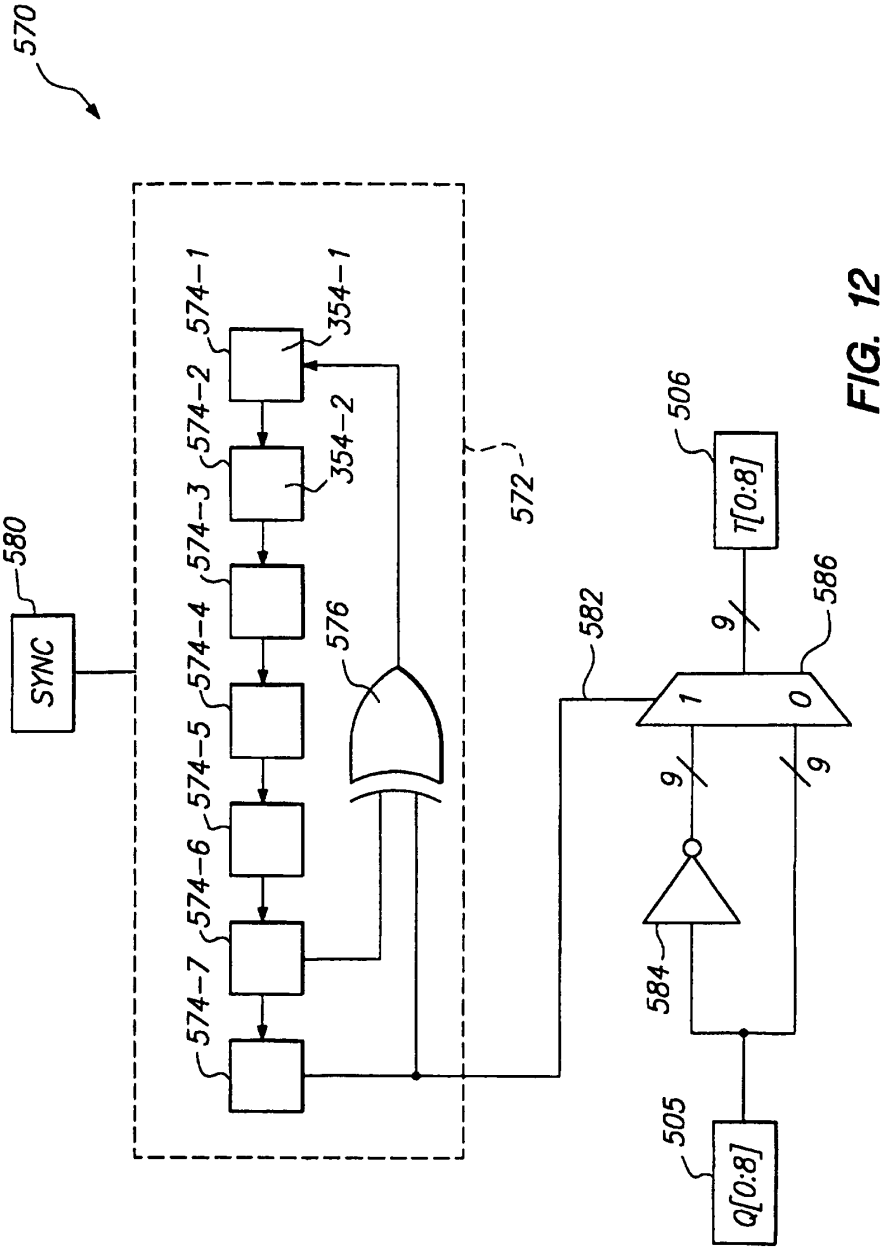


FIG. 12

IPR2024-001428
HP – EX1002, Page 188 of 487

SUBSTITUTE SHEET (RULE 26)

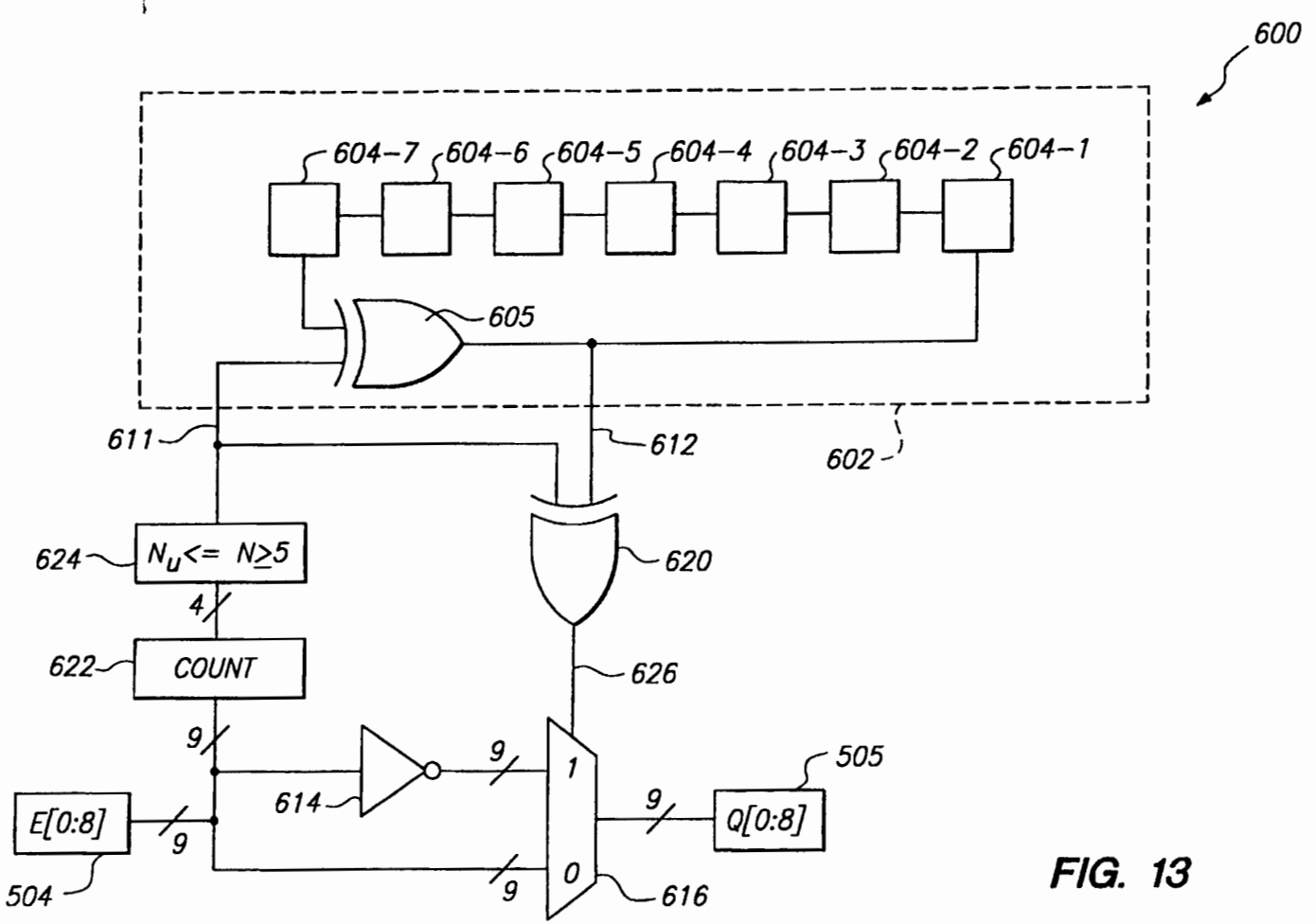


FIG. 13

SUBSTITUTE SHEET (RULE 26)

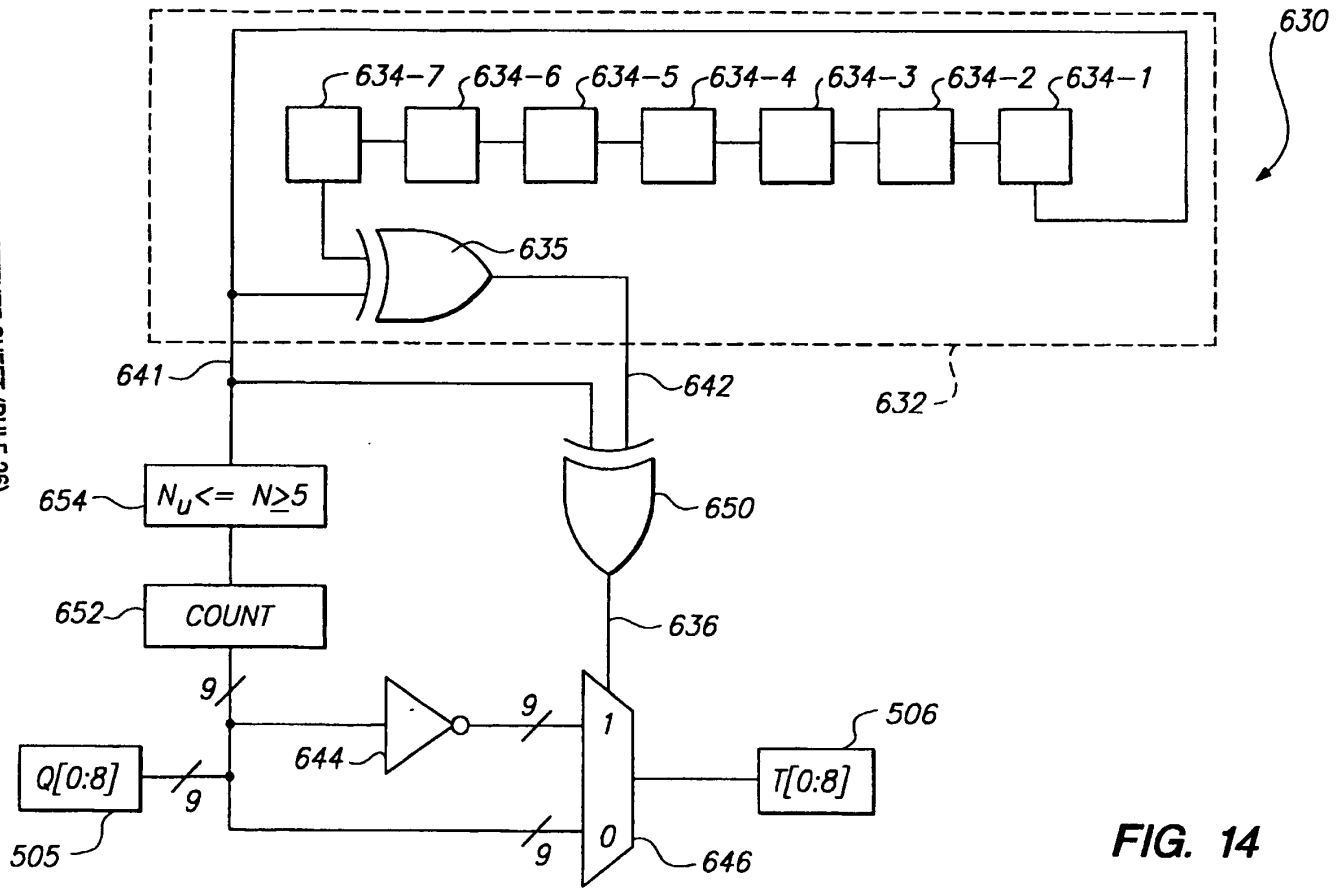


FIG. 14

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : **H04L 25/03, 25/49**	**A3**	**(11) International Publication Number:** **WO 97/36408** **(43) International Publication Date:** 2 October 1997 (02.10.97)

(21) International Application Number: PCT/US97/05220

(22) International Filing Date: 25 March 1997 (25.03.97)

(30) Priority Data:
08/622,810 27 March 1996 (27.03.96) US

(71) Applicant: SILICON IMAGE, INC. [US/US]; Suite 222, 1032 Elwell Court, Palo Alto, CA 94303 (US).

(72) Inventors: LEE, Kyeongho; 523 Walker Drive #9, Mountain View, CA 94043 (US). JEONG, Deog-Kyoon; 3351 Alma Street #307, Palo Alto, CA 94306 (US).

(74) Agents: OKAMOTO, James, K. et al.; Fenwick & West L.L.P., Suite 700, Two Palo Alto Square, Palo Alto, CA 94306 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ARIPO patent (GH, KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

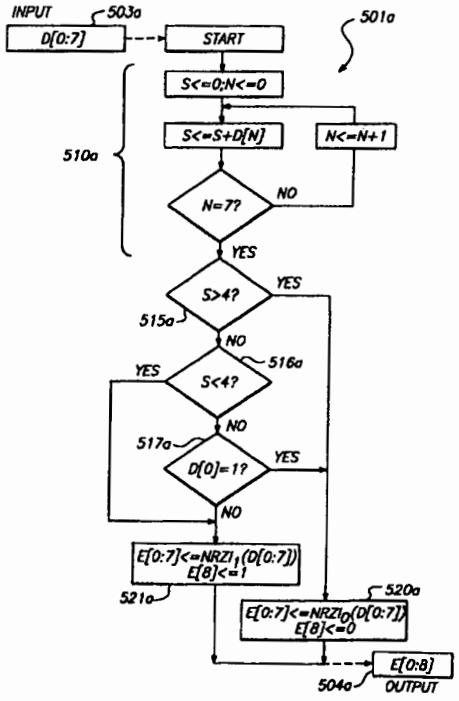
Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report:
12 February 1998 (12.02.98)

(54) Title: TRANSITION CONTROLLED BALANCED ENCODING SCHEME

(57) Abstract

A method and apparatus for producing a transition-controlled, DC-balanced sequence of characters from an input sequence of data bytes is disclosed herein. The bits in each of the data bytes are selectively complemented in accordance with the number of logical '1' signals in each data byte in order to produce selectively complemented data blocks. A cumulative disparity is then determined between the logical values of different type included within ones of the selectively complemented data blocks previously encoded into characters. In addition, a current disparity in a candidate character associated with a current one of the selectively complemented data blocks being encoded is also determined. The candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of a polarity opposite to a first polarity of the cumulative disparity. Alternately, the complement of the candidate character is assigned to the current one of the selectively complemented data blocks if the current disparity is of the first polarity. In a high-transition mode of operation, the bits within data blocks including fewer than a minimum number of logical '1' signals are selectively complemented so that each such selectively complemented data block includes in excess of the minimum number of logical transitions. In a low-transition mode of operation, the bits within data blocks having more than a predefined number of logical '1' signals are selectively complemented so that each such selectively complemented data block includes less than the maximum number of logical transitions.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

INTERNATIONAL SEARCH REPORT

Interi. ..onal Application No
PCT/US 97/05220

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04L25/03 H04L25/49

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)
IPC 6 H04L

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)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	FAIR ET AL.: "Guided scrambling: a new line coding technique for high bit rate fiber optic transmission systems" IEEE TRANSACTIONS ON COMMUNICATIONS, vol. 39, no. 2, February 1991, NEW YORK, US, pages 289-296, XP000225307 see abstract; figures 1,3,4 see page 289, left-hand column, paragraph 2 - right-hand column, paragraph 2 see page 290, left-hand column, paragraph 2 see page 290, right-hand column, paragraph 1 see page 290, right-hand column, paragraph 3 --- -/--	1-9, 29-31, 34-36

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

[X] Patent family members are listed in 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

2

Date of the actual completion of the international search	Date of mailing of the international search report
4 September 1997	12. 12. 97
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040. Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer SCRIVEN P.

Form PCT/ISA/210 (second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 193 of 487

INTERNATIONAL SEARCH REPORT

Inter .ional Application No
PCT/US 97/05220

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	WO 92 22160 A (NORTHERN TELECOM) 10 December 1992 see abstract; figures 3-5 see page 3, line 4 - line 17 see page 4, line 11 - line 21	1-9, 29-31, 34-36
A	US 5 410 600 A (TOY) 25 April 1995 see abstract see column 2, line 41 - line 60 see column 2, line 67 - line 10 see column 3, line 23 - line 36	1,4,7, 29,34

2

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 194 of 487

INTERNATIONAL SEARCH REPORT

I ational application No.
PCT/US 97/05220

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see continuation sheet

1. ☐ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1-9,29-31,34-36

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest.

☐ No protest accompanied the payment of additional search fees.

IPR2024-001428
HP – EX1002, Page 195 of 487

INTERNATIONAL SEARCH REPORT

International Application No. PCT/ US 97/05220

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

1. claims 1-9,29-31,34-36:
production of a DC-balanced sequence, using PN sequences

2. claims 10-25:
production of a DC-balanced sequence using two stages of controlled inversion

3. claim 26 :
transmission system using selective inversion

4. claims 27,28,32,33:
production of a DC-balanced sequence, without introducing redundant symbols

INTERNATIONAL SEARCH REPORT

Information on patent family members

Inter. ...onal Application No
PCT/US 97/05220

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9222160 A	10-12-92	US 5200979 A	06-04-93
		CA 2109428 A	07-12-92
		DE 69216111 D	30-01-97
		DE 69216111 T	17-04-97
		EP 0610204 A	17-08-94
		JP 6503217 T	07-04-94
US 5410600 A	25-04-95	NONE	

Form PCT/ISA/210 (patent family annex) (July 1992)

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
25 July 2002 (25.07.2002)

PCT

(10) International Publication Number
WO 02/058318 A1

(51) International Patent Classification[7]: H04L 7/033

(21) International Application Number: PCT/US01/50149

(22) International Filing Date: 26 October 2001 (26.10.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/243,855 27 October 2000 (27.10.2000) US

(71) Applicant *(for all designated States except US)*: **SILICON IMAGE** [US/US]; 1060 East Arques Avenue, Sunnyvale, CA 94085 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **LI, Hung, Sung** [CN/US]; 825 E. Evelyn Avenue, #407, Sunnyvale, CA 94086 (US). **KIM, Ook** [KR/US]; 1094 Tanland Drive, #105, Palo Alto, CA 94303 (US).

(74) Agents: **DALEY-WATSON, Christopher, J.** et al.; Perkins Coie LLP, P.O. Box 1247, Seattle, WA 98111-1247 (US).

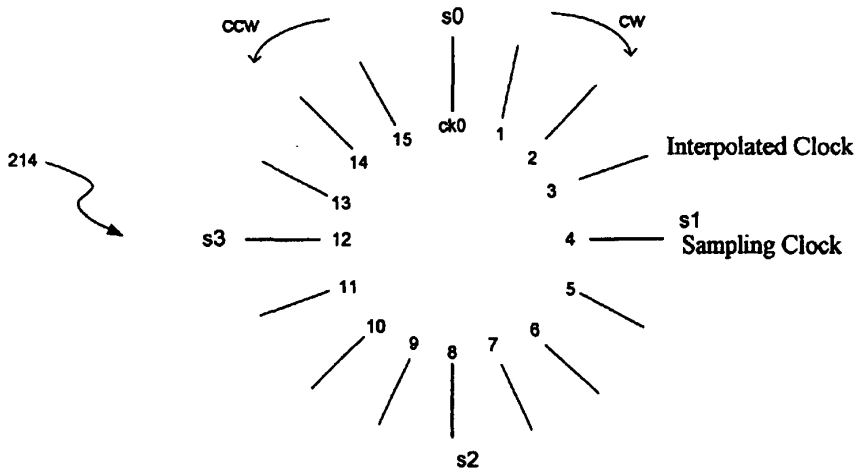
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR RECOVERING N TIMES OVERSAMPLED DATA BY SELECTION OF N PHASE SHIFTED SAMPLING CLOCKS OUT OF N*X INTERPOLATED CLOCKS ACCORDING TO THE RELATIVE VALUES OF THE SAMPLED DATA



(57) Abstract: N-times oversampled digital data is recovered by selection of N oversampling clocks (s0-s3) out of a totality of N*X interpolated clocks (ck0...ck15) depending on the relative values of the sampled data. The N oversampling clocks (s0-s3) are advanced or delayed in phase by an increment or decrement of 360°/(N*X) to achieve sampling at the center of the received data. Clock selection is performed via an N*(N*X) transistor switching matrix.

WO 02/058318 A1

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHOD AND APPARATUS FOR RECOVERING N TIMES OVERSAMPLED DATA BY SELECTION OF N PHASE SHIFTED SAMPLING CLOCKS OUT OF N*X INTERPOLATED CLOCKS ACCORDING TO THE R ELATIVE VALUES OF THE SAMPLED DATA

TECHNICAL FIELD

[0001] Embodiments of the invention are in the field of processing high-speed digitally encoded analog data.

DESCRIPTION OF THE BACKGROUND

[0002] When performing parallel sampling of a serial data stream, clock skew must be compensated for. Clock skew occurs when there is a difference in clock signal timing between the signal being sampled and the sampling circuit. Clock skew can cause the sampling circuit to sample the signal at times during which the received signal does not represent meaningful data. Bit error rate ("BER") is a measure of transmission errors, typically the number of erroneous data bits over the total number of bits transmitted. In most systems, there is a maximum BER that cannot be tolerated.

[0003] In serially transmitted binary data, one way to extract original data is to sample more bits during one bit period. This technique is called oversampling. From oversampled data, the receiver can extract the best samples with minimum error, due to nearby data and clock information. Oversampling includes sampling multiple times in a period of time during which the received signal is expected to represent meaningful data. The multiple samples collected, however, include bad samples and good samples, and the bad samples must be discarded. In some prior circuits that perform oversampling, the received signal frequency is too high for a single sampling circuit to perform the required number of sampling operations per time period. Therefore, several parallel sampling circuits are used. Each of the sampling circuits is "fired" by a clock signal with a

IPR2024-001428
HP – EX1002, Page 200 of 487

distinct point in time, so the sampling circuits are fired in sequence. For example, an oversampling ratio of 4 implies that 4 data samples are to be sampled per bit time period. If there are 16 parallel sampling circuits which sample input data based on clock data, 16 equally spaced clock phases are required. This proliferation of clocks can be expensive in hardware and power, and the majority of the samples collected are not actually used for extracting transmitted data samples.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] **Figure 1** is block diagram of an embodiment of a clock and data recovery circuit.

[0005] **Figure 2A** is a diagram illustrating sixteen clock signals and their relative phases.

[0006] **Figure 2B** illustrates the order and frequency of sampling circuits "firing" in the embodiment of Figure 1.

[0007] **Figure 3** illustrates a sampling operation in the embodiment of Figure 1.

[0008] **Figure 4** is a simplified diagram of a switching matrix in one embodiment of a sampling clock generating circuit for use in the circuit of Figure 1.

[0009] **Figure 5** is another diagram of the switching matrix of **Figure 4**.

[0010] **Figure 6** is block diagram showing part of the switching matrix of **Figure 4**.

[0011] **Figure 7** is a timing diagram that illustrates the synchronization of a clock switching operation of the embodiment of Figure 6.

DETAILED DESCRIPTION

[0012] The invention will now be described with respect to various embodiments. The following description provides specific details for a

IPR2024-001428
HP – EX1002, Page 201 of 487

thorough understanding of, and enabling description for, these embodiments of the invention. However, one skilled in the art will understand that the invention may be practiced without these details. In other instances, well known structures and functions have not been shown or described in detail to avoid unnecessarily obscuring the description of the embodiments of the invention. For each embodiment, the same reference numbers and acronyms identify elements or acts with the same or similar functionality for ease of understanding and convenience.

[0013] A clock and data recovery method and apparatus described herein is applicable to various types of data transmission. Embodiments include a method and apparatus for receiving a multi-channel serial digitally encoded signal and converting the received signal to digital data, or a set of binary characters. One embodiment includes determining whether a phase of a sampling circuit is appropriate to sample meaningful data from a received signal. If it is determined that the phase of the sampling circuit is not appropriate, the phase is shifted so that sampling occurs sooner or later in relationship to the received signal. The determination is based, in one embodiment, on the order and value of the samples taken, which indicate whether the samples are taken too close to a transition of the received signal. Determining an appropriate sampling phase allows fewer samples to be taken and therefore fewer sampling circuits to be used without corresponding degradation in sampling resolution. This can be illustrated as follows.

[0014] Let B = bit rate in samples/second, and

[0015] OS = oversampling ratio, that is, the received signal is sampled at OS times the frequency of the transmitted data rate.

[0016] If N = the number of parallel sampling circuits. The sampling frequency per sampling circuit is (B x OS) / N. If N sampling clock phases are further divided by X phases, then the total number of phases becomes

IPR2024-001428
HP – EX1002, Page 202 of 487

$N \cdot X$. If each of N sampling circuits may receive one of N clock phases (that is, each sampling circuit is supplied with a phase clock among N clock phases) then the sampling resolution is $1/B \cdot OS \cdot X$, and the effective sampling frequency per sampling circuit becomes $B \cdot OS \cdot X$. According to embodiments of the invention, the effective sampling frequency is further increased without increasing the number of sampling circuits. This is because each N phase clock has a resolution of $1/B \cdot OS \cdot X$. In one embodiment, N is four, X is four, so that each phase of a four phase clock has a resolution of four phases. The sampling points are intelligently chosen, which allows four of the sixteen available phases to be used instead of using all sixteen phases.

[0017] One embodiment uses four generated clock signals to produce sixteen sampling clock phases from which to choose. The sixteen sampling clock phases are evenly distributed about 360 degrees. The four generated clock signals actually chosen to drive the four sampling circuits have phase increments of 90 degrees. There are sixteen possible sets of four clock signals that can be chosen. The sampled signals are evaluated to intelligently determine whether the sampling circuits are sufficiently in phase with the received signal. The evaluation also indicates whether to adjust the sampling circuit phases to occur sooner or later. The effective sampling resolution is thus greater than the actual sampling resolution. For example, in the case just described, the actual sampling resolution is four, but the quality of the converted signal, as reflected for example in the BER of the converted signal, is comparable to the quality achieved with sampling resolution of sixteen.

[0018] In the Figure 1 block diagram, an embodiment of a clock and data recovery circuit 104 provides sampling of a digitally encoded signal with a sampling resolution greater than the oversampling ratio. The circuit 104 includes four sampling circuits 100, 101, 102, and 103. The sampling circuits 100-103 are any one of several known circuits for collecting

IPR2024-001428
HP – EX1002, Page 203 of 487

samples of a signal over time or periodically for the purpose of reconstructing or processing the signal. In one embodiment, the sampling circuits 100-103 are arranged in parallel to sample a digitally encoded analog signal and accurately produce digital data from the signal. The sampling circuits are each clocked separately by a respective sampling clock signal s0, s1, s2, and s3, with each having a different phase. The sampling clock signals s0-s3 each activate a respective sampling circuit 100-103 to collect a sample of the received signal. The value of the signals and their order of occurrence determine the encoding of the received signal. In one embodiment, the received signal has different voltage values that determine a data value, such as 1, 0. When the received signal makes a transition from one voltage to another, the signal crosses a transition point at which samples are contaminated by noise and nearby data values. It is therefore important to synchronize the sampling clock signals to the received data so that samples of meaningful data are taken because they represent no meaningful data encoded in the signal. The synchronization of the sampling clock signals is discussed in more detail below.

[0019] A clock generating circuit 108 generates four clock signals 110 that each have different phases. In one embodiment, the four clock signals 110 are 90 degrees out of phase with their closest neighbors so that the four clock phases are equally distributed about 360 degrees. In one embodiment, the clock generating circuit 108 is an analog phase locked loop (PLL), although other embodiments may employ other clock generating circuits such as a delay lock loop (DLL), multiple clocks (e.g., oscillators) or a single clock with delay elements. A clock divider circuit 112 receives the four clock signals 110 and creates sixteen clock signals 114 from the four clock signals. The clock divider circuit 112 is a known circuit, such as in one embodiment, a multi-phase interpolator circuit. In another embodiment, the clock divider circuit 112 is a resistor network. In

IPR2024-001428
HP – EX1002, Page 204 of 487

yet another embodiment, the clock divider circuit 112 is a circuit including voltage controlled oscillator (VCO) delay elements.

[0020] The sixteen clock signals 114 each have the same frequency and are synchronized, but each have a unique phase such that the sixteen phases are equally distributed about 360 degrees. A sampling clock generating circuit 116 receives the sixteen clock signals 114 and generates in response the four sampling clock signals s0, s1, s2, and s3 as will be more fully described.

[0021] A switching signal generation circuit 122 generates a control signal 120 and provides it to the sampling clock signal generating circuit 116. The control signal 120 determines which of the sixteen clock signals 114 the sampling clock generating circuit 116 generates as the four sampling clock signals s0-s3. The control signal 120 are composed of sixteen signals st0-st15, and only one is active at one time. The control signal generating circuit 122 activates a particular control signal 120 after determining whether the sampling circuits 100-103 are sufficiently in phase with the received signal so as to sample effectively and allow meaningful data to be recovered from the received signal. The switching signal generating circuit 122 receives and evaluates sampled data signals 104, 105, 106, and 107 to determine whether sampling phases of the sampling circuits 100-103 as dictated by the sampling clock signals s0-s3 must be shifted. (The signals (e.g., s0-s3, 104-107, etc.) are shown as lines between blocks in Figure 1.) The control signal generating circuit 122 further receives the sampling clock signals 126 to control signal generating logic. In one embodiment, the control signal generating circuit 122 includes a digital clock and data recovery (CDR). The control signal generating circuit 122 transmits a latching clock signal 124 to synchronize logic in the sampling clock generating circuit 116. The logic in the sampling clock generating circuit 116 must be synchronized, as discussed below, so that when the sampling clock signals s0-s3 are shifted, no clock

IPR2024-001428
HP – EX1002, Page 205 of 487

skew effects occur in the sampling clock signals s0-s3 due to the fact that shifting the phases of the sampling clock signals s0-s3 includes switching clock sources.

[0022] **Figure 2A** is a diagram illustrating the sixteen clock signals 114 labeled ck0, 1,2, etc., and their relative phases. The sixteen clock signals 114 are separated by uniform phase increments about 360 degrees (i.e., 22.5 degrees apart). The four sampling clock signals s0-s3, in this example, are ck0, ck4, ck8, and ck12. The four sampling clock signals s0-s3 are separated by 90 degree increments. **Figure 2B** shows the order and frequency of the sampling circuits s0-s3 "firing". The first sampling clock signal, s0, is ck0 in this example. The second sampling clock signal, s1, is ck4, and so on. When the switching signal generating circuit 122 determines that the phases of the sampling clock signals s0-s3 are offset from the optimum center of the received signal, a control signal 120 is generated to shift the phases of the sampling clock signals s0-s3 forward one phase increment. This is illustrated in **Figure 2B**, which shows that after s3 (which is ck12) occurs, the next s0 is ck1. A phase shift of one increment "late" or clockwise has occurred. After the shift, s1 is ck5, s2 is ck9, and s3 is ck13. The occurrence of ck1 (the "new" s0) is delayed one phase increment, as shown, to provide consistent sampling clock signals s0-s3 to the sampling clock circuits 100-103. When the switching signal generating circuit 122 determines that the phases of the sampling clock signals s0-s3 are too far ahead of the received signal, it generates a control signal 120 to shift the phases of the sampling clock signals s0-s3 back one phase increment in a manner corresponding to the illustrated example of a shift forward. For example, s0 becomes ck15, and so on.

[0023] The switching signal generating circuit 122, in one embodiment, executes an algorithm to determine whether the sampling circuits 100-103 are sufficiently in phase with the received signal. **Figure 3** illustrates

IPR2024-001428
HP – EX1002, Page 206 of 487

sampling in one embodiment. A waveform of the received data signal is shown along with logic values represented by different signal levels. A highest positive signal level represents a logic 1, and a lowest negative signal level represents a logic level 0. Times A, B, C, D, and E are times at which samples are taken, or sample points. "Midbit" samples a, c, and e are taken at times A, C, and E. "Transition" samples b and d are taken at times B and D, which are close to signal transitions of the zero-crossing. The switching signal generating circuit 122 evaluates samples 104-107 to determine at which times the samples were taken. In one embodiment, the switching signal generating circuit 122 determines whether the phases of the sampling circuits 100-103 are early or late as follows.

[0024] 1) If a equals b and b does not equal c, sampling is late

[0025] 2) If a does not equal b and b equals c, sampling is early

[0026] 3) If a equals b equals c, no decision is possible

[0027] 4) If a equals c and c does not equal b, no decision is possible

[0028] In case 1, the switching signal generating circuit 122 generates a control signal 120 to change the sampling phases to an earlier phase. In case 2, the switching signal generating circuit 122 generates a control signal 120 to change the sampling phases to a later phase. In cases 3 and 4, no new control signal 120 is generated and the current control signal 120 remains active. When no decision is possible, the algorithm assumes that the sampling phases are effectively correct and that no changes are necessary. Those skilled in the relevant art will recognize that various circuits or systems may be employed in the switching signal generating circuit 122 to implement the algorithm (e.g., field programmable logic, an array of logic gates corresponding to logic rules, a programmed microcontroller, etc.). In other embodiments, other known algorithms or methods are used to determine whether the sampling circuits 100-103 are sufficiently synchronized.

IPR2024-001428
HP – EX1002, Page 207 of 487

[0029] **Figure 4** is a simplified diagram of a switching matrix in one embodiment of the sampling clock generating circuit 116. In various embodiments, the sampling clock generation circuit can be a barrel shifter or any other logic circuit whose outputs can be shifted. As described herein, the switching signal generating circuit 122 effectively processes the four samples it receives from the four sampling clock circuits 100-103 to decide whether the phase of the clock should be moved forward or backward (e.g., clockwise or counter clockwise through control signals st0-st15 if a barrel shifter is employed as the sampling clock generating circuit 116). The clock divider circuit 112 includes sixteen outputs connected to the sampling clock signal generator circuit 116 for carrying sixteen clock signals ck0-ck15 as shown. The clock signals ck0-ck15 each have a distinct phase as illustrated in **Figure 2A**. The ck0-ck15 outputs of the clock divider circuit 112 are each electrically connected to each of four sampling clock outputs that carry the sampling clock signals s0-s3 so as to form the matrix shown. The sampling clock generating circuit 116 receives an active control signal that determines which of the sixteen clock signals ck0-ck15 become the four sampling clock signals s0-s3. Two cases are shown in **Figure 4**. In one case, the st0 signal is active. In this case, the switching matrix connects the ck0 output to the s0 output, the ck4 output to the s1 output, the ck8 output to the s2 output, and the ck12 output to the s3 output. In another case, the phases of the sampling clock signals s0-s3 are shifted forward one phase increment. Thus, the switching matrix connects the ck1 output to the s0 output, the ck5 output to the s1 output, the ck9 output to the s2 output, and the ck13 output to the s3 output. The other control signal lines, such as for control signals st2-st15, are not shown for clarity.

[0030] **Figure 5** is a diagram of the switching matrix of **Figure 4** in an embodiment that shows transistor switches at intersection points. The control signal lines for st0 and st1 are shown connected to the gates of

IPR2024-001428
HP – EX1002, Page 208 of 487

transistor switches. For example, the control signal line for the control signal st0 is shown connected to the gates of transistor 50 and 51. When st0 is active, the transistors 50 and 51 are "on" and electrically connect the ck0 line with the s0 line, and the ck4 line with the s1 line. (While not shown in Figure 5, st0 likewise activates two other transistors to electrically connect the ck8 and ck12 lines with the s2 and s3 lines, respectively.) Similarly, the st1 line is connected to the gates of transistors 52 and 53 (and other transistors not shown), so that when st1 is active, the ck1 line and the ck5 line are connected to the s0 and s1 lines, respectively. The other control signal lines st2-st15 and clock signal lines ck6-ck15 are not shown for clarity.

[0031] **Figure 6** is block diagram showing part of the switching matrix of Figure 4 at yet a further level of detail. The ck0-ck15 outputs of the clock divider circuit 112 are in effect candidate sources for the sampling clock signals s0-s3. As previously described, the phases of the sampling clock signals s0-s3 are shifted when it is determined that the sampling circuits 100-103 are not sufficiently synchronized with the received signal. This shifting involves switching sampling clock sources from one set of four of the ck0-ck15 outputs to another set of four. **Figure 6** illustrates latching circuitry that allows the sampling clock source switching to occur when the clocks are stable so that the s0-s3 outputs remain consistent. Flip-Flops (FFs) are used to latch the active control signal 120. The diagram is simplified so that only part of the st0 control signal line is shown, including the FFs 60, 61, and 62; intervening FFs are not show for simplicity. The FFs 60, 61 and 62 latch the st0 signal for the ck0 line, the ck1 line, and the ck15 line. Similarly, a corresponding portion of the st15 switching line is shown with FFs 63, 64, and 65. The st0 switching line will be discussed. Other portions of the switching matrix that correspond to the portion discussed operate similarly.

IPR2024-001428
HP – EX1002, Page 209 of 487

[0032] The outputs of the FFs 60-62 are each electrically connected to a transistor gate as shown. The FFs and transistors are clocked by a common clock signal 124, designated Sn, which is one of the sampling clock signals s0-s3. One of the s0-s3 signals is chosen to appropriately synchronize the switching operation according to its phase. For example, an appropriate phase for s1 is chosen to be Sn after simulation reveals that s2 is too late and s0 is too early, as described further below. The signals sw0-sw5 are latched, or synchronized switching signals that have been clocked through a FF by the Sn common FF clock signal. When st0 is active, and the Sn signal latches the st0 signal through the FF 60, the sw0 signal turns on the transistor connected to the FF 60 and allows the ck0 signal to generate or source the sampling clock signal s0.

[0033] **Figure 7** is a timing diagram that illustrates the synchronization of the clock switching operation further. **Figure 7** can be referred to along with **Figure 6**. **Figure 7** illustrates a switch of clock source for the sampling clock signal s0 from ck0 to ck1. Initially, the source of the sampling clock signal s0 is ck0, but after the switching operation the source of the sampling clock signal s0 is ck1. In this example, the sampling clock signal s0 also serves as the common FF clock signal 124, or signal Sn. The control signals st0 and st1 are shown below the s0 signal. Latched switching signals sw0 and sw1 are shown below the control signals st0 and st1.

[0034] At transition time 702, the current active switching signal changes from st0 to st1. At rising edge 704 of the sampling clock signal s0, the st1 control signal is latched by the FF 60, causing the active latched sw signal to change from sw0 to sw1 at the transition time 710. This transition, as shown at 708, causes the clock source of the sampling clock signal s0 to change from ck0 to ck1. The change in clock source happens during a stable period of the s0 signal (in this case, a high signal period) so that no sampling clock disruption occurs. The falling edge 714

IPR2024-001428
HP – EX1002, Page 210 of 487

of s0 is extended as shown at 712, after the st0, st1 transition Thus, switching of clocks may be completed at the end of a clock cycle. Switching the clock signal lines within one clock cycle prevents disturbing operation of the sampling circuits 100-103. Before and after the switching operation just described, the other control signals st2-st15 are inactive.

[0035] In some instances, different sampling clock signals s0-s3 may be used to clock the FFs. It may happen, for example due to physical circuit particularities, that the s0 signal is too late or too early to serve as the common FF clock signal 124. This can be discovered during simulation or testing of the circuit. In that case, another of the sampling clock signals s0-s3 can be chosen to be the common FF clock signal 124.

[0036] Embodiments of the invention have been described with reference to particular examples. Many alternatives are possible under the invention described herein. For example, the number of sampling circuits could be more or less than four. Similarly, the number of clock phases provided by the clock divider could be more or less than sixteen. Other variations within the scope of the claimed invention include different circuits or methods for synchronizing the switching operation from one set of sampling clock signals to another.

[0037] Aspects of the invention may be implemented as functionality programmed into any of a variety of circuitry, including programmable logic devices (PLDs), such as field programmable gate arrays (FPGAs), programmable array logic (PAL) devices, electrically programmable logic and memory devices and standard cell-based devices, as well as application specific integrated circuits (ASICs). Some other possibilities for implementing aspects of the invention include: microcontrollers with memory (such as electronically erasable programmable read only memory (EEPROM)), embedded microprocessors, firmware, software, etc. If aspects of the invention are embodied as software at at least one stage during manufacturing (e.g. before being embedded in firmware or in a

IPR2024-001428
HP – EX1002, Page 211 of 487

PLD), the software may be carried by any computer readable medium, such as magnetically- or optically-readable disks (fixed or floppy), modulated on a carrier signal or otherwise transmitted, etc. Furthermore, aspects of the invention may be embodied in microprocessors having software-based circuit emulation, discrete logic (sequential and combinatorial), custom devices, fuzzy (neural) logic, quantum devices, and hybrids of any of the above device types. The underlying device technologies may be provided in a variety of component types, e.g., metal-oxide semiconductor field-effect transistor (MOSFET) technologies like complementary metal-oxide semiconductor (CMOS), bipolar technologies like emitter-coupled logic (ECL), and polymer technologies.

[0038] Unless the context clearly requires otherwise, throughout the description and the claims, the words "comprise," "comprising," and the like are to be construed in an inclusive sense as opposed to an exclusive or exhaustive sense; that is to say, in a sense of "including, but not limited to." Words using the singular or plural number also include the plural or singular number respectively. Additionally, the words "herein," "above," "below," and words of similar import, when used in this application, shall refer to this application as a whole and not to any particular portions of this application. When the claims use the word "or" in reference to a list of two or more items, that word covers all of the following interpretations of the word: any of the items in the list, all of the items in the list and any combination of the items in the list.

[0039] The above detailed descriptions of embodiments of the invention are not intended to be exhaustive or to limit the invention to the precise form disclosed above. While specific embodiments of, and examples for, the invention are described above for illustrative purposes, various equivalent modifications are possible within the scope of the invention, as those skilled in the relevant art will recognize. For example, while steps are presented in a given order, alternative embodiments may perform

IPR2024-001428
HP – EX1002, Page 212 of 487

routines having steps in a different order. The teachings of the invention provided herein can be applied to other systems, not necessarily the system described herein. These and other changes can be made to the invention in light of the detailed description. The elements and acts of the various embodiments described above can be combined to provide further embodiments.

[0040] These and other changes can be made to the invention in light of the above detailed description. In general, the terms used in the following claims, should not be construed to limit the invention to the specific embodiments disclosed in the specification, unless the above detailed description explicitly defines such terms. Accordingly, the actual scope of the invention encompasses the disclosed embodiments and all equivalent ways of practicing or implementing the invention under the claims.

[0041] While certain aspects of the invention are presented below in certain claim forms, the inventors contemplate the various aspects of the invention in any number of claim forms. For example, while only one aspect of the invention is recited as embodied in a semiconductor chip, other aspects may likewise be embodied in a chip. Accordingly, the inventors reserve the right to add additional claims after filing the application to pursue such additional claim forms for other aspects of the invention.

IPR2024-001428
HP – EX1002, Page 213 of 487

CLAIMS

We claim:

[c1] 1. A clock and data recovery system, comprising:

a first number of parallel sampling circuits for collecting a first number of samples of a received digitally encoded signal;

a switching signal generating circuit configured to receive the first number of samples and to determine whether the first number of parallel sampling circuits are sufficiently in phase with the received signal so as to collect a meaningful sample; and

a sampling clock signal generating circuit configured to receive a control signal from the switching signal generating circuit, wherein the control signal causes the clock signal generating circuit to generate a first number of clock phases to clock the first number of parallel sampling circuits, wherein the first number of clock phases are chosen from a second number of clock phases, the second number being an integer multiple of the first number.

IPR2024-001428
HP – EX1002, Page 214 of 487

[c2] 2 The system of claim 1, wherein the first number is four, and wherein the switching signal generating circuit evaluates the four samples to determine an order and a logic value for each of the four samples, and based upon the order and the logic value of each of the four sample, determines whether the four clock phases must be shifted ahead or behind.

[c3] 3. The system of claim 1, wherein the first number is four and the second number is sixteen, wherein the four clock phases are equally spaced about 360 degrees, and wherein the control signal causes the four clock phases to shift clockwise or counterclockwise one phase increment.

[c4] 4. The system of claim 1, wherein:

the switching signal generating circuit to determine an order and a logic value for each of the first number of samples, and based upon the order and the logic value of each of the first number of samples, determines whether the first number of clock phases must be shifted ahead or behind, and

the sampling clock generating circuit comprises a switch matrix including a second number of clock signal lines that are each coupled so as to clock each of the first number of parallel sampling circuits, and wherein a first number of the clock signal lines is coupled to the first number of parallel sampling circuits at one time through the control signal.

[c5] 5. A method for converting a received digital-encoded serial signal into digital data comprising:

generating N sampling clock signals with N distinct phases equally distributed about 360 degrees;

IPR2024-001428
HP – EX1002, Page 215 of 487

dividing each of the N sampling clock signals by X to produce N*X sampling clock signals with N*X distinct phases equally distributed about 360 degrees;

sampling the received signal, including collecting N samples using N sampling circuits, wherein each of the N sampling circuits is clocked by one of N sampling clock signals chosen from among the N*X sampling clock signals, and wherein the N sampling clock signals chosen have distinct phases equally distributed about 360 degrees;

determining whether the N sampling clock signals chosen are synchronized with the received signal so as to sample meaningful data; and

if it is determined that the N sampling clock signals chosen are not synchronized, choosing N new sampling clock signals from among the N*X sampling clock signals for clocking the N sampling circuits.

[c6] 6. The method of claim 5, wherein determining whether the N sampling clock signals chosen are synchronized includes evaluating the N samples collected to determine a relationship between the distinct phases of the N sampling clocks and a signal transition point of the received signal, and wherein when the distinct phases of the N sampling clocks are determined to be too close to the signal transition point, the N new sampling clocks are chosen so that the distinct phases of the N new sampling clocks are farther from the signal transition point.

[c7] 7. The method of claim 5, wherein when the N equals four and wherein X equals four.

IPR2024-001428
HP – EX1002, Page 216 of 487

[c8] 8. The method of claim 5, wherein the N sampling clock signals are not synchronized when the distinct phases of the N sampling clock signals are determined to be too close to a signal transition point of the received signal, and wherein the distinct phases of the N new sampling clock signals are chosen from a group comprising N phases each shifted one phase increment ahead relative to the phases determined to be too close and N phases each shifted one phase increment behind relative to the phases determined to be too close.

[c9] 9. The method of claim 5, wherein determining whether the N sampling clock signals chosen are synchronized includes determining whether the distinct phases of the N sampling clock signals are too close to a signal transition point of the received signal, and wherein choosing N new sampling clock signals includes performing a switching operation that switches from N clock signal sources to N new clock signal sources in a controlled manner such that the N sampling circuits receive uninterrupted sampling clock signals.

[c10] 10. The method of claim 9, wherein determining includes determining whether the disctinct phases of the N sampling clock signals are too far ahead of the received signal or too far behind the received signal, and in response, generating one of a set of control signals to perform the switching operation, wherein the set of control signals includes one control signal for each possible group of N sampling click signals among the N*X sampling clock signals.

[c11] 11. The method of claim 10, wherein:

when the distinct phases are too far ahead of the received signal the control signal generated is for a group of N sampling clock signals each at least one phase increment behind the

IPR2024-001428
HP – EX1002, Page 217 of 487

N sampling clock signals determined to be too far ahead; and

when the distinct phases are too far behind the received signal the control signal generated is for a group of N sampling clock signals each at least one phase increment ahead of the N sampling clock signals determined to be too far behind.

[c12] 12. A machine-readable medium carrying instructions for performing a method for converting a received digital-encoded serial signal into digital data, the method comprising:

establishing a sampling frequency;

periodically taking two or more samples of the received digitally-encoded serial signal based on the established sampling frequency to oversample the received digitally-encoded serial signal;

determining if the periodically taken samples correspond to meaningful data;

if at least one of the samples fails to correspond to meaningful data, establishing a new sampling frequency, wherein the new sampling frequency has an altered phase with respect to the established sampling frequency; and

again periodically taking two or more samples of the received digitally-encoded serial signal based on the new sampling frequency to oversample the received digitally-encoded serial signal.

IPR2024-001428
HP – EX1002, Page 218 of 487

[c13] 13. The machine-readable medium of claim 12 wherein the machine-readable medium is a computer-readable disk.

[c14] 14. The machine-readable medium of claim 12 wherein the machine-readable medium is a is a data transmission medium transmitting a generated data signal containing the instructions.

[c15] 15. The machine-readable medium of claim 12 wherein the machine-readable medium is a logical node in a computer network receiving the instructions.

[c16] 16. The machine-readable medium of claim 12 wherein the machine-readable medium is a semiconductor memory.

[c17] 17. The machine-readable medium of claim 12 wherein the machine-readable medium is a programmable logic device.

[c18] 18. A semiconductor chip, comprising:

a first number of parallel sampling circuits for collecting a first number of samples of a received digitally encoded signal;

a switching signal generating circuit configured to receive the first number of samples and to determine whether the first number of parallel sampling circuits are sufficiently in phase with the received signal so as to collect a meaningful sample; and

a sampling clock signal generating circuit configured to receive a control signal from the switching signal generating circuit, wherein the control signal causes the clock signal generating circuit to generate a first number of clock phases to clock the first number of parallel sampling circuits,

IPR2024-001428
HP – EX1002, Page 219 of 487

wherein the first number of clock phases are chosen from a second number of clock phases, the second number being an integer multiple of the first number.

[c19] 19. A system for converting a received digital-encoded serial signal into digital data comprising:

means for generating N sampling clock signals with N distinct phases;

means for dividing each of the N sampling clock signals by to produce N*X sampling clock signals with N*X distinct phases;

means for sampling the received signal, including collecting N samples using N sampling circuits, wherein each of the N sampling circuits is clocked by one of N sampling clock signals chosen from among the N*X sampling clock signals;

means for determining whether the N sampling clock signals chosen are synchronized with the received signal so as to sample meaningful data; and

means for choosing N new sampling clock signals from among the N*X sampling clock signals for clocking the N sampling circuits if the means for determining determines that the N sampling clock signals chosen are not synchronized.

[c20] 20. An apparatus for recovering a data signal from a received signal, comprising:

N sampling circuits which each samples the received signal periodically with a unique phase;

a sampling clock signal generating circuit coupled to the N sampling circuits, wherein the sampling clock generating

IPR2024-001428
HP – EX1002, Page 220 of 487

circuit generates a sampling clock signal with the unique phase for each of the N sampling circuits;

a clock divider circuit coupled to the sampling clock generating circuit, wherein the clock divider circuit supplies N*X clock signals to the sampling clock generating circuit, the N*X clock signals having N*X unique phases separated by substantially equal phase increments totaling 360 degrees; and

a switching signal generation circuit coupled to the N sampling circuits and to the sampling clock generating circuit to,

determine whether the N sampling circuits are sufficiently in phase with the received signal; and

generate a control signal to shift phases of the N sampling circuits to cause the N sampling circuits to be sufficiently in phase with the received signal, such that an effective sampling resolution is N*X and an actual sampling resolution is N.

IPR2024-001428
HP – EX1002, Page 221 of 487

[c21] 21. The apparatus of claim 20, further comprising an analog phase locked loop coupled to the clock divider circuit to supply the clock divider circuit with N clock signals.

[c22] 22. The apparatus of claim 20, wherein the clock divider circuit comprises a multi-phase interpolator circuit.

[c23] 23. The apparatus of claim 20, wherein the clock divider circuit comprises multiple delay elements that comprise voltage controlled oscillator cells.

[c24] 24. The apparatus of claim 20, wherein the clock divider circuit comprises a resistive network.

[c25] 25. The apparatus of claim 20, wherein the clock divider circuit comprises inverters and switches.

[c26] 26. The apparatus of claim 20, wherein the switching signal generation circuit comprises logic that receives samples of the received signal from each of the N sampling circuits, and evaluates the samples as a sequence of ones and zeros to establish one state chosen from a group comprising, the N sampling circuits are sampling too late, the N sampling circuits are sampling too early, and no decision is possible.

[c27] 27. The apparatus of claim 20, wherein the one state established determines the control signal generated, and wherein there are N*X possible control signals that each cause the sampling clock generating circuit to generate a unique set of N sampling clock signals.

[c28] 28. The apparatus of claim 20, wherein:

IPR2024-001428
HP – EX1002, Page 222 of 487

N equals four and X equals four;

the switching signal generating circuit comprises logic that receives samples of the received signal from each of the four sampling circuits, and evaluates the samples as a sequence of ones and zeros to establish one state chosen from a group comprising, the four sampling circuits are sampling too late, the four sampling circuits are sampling too early, and no decision is possible; and

the control signal is one of sixteen possible control signals that cause the sampling clock generating circuit to change phases of four sampling clock signals it generates for the four sampling circuits.

[c29] 29. The apparatus of claim 28, wherein the sampling clock circuit comprises a switching matrix that determines which four of sixteen possible clock signals drive the four sampling circuits at one time; and wherein the sampling clock circuit further uses one of the four clock signals driving the four sampling circuits to synchronize a switching operation initiated by the control signal, such that the four sampling circuits receive uninterrupted sampling clock signals during the switching operation.

[c30] 30. The apparatus of claim 28, wherein the switching signal generating circuit receives the four sampling clock signals generated by the sampling clock generating circuit for clocking the logic.

IPR2024-001428
HP – EX1002, Page 223 of 487

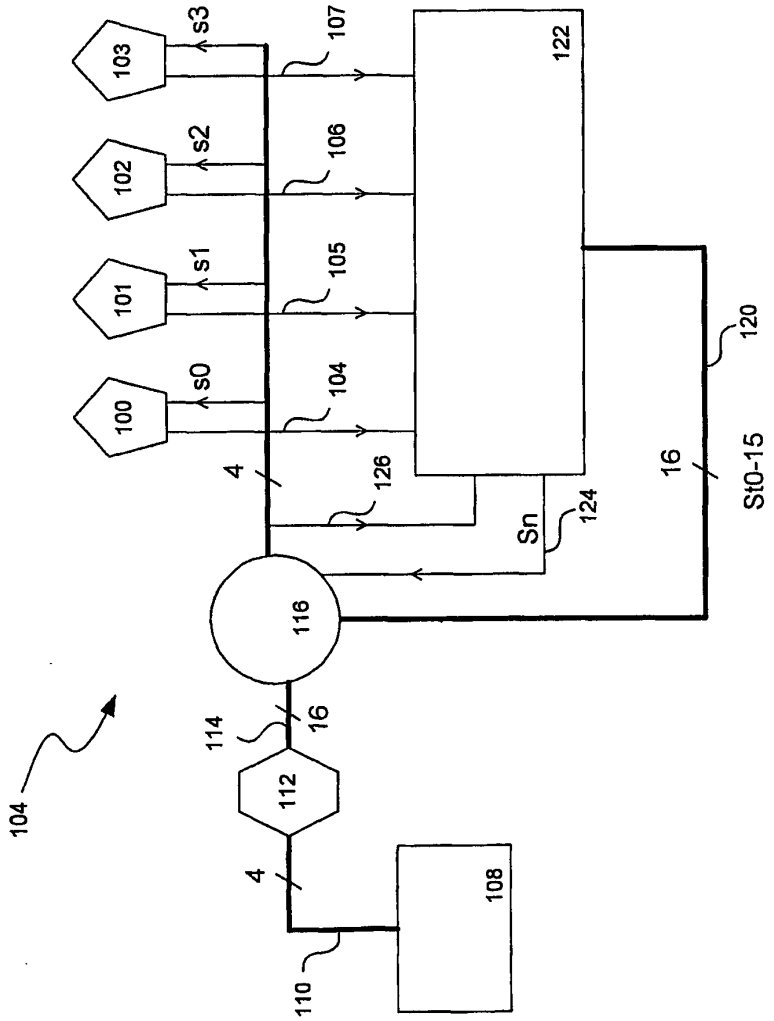


Fig. 1

IPR2024-001428
HP – EX1002, Page 224 of 487

Fig. 2A

Fig. 2B

IPR2024-001428
HP – EX1002, Page 225 of 487

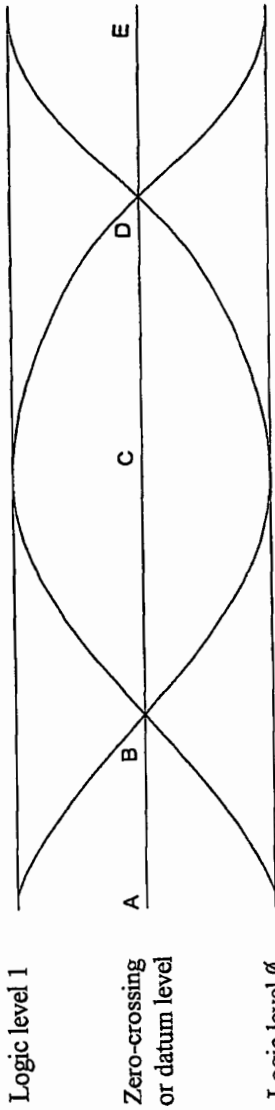


Fig. 3

IPR2024-001428
HP – EX1002, Page 226 of 487

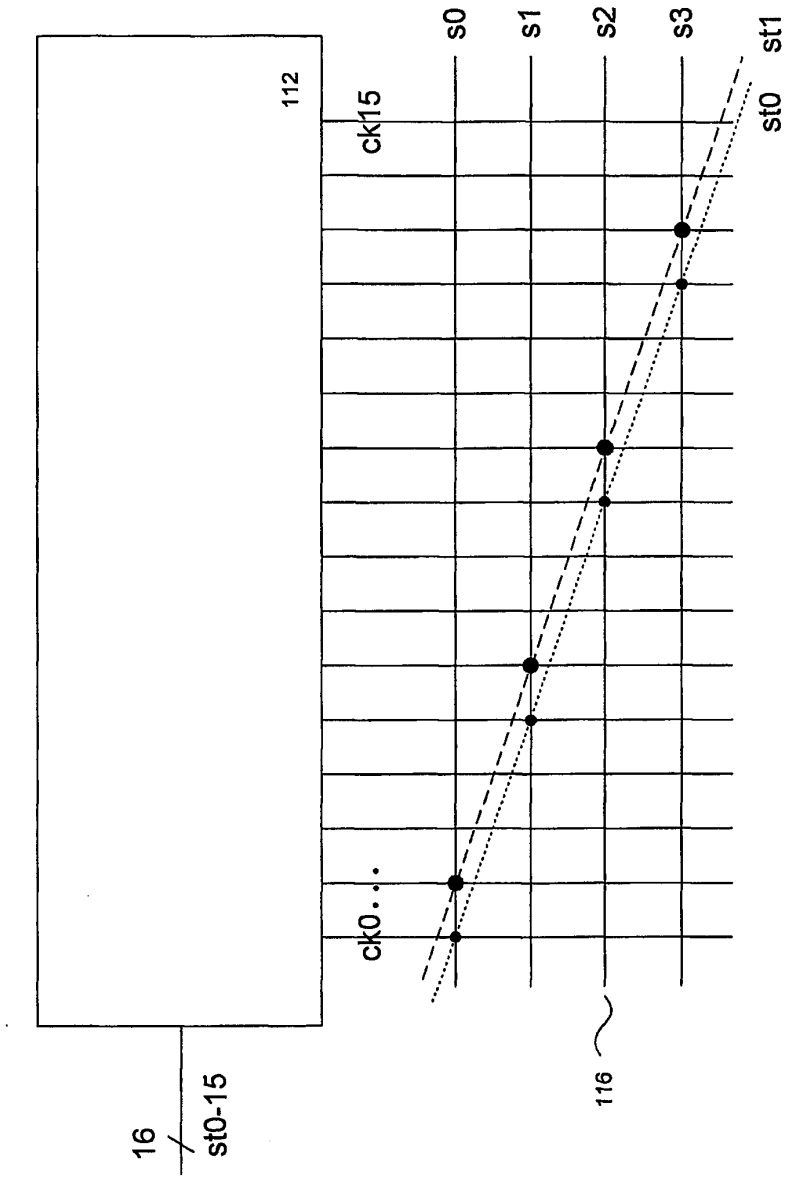


Fig. 4

IPR2024-001428
HP – EX1002, Page 227 of 487

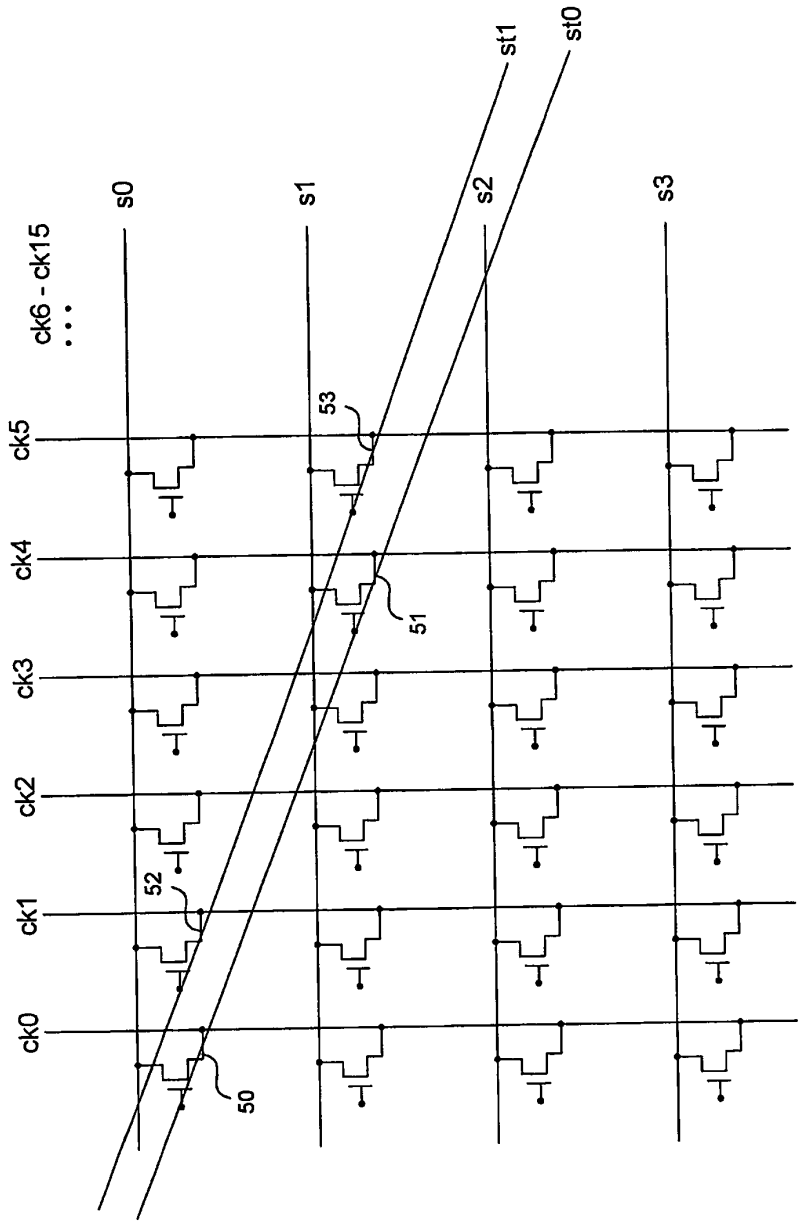


Fig. 5

IPR2024-001428
HP – EX1002, Page 228 of 487

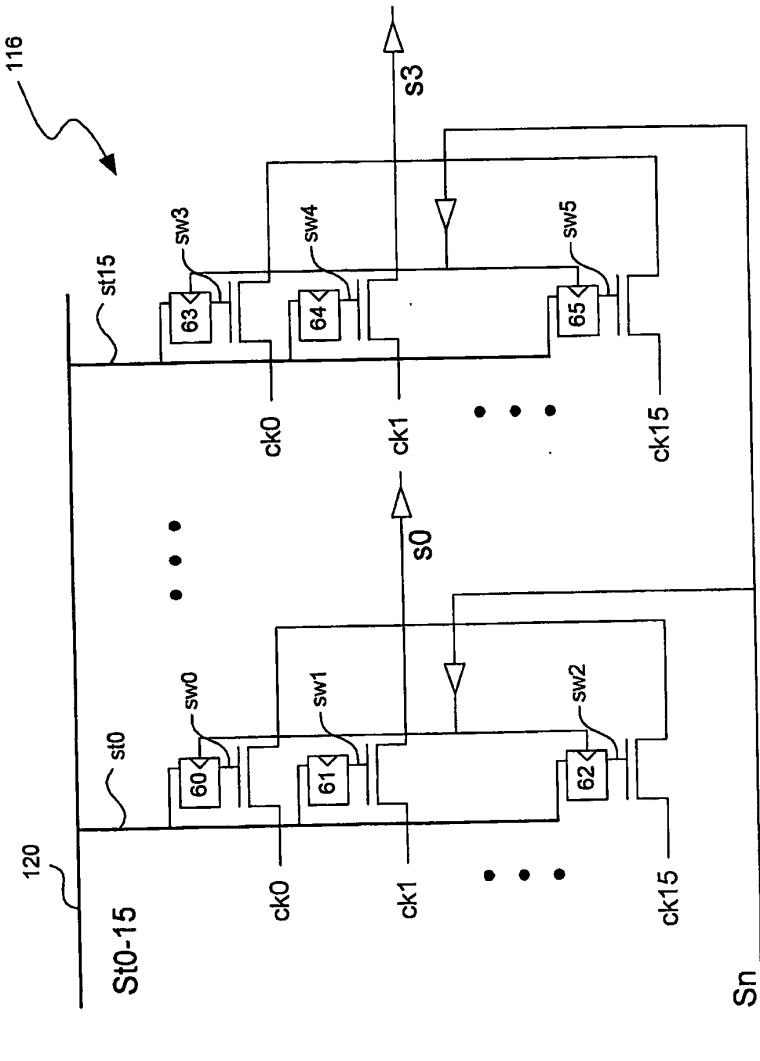

116
s3
sw3
sw4
sw5
st15
63
64
65
ck0
ck1
ck15
s0
sw0
sw1
sw2
st0
60
61
62
120
St0-15
124
Sn

Fig. 6

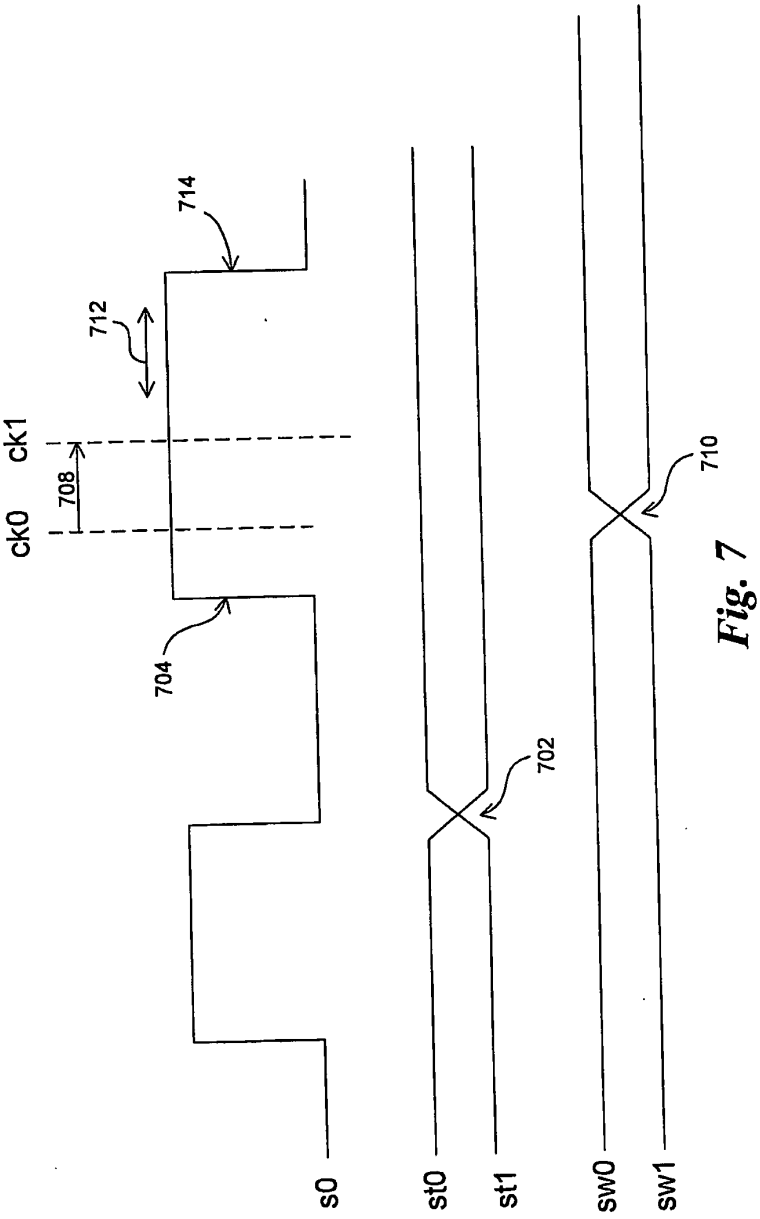


Fig. 7

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 01/50149

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L7/033

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)
IPC 7 H04L 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, INSPEC, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 317 159 A (AMERICAN TELEPHONE & TELEGRAPH) 24 May 1989 (1989-05-24) page 3, line 4 -page 4, line 19 page 5, line 13 - line 49; figures 1,3,5 ---	1-30
A	CHIH-KONG KEN YANG ET AL: "A 0.6/spl mu/m CMOS 4Gb/s Transceiver With Data Recovery Using Oversampling" SYMPOSIUM ON VLSI CIRCUITS DIGEST OF TECHNICAL PAPERS, pages 71-72, XP010245952 the whole document --- -/--	1-30

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

[X] Patent family members are listed in 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
15	23/04/2002

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, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Marzenke, M

Form PCT/ISA/210 (second sheet) (July 1992)

IPR2024-001428
HP – EX1002, Page 231 of 487

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	KYEONGHO LEE ET AL: "High Speed Low EMI Digital Video Interface With Cable Deskewing and transition Minimization Coding" SYMPOSIUM ON VLSI CIRCUITS DIGEST OF TECHNICAL PAPERS, 1997, pages 33-34, XP010245963 the whole document	1-30
A	KANG J-K ET AL: "A CMOS HIGH-SPEED DATA RECOVERY CIRCUIT USING THE MATCHED DELAY SAMPLING TECHNIQUE" IEEE JOURNAL OF SOLID-STATE CIRCUITS, IEEE INC. NEW YORK, US, vol. 32, no. 10, 1 October 1997 (1997-10-01), pages 1588-1596, XP000751031 ISSN: 0018-9200 the whole document	1-30

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 01/50149

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0317159 A	24-05-1989	US 4821297 A	11-04-1989
		CA 1288839 A1	10-09-1991
		DE 3888927 D1	11-05-1994
		DE 3888927 T2	20-10-1994
		EP 0317159 A2	24-05-1989
		ES 2050710 T3	01-06-1994
		JP 1161936 A	26-06-1989
		KR 9107714 B1	30-09-1991

Form PCT/ISA/210 (patent family annex) (July 1992)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 582 537 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **93480096.2**

(51) Int. Cl.[5]: **H04L 29/06, H04L 12/64**

(22) Date of filing: **16.07.93**

(30) Priority: **07.08.92 US 927697**

(43) Date of publication of application:
09.02.94 Bulletin 94/06

(84) Designated Contracting States:
AT BE CH DE ES FR GB IT LI NL SE

(71) Applicant: **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor: **Cidon, Israel**
Technion - I.I.T.
Haifa 32000 (IL)

Inventor: **Doney, Richard M.**
5724 Fortunes Ridge Drive
Durham, NC 27713 (US)
Inventor: **Drake, John Ellis, Jr.**
321 Fearrington
Pittsboro, NC 27312 (US)
Inventor: **Hervatic, Elizabeth Anne**
4908 Matlock Street
Apex, NC 27502 (US)
Inventor: **Potter, Kenneth Harvey, Jr.**
5404 Amsterdam Place
Raleigh, NC 27606-9708 (US)
Inventor: **Tedijanto, Theodore Ernest**
106 Tasman Court
Cary, NC 27513 (US)

(74) Representative: **de Pena, Alain**
Compagnie IBM France Département de Propriété Intellectuelle
F-06610 La Gaude (FR)

(54) Transmission of high-priority, real-time traffic on low-speed communications links.

(57) A protocol is defined for mixed data/voice/multimedia communications systems to transmit and receive high-priority, real-time traffic over low-speed digital communication links by embedding such high-priority traffic in low-priority, non-real-time traffic. High-priority, real-time packets are thus transmitted without delay by preempting low-priority packets. Low-priority, non-real-time packets are held during preemption, and low-priority transmission is automatically resumed after transmission of high-priority packets has been completed. A protocol is defined for communications systems to exchange information, at the time that a communication link is activated, defining their link capabilities for handling high-priority, real-time packets and to agree on how this high-priority traffic will be transmitted on the communication link.

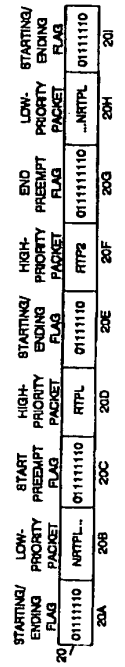


FIG. 2

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 234 of 487

Field of the Invention

The present invention relates to data communications networks and more particularly to a data communications network having the capability of processing both high priority and low priority packets using a preempt-resume protocol.

Background of the Invention

Communications systems traditionally have used packet switching techniques for carrying bursty data traffic and have used circuit switching techniques for carrying multiplexed real-time traffic such as voice and video. Circuit switching techniques are typified by a time-division multiplexed voice telephone network in which the traffic is sent as a continuous stream of bits. Packet switching techniques, on the other hand, have been developed to handle bursty data over digital networks in which destination and drop-off addresses are combined with the message data. Each packet is delimited by flags and contains address/routing headers, priority definers and error checkers. Traditional packet networks are characterized by significant per packet processing in the intermediate nodes of a network. This processing has limited the throughput of packet nodes and introduced high delays for packets. To achieve higher throughput and to reduce this delay, fast packet switching networks have been defined which minimize the amount of processing required in intermediate nodes.

This simplified intermediate node processing now makes it feasible for packet networks to carry, in the form of packets, traffic traditionally carried only over circuit switched networks. In addition, this traffic in packetized form can share the same packet network including communication links with the bursty data traffic. However since the traditional circuit switched traffic had stringent bounds on total allowable delay across the network as well as variability of delay, nodes in the packet network must ensure that this traffic receives priority handling. To accomplish this, packets carrying bursty data traffic can be assigned a non real-time priority while packets carrying the traditional circuit model traffic can be assigned a higher, real time priority. A node in a fast packet network contains buffers for holding packets waiting for transmission on its communication links. Packets waiting for transmission can be held in buffers managed differently , depending on the priority, assigned to the packets.

A communication node in a network can adopt a number of different service policies in order to transmit packets from the different priority buffers: priority with no preemption, preemption with retransmission, and preemption with resume. When do preemption is used, the packet priority is only examined to determine from which buffer to select the next packet for transmission. If a high-priority packet is placed in the buffer while a low-priority packet is being transmitted, the high-priority packet must wait until the current transmission is completed. A preemption with retransmission service policy means that the node will abort the transmission of a low-priority packet upon the arrival of a high-priority packet and immediately transmit the high-priority packet. Once all high-priority packets have been transmitted, transmission of the preempted low-priority packet will be restarted from the beginning of the packet. A preemption with resume service policy is similar except the preempted low-priority packet is restarted from the point of interruption rather than the beginning.

The selection of the appropriate service policy is dependent on the characteristics of the communication link, the delay requirements of the high-priority packets, and the size of the low-priority packets. If the transmission rate of the communication link is high enough compared to the size of the longest low-priority packets, then the delay incurred by a high-priority packet waiting for a low-priority one to complete may be acceptable. In this case, the priority with no preemption service policy is preferable since it is easier to implement and may have slightly lower link overhead. If the usage efficiency of the communication link is not important but the delay associated with waiting for completion of the low-priority packets is too high, than the preemption with retransmission service policy may be acceptable. However, if the usage efficiency of the communication link is important and the priority with no preemption service policy does not meet the delay requirements, then the preempt with resume service policy may be required.

Various schemes exist for transmitting packetized information over communication links. The typical scheme used over low speed serial links up to T3 speeds is based ont he HDLC MAC-layer protocol. Each packet is delimited by starting and ending flags (X'7E'). The ending flag of one packet may also be the starting flag for the next packet. The packet itself consists of an integral number of bytes of data. Since the contents of the packet may include bit patterns that are the same as the flag pattern, a technique known as bit stuffing is used to differentiate the data from the flags. The transmitter inserts a '0' bit after any sequence of five contiguous '1' bits int he packet data. Likewise, the receiver removes any '0' bit immediately following a sequence of five '1' bits in the received bit stream. When no packets are waiting to be transmitted, flags are repeatedly transmitted.

Both the priority with no preemption and the preemption with retransmission service policies can be im-

IPR2024-001428
HP – EX1002, Page 235 of 487

plemented using the existing HDLC MAC-layer protocol.

Summary of the invention

The present invention relates to method and apparatus for effecting a modified HDLC MAC-layer protocol providing preemptive priority with resume over a serial communication link.

It is, therefore, an object of this invention to provide method and apparatus for embedding high-priority traffic in low-priority for serial transmission though low speed communication links, without delay incurred by having first to complete transmission of a low-priority traffic. It is another object of this invention to preempt low-priority packet traffic with a high-priority packet and later resume transmission of the preempted low-priority packet automatically, with minimal overhead on the communication link. It is another object of this invention to detect bit transmission errors affecting packet boundaries and to recover from said errors with minimal loss of packets. It is another object of this invention to provide a method and apparatus for determining whether preemption of low-priority packets is required on a given communication link as a function of link speed, acceptable delay and maximum packet size, and therefore whether preemption should be enabled. It is another object of this invention to allow a mode of operation that is compatible with HDLC MAC-layer, either priorit to enabling preemption or when it is determined that preemption is not required.

According to this invention, packets of information are embedded in frames having starting and ending flags, control headers to designate the packet's priority, and any required routing information. The flags, in addition to defining the boundaries of a packet, also define byte alignment of the packet data. The flags also indicate when a low-priority packet has been preempted by a high-priority packet and when the transmission of a low-priority packet is being resumed. The control bits designate whether or not a particular packet is high-priority or low-priority and hence whether or not the packet is preemptable.

Also when a communication link is activated, the two communications systems at each end of the link exchange control information about their respective packet capabilities. The control information describes the maximum supported low-priority packet size supported in the transmit and receive direction and the ability of the communications system to support high-priority packets. Using the information exchanged along with the communication link's data rate and the maximum acceptable delay for high-priority packets, each communications system independently determines whether the preempt/resume protocol should be enabled or whether a simple priority with no preemption the protocol should be used.

Brief Description of the Drawings

Figure 1 shows a basic packet frame used in the practice of this invention, where packet containing header and data is delimited by flags;
Figure 2 shows the valid combination of formatted packet frames in which a low-priority packet is preempted by a high-priority packet with subsequent automatic resumption;
Figure 3 shows a combination of packet frames containing a bit error which causes a transmission abort;
Figure 4 shows a block diagram of the transmit portion of a communication link interface of a communications systems to which this invention is applicable;
Figure 5 shows a block diagram of the receive portion of a communication link interface of a communications system to which this invention is applicable;
Figure 6 is a table used in explaining how maximum permissible packet size and the need for preempt/resume protocols is determined in each direction of a communication link; and
Figure 7 is a finite state machine table showing preempt/resume states.

Detailed Description of the Invention

This invention defines a preempt/resume protocol extension to the existing HDLC MAC-layer packet framing protocol used on serial communication links, to allow the preemption of low-priority packets so that high-priority packets may be transmitted with minimal delay. A link activation protocol is also defined to determine whether the preempt/resume protocol extension should be enabled. During the link activation , normal HDLC MAC-layer framing protocol is used to transmit packets across the communication link. Also if the link activation determines that preempt/resume should not be used, all packets are sent using the HDLC MAC-layer framing protocol with simple priority without preemption.

IPR2024-001428
HP – EX1002, Page 236 of 487

Enabling Preempt/Resume Protocol Extension

Determination of when to use the preempt/resume protocol extension is based on the support of high priority traffic, low-priority packet sizes supported, communication link speed, and maximum acceptable delay of high-priority packets. The two communications systems on each end of the communication link can determine the need for the preempt/resume protocol independently for their direction of the link. During link-activation, each communications system sends a control message to its neighbor at the other end over the communication link using the normal HDLC MAC-layer framing. The control message contains the following fields :

- . A High-priority Traffic Supported filed indicates whether the sending system supports high-priority traffic on its link.
- . A Maximum Received Low-Priority Packet Size Supported field defines the maximum packet size that the sender can receive.
- . A Maximum Transmitted Low Priority Packet Size Supported field defines the maximum packet size that the sender can transmit .

If high-priority traffic is not supported, the receiver must verify that each packet received is a low-priority packet. If a packet identified as high-priority packet is received, it is discarded by the receiver.

The following example, described with reference to Figure 6, shows how communications systems A and B each determine the maximum low-priority packet size supported in each direction and whether preempt/resume protocol extension should be enabled.

Transmission of a high-priority packet may not be delayed by a low-priority packet for more than T, where in the following example T = 0.5 msecs. A sender that supports high-priority traffic must either transmit :

. without preempt/resume protocol, which constrains the low-priority packet size to satisfy the equation

$$\frac{\text{packet size}}{\text{link speed}} < T$$

. or with preempt/resume.

For link direction A to B, the maximum low-priority packet size is 8KB. Assuming a link speed of 18.432Mbps, the above inequality is not satisfied since

$$\frac{8\ \text{Kbytes}}{18.432\ \text{kbits per msec}} = 3.56\ \text{msecs} > 0.5\ \text{msecs}$$

Therefore, to satisfy the delay requirements, the preempt/resume protocol is required. This means that communications system A will transmit preempted packets and that communications system B must support receipt of preempted packets.

For link direction B to A, the maximum low-priority packet size is 1KB. The above inequality is satisfied as

$$\frac{\text{Kbytes}}{18.432\ \text{kbits per msec}} = 0.44\ \text{msecs} < 0.5\ \text{msecs}$$

and the preempt/resume protocol is not be used. This means that communications system B will not transmit preempted packets and that communications system A will interpret preempt flags as error conditions.

If a particular communications system has not implemented the preempt/resume protocol extension but supports high-priority packets, it will select a maximum low-priority packet size to greater than the value of link speed x T. Using this value, the communications system on the other end of the communication link will correctly determine that the preempt/resume protocol is not to be used.

Preempt/resume Protocol Extension

In the following description, bit sequences may be described using either conventional binary representation or, for the sake of convenience, hexadecimal representation.

The protocol for allowing high-priority packets to temporarily preempt low-priority packets uses three types of flags to delimit packets : a normal flag which can be a starting, endingor idle flag, a start-preempt flag and an end-preempt flag. The normal flag is defined as the 8-bit sequence B'01111110'(X'7e'). The star-preempt flag is defined as the 9-bit sequence B'011111110' and the end-preempt flag is defined as the ten-bit sequence B'0111111110'. All flags are on byte boundaries with respect to the packet data that they delineate. To differentiate flag bit sequences from bit sequences within the packets, zero bit stuffing is used in the packet data. An extra '0' bit is inserted in the transmitted bit stream after each occurrence of five consecutive '1' bits in a packet. A sequence of more than eight '1' bits indicates an error condition aborting the current packet being transmitted and received. Also a sequence of more than six '1' bits indicates an abort condition if the preempt/resume protocol is not enabled.

IPR2024-001428
HP – EX1002, Page 237 of 487

The following is a set of rules adopted for a practical preempt/resume protocol.

. The bit sequence B'01111110' (X'7E') always defines byte alignment and may occur any number of times before and after complete packets.
. Sic '1' bits preceded by a '0' bit that is not byte aligned with received packet data is an invalid code.
. Nine '1' bits preceded by a '0' bit that is byte aligned is also an invalid code.
. Receipt of an invalid code aborts the current packet and all subsequent packets are aborted until X'7E' occurs.
. Verification that the preempted packet is a low priority packet is performed.
. Verification that packets received during preemption are high-priority packets is also performed.
. A low-priority packet cannot be preempted until the first byte is transmitted.
. Receipt of the start-preempt flag (B'011111110') immediately followed by the end-preempt flag (B'0111111110') aborts the preempted packet and ends preempt mode.

Under the foregoing rules, the following is a valid combination of packets and flags when preempt/resume is not enabled.

7E { [7E] [RTP 7E] [7E] [NRTP 7E] }

Under the foregoing rules, the following is a valid combination of packets and flags when preempt/resume is enabled :

```
7E { [ 7E ] [ RTP 7E ] [ 7E ] [ NRTP 7E ]
     <----------------------preempted packet---------------------->
   [ pNRTP { [ SP [ 7E ] RTP [ 7E RTP ] [ 7E ]  EP ]  pNRTP } ] }
                 <-----------preemption----------->
```

where

. [] denotes optional and repeatable fields
. {} denotes required, repeatable fields
. 7E represents the byte-aligned flag (B'01111110', X'7E')
. RTP represents a high-priority packet
. NRTP represents a low-priority packet
. pNRTP represents portions of a preempted low-priority packet
. SP represents a start-preempt flag (B'011111110')
. EP represents an end-preempt flag (B'0111111110')

Figure 1 shows a conventional frame 10 delimited by normal (starting and ending) 7E flags 10a and containing both a control header 10b field and a data 10c field.

Figure 2 illustrates in frame sequence 20 a preempt valid operation in more detail with the case of a low priority packet being preempted by two consecutive high-priority packets. The first field 20a shows the normal byte-aligned starting flag X'7E'. The second field 20b is an ongoing low-priority packet NRTP1. The third field 20c shows a start-preempt or SP flag bit by bit. This SP flag interrupts the low-priority packet and indicates the transmission of the remainder of the low-priority packet NRTP1 is suspended. The fourth field 20d represents the first high-priority packet RTP1. the fifth field 20c shows the recurrence of a normal flag indicating the completion of the RTP1 packet. The transmission of the second high-priority packet RTP2 begins immediately in the sixth field 20f, without reversion to NRTP1. the seventh field 20g contains an end-preempt flag EP.

Thereafter the remainder of the preempted low-priority packet NRTP1 is completed, as shown in the eighth field 20h. Finally, in the ninth field 20i, the normal flag X'7E' indicates the end of NRTP1 and returns the system to the ready state.

Figure 3 illustrates in frame sequence 30 the case of a bit error corrupting the start-preempt flag SP in the third field 30c. The bit error would typically be caused by a transmission error on the communication link. The first and second fields 30a and 30b are identical with the same fields in Figure 2. The third field 30c shows a double '0' at the beginning of what otherwise would be a normal flag. Since the flag does not occur on a byte-aligned boundary with respect to the low-priority packet NRTP1 in the second field 30b, then NRTP1 is invalid and discarded. The high-priority packet RTP1 in the fourth field 30d is saved because it is surrounded by valid flags in the third and fifth fields 30c and 30c. The second high-priority packet RTP2 in the sixth field 30f is discarded, however, because it is followed by an end-preempt flag EP in the seventh field 3g, when no

IPR2024-001428
HP – EX1002, Page 238 of 487

valid start-preempt flag SP was detected due to the bit error. All packets thereafter are discarded until the normal flag X'7E' appears, as in the ninth field 30i.

The preempt/resume states of this invention can be summarized by a Finite State Machine (FSM) Table shown in Figure 7. Since this preempt/resume protocol is a "bit-oriented" protocol, the complete FSM describing the protocol machine (the sender or the receiver) is also at the bit level. However, for the sake of clarity, the following describes only the FSM in terms of detected "sequences of bits", which captures those state transitions which are associated with preemption. Furthermore, for the purpose of describing the protocol, is believed to suffice to show the FSM for a receiver only.

The following is a list of the FSM states, inputs and outputs (actions) along with their description.

STATE	DESCRIPTION
idl	Idle, expecting '7E'.
rdy	Just received '7E' or non-byte-aligned '7E*', ready to receive either high-priority packet or low priority packet
rtp	Receiving high-priority packet
rnrtp	Receiving low-priority packet
p_rdy1	Just entered preemption mode (just received start-preempt flag SP), ready to receive high-priority
packet	or '7E'
p_rdy2	In preemption mode, just received either '7E' or
'7E*',	no high-priority packet has been received yet during current preemption
p_rdy3	In preemption mode, just received either '7E' or
'7E*',	ready to receive a high-priority packet, at least one high-priority packet has already been received during current preemption
p_rrtp	Receiving high-priority packet in preemption mode
p_idl2	Idle in preemption mode, expecting '7E', no high-priority packet has been received yet during the
current	preemption
p_idl3	Idle in preemption mode, expecting '7E', at least one high-priority packet has already been received during the current preemption

IPR2024-001428
HP – EX1002, Page 239 of 487

p_end Just exited preemption mode (just received end-preempt flag EP), expecting data (the continuation of the preempted low-priority packet)

INPUT	DESCRIPTION
'7E' share	Normal starting/ending/idle flag, X'7E' (does not bit with prior flag)
'7E*'	X'7E' flag not byte-aligned with received data bytes
RTP	Data (non-flag) byte from a high-priority packet, distinguished from NRTP by a control bit in the first byte of a string, indicating packet priority. Zero bit stuffing is performed on sequences of these bytes.
NRTP	Data (non-flag) byte from a low-priority packet, distinguished from RTP by a control bit in the first byte of a string indicating packet priority. Zero bit stuffing is performed on sequences of these bytes.
SP	Start-preempt flag B'011111110' (seven 1's)
EP	End-preempt flag B'0111111110' (eight 1's)
IC	Invalid codes, include non-byte-aligned SP, non-byte-aligned EP, a run of six '1' bits right after a flag
or	B'0111111111 (nine 1's), or B'01111111 (seven 1's)
when	preempt/resume is not enabled

OUTPUT DESCRIPTION

IPR2024-001428
HP – EX1002, Page 240 of 487

strt_R	Indicate start of received high-priority packet and forward first byte
strt_N	Indicate start of received low-priority packet and forward first byte
more_R	Forward another byte of high-priority packet
more_N	Forward another byte of low-priority packet
end_R	Indicate end of received high-priority packet
end_N	Indicate end of received low-priority packet
abrt_R	Abord/discard current received high-priority packet
abrt_RN	abord/discard both preempting high-priority packet and the preempted low-priority packet

Note that states p_rdy1, p_rdy2 and p_rdy3 are similar in that the receiver is in the preempt mode and expecting a high-priority packet in all these states. However, it is necessary to have p_rdy1 and p_rdy2 to ensure that all packets are recovered when surrounded by normal flags. Likewise, p_rdy2 and p_rdy3 are needed to tell whether or not it is legal to receive the end-preempt (EP) flag.

Embodiment of Communication Link Interfaces

Figure 4 shows a block diagram of the transmitter 40 portion of the communication link interface of that communications system. Packets arrive from the communications system's packet source 41 for transmission on communication link 48. The packets may have been generated locally by this system or may have been received from another communication link on this system (e.g. an intermediate node in a packet network). The communications system places the high-priority packets into a high priority buffer 42 and places the low-priority packets into a low priority buffer 43. If no packets are stored in either high priority buffer 42 or low priority buffer 43, a flag generator 46 is connected to the communication link 48 via a bit multiplexer 47. The flag generator 46 repeatedly generates an idle flag X'7E', when no packets are stored for transmission.

When a low-priority packet arrives in the low priority buffer 43 and a packet is currently being transmitted, the transmitter 40 waits until all earlier packets in the low priority buffer 43 have been transmitted and the high priority buffer 42 is empty. When a low-priority packet is at the head of the low priority buffer 43 and no other packet is being transmitted on the communication link 48, bytes from the low priority buffer 43 are transferred one at a time through a byte multiplexer 44 to a parallel serial converter 45. The parallel serial converter 45 serializes the data and monitors the outgoing data for sequences of five consecutive '1' bits. It also inserts a single '0' bit immediately after each set of five '1' bits. The resulting bit stream is routed through the bit to the communication link 48. When the transmission of the low-priority packet is complete, the bit multiplexer 47 selects the flag generator 46 for transmission to send at least one or more normal flags until the next packet is ready to be transmitted. Note that each time a flag is sent, the parallel serial converter 45 resets its internal count of the number of consecutive '1' bits.

If a low-priority packet is being transmitted from low priority buffer 43 and a high-priority packet arrives in the high priority buffer 42, then the transmission of the low-priority packet is preempted. The remaining bits int he parallel serial converter 45 along with any stuffed zero bits are transmitted guaranteeing a data byte

IPR2024-001428
HP – EX1002, Page 241 of 487

boundary for the preempted packet, and then the flag generator 46 sends special start-preempt flag described earlier. Bytes from the high priority buffer 42 are then transferred through the byte multiplexer 44 to the parallel serial converter 45 which performs serialization and zero bit stuffing. The resulting high-priority packet is then transferred to the communication link 48. If, during the transmission of the high-priority packet, another high-priority packet arrives in the high-priority buffer 42, the n the flag generator 46 sends a normal flag when the first high-priority packet is completed and transmitter 40 begins transmission of the next high-priority packets without exiting the preempt mode. When the last of the series of high-priority packets has been sent (there are not more packets waiting in the high priority buffer 42), the flag generator 46 sends the end-preempt flag described earlier. The remaining bytes from the preempted low-priority packet in the low priority buffer 43 are the released to the parallel serial converter 45 and the communication link 48. If a subsequent high-priority packet arrives at the high-priority buffer 42 prior to the completion of the preempted low-priority packet, the preemption and resume sequence is repeated. When the transmission of the low-priority packet is completed, the flag generator 46 transmits a normal ending flag.

Figure 5 shows a block diagram of the receiver 50 portion of the communication link interface of the communications system up to a point at which received whole packets are passed to a packet target 56 within the communications system. The packet target 56 could be the final destination for the received packets or could be a packet switch used to route packets to other communication links for transmission to other nodes in a packet network. Any buffering associated with the packet target 56 is outside the receiver 50 and is not included in Figure 5.

A flag detector 52 continuously monitors the bit stream received from a communication link 51 for normal, start-preempt and end-preempt flags. If a sequence of bits other than a flag is detected immediately following a normal flag, it indicates the beginning of a new frame. A serial parallel converter 53 receives the bit stream, discard any '0' bit if it immediately follows five consecutive '1' bits, and converts the remaining bits into byte-parallel form. If the received packet is a high-priority , packet, the parallel byte data is passed directly through a multiplexer 59 to a multiplexer 55 connected to the packet target 56 until a normal ending flag is detected by the flag detector 52. The receiver 50 indicates the end of the packet to the packet target 56.

If the received packet can be preempted (i.e. low-priority, non-real-time packet), then the parallel byte data is instead passed through byte multiplexer 59 to the preemptable packet buffer 54 in order to permit the entire packet to be accumulated before passing it to the packet target 56. If flag detector 52 detects a start-preempt flag and there is no partial byte in the serial parallel converter 53, then it indicates the beginning of a high-priority preempting packet and therefore the beginning of preempt mode. The bit stream is passed through the serial parallel converter 53 as before but this time the parallel byte data is passed directly through the multiplexers 59 and 55 to the packet target 56 within the communications system.

When the flag detector 52 detects either an normal ending flag or an end-preempt flag, then the receiver 50 indicates the end of the packet to the packet target 56. If a normal ending flag is detected, then the serial parallel converter 53 will continue to route the parallel byte data from subsequent packets directly to the multiplexer 55. If an end-preempt flag is detected, the receiver 50 will end preempt mode. The received bit stream will be routed through the serial parallel converter 53 and multiplexer 59 to the preemptable packet buffer 54 thus resuming reception of the preempted low-priority packet . If the flag detector 52 detects a normal ending flag indicating the end of the low-priority packet, the receiver 50 transfers the entire low-priority packet stored in the preemptable packet buffer 54 through the multiplexer 55 to the packet target 56.

Figure 5 assumes that the transfer of a whole packet from the preemptable packet buffer 54 to the packet target 56 is accomplished before the first parallel data byte from a subsequent packet can be received. If the transfer takes longer than this in a particular implementation, a FIFO buffer can be placed after the serial parallel converter 53 to temporarily hold the receive data bytes until the transfer from the preemtable packet buffer 54 is complete. Also in some implementations, high-priority packets may be passed to separate packet targets. In this case, the preemptable packet buffer 54 is not required.

Claims

1. A packet-switched communications system for mixed low-priority and high-priority traffic characterized in that it comprises:
   means for embedding high-priority packets within low-priority packets based on distinctive pattern codes attached to said packets,
   means responsive to a distinctive start-preempt pattern
   for transmitting the high-priority packet immediately,
   means for buffering a low-priority packet that may be preempted by a high-priority packet, and

IPR2024-001428
HP – EX1002, Page 242 of 487

means responsive to a distinctive end-preempt pattern for resuming transmission of a preempted low-priority packet temporarily stored in said buffering means.

2. A communications system for transmitting mixed low-priority and high-priority traffic characterized in that it comprises:
means for distinguishing between low-priority and high-priority packets,
means for generating a start-preempt pattern to interrupt transmission of a low-priority packet,
means for buffering remainder of low-priority packet that has been preempted,
means for generating an end-preempt pattern at the completion of transmission of all high-priority packets and the resumption of transmission of the interrupted low-priority packet.

3. A communication system according to claim 2 in which:
said distinguishing means is a pattern detector having stored therein the patterns identifying respectively a low-priority packet and a high-priority packet and further having comparative means for matching received patterns with said stored patterns.

4. A communications system for receiving mixed low-priority and high-priority traffic characterized in that it comprises:
means for distinguishing between low-priority and high-priority received packets,
means for detecting start of preempting high-priority packets,
means for comparing bit patterns in received signals with stored distinctive flag patterns to determine whether a particular pattern is a complete packet or an interrupted low-priority packet,
means for receiving multiple high-priority packet without resuming preempted low-priority packet,
means for buffering low-priority packet that may be preempted by a high-priority packet,
means for resuming reception of preempted low-priority packet,
means for completing low-priority packet temporarily stored in said buffering means.

5. A method for embedding high-priority communication packets within low-priority communication packet in a packet-switched communications system characterized in that it comprises the steps of:
a) prefixing each low-priority packet with a characteristic start flag;
b) prefixing each high-priority packet with a characteristic start-preempt pattern of bits;
c) suffixing each low-priority packet with a characteristic ending flag;
d) suffixing each high-priority packet with a characteristic end-preempt pattern of bits;
e) using the starting flag to initiate transmission of packets and to establish byte alignment with a packet frame;
f) using the start-preempt pattern of bits to interrupt
the transmission of low-priority packets and buffer non-transmitted low-priority bits while transmission of high-priority bits continues;
g) using the end-preempt pattern of bits following the completion of transmission of a high-priority packet to indicate the resumption of the transmission of the interrupted low-priority packet; and
h) using the ending flag to return the communications system to the idle state.

6. A method for receiving packet-switched communication signals in which high-priority packets are embedded in low-priority packets characterized in that it comprises the steps of:
a) monitoring incoming signal bit streams for the presence of preassigned patterns of starting and ending
flags and start-preempt and end-preempt flags;
b) responding to a starting flag to effect byte alignment and to enter a ready state for receiving and buffering low-priority packets;
c) responding to a start-preempt flag to suspend reception of low-priority packet and accept and relay a high-priority packet immediately;
d) responding to an end-preempt flag to resume reception of the suspended low-priority packet;
e) responding to an ending flag to relay the completed low-priority packet,
f) responding to the failure to detect an end-preempt flag after a high-priority packet by aborting further reception until a normal flag occurs.

7. A method for receiving packet-switched communication signals according claim 6 further comprising:
set of flag conditions that when received return the receiver to a non-error state,
set of error states that allow recovering of some packets prior to returning to a non-error state, and

IPR2024-001428
HP – EX1002, Page 243 of 487

a finite state machine defining the conditions under which transitions occur between non-error and error states.

8. A method for communications systems at each end of a communication links to determine whether preempt/resume is required for each link-direction at the time of link activation characterized in that it comprises:
means to exchange link capabilities between adjacent communication systems over the communications systems over the communication link at link activation time,
means for each communications system to independently computer whether preempt/resume protocol extension should be enable, and
means for communications systems not supporting preempt/resume protocol extension to activate link with communications system supporting extension.

IPR2024-001428
HP – EX1002, Page 244 of 487

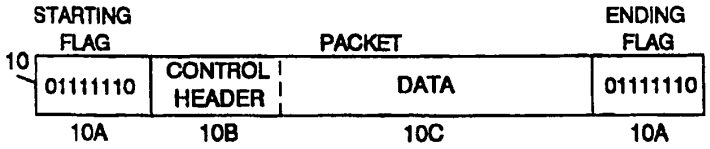


FIG. 1

FIG. 2

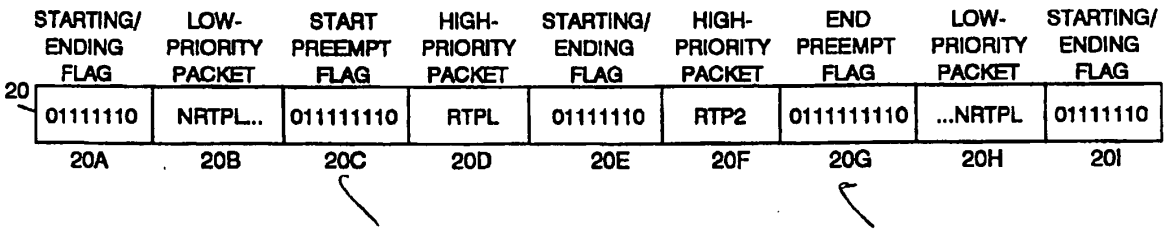


FIG. 3

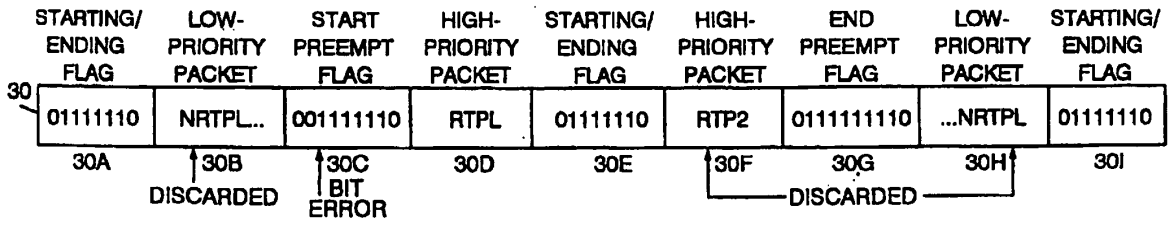


FIG. 4

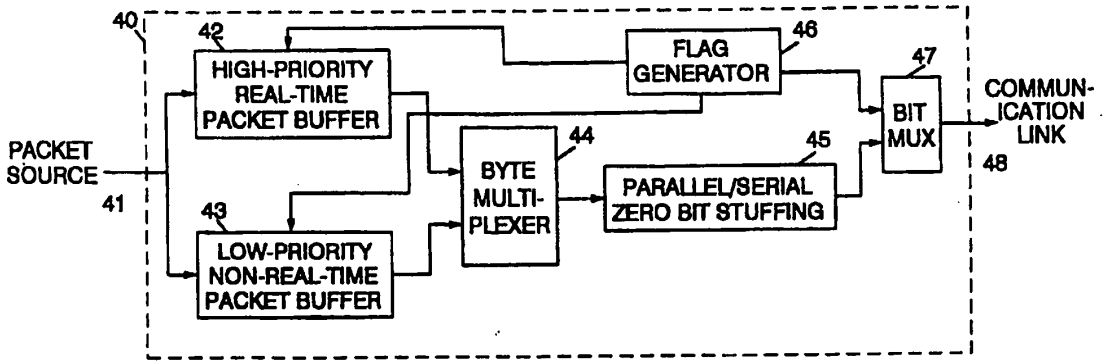

40
42
HIGH-PRIORITY REAL-TIME PACKET BUFFER
FLAG GENERATOR
46
47
BIT MUX
COMMUN-ICATION LINK
48
PACKET SOURCE
41
44
BYTE MULTI-PLEXER
45
PARALLEL/SERIAL ZERO BIT STUFFING
43
LOW-PRIORITY NON-REAL-TIME PACKET BUFFER

FIG. 5

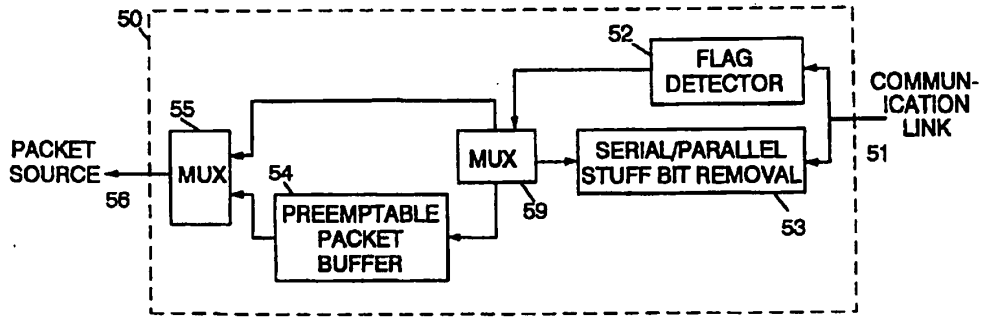

50
52
FLAG DETECTOR
COMMUN-ICATION LINK
51
55
PACKET SOURCE
56
MUX
54
PREEMPTABLE PACKET BUFFER
MUX
59
SERIAL/PARALLEL STUFF BIT REMOVAL
53

FIG. 6

FIELD IN LINK ACTIVATION CONTROL MESSAGE	CONTROL MESSAGE A→B	CONTROL MESSAGE B→A	LINK DIRECTION DESCRIBED BY FIELD IN CONTROL MESSAGE	RESOLUTION	COMMENT
HIGH-PRIORITY TRAFFIC SUPPORTED	YES	YES	A ⟷ B	YES	HIGH-PRIORITY PACKETS WILL BE SUPPORTED ON THIS LINK, SINCE BOTH ENDS SUPPORT IT.
MAXIMUM LOW-PRIORITY PACKET SIZE SUPPORTED	RECEIVE = 5KB	TRANSMIT= 1KB	B→A	1KB	B WILL TRANSMIT A MAXIMUM LOW-PRIORITY PACKET SIZE OF 1KB, WHICH IS MIN (5KB, 1KB).
MAXIMUM LOW-PRIORITY PACKET SIZE SUPPORTED	TRANSMIT= 9KB	RECEIVE= 8KB	A→B	8KB	A WILL TRANSMIT A MAXIMUM LOW-PRIORITY PACKET SIZE OF 8KB, WHICH IS MIN (8KB, 9KB).

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 247 of 487

INPUT → / ↓STATE	7E	7E*	RTP	NRTP	SP	EP	IC
IDL	RDY		IDL	IDL	IDL	IDL	IDL
RDY	RDY		RRTP STRT_R	RNTP STRT_N	IDL	IDL	IDL
RRTP	RDY END_R	RDY ABRT_R	RRTP MORE_R		IDL ABRT_R	IDL ABRT_R	IDL ABRT_R
RNRTP	RDY END_N	RDY ABRT_N		RNRTP MORE_N	P_RDY1	IDL ABRT_N	IDL ABRT_N
P_RDY1	P_RDY2		P_RRTP STRT_R	IDL ABRT_N	P_IDL2	IDL ABRT_N	P_IDL2
P_RDY2	P_RDY2		P_RRTP STRT_R	RNRTP ABRT_N STRT_N	P_IDL2	IDL ABRT_N	P_IDL2
P_RDY3	P_RDY3		P_RRTP STRT_R	RNRTP ABRT_N STRT_N	P_IDL3	P_END	P_IDL3
P_RRTP	P_RDY3 END_R	P_RDY3 ABRT_R	P_RRTP MORE_R		P_IDL3 ABRT_R	P_END END_R	P_IDL3 ABRT_R
P_IDL2	P_RDY2		P_IDL2	P_IDL2	P_IDL2	P_IDL2	P_IDL2
P_IDL3	P_RDY3		P_IDL3	P_IDL3	P_IDL3	P_IDL3	P_IDL3
P_END	RDY ABRT_N			RNTP MORE_N	P_RDY1	IDL ABRT_N	IDL ABRT_N

NOTE: ENTRIES WITHOUT VALUES REPRESENT INPUTS THAT CAN NOT BE DETECTED WHILE IN THE CURRENT STATE.

FIG. 7

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 248 of 487

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2001

Application or Docket Number

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	30	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	30 minus 20=	* 10
INDEPENDENT CLAIMS	4 minus 3 =	* 1
MULTIPLE DEPENDENT CLAIM PRESENT		☐

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐		OR	OTHER THAN SMALL ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	370.00	OR	BASIC FEE	740.00
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL		OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	* 26	Minus	** 30	= —
Independent	* 4	Minus	*** 4	= —
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. 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 number found in the appropriate box in column 1.

IPR2024-001428
HP – EX1002, Page 249 of 487

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.
10/035,911	11/07/2001	Yeshik Shin	594728119US	2160

25096 7590 11/15/2005

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

EXAMINER	
BLOUNT, STEVEN	
ART UNIT	**PAPER NUMBER**
2668	

DATE MAILED: 11/15/2005

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

PTO-90C (Rev. 10/03)

AK

Office Action Summary	Application No.	Applicant(s)	
	10/035,911	SHIN ET AL.	
	Examiner	Art Unit	
	Steven Blount	2668	

-- 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 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, 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 *03 July 2003*.

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

3)☐ 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

4)☒ Claim(s) *1 - 30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1 - 2, 5 - 10, 13 - 18, 21 - 25, 28 - 30* is/are rejected.

7)☒ Claim(s) *3 - 4, 11 - 12, 19 - 20, and 26 - 27* is/are objected to.

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

Application Papers

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

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

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

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

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

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).

a)☐ All b)☐ Some * c)☐ None of:

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) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1, 2, 6, 9, 10, 14, 17, 18, 24, and 25 are rejected under 35 U.S.C. 102(b) as being anticipated by European Patent Application 0582537 to Cidon.

With regard to claim 1, Cidon teaches receiving a low priority data packet and beginning its transmission, followed by receiving a second, high priority packet, followed by stopping the transmission of the first, data packet. See page 4 lines 45+ and page 5 lines 38+. Cidon also teaches that this is followed by transmitting a preempt indicator. See the SP flag mentioned on page 5, line 43. Following the preempt indicator, the second, high priority packet is transmitted. See line 44. The end preempt indicator is the equivalent of the continue indicator as claimed. This is followed by transmission of the remainder of the low priority packet.

With regard to claim 2, the indicators are primitives.

With regard to claim 6, the first and second packets are transmitted through the same link.

With regard to claims 9, 10, and 14, see the rejection of claims 1, 2, and 6 and note that all the method steps in these claims are discussed therein.

With regard to claim 17, see the rejection of claim 1 and note that the indicators are transmitted via the same communications link.

With regard to claim 18, the indicators are, as noted above, primitives.

With regard to claims 24 and 25, see the rejection of claim 1 and note that the "components" as claimed are discussed on page 8 lines 38+, page 8 lines 55+, and page 9 lines 8+.

Claim Rejections - 35 USC § 103

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

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

4. Claims 5, 7 – 8, 13, 15 – 16, 21 – 23, and 28 - 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over European patent Application 0582537 to Cidon.

With regard to claims 5, 13, 21, and 28, the first packet of Cidon is a data packet, and while it is not explicitly stated that the second packet is a control packet, one of ordinary skill in the art would find it obvious that the "real time traffic" mentioned in Cidon as being the second data packet (see the abstract) would have control information associated with it.

With regard to claims 7, 15, 22, and 29, it would have been obvious at the time of the invention to have applied the teachings above with respect to the first and second packets to a third packet so that a higher priority packet of this third type can be transmitted before the first and second packets.

With regard to claims 8, 16, 23, and 30, the examiner takes Official Notice that communication packets have headers. Further, the sender/receiver type of system taught in Cidon would make obvious having the header being transmitted only once.

5. Claims 3 – 4, 11 – 12, 19 – 20, and 26 – 27 are objected to as being dependent upon rejected independent claims, would be allowable if rewritten to include the limitations of the base claims and any intervening claims.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Steven Blount whose telephone number is 571 - 272 - 3071. The examiner can normally be reached on M-F 9:00 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Mr. Chieh Fan, can be reached on 571 – 272 - 3042. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SB

11/05/2005

CHIEH M. FAN
PRIMARY EXAMINER

Notice of References Cited	Application/Control No. 10/035,911	Applicant(s)/Patent Under Reexamination SHIN ET AL.	
	Examiner Steven Blount	Art Unit 2668	Page 1 of 1

U.S. PATENT DOCUMENTS

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

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N	EP 0582537	09-1994	EPO	Cidon	H04L 29/06
	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.

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 582 537 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **93480096.2**

(51) Int. Cl.[5]: **H04L 29/06**, H04L 12/64

(22) Date of filing: **16.07.93**

(30) Priority: **07.08.92 US 927697**

(43) Date of publication of application:
09.02.94 Bulletin 94/06

(84) Designated Contracting States:
AT BE CH DE ES FR GB IT LI NL SE

(71) Applicant: **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor: **Cidon, Israel**
Technion - I.I.T.
Haifa 32000 (IL)

Inventor: **Doney, Richard M.**
5724 Fortunes Ridge Drive
Durham, NC 27713 (US)
Inventor: **Drake, John Ellis, Jr.**
321 Fearrington
Pittsboro, NC 27312 (US)
Inventor: **Hervatic, Elizabeth Anne**
4908 Matlock Street
Apex, NC 27502 (US)
Inventor: **Potter, Kenneth Harvey, Jr.**
5404 Amsterdam Place
Raleigh, NC 27606-9708 (US)
Inventor: **Tedijanto, Theodore Ernest**
106 Tasman Court
Cary, NC 27513 (US)

(74) Representative: **de Pena, Alain**
Compagnie IBM France Département de Propriété Intellectuelle
F-06610 La Gaude (FR)

(54) Transmission of high-priority, real-time traffic on low-speed communications links.

(57) A protocol is defined for mixed data/voice/multimedia communications systems to transmit and receive high-priority, real-time traffic over low-speed digital communication links by embedding such high-priority traffic in low-priority, non-real-time traffic. High-priority, real-time packets are thus transmitted without delay by preempting low-priority packets. Low-priority, non-real-time packets are held during preemption, and low-priority transmission is automatically resumed after transmission of high-priority packets has been completed. A protocol is defined for communications systems to exchange information, at the time that a communication link is activated, defining their link capabilities for handling high-priority, real-time packets and to agree on how this high-priority traffic will be transmitted on the communication link.

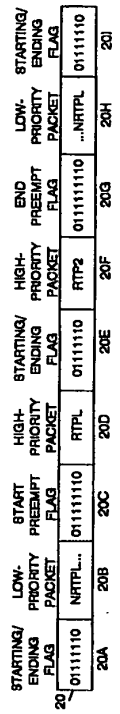


FIG. 2

BEST AVAILABLE COPY

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 256 of 487

Field of the Invention

The present invention relates to data communications networks and more particularly to a data communications network having the capability of processing both high priority and low priority packets using a preempt-resume protocol.

Background of the Invention

Communications systems traditionally have used packet switching techniques for carrying bursty data traffic and have used circuit switching techniques for carrying multiplexed real-time traffic such as voice and video. Circuit switching techniques are typified by a time-division multiplexed voice telephone network in which the traffic is sent as a continuous stream of bits. Packet switching techniques, on the other hand, have been developed to handle bursty data over digital networks in which destination and drop-off addresses are combined with the message data. Each packet is delimited by flags and contains address/routing headers, priority definers and error checkers. Traditional packet networks are characterized by significant per packet processing in the intermediate nodes of a network. This processing has limited the throughput of packet nodes and introduced high delays for packets. To achieve higher throughput and to reduce this delay, fast packet switching networks have been defined which minimize the amount of processing required in intermediate nodes.

This simplified intermediate node processing now makes it feasible for packet networks to carry, in the form of packets, traffic traditionally carried only over circuit switched networks. In addition, this traffic in packetized form can share the same packet network including communication links with the bursty data traffic. However since the traditional circuit switched traffic had stringent bounds on total allowable delay across the network as well as variability of delay, nodes in the packet network must ensure that this traffic receives priority handling. To accomplish this, packets carrying bursty data traffic can be assigned a non real-time priority while packets carrying the traditional circuit model traffic can be assigned a higher, real time priority. A node in a fast packet network contains buffers for holding packets waiting for transmission on its communication links. Packets waiting for transmission can be held in buffers managed differently , depending on the priority, assigned to the packets.

A communication node in a network can adopt a number of different service policies in order to transmit packets from the different priority buffers: priority with no preemption, preemption with retransmission, and preemption with resume. When do preemption is used, the packet priority is only examined to determine from which buffer to select the next packet for transmission. If a high-priority packet is placed in the buffer while a low-priority packet is being transmitted, the high-priority packet must wait until the current transmission is completed. A preemption with retransmission service policy means that the node will abort the transmission of a low-priority packet upon the arrival of a high-priority packet and immediately transmit the high-priority packet. Once all high-priority packets have been transmitted, transmission of the preempted low-priority packet will be restarted from the beginning of the packet. A preemption with resume service policy is similar except the preempted low-priority packet is restarted from the point of interruption rather than the beginning.

The selection of the appropriate service policy is dependent on the characteristics of the communication link, the delay requirements of the high-priority packets, and the size of the low-priority packets. If the transmission rate of the communication link is high enough compared to the size of the longest low-priority packets, then the delay incurred by a high-priority packet waiting for a low-priority one to complete may be acceptable. In this case, the priority with no preemption service policy is preferable since it is easier to implement and may have slightly lower link overhead. If the usage efficiency of the communication link is not important but the delay associated with waiting for completion of the low-priority packets is too high, than the preemption with retransmission service policy may be acceptable. However, if the usage efficiency of the communication link is important and the priority with no preemption service policy does not meet the delay requirements, then the preempt with resume service policy may be required.

Various schemes exist for transmitting packetized information over communication links. The typical scheme used over low speed serial links up to T3 speeds is based ont he HDLC MAC-layer protocol. Each packet is delimited by starting and ending flags (X'7E'). The ending flag of one packet may also be the starting flag for the next packet. The packet itself consists of an integral number of bytes of data. Since the contents of the packet may include bit patterns that are the same as the flag pattern, a technique known as bit stuffing is used to differentiate the data from the flags. The transmitter inserts a '0' bit after any sequence of five contiguous '1' bits int he packet data. Likewise, the receiver removes any '0' bit immediately following a sequence of five '1' bits in the received bit stream. When no packets are waiting to be transmitted, flags are repeatedly transmitted.

Both the priority with no preemption and the preemption with retransmission service policies can be im-

IPR2024-001428
HP – EX1002, Page 257 of 487

plemented using the existing HDLC MAC-layer protocol.

Summary of the invention

The present invention relates to method and apparatus for effecting a modified HDLC MAC-layer protocol providing preemptive priority with resume over a serial communication link.

It is, therefore, an object of this invention to provide method and apparatus for embedding high-priority traffic in low-priority for serial transmission though low speed communication links, without delay incurred by having first to complete transmission of a low-priority traffic. It is another object of this invention to preempt low-priority packet traffic with a high-priority packet and later resume transmission of the preempted low-priority packet automatically, with minimal overhead on the communication link. It is another object of this invention to detect bit transmission errors affecting packet boundaries and to recover from said errors with minimal loss of packets. It is another object of this invention to provide a method and apparatus for determining whether preemption of low-priority packets is required on a given communication link as a function of link speed, acceptable delay and maximum packet size, and therefore whether preemption should be enabled. It is another object of this invention to allow a mode of operation that is compatible with HDLC MAC-layer, either priorit to enabling preemption or when it is determined that preemption is not required.

According to this invention, packets of information are embedded in frames having starting and ending flags, control headers to designate the packet's priority, and any required routing information. The flags, in addition to defining the boundaries of a packet, also define byte alignment of the packet data. The flags also indicate when a low-priority packet has been preempted by a high-priority packet and when the transmission of a low-priority packet is being resumed. The control bits designate whether or not a particular packet is high-priority or low-priority and hence whether or not the packet is preemptable.

Also when a communication link is activated, the two communications systems at each end of the link exchange control information about their respective packet capabilities. The control information describes the maximum supported low-priority packet size supported in the transmit and receive direction and the ability of the communications system to support high-priority packets. Using the information exchanged along with the communication link's data rate and the maximum acceptable delay for high-priority packets, each communications system independently determines whether the preempt/resume protocol should be enabled or whether a simple priority with no preemption the protocol should be used.

Brief Description of the Drawings

Figure 1 shows a basic packet frame used in the practice of this invention, where packet containing header and data is delimited by flags;
Figure 2 shows the valid combination of formatted packet frames in which a low-priority packet is preempted by a high-priority packet with subsequent automatic resumption;
Figure 3 shows a combination of packet frames containing a bit error which causes a transmission abort;
Figure 4 shows a block diagram of the transmit portion of a communication link interface of a communications systems to which this invention is applicable;
Figure 5 shows a block diagram of the receive portion of a communication link interface of a communications system to which this invention is applicable;
Figure 6 is a table used in explaining how maximum permissible packet size and the need for preempt/resume protocols is determined in each direction of a communication link; and
Figure 7 is a finite state machine table showing preempt/resume states.

Detailed Description of the Invention

This invention defines a preempt/resume protocol extension to the existing HDLC MAC-layer packet framing protocol used on serial communication links, to allow the preemption of low-priority packets so that high-priority packets may be transmitted with minimal delay. A link activation protocol is also defined to determine whether the preempt/resume protocol extension should be enabled. During the link activation , normal HDLC MAC-layer framing protocol is used to transmit packets across the communication link. Also if the link activation determines that preempt/resume should not be used, all packets are sent using the HDLC MAC-layer framing protocol with simple priority without preemption.

IPR2024-001428
HP – EX1002, Page 258 of 487

Enabling Preempt/Resume Protocol Extension

Determination of when to use the preempt/resume protocol extension is based on the support of high priority traffic, low-priority packet sizes supported, communication link speed, and maximum acceptable delay of high-priority packets. The two communications systems on each end of the communication link can determine the need for the preempt/resume protocol independently for their direction of the link. During link-activation, each communications system sends a control message to its neighbor at the other end over the communication link using the normal HDLC MAC-layer framing. The control message contains the following fields :

. A High-priority Traffic Supported filed indicates whether the sending system supports high-priority traffic on its link.

. A Maximum Received Low-Priority Packet Size Supported field defines the maximum packet size that the sender can receive.

. A Maximum Transmitted Low Priority Packet Size Supported field defines the maximum packet size that the sender can transmit .

If high-priority traffic is not supported, the receiver must verify that each packet received is a low-priority packet. If a packet identified as high-priority packet is received, it is discarded by the receiver.

The following example, described with reference to Figure 6, shows how communications systems A and B each determine the maximum low-priority packet size supported in each direction and whether preempt/resume protocol extension should be enabled.

Transmission of a high-priority packet may not be delayed by a low-priority packet for more than T, where in the following example T = 0.5 msecs. A sender that supports high-priority traffic must either transmit :

. without preempt/resume protocol, which constrains the low-priority packet size to satisfy the equation

$$\frac{\text{packet size}}{\text{link speed}} < T$$

. or with preempt/resume.

For link direction A to B, the maximum low-priority packet size is 8KB. Assuming a link speed of 18.432Mbps, the above inequality is not satisfied since

$$\frac{8\ \text{Kbytes}}{18.432\ \text{kbits per msec}} = 3.56\ \text{msecs} > 0.5\ \text{msecs}$$

Therefore, to satisfy the delay requirements, the preempt/resume protocol is required. This means that communications system A will transmit preempted packets and that communications system B must support receipt of preempted packets.

For link direction B to A, the maximum low-priority packet size is 1KB. The above inequality is satisfied as

$$\frac{\text{Kbytes}}{18.432\ \text{kbits per msec}} = 0.44\ \text{msecs} < 0.5\ \text{msecs}$$

and the preempt/resume protocol is not be used. This means that communications system B will not transmit preempted packets and that communications system A will interpret preempt flags as error conditions.

If a particular communications system has not implemented the preempt/resume protocol extension but supports high-priority packets, it will select a maximum low-priority packet size to greater than the value of link speed x T. Using this value, the communications system on the other end of the communication link will correctly determine that the preempt/resume protocol is not to be used.

Preempt/resume Protocol Extension

In the following description, bit sequences may be described using either conventional binary representation or, for the sake of convenience, hexadecimal representation.

The protocol for allowing high-priority packets to temporarily preempt low-priority packets uses three types of flags to delimit packets : a normal flag which can be a starting, endingor idle flag, a start-preempt flag and an end-preempt flag. The normal flag is defined as the 8-bit sequence B'01111110'(X'7e'). The star-preempt flag is defined as the 9-bit sequence B'011111110' and the end-preempt flag is defined as the ten-bit sequence B'0111111110'. All flags are on byte boundaries with respect to the packet data that they delineate. To differentiate flag bit sequences from bit sequences within the packets, zero bit stuffing is used in the packet data. An extra '0' bit is inserted in the transmitted bit stream after each occurrence of five consecutive '1' bits in a packet. A sequence of more than eight '1' bits indicates an error condition aborting the current packet being transmitted and received. Also a sequence of more than six '1' bits indicates an abort condition if the preempt/resume protocol is not enabled.

IPR2024-001428
HP – EX1002, Page 259 of 487

The following is a set of rules adopted for a practical preempt/resume protocol.

. The bit sequence B'01111110' (X'7E') always defines byte alignment and may occur any number of times before and after complete packets.
. Sic '1' bits preceded by a '0' bit that is not byte aligned with received packet data is an invalid code.
. Nine '1' bits preceded by a '0' bit that is byte aligned is also an invalid code.
. Receipt of an invalid code aborts the current packet and all subsequent packets are aborted until X'7E' occurs.
. Verification that the preempted packet is a low priority packet is performed.
. Verification that packets received during preemption are high-priority packets is also performed.
. A low-priority packet cannot be preempted until the first byte is transmitted.
. Receipt of the start-preempt flag (B'011111110') immediately followed by the end-preempt flag (B'0111111110') aborts the preempted packet and ends preempt mode.

Under the foregoing rules, the following is a valid combination of packets and flags when preempt/resume is not enabled.

7E { [7E] [RTP 7E] [7E] [NRTP 7E] }

Under the foregoing rules, the following is a valid combination of packets and flags when preempt/resume is enabled :

```
7E { [ 7E ] [ RTP 7E ] [ 7E ] [ NRTP 7E ]
     <----------------------preempted packet---------------------->
   [ pNRTP { [ SP [ 7E ] RTP [ 7E RTP ] [ 7E ]  EP ]  pNRTP } ] }
                 <-----------preemption----------->
```

where

. [] denotes optional and repeatable fields
. {} denotes required, repeatable fields
. 7E represents the byte-aligned flag (B'01111110', X'7E')
. RTP represents a high-priority packet
. NRTP represents a low-priority packet
. pNRTP represents portions of a preempted low-priority packet
. SP represents a start-preempt flag (B'011111110')
. EP represents an end-preempt flag (B'0111111110')

Figure 1 shows a conventional frame 10 delimited by normal (starting and ending) 7E flags 10a and containing both a control header 10b field and a data 10c field.

Figure 2 illustrates in frame sequence 20 a preempt valid operation in more detail with the case of a low priority packet being preempted by two consecutive high-priority packets. The first field 20a shows the normal byte-aligned starting flag X'7E'. The second field 20b is an ongoing low-priority packet NRTP1. The third field 20c shows a start-preempt or SP flag bit by bit. This SP flag interrupts the low-priority packet and indicates the transmission of the remainder of the low-priority packet NRTP1 is suspended. The fourth field 20d represents the first high-priority packet RTP1. the fifth field 20c shows the recurrence of a normal flag indicating the completion of the RTP1 packet. The transmission of the second high-priority packet RTP2 begins immediately in the sixth field 20f, without reversion to NRTP1. the seventh field 20g contains an end-preempt flag EP.

Thereafter the remainder of the preempted low-priority packet NRTP1 is completed, as shown in the eighth field 20h. Finally, in the ninth field 20i, the normal flag X'7E' indicates the end of NRTP1 and returns the system to the ready state.

Figure 3 illustrates in frame sequence 30 the case of a bit error corrupting the start-preempt flag SP in the third field 30c. The bit error would typically be caused by a transmission error on the communication link. The first and second fields 30a and 30b are identical with the same fields in Figure 2. The third field 30c shows a double '0' at the beginning of what otherwise would be a normal flag. Since the flag does not occur on a byte-aligned boundary with respect to the low-priority packet NRTP1 in the second field 30b, then NRTP1 is invalid and discarded. The high-priority packet RTP1 in the fourth field 30d is saved because it is surrounded by valid flags in the third and fifth fields 30c and 30c. The second high-priority packet RTP2 in the sixth field 30f is discarded, however, because it is followed by an end-preempt flag EP in the seventh field 3g, when no

IPR2024-001428
HP – EX1002, Page 260 of 487

valid start-preempt flag SP was detected due to the bit error. All packets thereafter are discarded until the normal flag X'7E' appears, as in the ninth field 30i.

The preempt/resume states of this invention can be summarized by a Finite State Machine (FSM) Table shown in Figure 7. Since this preempt/resume protocol is a "bit-oriented" protocol, the complete FSM describing the protocol machine (the sender or the receiver) is also at the bit level. However, for the sake of clarity, the following describes only the FSM in terms of detected "sequences of bits", which captures those state transitions which are associated with preemption. Furthermore, for the purpose of describing the protocol, is believed to suffice to show the FSM for a receiver only.

The following is a list of the FSM states, inputs and outputs (actions) along with their description.

STATE	DESCRIPTION
idl	Idle, expecting '7E'.
rdy	Just received '7E' or non-byte-aligned '7E*', ready to receive either high-priority packet or low priority packet
rtp	Receiving high-priority packet
rnrtp	Receiving low-priority packet
p_rdy1	Just entered preemption mode (just received start-preempt flag SP), ready to receive high-priority packet or '7E'
p_rdy2	In preemption mode, just received either '7E' or '7E*', no high-priority packet has been received yet during current preemption
p_rdy3	In preemption mode, just received either '7E' or '7E*', ready to receive a high-priority packet, at least one high-priority packet has already been received during current preemption
p_rrtp	Receiving high-priority packet in preemption mode
p_idl2	Idle in preemption mode, expecting '7E', no high-priority packet has been received yet during the current preemption
p_idl3	Idle in preemption mode, expecting '7E', at least one high-priority packet has already been received during the current preemption

IPR2024-001428
HP – EX1002, Page 261 of 487

p_end Just exited preemption mode (just received end-preempt flag EP), expecting data (the continuation of the preempted low-priority packet)

INPUT	DESCRIPTION
'7E' share	Normal starting/ending/idle flag, X'7E' (does not bit with prior flag)
'7E*'	X'7E' flag not byte-aligned with received data bytes
RTP	Data (non-flag) byte from a high-priority packet, distinguished from NRTP by a control bit in the first byte of a string, indicating packet priority. Zero bit stuffing is performed on sequences of these bytes.
NRTP	Data (non-flag) byte from a low-priority packet, distinguished from RTP by a control bit in the first byte of a string indicating packet priority. Zero bit stuffing is performed on sequences of these bytes.
SP	Start-preempt flag B'011111110' (seven 1's)
EP	End-preempt flag B'0111111110' (eight 1's)
IC	Invalid codes, include non-byte-aligned SP, non-byte-aligned EP, a run of six '1' bits right after a flag
or	B'0111111111 (nine 1's), or B'01111111 (seven 1's)
when	preempt/resume is not enabled

OUTPUT	DESCRIPTION

IPR2024-001428
HP – EX1002, Page 262 of 487

strt_R	Indicate start of received high-priority packet and forward first byte
strt_N	Indicate start of received low-priority packet and forward first byte
more_R	Forward another byte of high-priority packet
more_N	Forward another byte of low-priority packet
end_R	Indicate end of received high-priority packet
end_N	Indicate end of received low-priority packet
abrt_R	Abord/discard current received high-priority packet
abrt_RN	abord/discard both preempting high-priority packet and the preempted low-priority packet

Note that states p_rdy1, p_rdy2 and p_rdy3 are similar in that the receiver is in the preempt mode and expecting a high-priority packet in all these states. However, it is necessary to have p_rdy1 and p_rdy2 to ensure that all packets are recovered when surrounded by normal flags. Likewise, p_rdy2 and p_rdy3 are needed to tell whether or not it is legal to receive the end-preempt (EP) flag.

Embodiment of Communication Link Interfaces

Figure 4 shows a block diagram of the transmitter 40 portion of the communication link interface of that communications system. Packets arrive from the communications system's packet source 41 for transmission on communication link 48. The packets may have been generated locally by this system or may have been received from another communication link on this system (e.g. an intermediate node in a packet network). The communications system places the high-priority packets into a high priority buffer 42 and places the low-priority packets into a low priority buffer 43. If no packets are stored in either high priority buffer 42 or low priority buffer 43, a flag generator 46 is connected to the communication link 48 via a bit multiplexer 47. The flag generator 46 repeatedly generates an idle flag X'7E', when no packets are stored for transmission.

When a low-priority packet arrives in the low priority buffer 43 and a packet is currently being transmitted, the transmitter 40 waits until all earlier packets in the low priority buffer 43 have been transmitted and the high priority buffer 42 is empty. When a low-priority packet is at the head of the low priority buffer 43 and no other packet is being transmitted on the communication link 48, bytes from the low priority buffer 43 are transferred one at a time through a byte multiplexer 44 to a parallel serial converter 45. The parallel serial converter 45 serializes the data and monitors the outgoing data for sequences of five consecutive '1' bits. It also inserts a single '0' bit immediately after each set of five '1' bits. The resulting bit stream is routed through the bit to the communication link 48. When the transmission of the low-priority packet is complete, the bit multiplexer 47 selects the flag generator 46 for transmission to send at least one or more normal flags until the next packet is ready to be transmitted. Note that each time a flag is sent, the parallel serial converter 45 resets its internal count of the number of consecutive '1' bits.

If a low-priority packet is being transmitted from low priority buffer 43 and a high-priority packet arrives in the high priority buffer 42, then the transmission of the low-priority packet is preempted. The remaining bits int he parallel serial converter 45 along with any stuffed zero bits are transmitted guaranteeing a data byte

IPR2024-001428
HP – EX1002, Page 263 of 487

boundary for the preempted packet, and then the flag generator 46 sends special start-preempt flag described earlier. Bytes from the high priority buffer 42 are then transferred through the byte multiplexer 44 to the parallel serial converter 45 which performs serialization and zero bit stuffing. The resulting high-priority packet is then transferred to the communication link 48. If, during the transmission of the high-priority packet, another high-priority packet arrives in the high-priority buffer 42, the n the flag generator 46 sends a normal flag when the first high-priority packet is completed and transmitter 40 begins transmission of the next high-priority packets without exiting the preempt mode. When the last of the series of high-priority packets has been sent (there are not more packets waiting in the high priority buffer 42), the flag generator 46 sends the end-preempt flag described earlier. The remaining bytes from the preempted low-priority packet in the low priority buffer 43 are the released to the parallel serial converter 45 and the communication link 48. If a subsequent high-priority packet arrives at the high-priority buffer 42 prior to the completion of the preempted low-priority packet, the preemption and resume sequence is repeated. When the transmission of the low-priority packet is completed, the flag generator 46 transmits a normal ending flag.

Figure 5 shows a block diagram of the receiver 50 portion of the communication link interface of the communications system up to a point at which received whole packets are passed to a packet target 56 within the communications system. The packet target 56 could be the final destination for the received packets or could be a packet switch used to route packets to other communication links for transmission to other nodes in a packet network. Any buffering associated with the packet target 56 is outside the receiver 50 and is not included in Figure 5.

A flag detector 52 continuously monitors the bit stream received from a communication link 51 for normal, start-preempt and end-preempt flags. If a sequence of bits other than a flag is detected immediately following a normal flag, it indicates the beginning of a new frame. A serial parallel converter 53 receives the bit stream, discard any '0' bit if it immediately follows five consecutive '1' bits, and converts the remaining bits into byte-parallel form. If the received packet is a high-priority , packet, the parallel byte data is passed directly through a multiplexer 59 to a multiplexer 55 connected to the packet target 56 until a normal ending flag is detected by the flag detector 52. The receiver 50 indicates the end of the packet to the packet target 56.

If the received packet can be preempted (i.e. low-priority, non-real-time packet), then the parallel byte data is instead passed through byte multiplexer 59 to the preemptable packet buffer 54 in order to permit the entire packet to be accumulated before passing it to the packet target 56. If flag detector 52 detects a start-preempt flag and there is no partial byte in the serial parallel converter 53, then it indicates the beginning of a high-priority preempting packet and therefore the beginning of preempt mode. The bit stream is passed through the serial parallel converter 53 as before but this time the parallel byte data is passed directly through the multiplexers 59 and 55 to the packet target 56 within the communications system.

When the flag detector 52 detects either an normal ending flag or an end-preempt flag, then the receiver 50 indicates the end of the packet to the packet target 56. If a normal ending flag is detected, then the serial parallel converter 53 will continue to route the parallel byte data from subsequent packets directly to the multiplexer 55. If an end-preempt flag is detected, the receiver 50 will end preempt mode. The received bit stream will be routed through the serial parallel converter 53 and multiplexer 59 to the preemptable packet buffer 54 thus resuming reception of the preempted low-priority packet . If the flag detector 52 detects a normal ending flag indicating the end of the low-priority packet, the receiver 50 transfers the entire low-priority packet stored in the preemptable packet buffer 54 through the multiplexer 55 to the packet target 56.

Figure 5 assumes that the transfer of a whole packet from the preemptable packet buffer 54 to the packet target 56 is accomplished before the first parallel data byte from a subsequent packet can be received. If the transfer takes longer than this in a particular implementation, a FIFO buffer can be placed after the serial parallel converter 53 to temporarily hold the receive data bytes until the transfer from the preemtable packet buffer 54 is complete. Also in some implementations, high-priority packets may be passed to separate packet targets. In this case, the preemptable packet buffer 54 is not required.

Claims

1. A packet-switched communications system for mixed low-priority and high-priority traffic characterized in that it comprises:
   means for embedding high-priority packets within low-priority packets based on distinctive pattern codes attached to said packets,
   means responsive to a distinctive start-preempt pattern
   for transmitting the high-priority packet immediately,
   means for buffering a low-priority packet that may be preempted by a high-priority packet, and

IPR2024-001428
HP – EX1002, Page 264 of 487

means responsive to a distinctive end-preempt pattern for resuming transmission of a preempted low-priority packet temporarily stored in said buffering means.

2. A communications system for transmitting mixed low-priority and high-priority traffic characterized in that it comprises:
means for distinguishing between low-priority and high-priority packets,
means for generating a start-preempt pattern to interrupt transmission of a low-priority packet,
means for buffering remainder of low-priority packet that has been preempted,
means for generating an end-preempt pattern at the completion of transmission of all high-priority packets and the resumption of transmission of the interrupted low-priority packet.

3. A communication system according to claim 2 in which:
said distinguishing means is a pattern detector having stored therein the patterns identifying respectively a low-priority packet and a high-priority packet and further having comparative means for matching received patterns with said stored patterns.

4. A communications system for receiving mixed low-priority and high-priority traffic characterized in that it comprises:
means for distinguishing between low-priority and high-priority received packets,
means for detecting start of preempting high-priority packets,
means for comparing bit patterns in received signals with stored distinctive flag patterns to determine whether a particular pattern is a complete packet or an interrupted low-priority packet,
means for receiving multiple high-priority packet without resuming preempted low-priority packet,
means for buffering low-priority packet that may be preempted by a high-priority packet,
means for resuming reception of preempted low-priority packet,
means for completing low-priority packet temporarily stored in said buffering means.

5. A method for embedding high-priority communication packets within low-priority communication packet in a packet-switched communications system characterized in that it comprises the steps of:
a) prefixing each low-priority packet with a characteristic start flag;
b) prefixing each high-priority packet with a characteristic start-preempt pattern of bits;
c) suffixing each low-priority packet with a characteristic ending flag;
d) suffixing each high-priority packet with a characteristic end-preempt pattern of bits;
e) using the starting flag to initiate transmission of packets and to establish byte alignment with a packet frame;
f) using the start-preempt pattern of bits to interrupt
the transmission of low-priority packets and buffer non-transmitted low-priority bits while transmission of high-priority bits continues;
g) using the end-preempt pattern of bits following the completion of transmission of a high-priority packet to indicate the resumption of the transmission of the interrupted low-priority packet; and
h) using the ending flag to return the communications system to the idle state.

6. A method for receiving packet-switched communication signals in which high-priority packets are embedded in low-priority packets characterized in that it comprises the steps of:
a) monitoring incoming signal bit streams for the presence of preassigned patterns of starting and ending
flags and start-preempt and end-preempt flags;
b) responding to a starting flag to effect byte alignment and to enter a ready state for receiving and buffering low-priority packets;
c) responding to a start-preempt flag to suspend reception of low-priority packet and accept and relay a high-priority packet immediately;
d) responding to an end-preempt flag to resume reception of the suspended low-priority packet;
e) responding to an ending flag to relay the completed low-priority packet,
f) responding to the failure to detect an end-preempt flag after a high-priority packet by aborting further reception until a normal flag occurs.

7. A method for receiving packet-switched communication signals according claim 6 further comprising:
set of flag conditions that when received return the receiver to a non-error state,
set of error states that allow recovering of some packets prior to returning to a non-error state, and

IPR2024-001428
HP – EX1002, Page 265 of 487

a finite state machine defining the conditions under which transitions occur between non-error and error states.

8. A method for communications systems at each end of a communication links to determine whether preempt/resume is required for each link-direction at the time of link activation characterized in that it comprises:
means to exchange link capabilities between adjacent communication systems over the communications systems over the communication link at link activation time,
means for each communications system to independently computer whether preempt/resume protocol extension should be enable, and
means for communications systems not supporting preempt/resume protocol extension to activate link with communications system supporting extension.

IPR2024-001428
HP – EX1002, Page 266 of 487

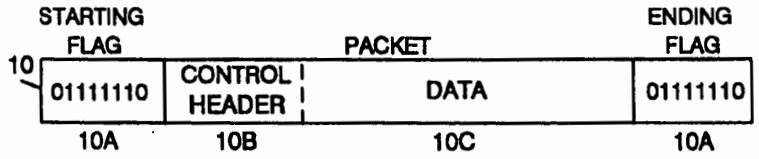


FIG. 1

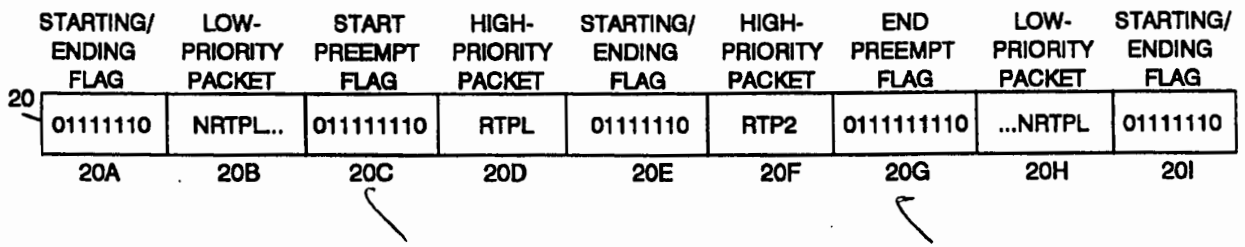


FIG. 2

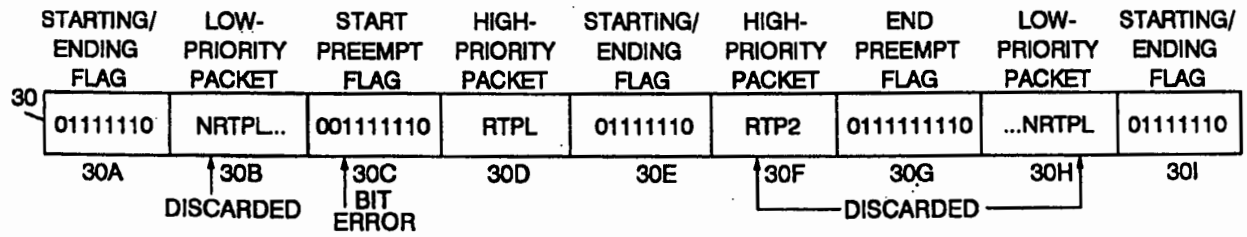


FIG. 3

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 267 of 487

FIG. 4

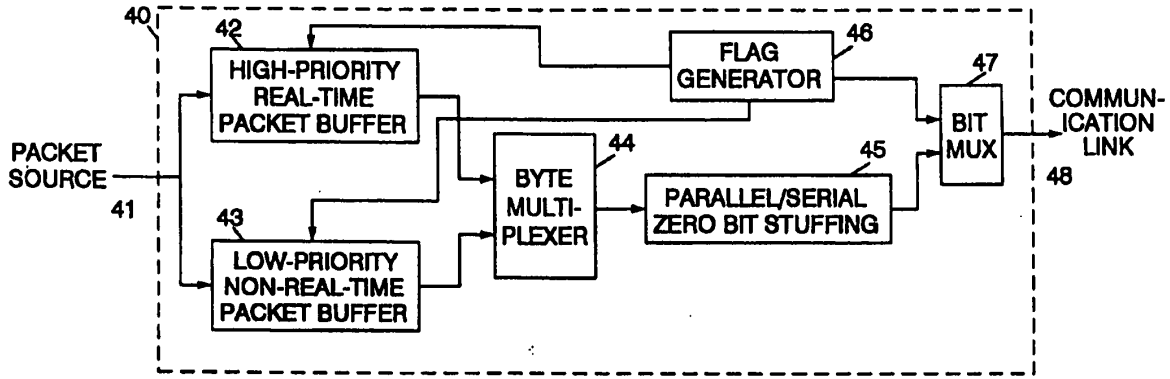
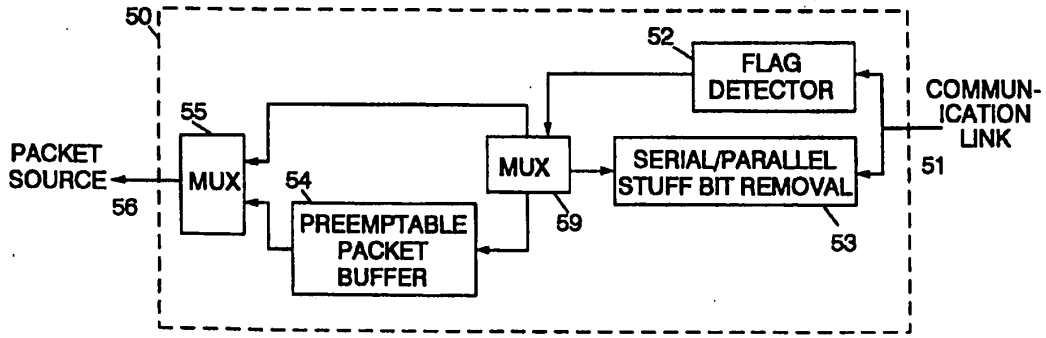


FIG. 5



EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 268 of 487

FIG. 6

FIELD IN LINK ACTIVATION CONTROL MESSAGE	CONTROL MESSAGE A → B	CONTROL MESSAGE B → A	LINK DIRECTION DESCRIBED BY FIELD IN CONTROL MESSAGE	RESOLUTION	COMMENT
HIGH-PRIORITY TRAFFIC SUPPORTED	YES	YES	A ↔ B	YES	HIGH-PRIORITY PACKETS WILL BE SUPPORTED ON THIS LINK, SINCE BOTH ENDS SUPPORT IT.
MAXIMUM LOW-PRIORITY PACKET SIZE SUPPORTED	RECEIVE = 5KB	TRANSMIT= 1KB	B → A	1KB	B WILL TRANSMIT A MAXIMUM LOW-PRIORITY PACKET SIZE OF 1KB, WHICH IS MIN (5KB, 1KB).
MAXIMUM LOW-PRIORITY PACKET SIZE SUPPORTED	TRANSMIT= 9KB	RECEIVE= 8KB	A → B	8KB	A WILL TRANSMIT A MAXIMUM LOW-PRIORITY PACKET SIZE OF 8KB, WHICH IS MIN (8KB, 9KB).

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 269 of 487

INPUT → / STATE ↓	7E	7E*	RTP	NRTP	SP	EP	IC
IDL	RDY		IDL	IDL	IDL	IDL	IDL
RDY	RDY		RRTP STRT_R	RNTP STRT_N	IDL	IDL	IDL
RRTP	RDY END_R	RDY ABRT_R	RRTP MORE_R		IDL ABRT_R	IDL ABRT_R	IDL ABRT_R
RNRTP	RDY END_N	RDY ABRT_N		RNRTP MORE_N	P_RDY1	IDL ABRT_N	IDL ABRT_N
P_RDY1	P_RDY2		P_RRTP STRT_R	IDL ABRT_N	P_IDL2	IDL ABRT_N	P_IDL2
P_RDY2	P_RDY2		P_RRTP STRT_R	RNRTP ABRT_N STRT_N	P_IDL2	IDL ABRT_N	P_IDL2
P_RDY3	P_RDY3		P_RRTP STRT_R	RNRTP ABRT_N STRT_N	P_IDL3	P_END	P_IDL3
P_RRTP	P_RDY3 END_R	P_RDY3 ABRT_R	P_RRTP MORE_R		P_IDL3 ABRT_R	P_END END_R	P_IDL3 ABRT_R
P_IDL2	P_RDY2		P_IDL2	P_IDL2	P_IDL2	P_IDL2	P_IDL2
P_IDL3	P_RDY3		P_IDL3	P_IDL3	P_IDL3	P_IDL3	P_IDL3
P_END	RDY ABRT_N			RNTP MORE_N	P_RDY1	IDL ABRT_N	IDL ABRT_N

NOTE: ENTRIES WITHOUT VALUES REPRESENT INPUTS THAT CAN NOT BE DETECTED WHILE IN THE CURRENT STATE.

FIG. 7

EP 0 582 537 A2

IPR2024-001428
HP – EX1002, Page 270 of 487

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☑ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

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

Bib Data Sheet

CONFIRMATION NO. 2160

SERIAL NUMBER	FILING DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
10/035,911	11/07/2001 RULE	370	2668	594728119US

APPLICANTS

Yeshik Shin, Sunnyvale, CA;

David D. Lee, Palo Alto, CA;
Deog-Kyoon Jeong, Seoul, KOREA, REPUBLIC OF;Shing Kong, Brisbane, CA;

** CONTINUING DATA **********
This appln claims benefit of 60/252,724 11/22/2000

** FOREIGN APPLICATIONS ********** none

IF REQUIRED, FOREIGN FILING LICENSE GRANTED
** 02/07/2002

Foreign Priority claimed	STATE OR	SHEETS	TOTAL	INDEPENDENT
35 USC 119 (a-d) conditions met	COUNTRY	DRAWING	CLAIMS	CLAIMS
Verified and Acknowledged	CA	59	30	4

☐ yes ☐ no
☐ yes ☐ no ☐ Met after Allowance
Examiner's Signature Initials

ADDRESS
25096
PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE , WA
98111-1247

TITLE
Method and system for nesting of communications packets

FILING FEE	FEES: Authority has been given in Paper No. _____ to charge/credit DEPOSIT ACCOUNT	☐ All Fees ☐ 1.16 Fees (Filing) ☐ 1.17 Fees (Processing Ext. of time)

IPR2024-001428
HP – EX1002, Page 272 of 487

Search Notes	Application/Control No.	Applicant(s)/Patent under Reexamination	
	10/035,911	SHIN ET AL.	
	Examiner	Art Unit	
	Steven Blount	2668	

SEARCHED			
Class	Subclass	Date	Examiner
370	465	11/2/05	SB
↓	468	↓	↓
↓	473	↓	↓
370	493	11/4/05	SB

INTERFERENCE SEARCHED			
Class	Subclass	Date	Examiner

SEARCH NOTES (INCLUDING SEARCH STRATEGY)		
	DATE	EXMR

Index of Claims

Application/Control No.	Applicant(s)/Patent under Reexamination
10/035,911	SHIN ET AL.
Examiner	**Art Unit**
Steven Blount	2668

√	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

Claim		Date								
Final	Original	11								
		03								
	1	✓								
	2	✓								
	3	o								
	4	o								
	5	✓								
	6	✓								
	7	✓								
	8	✓								
	9	✓								
	10	✓								
	11	o								
	12	o								
	13	✓								
	14	✓								
	15	✓								
	16	✓								
	17	✓								
	18	✓								
	19	o								
	20	o								
	21	✓								
	22	✓								
	23	✓								
	24	✓								
	25	✓								
	26	o								
	27	o								
	28	✓								
	29	✓								
	30	✓								
	31									
	32									
	33									
	34									
	35									
	36									
	37									
	38									
	39									
	40									
	41									
	42									
	43									
	44									
	45									
	46									
	47									
	48									
	49									
	50									

Claim		Date								
Final	Original									
	51									
	52									
	53									
	54									
	55									
	56									
	57									
	58									
	59									
	60									
	61									
	62									
	63									
	64									
	65									
	66									
	67									
	68									
	69									
	70									
	71									
	72									
	73									
	74									
	75									
	76									
	77									
	78									
	79									
	80									
	81									
	82									
	83									
	84									
	85									
	86									
	87									
	88									
	89									
	90									
	91									
	92									
	93									
	94									
	95									
	96									
	97									
	98									
	99									
	100									

Claim		Date								
Final	Original									
	101									
	102									
	103									
	104									
	105									
	106									
	107									
	108									
	109									
	110									
	111									
	112									
	113									
	114									
	115									
	116									
	117									
	118									
	119									
	120									
	121									
	122									
	123									
	124									
	125									
	126									
	127									
	128									
	129									
	130									
	131									
	132									
	133									
	134									
	135									
	136									
	137									
	138									
	139									
	140									
	141									
	142									
	143									
	144									
	145									
	146									
	147									
	148									
	149									
	150									

UNITED STATES PATENT AND TRADEMARK OFFICE

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

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO.
10/035,911	11/07/2001	Yeshik Shin	59472811 9US

CONFIRMATION NO. 2160

25096
PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

OC000000010949862

Date Mailed: 09/30/2003

Communication Regarding Rescission Of Nonpublication Request and/or Notice of Foreign Filing

Applicant's rescission of the previously-filed nonpublication request and/or notice of foreign filing is acknowledged. The paper has been reflected in the Patent and Trademark Office's (USPTO's) computer records so that the earliest possible projected publication date can be assigned.

The projected publication date is 01/08/2004.

If applicant rescinded the nonpublication request before or on the date of "foreign filing,"[1] then no notice of foreign filing is required.

If applicant foreign filed the application after filing the above application and before filing the rescission, and the rescission did not also include a notice of foreign filing, then a notice of foreign filing (not merely a rescission) is required to be filed within 45 days of the date of foreign filing. See 35 U.S.C. § 122(b)(2)(B)(iii), and Clarification of the United States Patent and Trademark Office's Interpretation of the Provisions of 35 U.S.C. § 122(b)(2)(B)(ii)-(iv), 1272 Off. Gaz. Pat. Office 22 (July 1, 2003).

If a notice of foreign filing is required and is not filed within 45 days of the date of foreign filing, then the application becomes abandoned pursuant to 35 U.S.C. § 122(b)(2)(B)(iii). In this situation, applicant should either file a petition to revive or notify the Office that the application is abandoned. See 37 CFR 1.137(f). Any such petition to revive will be forwarded to the Office of Petitions for a decision. Note that the filing of the petition will not operate to stay any period of reply that may be running against the application.

Questions regarding petitions to revive should be directed to the Office of Petitions at (703) 305-9282. Questions regarding publications of patent applications should be directed to the patent application publication hotline at (703) 605-4283 or by e-mail pgpub@uspto.gov.

[1] Note, for purpose of this notice, that "foreign filing" means "filing an application directed to the same invention in another country, or under a multilateral international agreement, that requires publication of applications 18 months after filing".

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
www.uspto.gov

Paper No. 7

PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE WA 98111-1247

COPY MAILED

SEP 3 0 2003

OFFICE OF PETITIONS

In re Application of
Yeshik Shin
Application No. 10/035,911
Filed: November 7, 2001
Attorney Docket No. : 594728119US
Title: METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

DECISION ON PETITION
UNDER 37 C.F.R. §1.137(f)

This is a decision on the petition filed on July 3, 2003, pursuant to 37 C.F.R. §1.137(f), to revive the above-identified application.

A grantable petition pursuant to 37 CFR 1.137(f) must be accompanied by:

(1) Notification of the filing of an application in a foreign country or under a multinational treaty that requires 18 month publication[1];
(2) The petition fee as set forth in 37 C.F.R. § 1.17(m), and;
(3) A statement that the entire delay in filing the notice from the date that the notice was due under 35 U.S.C. §122(b)(2)(B)(iii) until the date the notice was filed was unintentional. The Commissioner may require additional information where there is a question whether the delay was unintentional.

Petitioner states that the instant nonprovisional application is the subject of an application filed in an international application on November 21, 2001. However, the United States Patent and Trademark Office was unintentionally not notified of this filing within 45 days subsequent to the filing of the subject application in a foreign country.

1 See PTO/SB/36 and paragraph on PTO/SB/64a for further information. Both may be downloaded at http://www.uspto.gov/web/forms/index.html.

In view of the above, this application became abandoned pursuant to 35 U.S.C. §1.22(b)(2)(B)(iii) and 37 C.F.R. §1.213(c) for failure to timely notify the Office of the filing of an application under a multilateral international agreement that requires publication of applications 18 months after filing.

The petition under 37 C.F.R. §1.137(f) is **GRANTED**.

37 C.F.R. §1.137(f) requires a statement that the entire delay in filing the notice from the date that the notice was due under 35 U.S.C. §122(b)(2)(B)(iii) until the date the notice was filed was unintentional. Since the statement contained in the instant petition varies from the language required by 37 C.F.R. §1.137(f), the statement contained in the instant petition is being construed as the statement required by 37 C.F.R. §1.137(f) and petitioner must notify the Office if this is not a correct interpretation of the statement contained in the instant petition.

Petitioner has submitted the notification of an international filing, paid the petition fee, and has made a statement which is being construed as the proper statement of unintentional delay.

The instant petition has been found to be in compliance with 37 C.F.R. §1.137(f). Accordingly, the failure to timely notify the Office of a foreign or international filing within 45 days after the date of filing of such foreign or international application as provided by 35 U.S.C. §122(b)(2)(B)(iii) and 37 C.F.R. §1.213(c) is accepted as having been unintentionally delayed.

The previous Request and Certification under 35 U.S.C. §122(b)(2)(B)(i), filed with the original application papers, has been rescinded. A Notice Regarding Rescission of Nonpublication Request which sets forth the projected publication date of January 8, 2004 accompanies this decision on petition.

After this decision is mailed, the application will be forwarded to Technology Center 2600.

Telephone inquiries concerning *this decision* should be directed to Attorney Paul Shanoski at (703) 305-0011.

Paul Shanoski
Attorney
Office of Petitions
United States Patent and Trademark Office

OIPE
JUL 03 2003
PATENT & TRADEMARK OFFICE
JC36

07-07-03

#5 DAC/B

Attorney Docket No.: 594728819US
Express Mail Label EV 335524165 US

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

IN RE APPLICATION OF:	SHIN, ET AL.	EXAMINER:	
APPLICATION NO.:	10/035,911	ART UNIT:	2661
FILED:	11/07/2001	CONF. NO:	2160
FOR:	METHOD AND SYSTEM FOR NESTING COMMUNICATIONS PACKETS		

Petition to Revive Unintentionally Abandoned Application Under 37 C.F.R. § 1.137(b)

Mail Stop Petition
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

1. The applicants hereby petition for revival of the above-identified application. An international application directed to an invention disclosed in the above-referenced application was filed with the U.S. Receiving Office on November 21, 2001. On January 7, 2002, applicants filed a Request to Rescind Previous Nonpublication Request under 35 U.S.C. 122(b)(2)(B)(ii). However, it is our understanding that the United States Patent and Trademark Office has taken the position that the application became abandoned for failure to concurrently file a Notice of Foreign Filing in compliance with 35 U.S.C. 122(b)(2)(B)(iii).

2. Notice Under 37 C.F.R. § 1.137(b)(1)

   Enclosed herewith is a Notice of Rescission of Previous Nonpublication Request and Notice of Foreign Filing in compliance with 35 U.S.C. 122(b)(2)(B)(iii). An international application directed to an invention disclosed in the above-referenced application was filed with the U.S. Receiving Office on November 21, 2001.

3. Fee Under 37 C.F.R. § 1.137(b)(2)

   Enclosed is a check covering the fee of $1300.00 under 37 C.F.R. § 1.17(m).

4. Statement Under 37 C.F.R. § 1.137(b)(3)

   The entire delay in filing the required Notice until the filing of a grantable petition pursuant to 37 C.F.R. § 1.137(b) was unintentional.

07/09/2003 CNGUYEN 00000150 10035911
01 FC:1453 1300.00 OP

RECEIVED
JUL 10 2003
OFFICE OF PETITIONS

5. Duration of Abandonment (1203 OG 86-87, Oct. 21, 1997)

☒ This petition is being filed within 3 months of the date on which the Office clarified its position that a Request to Rescind Previous Nonpublication Request under 35 U.S.C. 122(b)(2)(B)(ii) [PTO/SB/36 (11-00)] is insufficient to comply with the requirements of 35 U.S.C. 122(b)(2)(B)(iii) [Notice of Foreign Filing]. See the "Reminder that Rescission of a Nonpublication Request is Not Itself a Notice of Foreign Filing" signed by Deputy Commissioner Kunin on April 11, 2003. Under current USPTO practice, further information regarding the abandonment is not required.

☐ This petition is being filed more than 3 months after the applicant was first notified of the abandonment, but within 1 year of the date of abandonment. Attached hereto is a statement by explaining how the abandonment occurred and how it was unintentional.

☐ This petition is being filed more than 1 year after the apparent date of abandonment. Attached hereto is a statement by explaining when the applicant or applicant's representative first became aware of the abandonment and how the delay in discovering the abandoned status occurred despite the exercise of due care or diligence on the part of the applicant or the applicant's representative.

6. Terminal Disclaimer under 37 C.F.R. § 1.137(b)(4) and (c)

☒ Since this application was filed on or after June 8, 1995, no terminal disclaimer is required.

☐ Since this application was filed before June 8, 1995, a terminal disclaimer is enclosed dedicating to the public a terminal part of the term of any patent granted hereon, and of any patent granted on a continuing application hereof, equivalent to the period of abandonment of this application.

7. Additional Fees

☒ Please charge any underpayment in fees for timely consideration of this Petition to Deposit Account No. 50-0665.

Respectfully submitted,
Perkins Coie LLP

Date: July 3, 2003

Maurice J. Pirio
Registration No. 33,273

Correspondence Address:
Customer No. 25096
Perkins Coie LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888

OIPE
JUL 0 3 2003
PATENT & TRADEMARK OFFICE
JC95

#6

PTO/SB/36 (05-03) (MODIFIED)
Approved for use through 04/30/2003. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

NOTICE OF RESCISSION OF PREVIOUS NONPUBLICATION REQUEST

(35 U.S.C. 122(b)(2)(B)(ii))
AND, IF APPLICABLE,

NOTICE OF FOREIGN FILING

(35 U.S.C. 122(b)(2)(B)(iii))

Send completed form to:
Mail Stop Petition
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450
FAX: (703) 305-8568

Application Number	10/035,911
Filing Date	11/07/2001
First Named Inventor	Yeshik Shin
Title	METHOD AND SYSTEM FOR NESTING COMMUNICATING PACKET
Attorney Docket Number	594728819US
Group Art Unit	2661
Examiner	

A request that the above-identified application not be published under 35 U.S.C. 122(b) (nonpublication request) was included with the above-identified application on filing pursuant to 35 U.S.C. 122(b)(2)(B)(i). I hereby **rescind** the previous nonpublication request.

☒ I hereby notify the Director under 35 U.S.C. 122(b)(2)(B)(iii) of a foreign or international filing. This notice is being provided later than forty-five (**45**) days after the date of such foreign or international filing. A petition to revive under 37 CFR 1.137(b) is enclosed. See 37 CFR 1.137(f).

*If a notice of subsequent foreign or international filing required by 35 U.S.C. 122(b)(2)(B)(iii) and 37 CFR 1.213(c) was not filed within forty-five (**45**) days after the date of filing of the foreign or international application, the application is ABANDONED, and a petition to revive under 37 CFR 1.137(b) is required. See 37 CFR 1.137(f).*

July 3, 2003
Date

Maurice J. Pirio
Signature

206-583-8548
Telephone Number

Maurice J. Pirio - Reg. No. 33,273
Typed or printed name

This request must be signed in compliance with 37 CFR 1.33(b).

If information or assistance is needed in completing this form, please contact the Pre-Grant Publication Division at (703) 605-4283 or by e-mail at PGPub@USPTO.gov.

RECEIVED
JUL 10 2003
OFFICE OF PETITIONS

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as Express Mail in an envelope addressed to: Mail Stop Petition, Commissioner For Patents, P.O. Box 1450, Alexandria, VA 22313-1450, or facsimile ((703) 305-8568) transmitted to the U.S. Patent and Trademark Office on the date shown below.

Name (Print/Type)	Douglas A. Gow		
Signature		*Date*	7/3/03

Burden Hour Statement: This collection of information is required by 37 CFR 1.213(b). The information is used by the public to rescind a previously filed request that an application not be published under 35 U.S.C. 122(b) (and the PTO to process that rescission). Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This form is estimated to take 6 minutes to complete. This time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Mail Stop PG Pub, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

[90020-0018/2nd rescind non-pub.doc]

COPY OF PAPERS ORIGINALLY FILED



Receipt

FILE COPY

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to Office of Initial Patent Examination, Customer Service Center, Commissioner for Patents, Washington, DC 20231. #4

7/26/02 2681

Date Jeanne Connelly

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Yeshik Shin et al.
Application No.	:	10/035,911 Confirmation No.: 2160
Filed	:	November 7, 2001
For	:	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS
Docket No.	:	371798019US
Date	:	July 26, 2002

RECEIVED
AUG 2 2 2002
Technology Center 2600

Office of Initial Patent Examination
Customer Service Center
Commissioner for Patents
Washington, DC 20231

REQUEST FOR CORRECTED FILING RECEIPT

Sir:

Attached is a copy of the official Filing Receipt received from the PTO in the above-identified application, for which issuance of a corrected Filing Receipt is respectfully requested.

There is an error with respect to the following data, which was incorrectly entered: In the "Applicant(s)" section, the residence of Yeshik Shin is misspelled. "Sunnyvile" should be "Sunnyvale."

Q:\Clients\Silicon Image (37179)\8019 (Nesting)\Correct Filing Receipt.doc

The corrections to be made have been marked in red on the enclosed copy of the Filing Receipt.

The corrections are not due to any errors by applicants, and therefore no fee is due.

Respectfully submitted,
Perkins Coie LLP

Maurice J. Pirio
Registration No. 33,273

MJP:jc
Enclosures:
Postcard
Copy of Filing Receipt with Corrections

P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888
Fax: (206) 583-8500

COPY OF PAPERS
ORIGINALLY FILED

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	DRAWINGS	TOT CLAIMS	IND CLAIMS
10/035,911	11/07/2001	2681	1134	371798019US	59	30	4

CONFIRMATION NO. 2160

UPDATED FILING RECEIPT

OC000000008291667

25096
PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

OIPE JUL 31 2002 PATENT & TRADEMARK OFFICE

Date Mailed: 06/17/2002

Receipt is acknowledged of this nonprovisional Patent Application. It will be considered in its order and you will be notified as to the results of the examination. Be sure to provide the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. 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 write to the Office of Initial Patent Examination's Filing Receipt Corrections, facsimile number 703-746-9195. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections 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 (if appropriate).**

Applicant(s)

Yeshik Shin, ~~Sunnyville~~ Sunnyvale, CA;
David D. Lee, Palo Alto, CA;
Deog-Kyoon Jeong, Seoul, KOREA, REPUBLIC OF;
Shing Kong, Brisbane, CA;

RECEIVED
AUG 2 2 2002
Technology Center 2600

Domestic Priority data as claimed by applicant

THIS APPLN CLAIMS BENEFIT OF 60/252,724 11/22/2000

Foreign Applications

If Required, Foreign Filing License Granted 02/07/2002

Projected Publication Date: Request for Non-Publication Acknowledged

Non-Publication Request: Yes

Early Publication Request: No

Title

Method and system for nesting of communications packets

Preliminary Class
455

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 Office of Export Administration, Department of Commerce (15 CFR 370.10 (j)); the Office of Foreign Assets Control, 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).

T.O /MPS



COPY OF PAPERS
ORIGINALLY FILED

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to Box Missing Parts, Commissioner for Patents, Washington, DC 20231.

5/24/02
Date

Jeanne Connelly

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants : Yeshik Shin et al.

Application No. : 10/035,911 Confirmation No.: 2160

Filed : November 7, 2001

For : METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

Docket No. : 371798019US

Date : May 24, 2002

Box Missing Parts
Commissioner for Patents
Washington, DC 20231

RESPONSE TO NOTICE TO FILE MISSING PARTS OF APPLICATION

Sir:

In response to the Notice to File Missing Parts dated February 7, 2002, please find enclosed a Declaration, Authorization for Extensions of Time Under 37 CFR § 1.136(a)(3), and a copy of the Notice to File Missing Parts for the above-identified application.

Also enclosed is a Transmittal of Formal Drawings (Figures 1-49B).

Please find enclosed a Petition for an Extension of Time, extending the time for response from April 7, 2002 to June 7, 2002.

IPR2024-001428
HP – EX1002, Page 285 of 487

The fees have been calculated as follows:

Basic Fee	$	740.00
Total Claims (30, 10 extra)		180.00
Independent Claims (4, 1 extra)		84.00
Missing Parts Surcharge		130.00
Total	$	1134.00
Extension-of-Time Fee		400.00
TOTAL	$	1534.00

Enclosed is a check in the amount of $1534.00 for the requisite fees. The Commissioner is hereby authorized to charge any additional filing fees or to credit any overpayment to Deposit Account No. 50-0665. A duplicate copy of this response is enclosed.

Respectfully submitted,

Perkins Coie LLP

Maurice J. Pirio
Registration No. 33,273

MJP:jc

Enclosures:
Postcard
Check
Copy of this Response
Declaration
Transmittal of Formal Drawings (Figures 1-49B)
Petition for Extension of Time (+ 2 copies)
Authorization for Extensions of Time Under 37 CFR § 1.136(a)(3)
Copy of Notice to File Missing Parts

PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888
FAX: (206) 583-8500

Express Mail Label No. EV035502544US

JUN 05 2002

COPY OF PAPERS
ORIGINALLY FILED

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
10/035,911	11/07/2001	Yeshik Shin	371798019US

CONFIRMATION NO. 2160

25096
PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

FORMALITIES LETTER

OC000000007439531

Date Mailed: 02/07/2002

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The statutory basic filing fee is missing.
  Applicant must submit $ **740** *to complete the basic filing fee for a non-small entity. If appropriate, applicant may make a written assertion of entitlement to small entity status and pay the small entity filing fee (37 CFR 1.27).*
- Total additional claim fee(s) for this application is $264.
  - **$180** for **10** total claims over 20.
  - **$84** for **1** independent claims over 3 .
- The oath or declaration is missing.
  A properly signed oath or declaration in compliance with 37 CFR 1.63, identifying the application by the above Application Number and Filing Date, is required.
- To avoid abandonment, a late filing fee or oath or declaration surcharge as set forth in 37 CFR 1.16(l) of $130 for a non-small entity, must be submitted with the missing items identified in this letter.
- **The balance due by applicant is $ 1134.**

06/07/2002 WABRHAM1 00000054 10035911
01 FC:101 740.00 DP
02 FC:103 180.00 DP
03 FC:102 84.00 DP
04 FC:105 130.00 DP

A copy of this notice MUST be returned with the reply.

Tran Chale

Customer Service Center
Initial Patent Examination Division (703) 308-1202
PART 2 - COPY TO BE RETURNED WITH RESPONSE

IPR2024-001428
HP – EX1002, Page 287 of 487

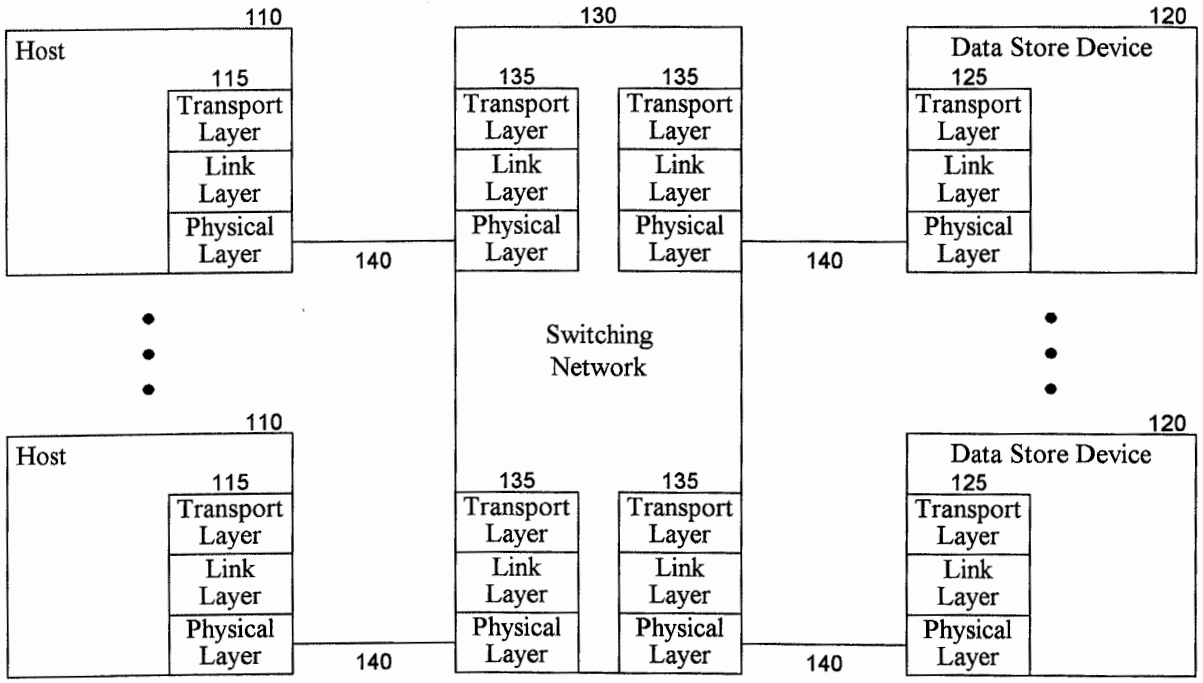

110
Host
115
Transport Layer
Link Layer
Physical Layer
140
130
135
Transport Layer
Link Layer
Physical Layer
135
Transport Layer
Link Layer
Physical Layer
Switching Network
120
Data Store Device
125
Transport Layer
Link Layer
Physical Layer
140
110
Host
115
Transport Layer
Link Layer
Physical Layer
140
135
Transport Layer
Link Layer
Physical Layer
135
Transport Layer
Link Layer
Physical Layer
120
Data Store Device
125
Transport Layer
Link Layer
Physical Layer

Fig. 1

20S090 TT6SE0T

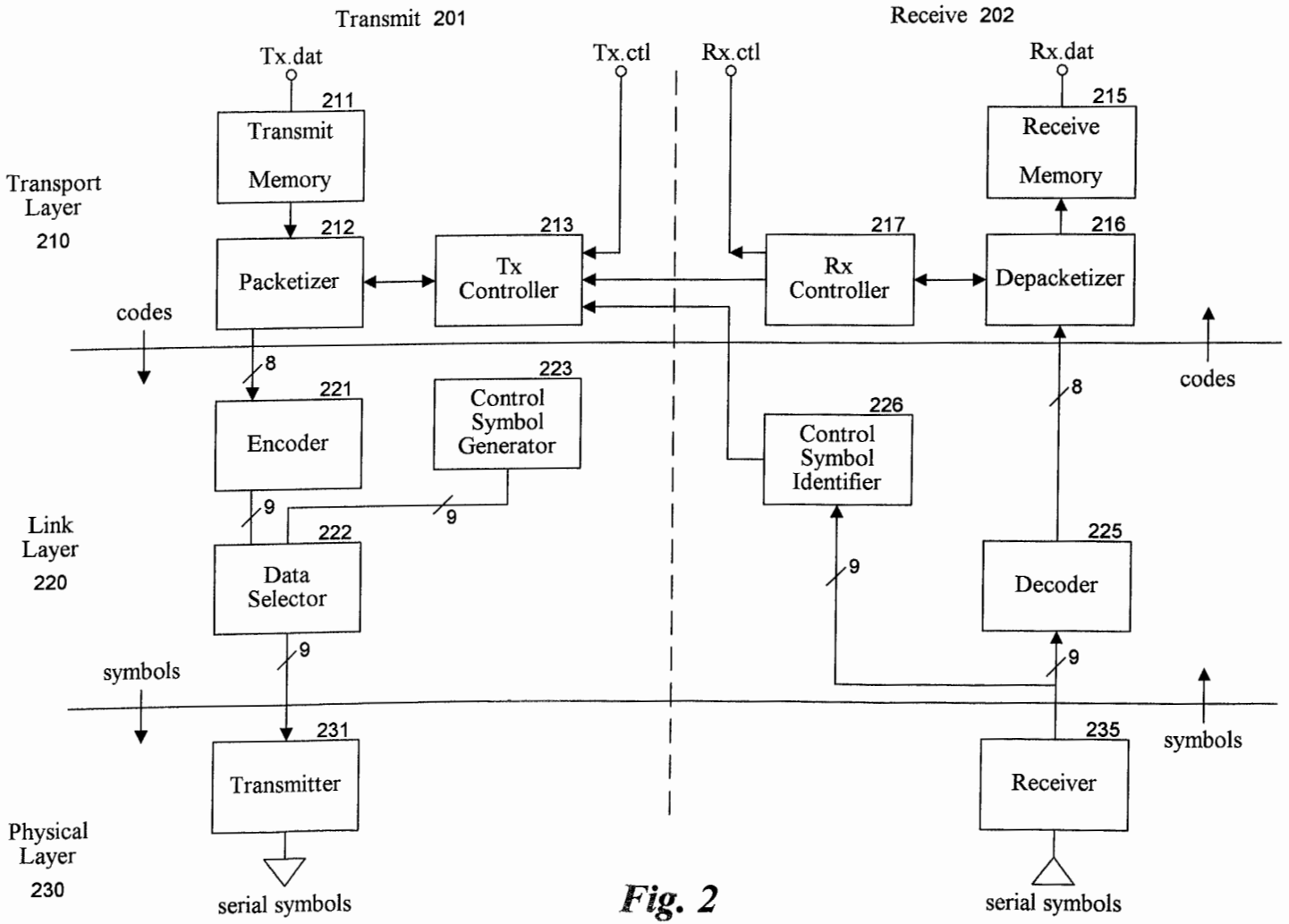


Fig. 2

20S090 TT6SE00T

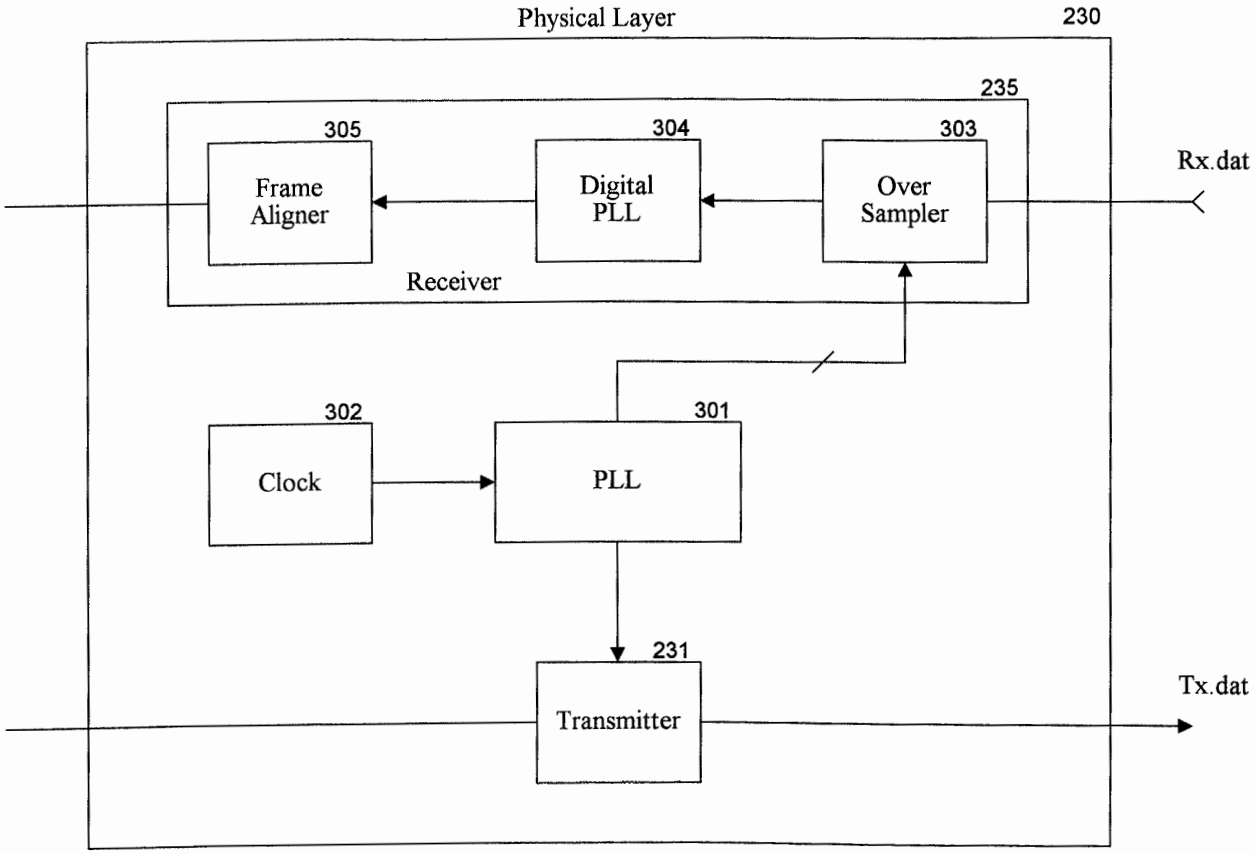


Fig. 3

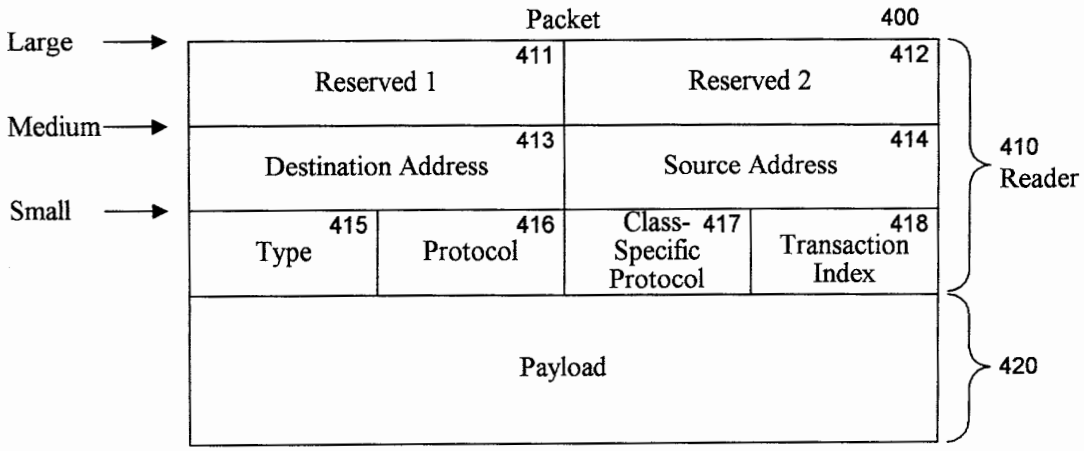


Fig. 4

P0035941.060502

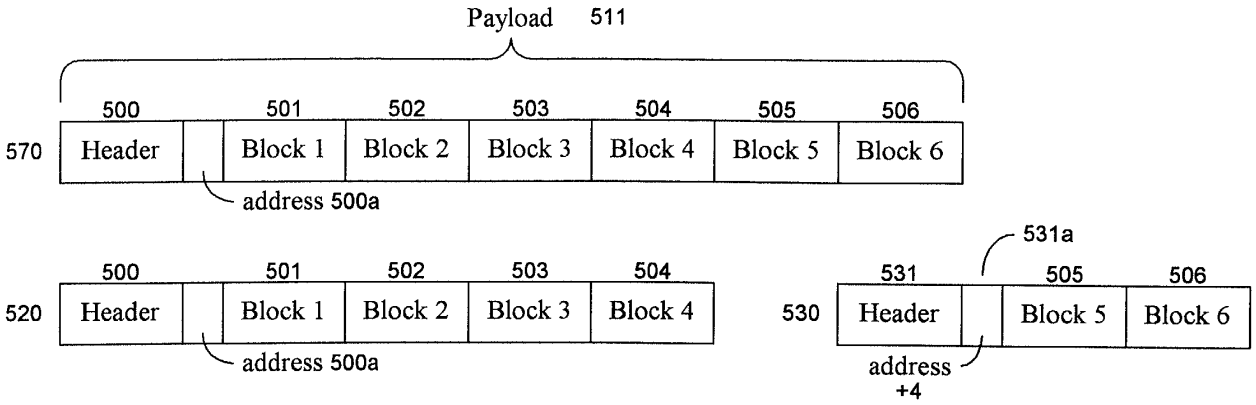

Payload 511
500
501
502
503
504
505
506
570
Header
Block 1
Block 2
Block 3
Block 4
Block 5
Block 6
address 500a
500
501
502
503
504
520
Header
Block 1
Block 2
Block 3
Block 4
address 500a
531a
531
505
506
530
Header
Block 5
Block 6
address
+4

Fig. 5

205090 "T69E00T

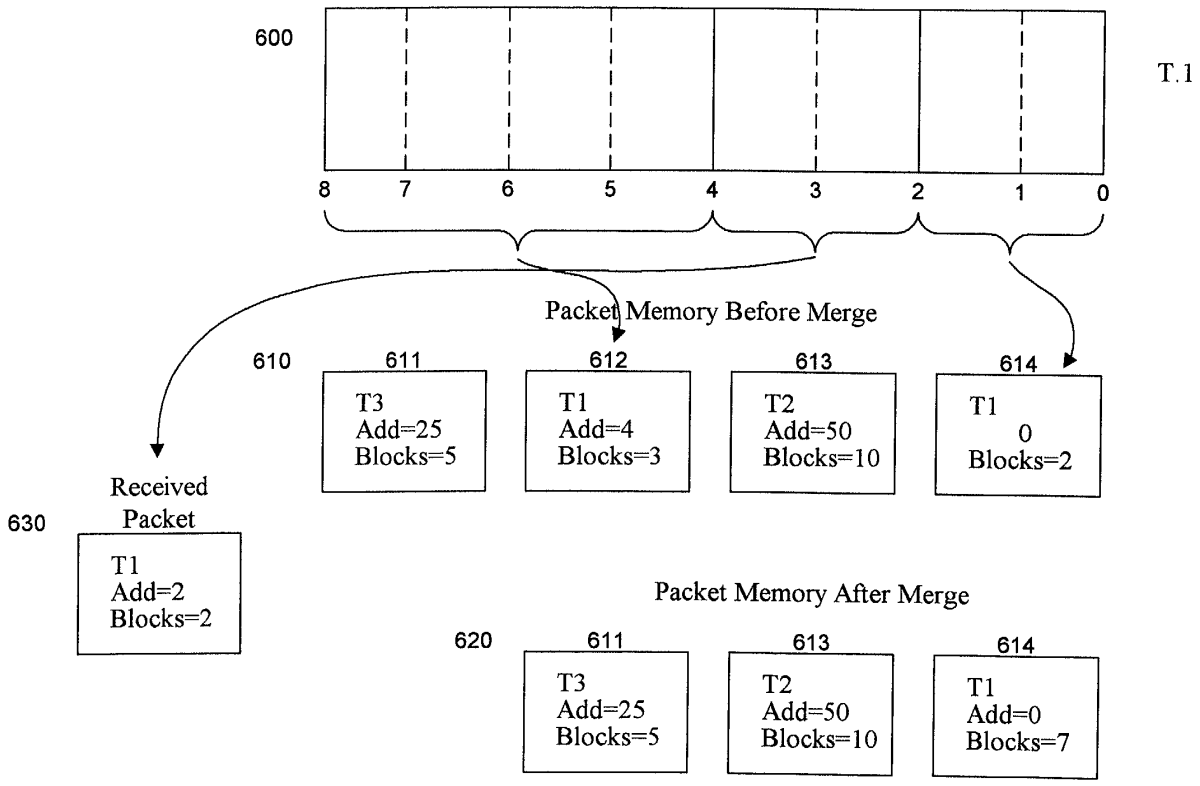


Fig. 6

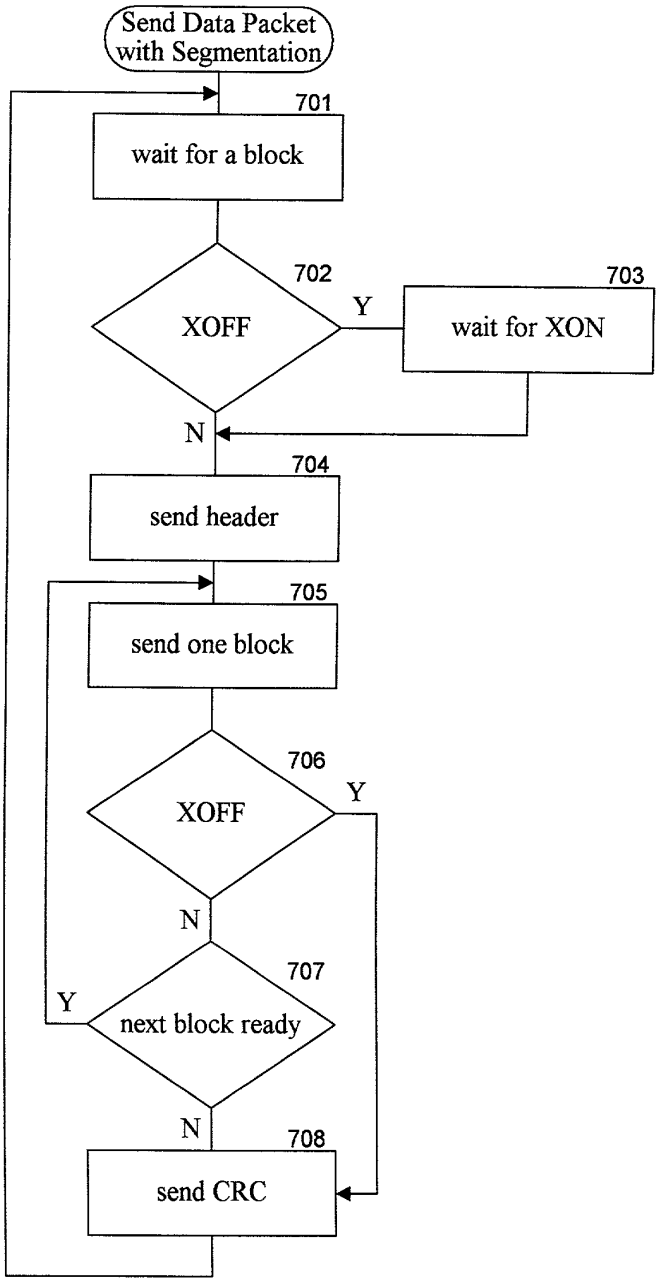


Fig. 7

IPR2024-001428
HP – EX1002, Page 294 of 487

EXHIBIT 1005, PAGE 294 OF 487

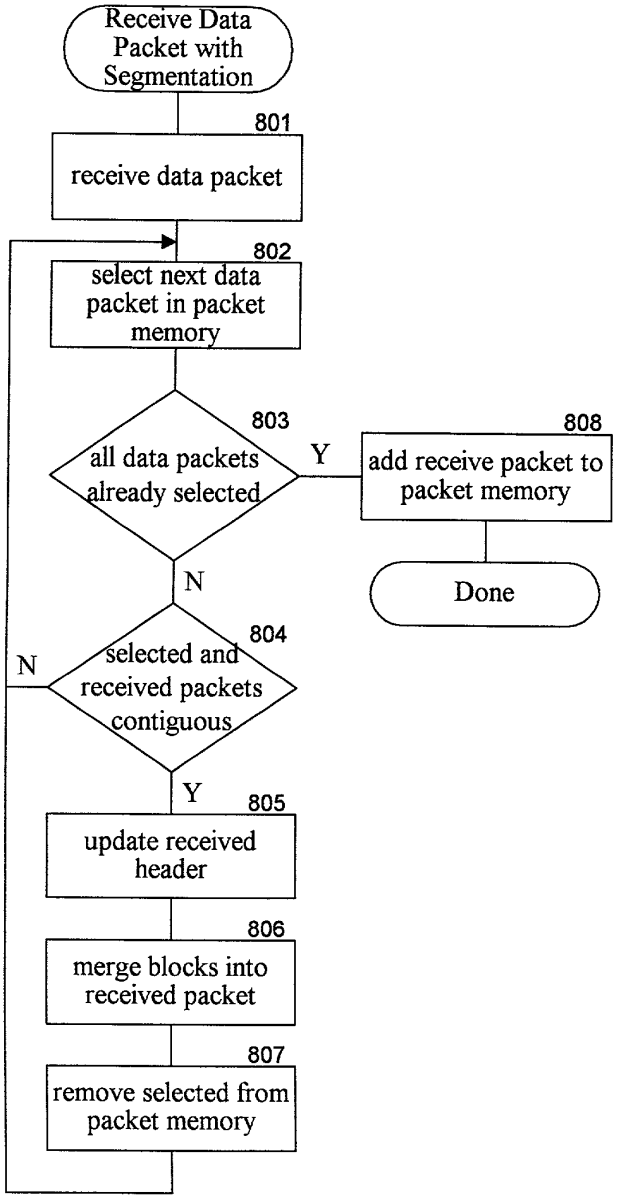
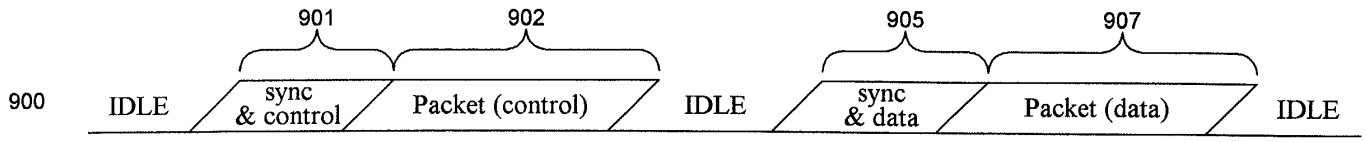


Fig. 8

205090 "TT6SE00T



sync & packet type

Fig. 9A

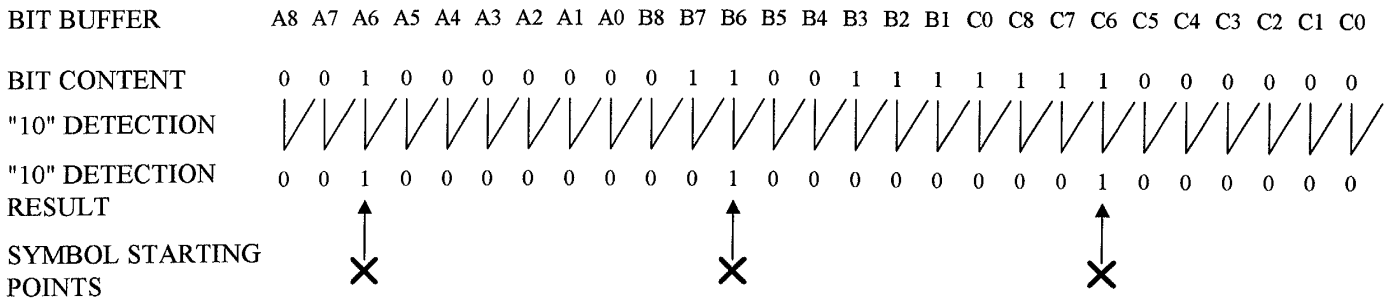

BIT BUFFER
A8 A7 A6 A5 A4 A3 A2 A1 A0 B8 B7 B6 B5 B4 B3 B2 B1 C0 C8 C7 C6 C5 C4 C3 C2 C1 C0
BIT CONTENT
0 0 1 0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0
"10" DETECTION
"10" DETECTION RESULT
0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
SYMBOL STARTING POINTS

Fig. 9B

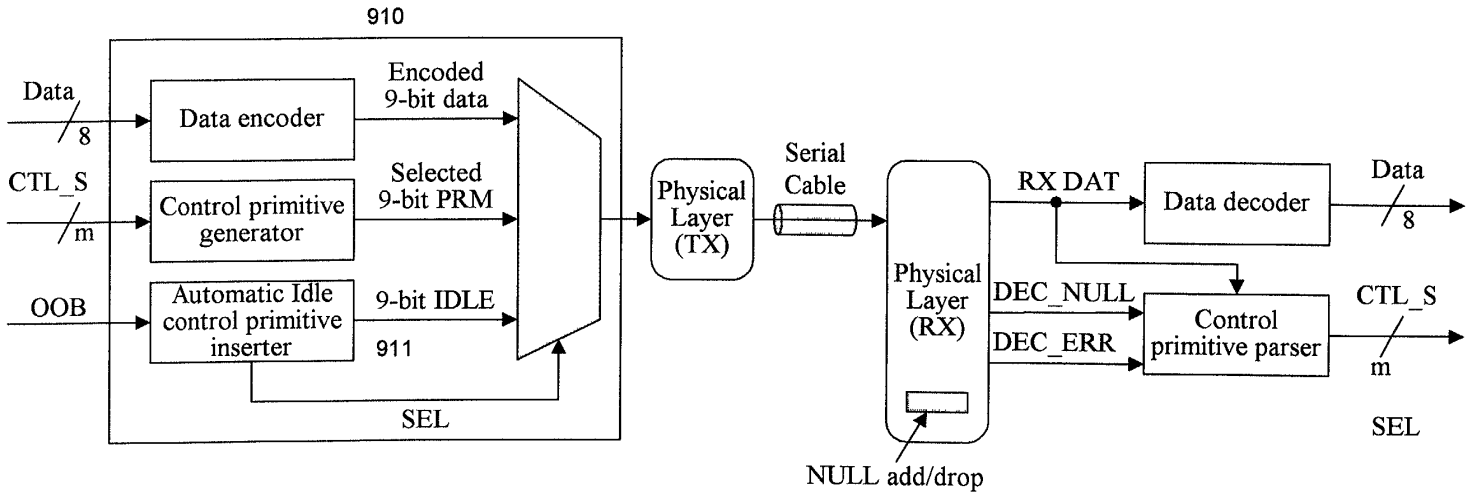

910
Data
8
CTL_S
m
OOB
Data encoder
Control primitive generator
Automatic Idle control primitive inserter
Encoded 9-bit data
Selected 9-bit PRM
9-bit IDLE
911
SEL
Physical Layer (TX)
Serial Cable
Physical Layer (RX)
NULL add/drop
RX DAT
DEC_NULL
DEC_ERR
Data decoder
Control primitive parser
Data
8
CTL_S
m
SEL

Fig. 9C

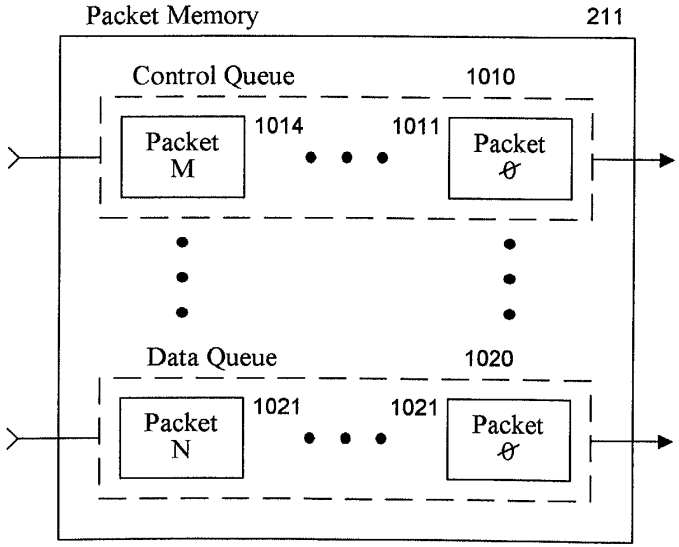

Packet Memory
211
Control Queue
1010
Packet M
1014
1011
Packet 0
Data Queue
1020
Packet N
1021
1021
Packet 0

Fig. 10

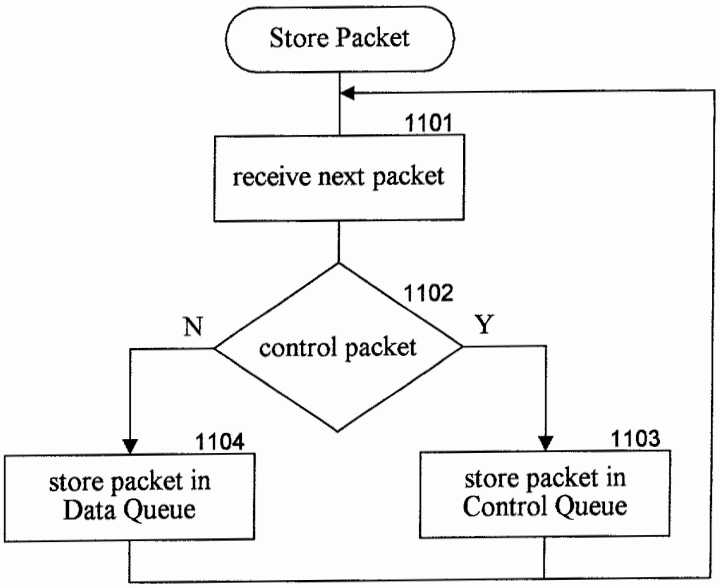


Fig. 11

Docket No. 10030644-1

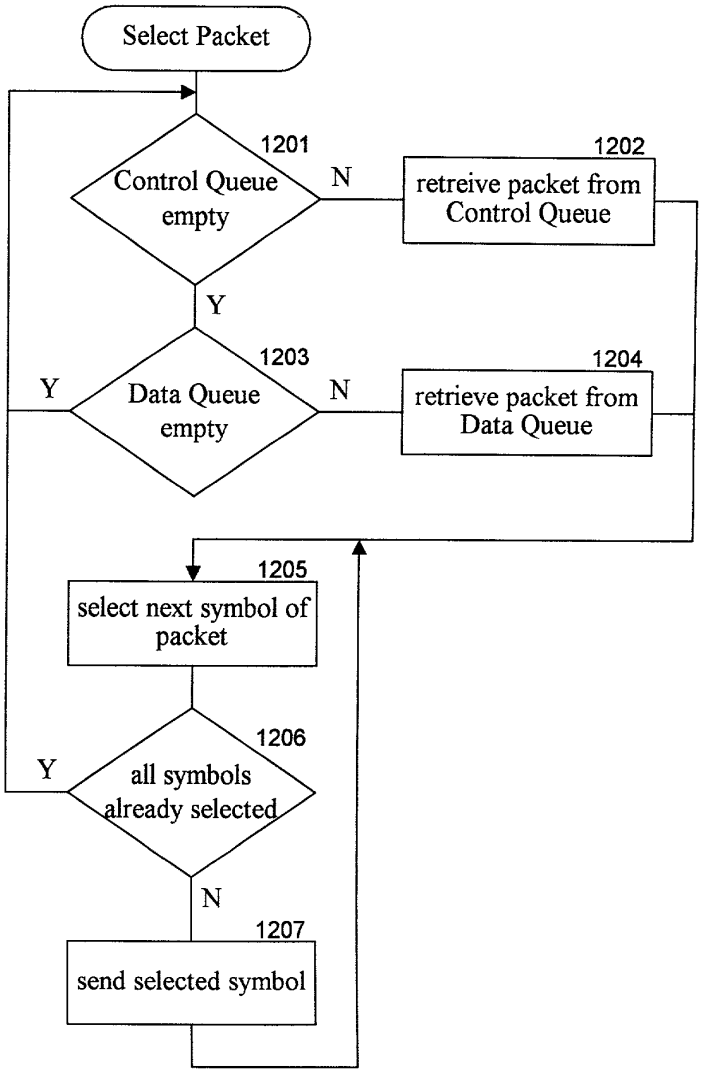


Fig. 12

IPR2024-001428
HP – EX1002, Page 301 of 487

2050901 TT6SE00T

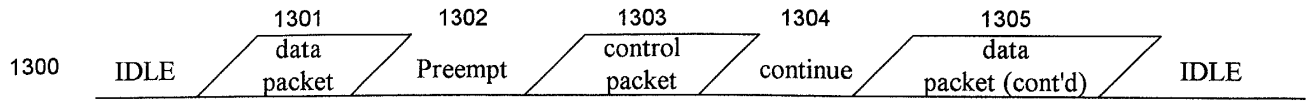


Fig. 13

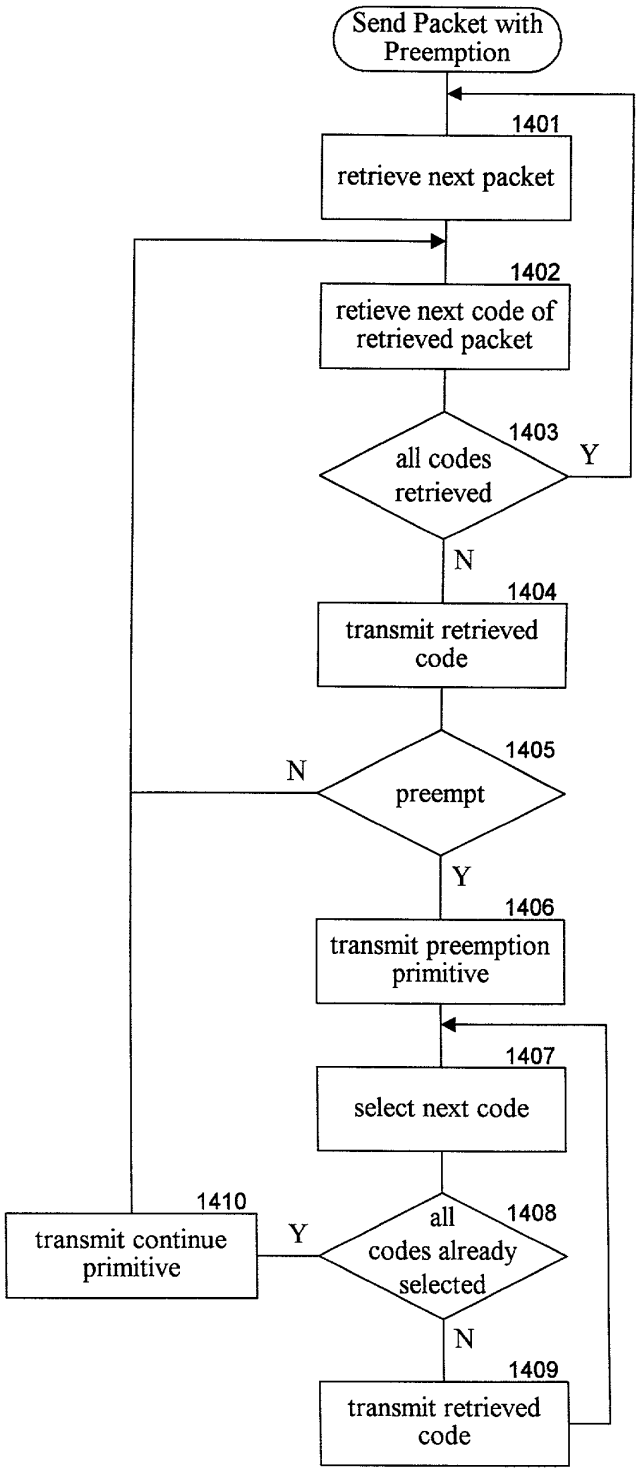


Fig. 14

20030502-1P-050502

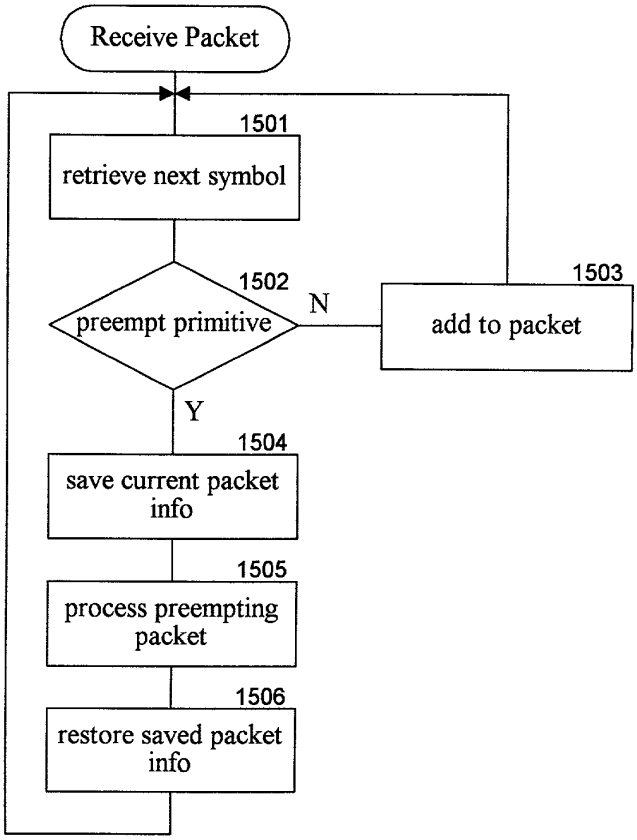


Fig. 15

P0035pt-060502

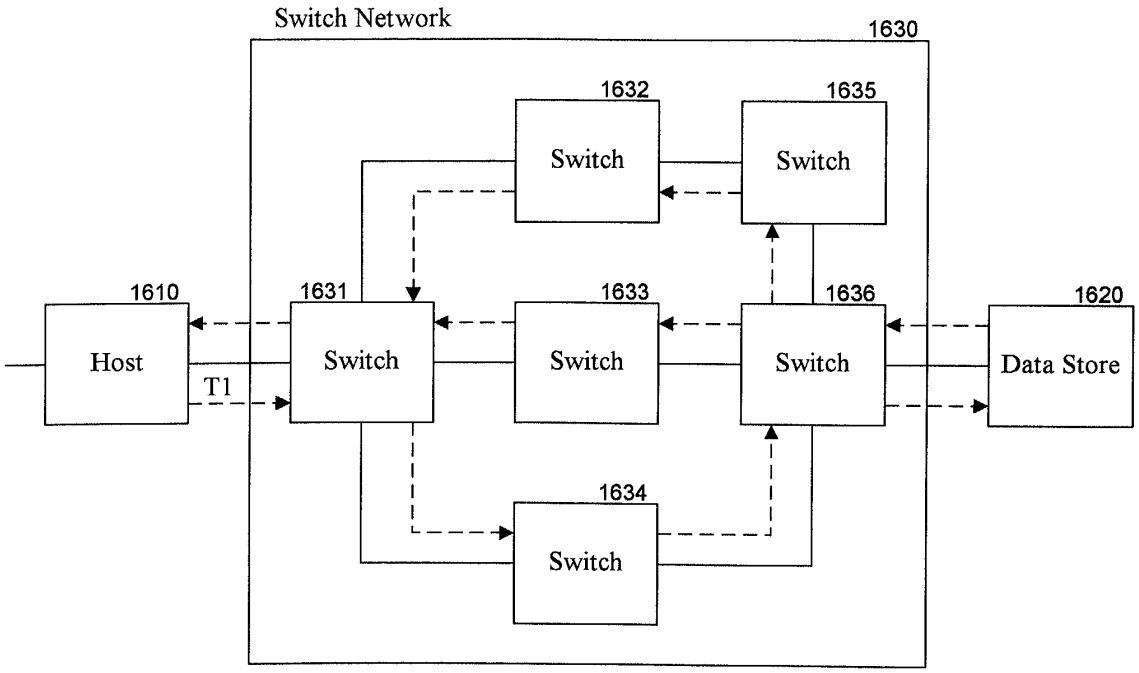

Switch Network
1630
1632
1635
Switch
Switch
1610
1631
1633
1636
1620
Host
T1
Switch
Switch
Switch
Data Store
1634
Switch

Fig. 16

20S090 TT6S200T

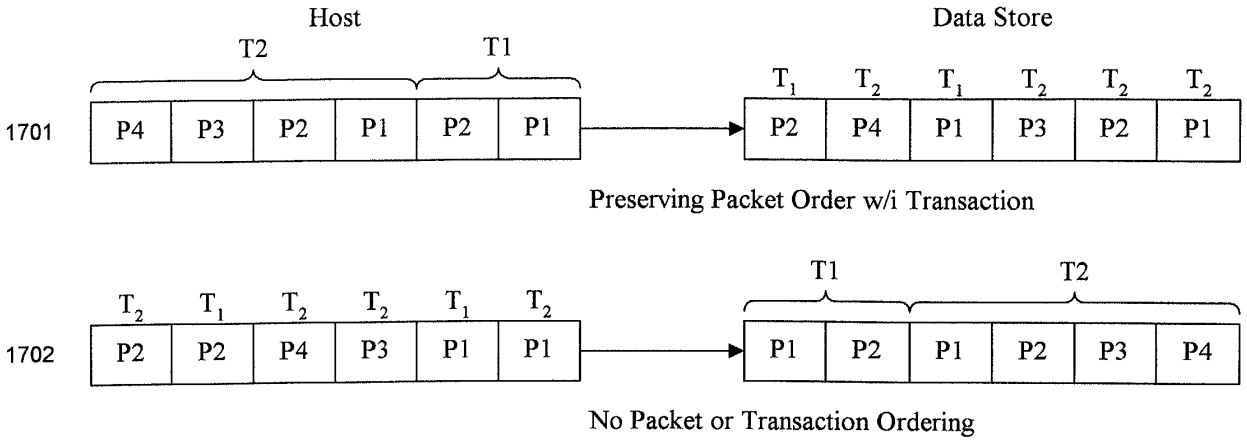


Fig. 17

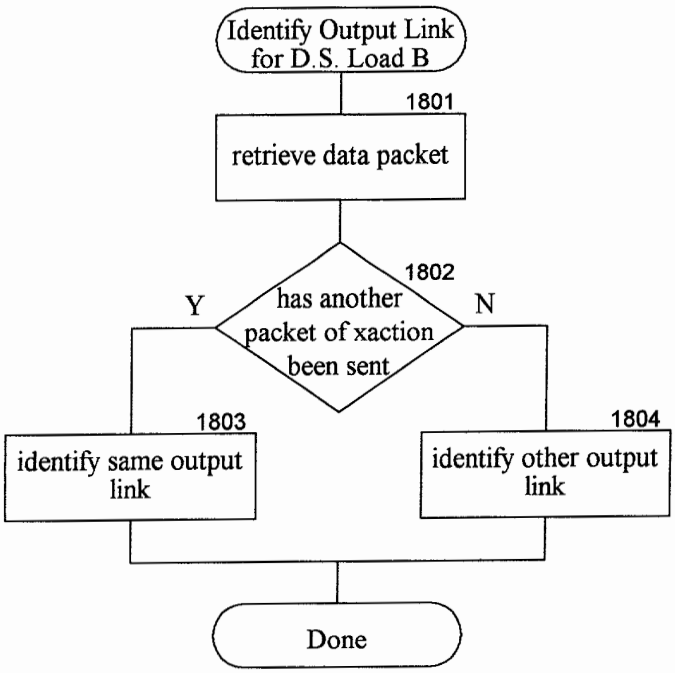


Fig. 18

HP 3505-4 .0505.2

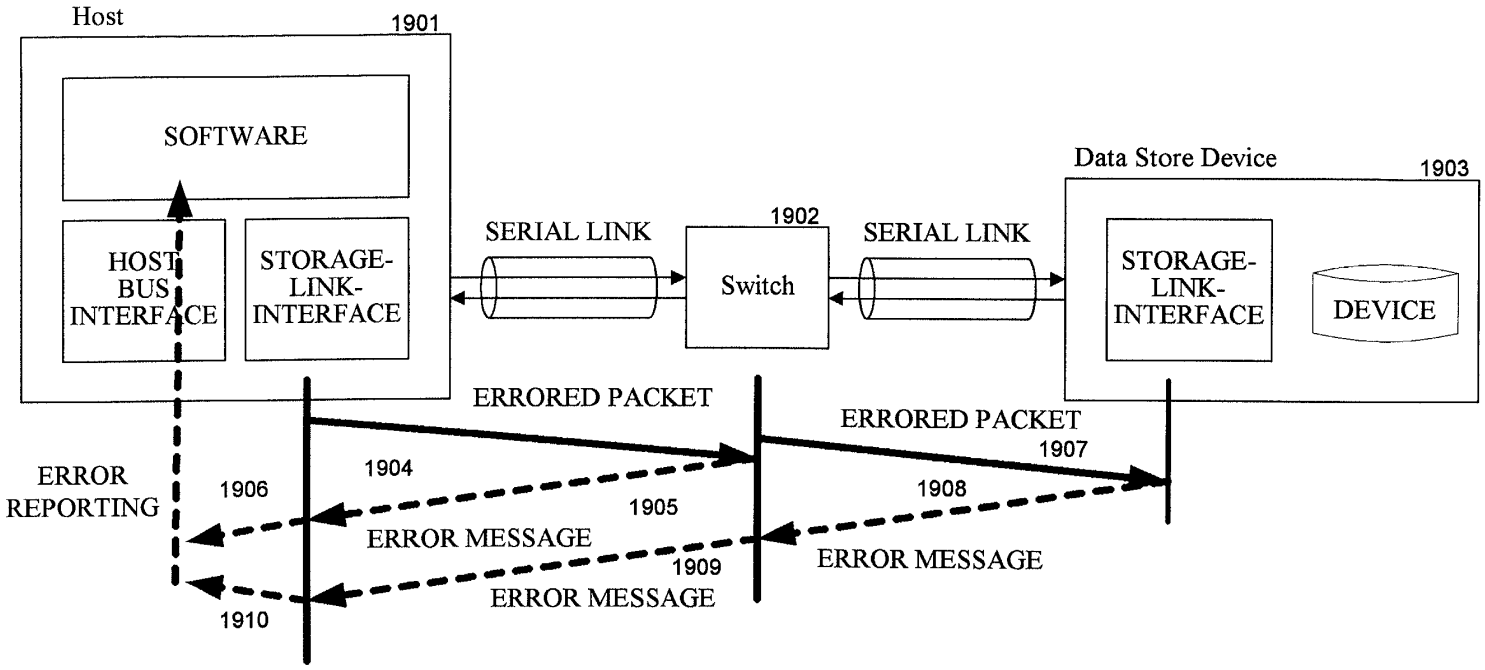

Host
1901
SOFTWARE
HOST BUS INTERFACE
STORAGE-LINK-INTERFACE
SERIAL LINK
1902
Switch
SERIAL LINK
Data Store Device
1903
STORAGE-LINK-INTERFACE
DEVICE
ERRORED PACKET
ERRORED PACKET
1907
ERROR REPORTING
1906
1904
1905
1908
ERROR MESSAGE
ERROR MESSAGE
1909
ERROR MESSAGE
1910

Fig. 19A

205090 "TT6SE00T

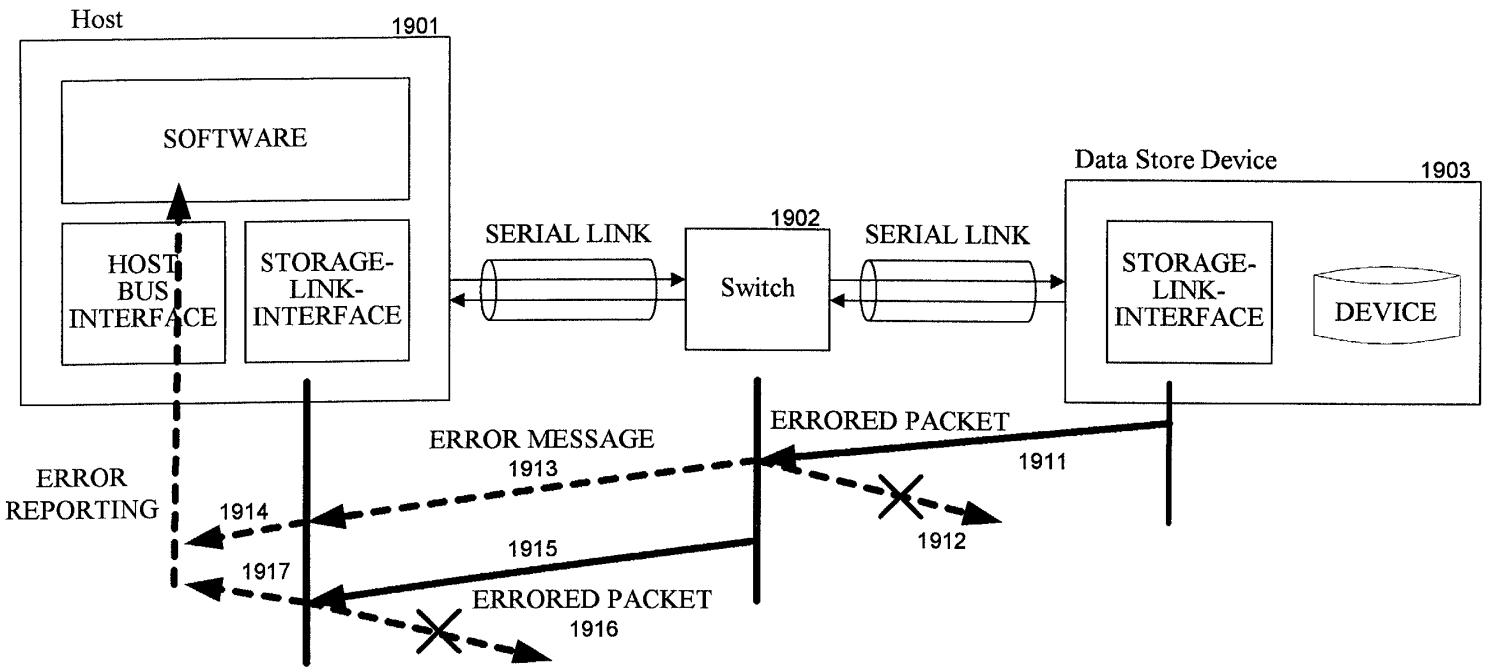


Fig. 19B

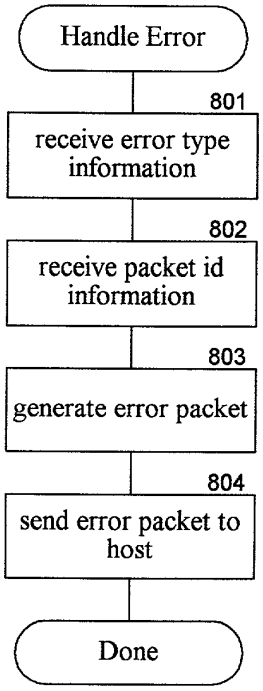


Fig. 19C

20050502.4956.0004

8b code	9 bit symbol
0000 0000	101010101
0000 0001	101010100
0000 0010	101010111
⋮	
0101 0101	001010101
⋮	
0111 0110	001110110
0111 0111	100100010
⋮	
1111 1111	110101010

Fig. 20

20050502.1760504

20090 TT6S60T

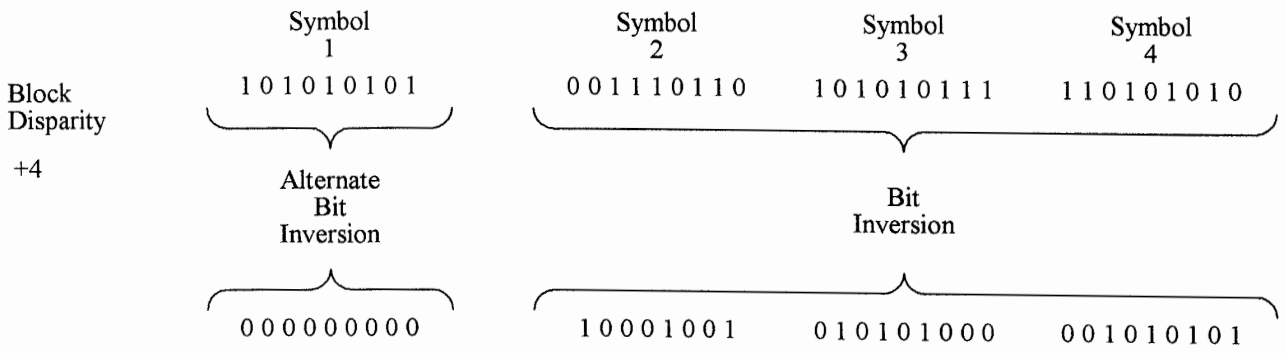


Fig. 21A

20S090 TT6SE00T

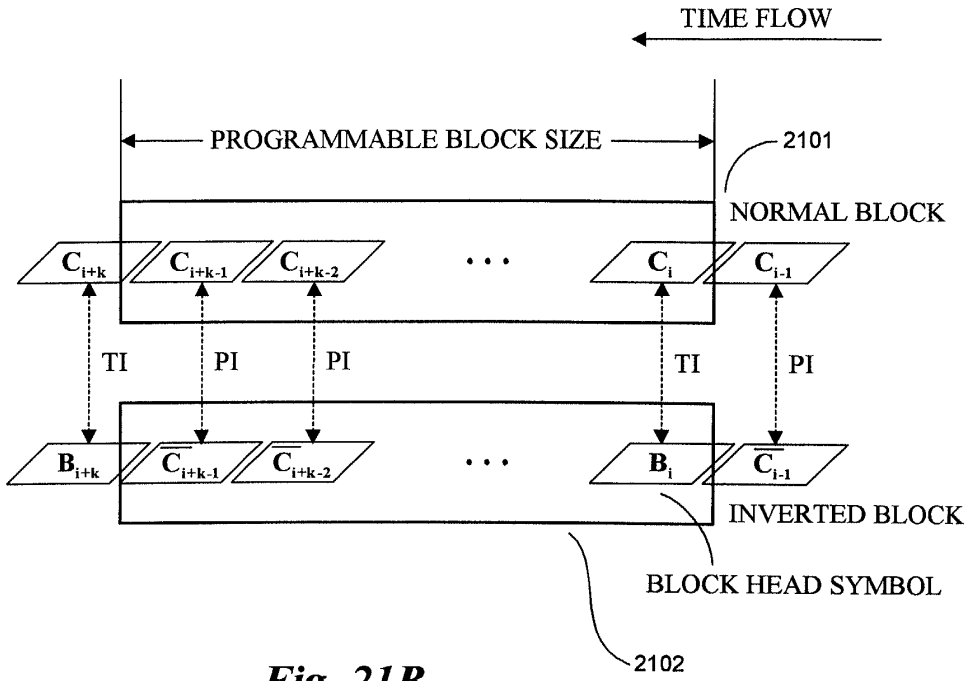


Fig. 21B

20090 TT6SE00T

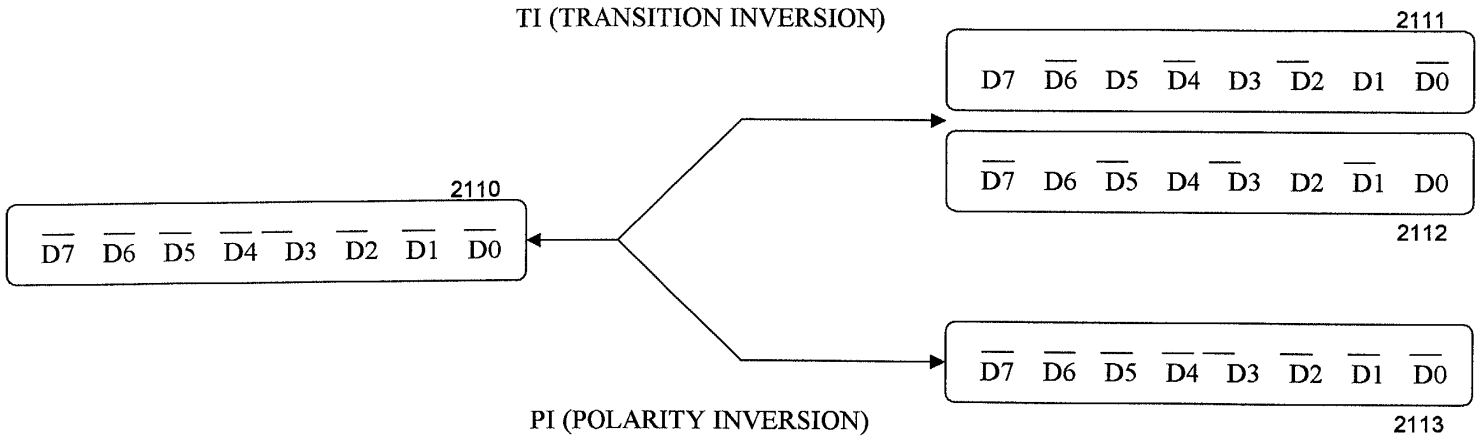


Fig. 21C

20S090 TT6SE00T

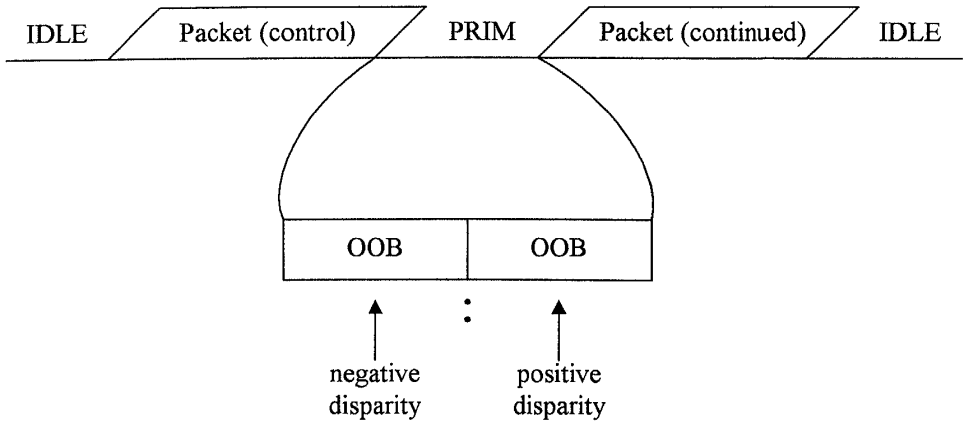


Fig. 22

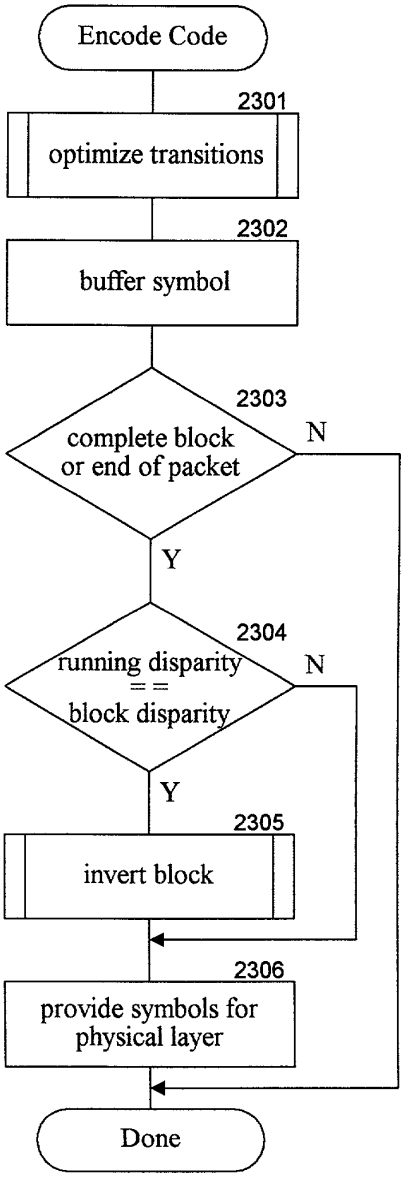


Fig. 23

20050502 - 4005994

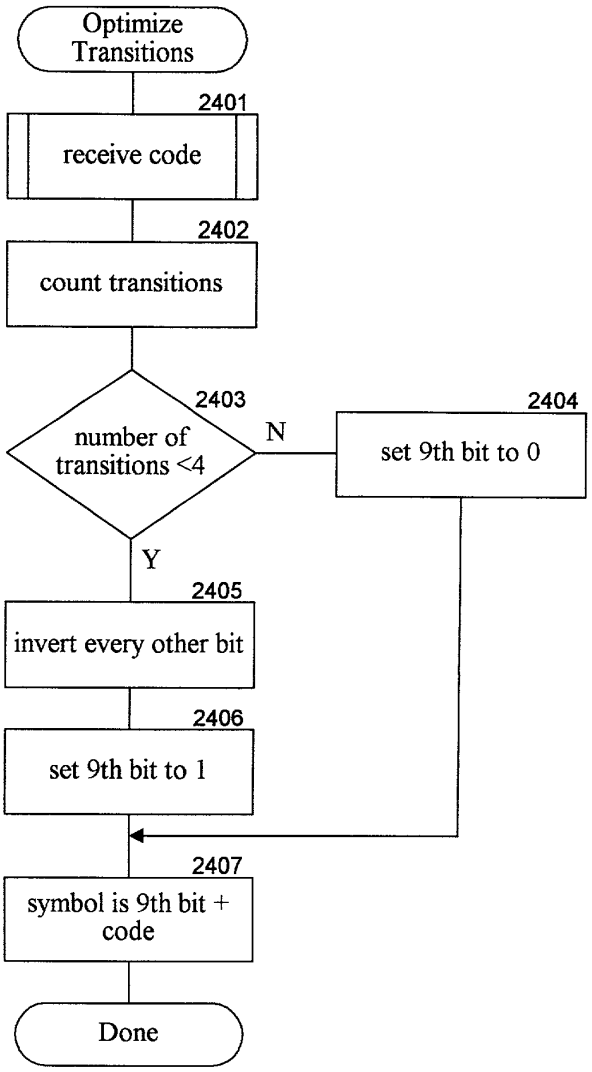


Fig. 24

20050502.4.0050.00

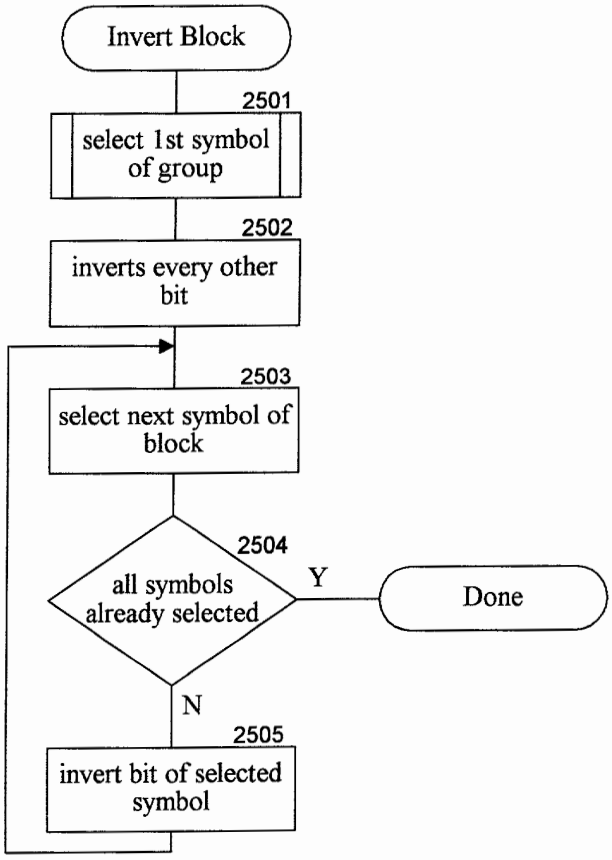


Fig. 25

HOUSE-0005502

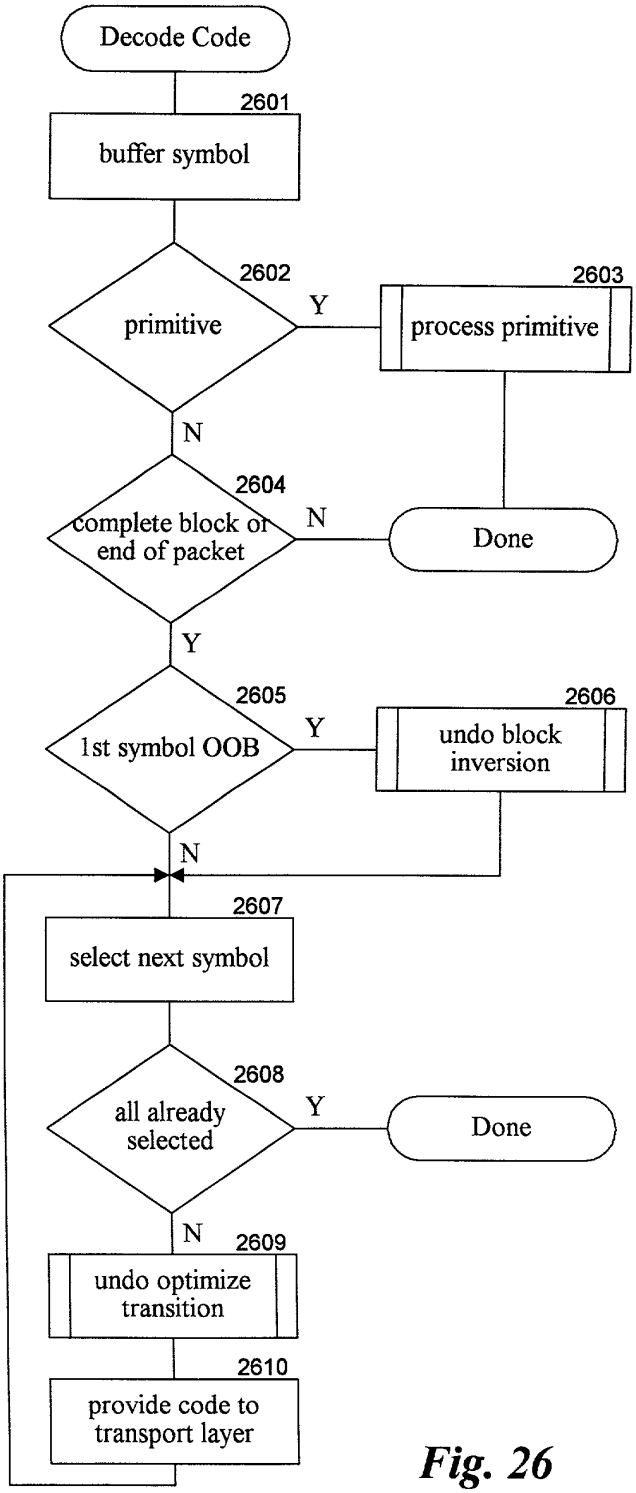


Fig. 26

IPR2024-001428
HP – EX1002, Page 319 of 487

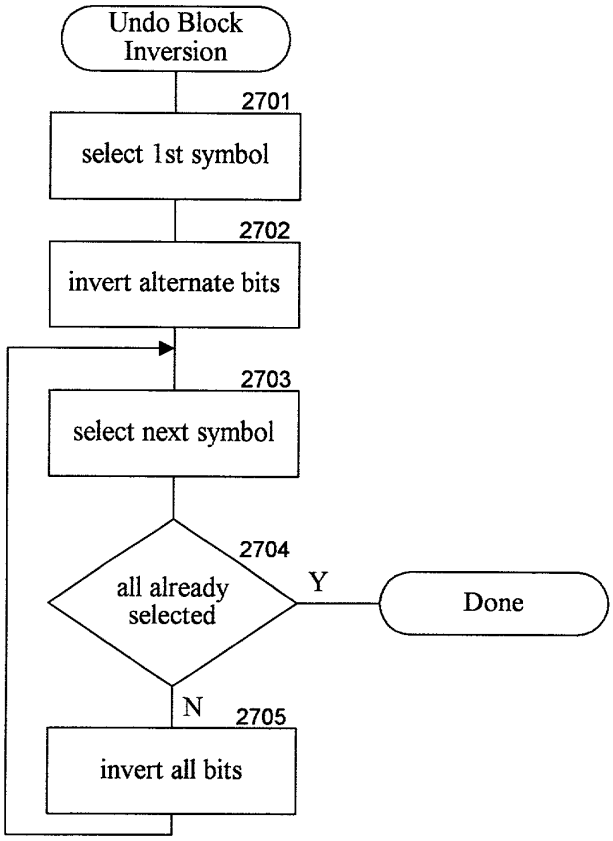


Fig. 27

20050502.1435.0054

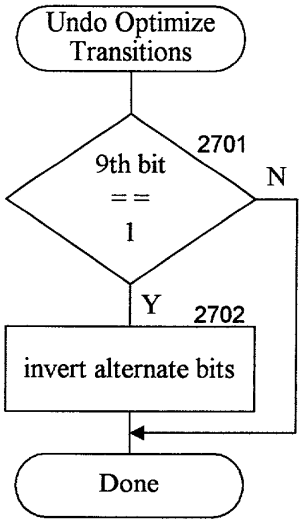


Fig. 28

20030194-060502

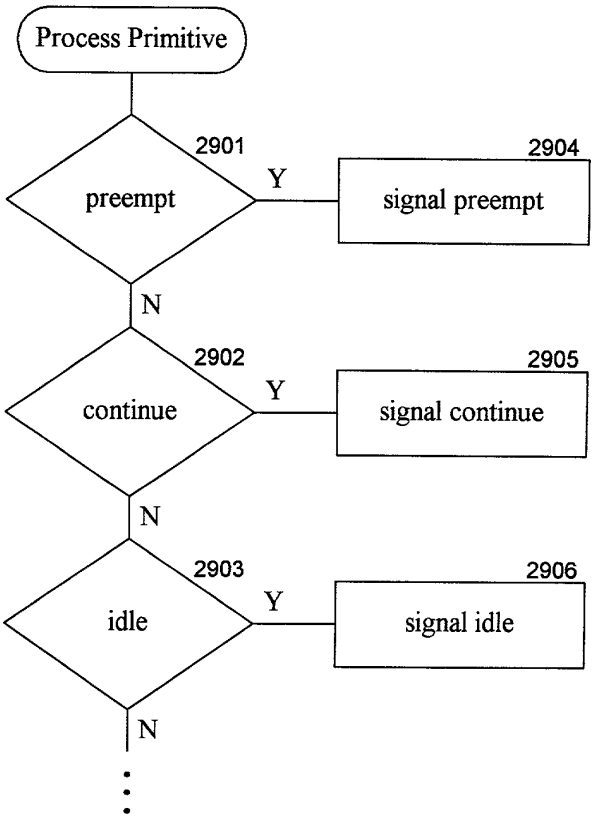


Fig. 29

3055.4P.005502

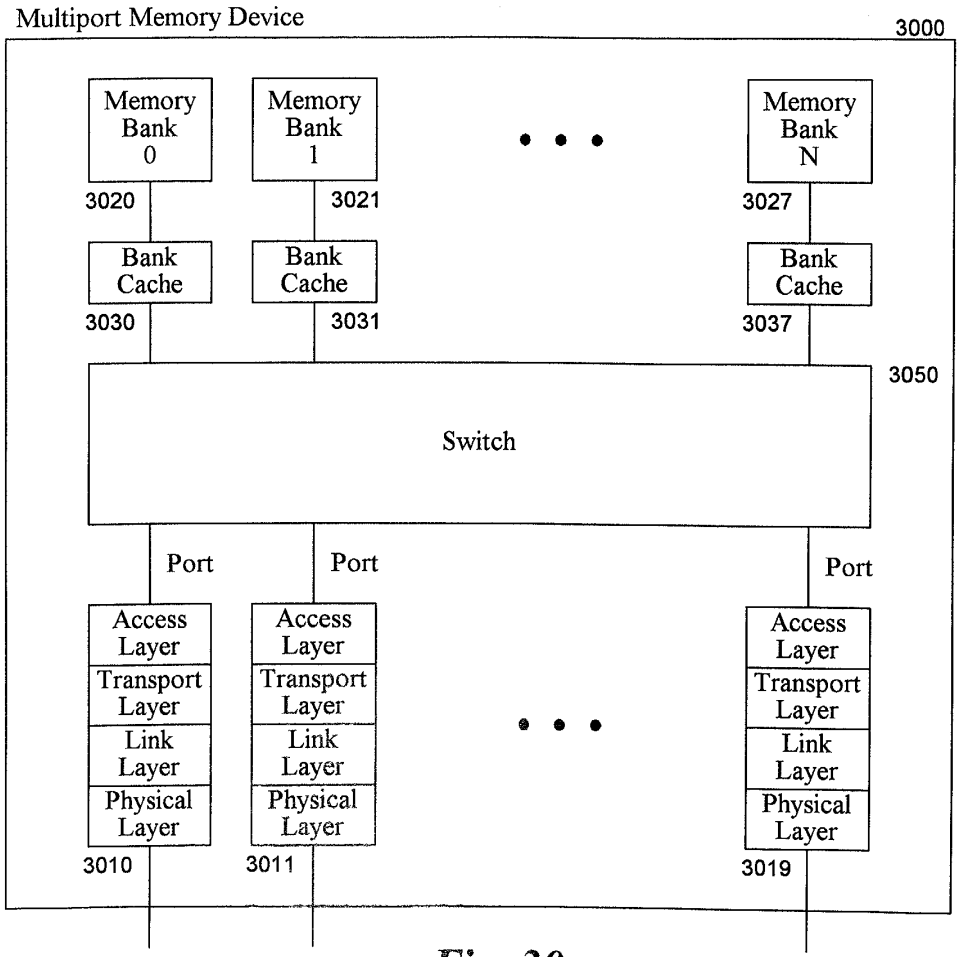

Multiport Memory Device
3000
Memory Bank 0
Memory Bank 1
Memory Bank N
3020
3021
3027
Bank Cache
Bank Cache
Bank Cache
3030
3031
3037
3050
Switch
Port
Port
Port
Access Layer
Transport Layer
Link Layer
Physical Layer
Access Layer
Transport Layer
Link Layer
Physical Layer
Access Layer
Transport Layer
Link Layer
Physical Layer
3010
3011
3019

Fig. 30

20S090 TT6SE00F

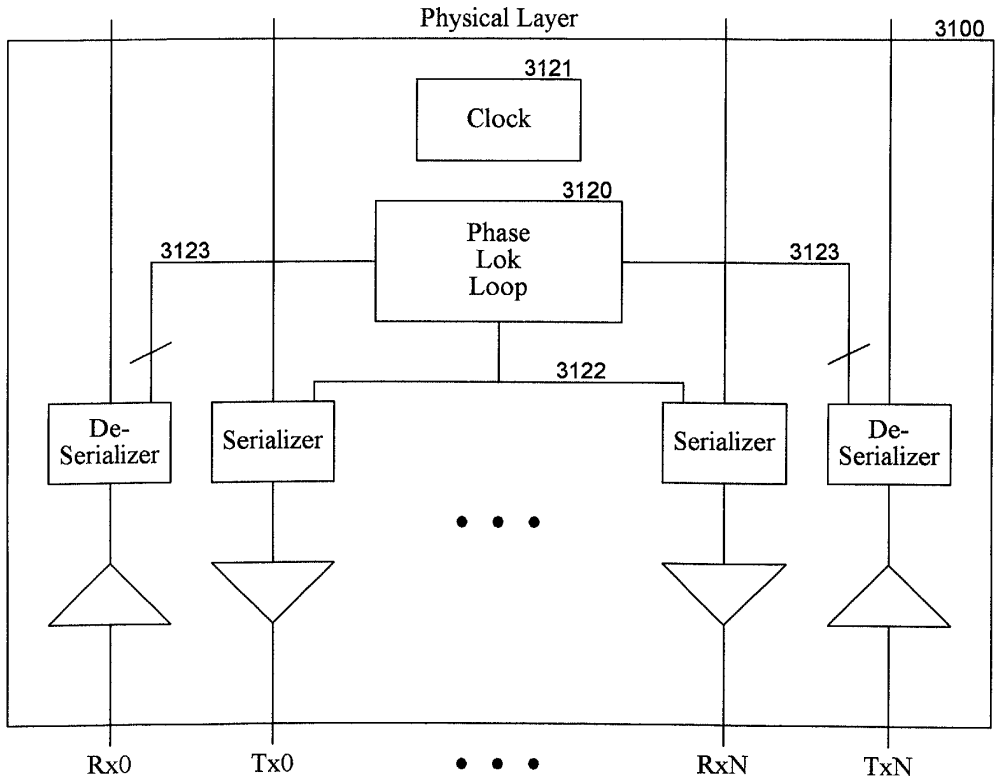


Fig. 31

Input Queue 3201

Port	R/W	Address	Data
3	R	1000	
4	W	4000	10....1
3	W	1000	111...0
3	R	2000	
	⋮		

Output Queue 3202

Valid	Port	Data
1	3	11...0
0		
0		
1	3	101...1
	⋮	

Fig. 32

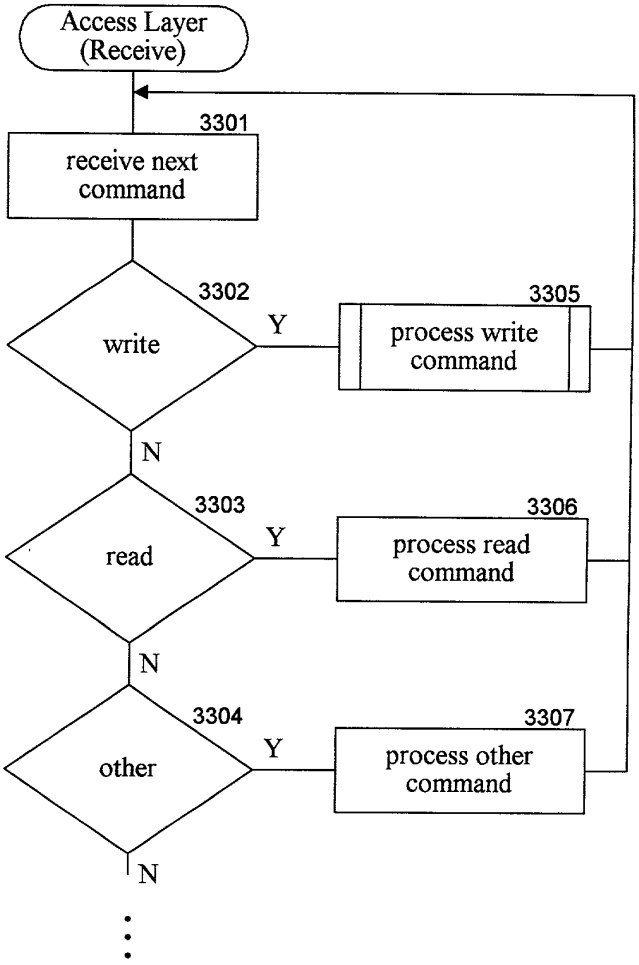


Fig. 33

HP 3PAR-00502

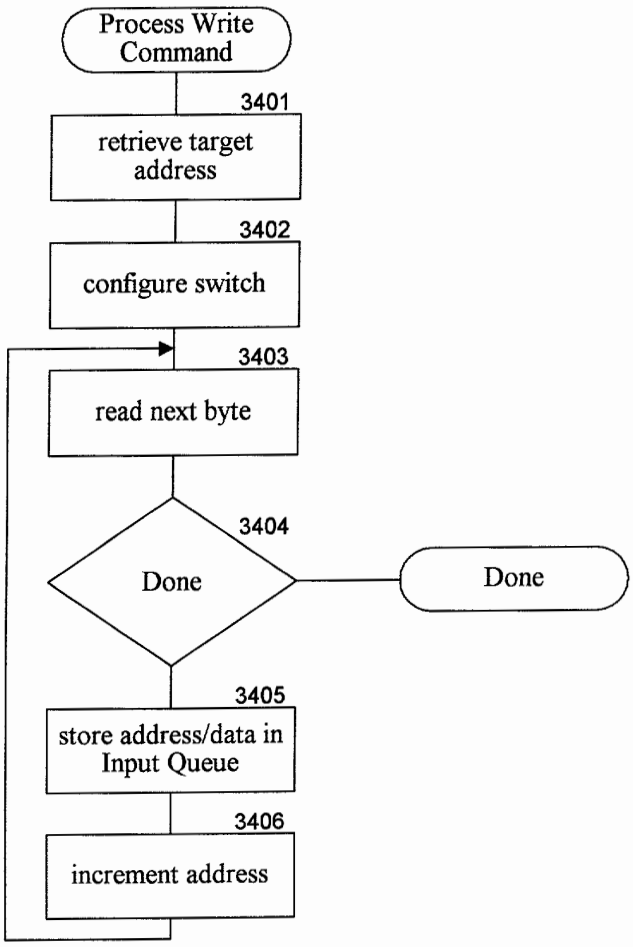


Fig. 34

3405-4 3405-5 3405-6

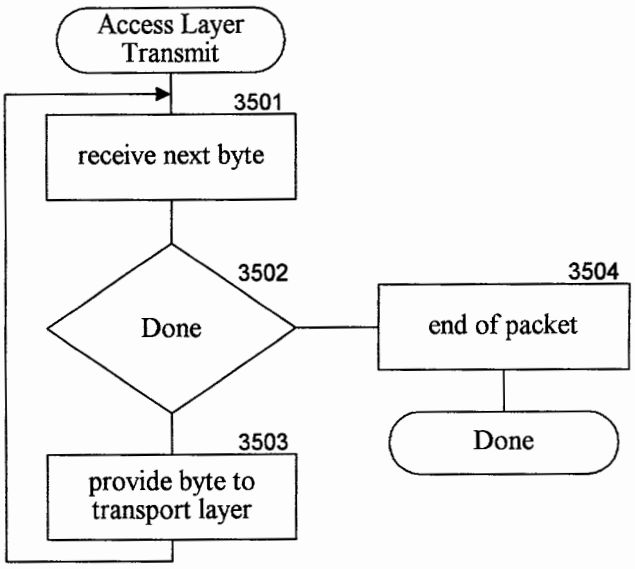


Fig. 35

2650502-4450502

20S090 TT6SE00T

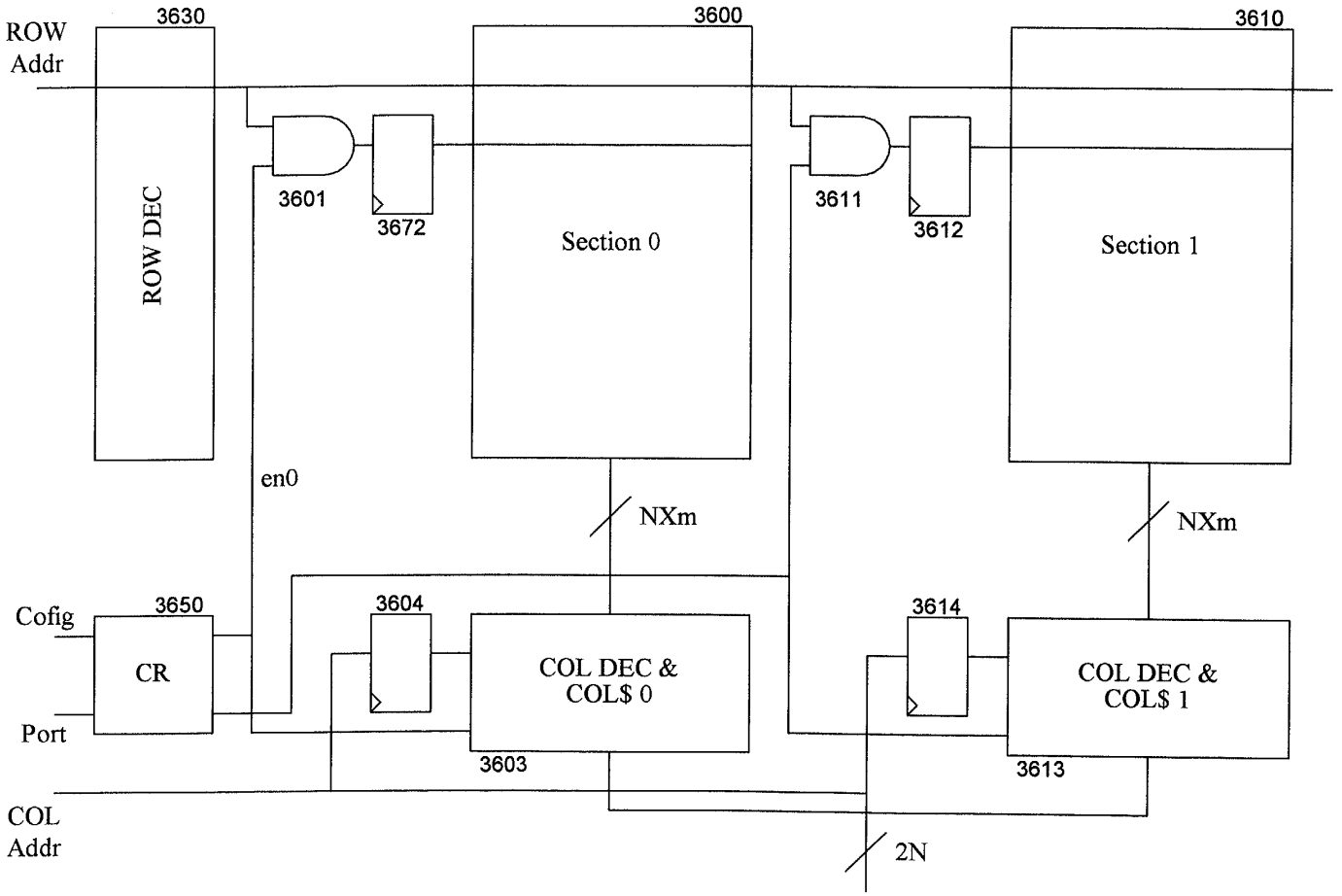


Fig. 36

205090 "TT6SE00T

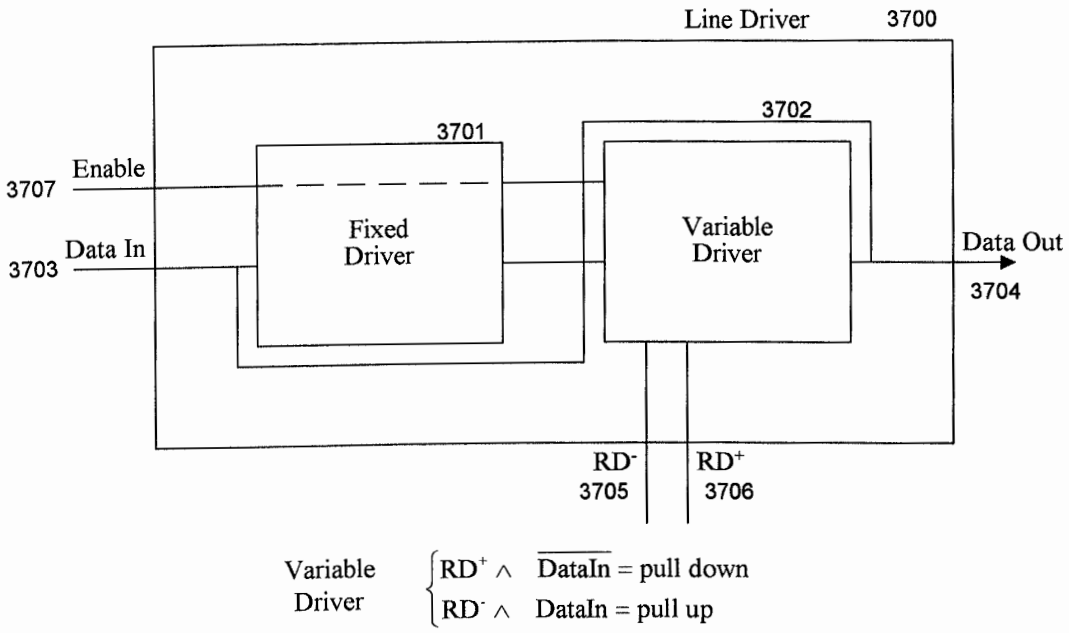


Fig. 37A

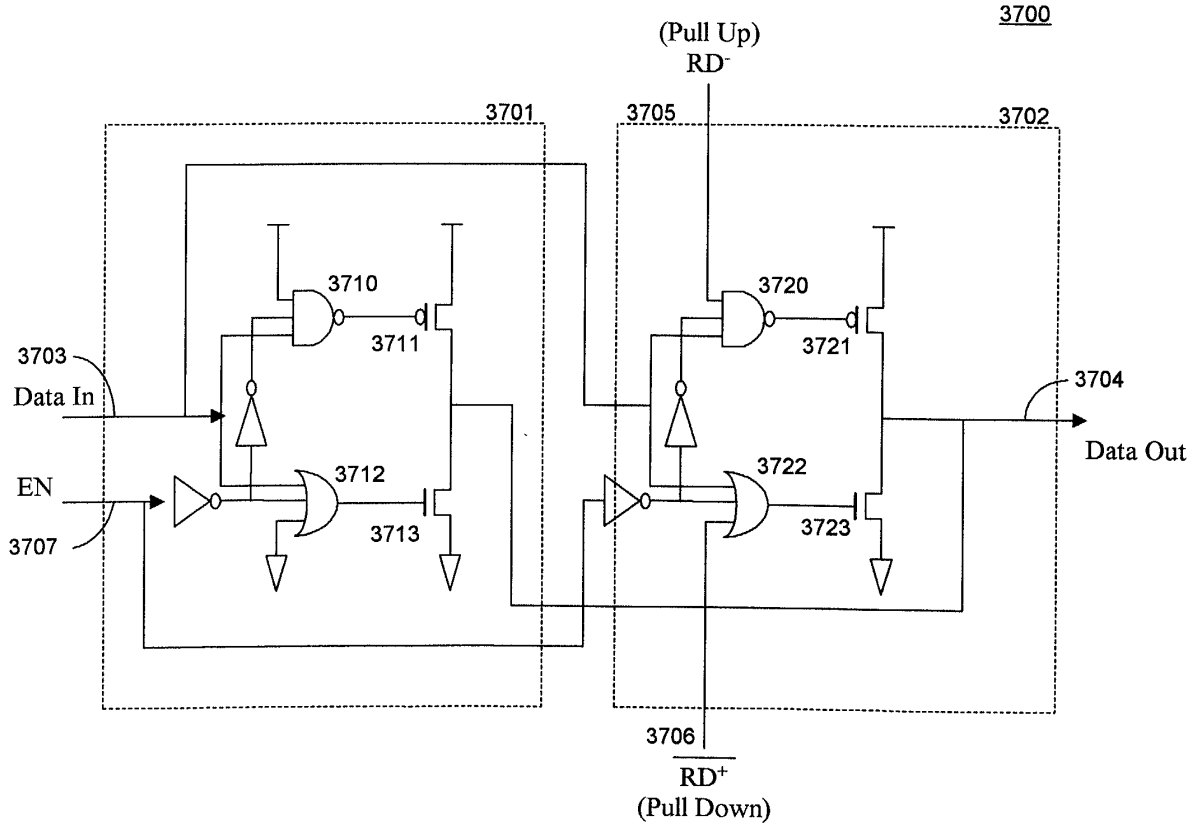

3700
(Pull Up)
RD-
3701
3705
3702
3710
3711
3720
3721
3703
Data In
3704
Data Out
3712
3722
EN
3713
3723
3707
3706
RD+
(Pull Down)

Fig. 37B

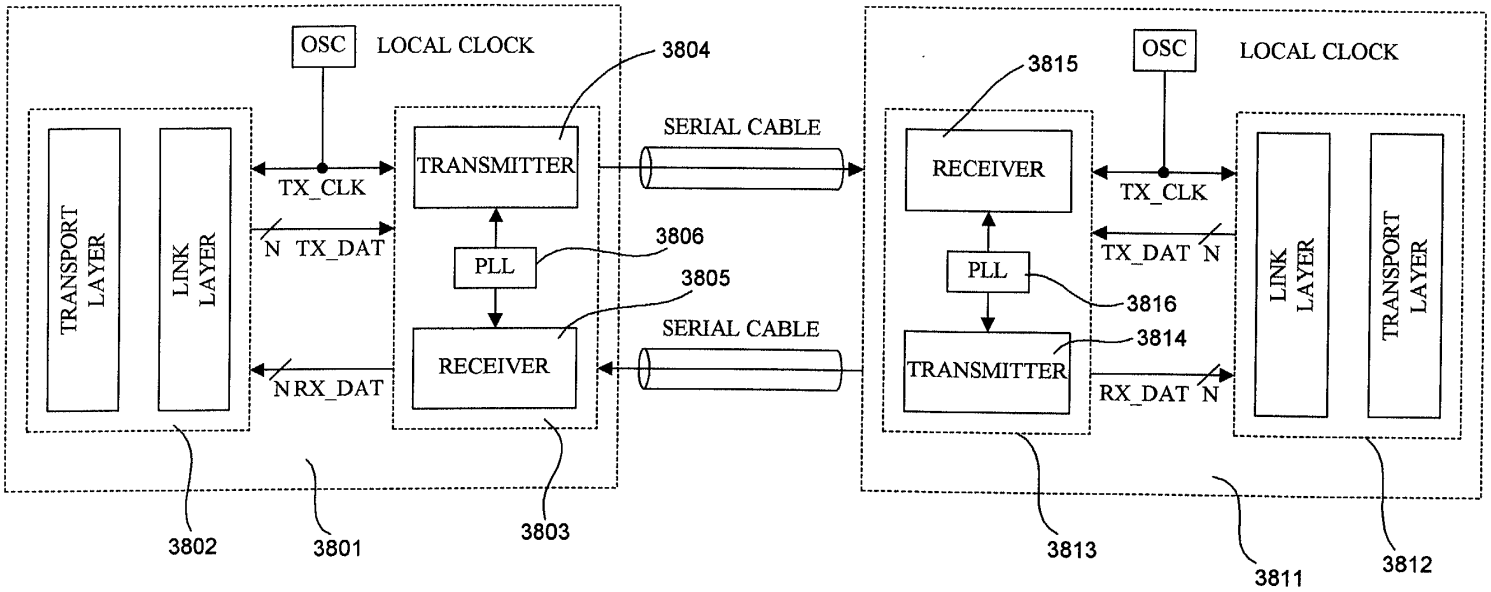

OSC
LOCAL CLOCK
3804
SERIAL CABLE
TRANSMITTER
TX_CLK
N TX_DAT
PLL
3806
3805
RECEIVER
NRX_DAT
SERIAL CABLE
TRANSPORT LAYER
LINK LAYER
3802
3801
3803
3815
OSC
LOCAL CLOCK
RECEIVER
TX_CLK
TX_DAT N
PLL
3816
3814
TRANSMITTER
RX_DAT N
LINK LAYER
TRANSPORT LAYER
3813
3811
3812

Fig. 38A

20S090 TT6SE00T

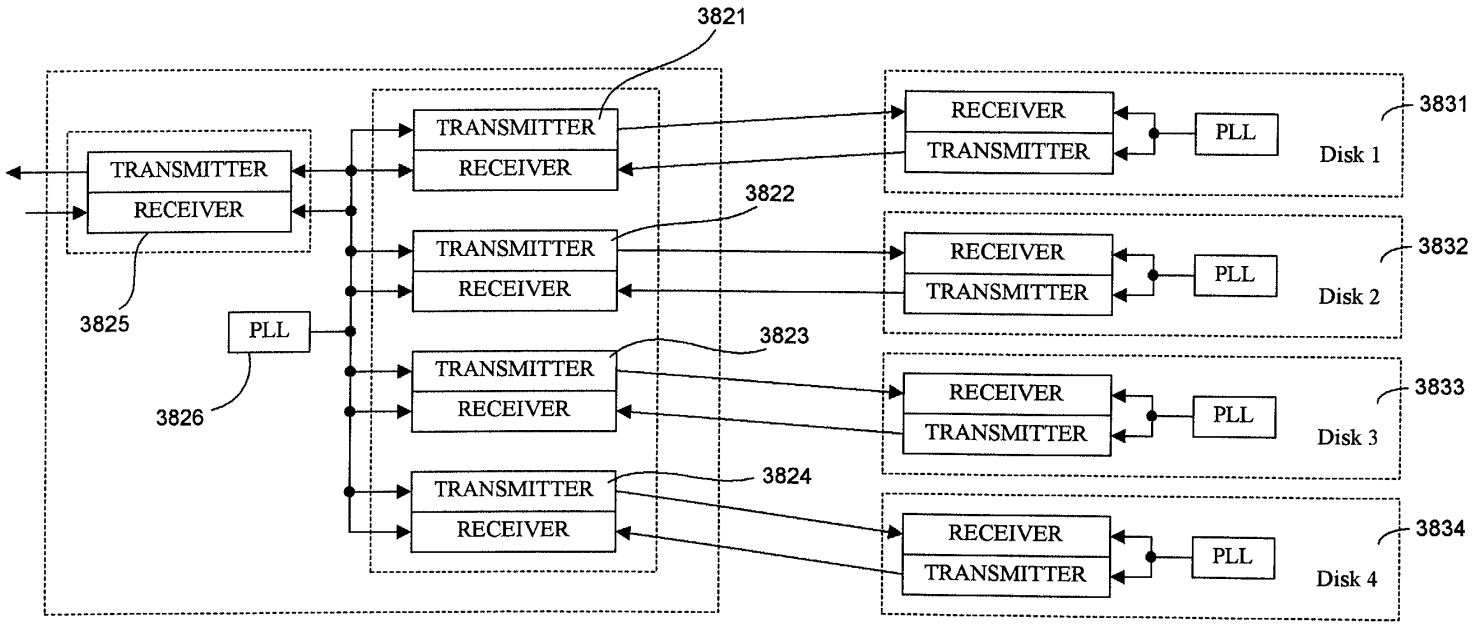


Fig. 38B

20S090 TT6SE00T

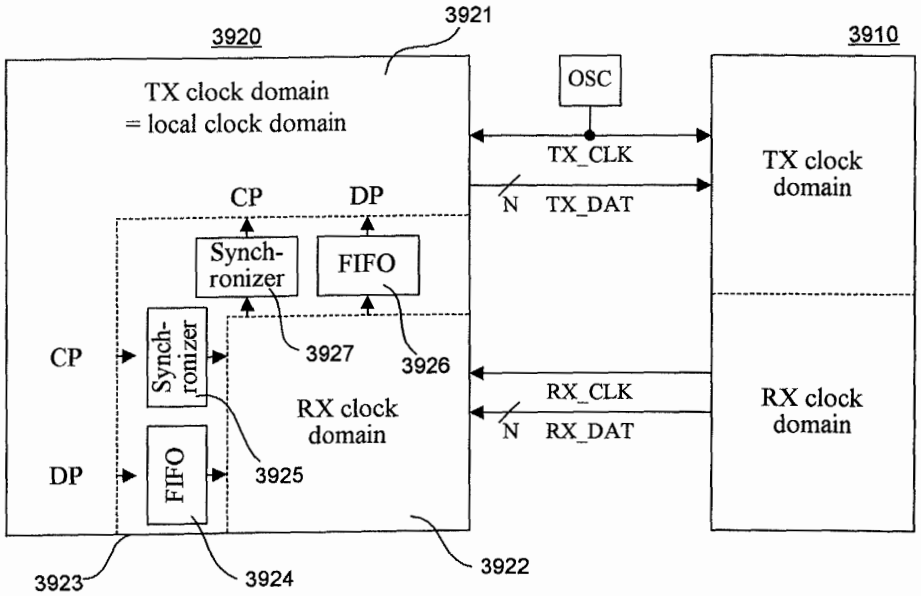


Fig. 39A

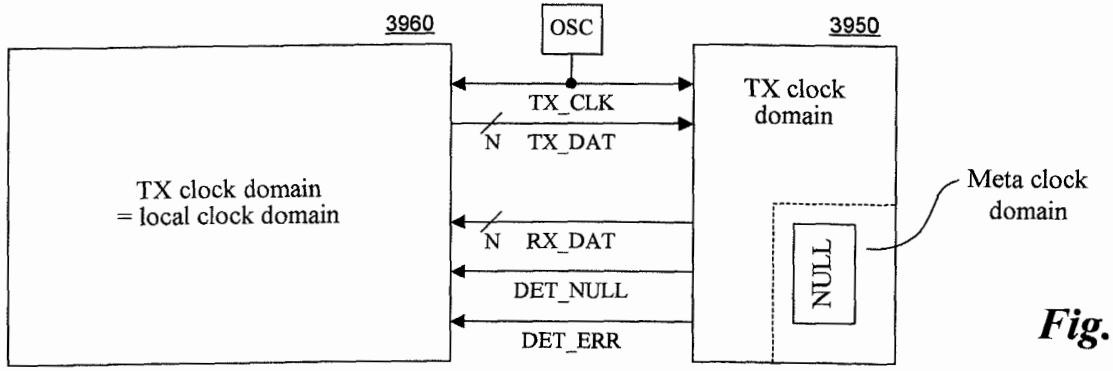


Fig. 39B

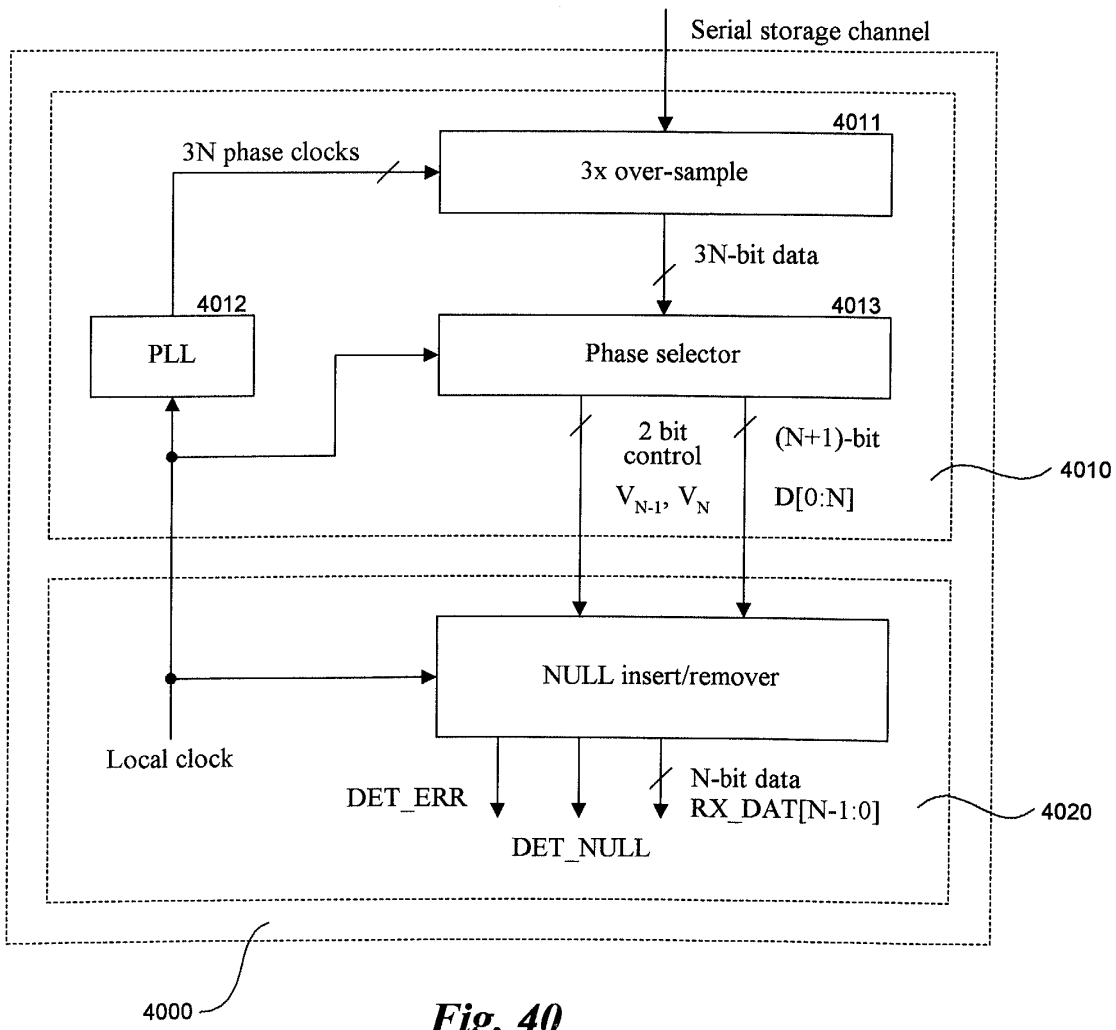

Serial storage channel
4011
3N phase clocks
3x over-sample
3N-bit data
4012
PLL
4013
Phase selector
2 bit control
V_{N-1}, V_N
(N+1)-bit
D[0:N]
4010
NULL insert/remover
Local clock
DET_ERR
N-bit data
RX_DAT[N-1:0]
4020
DET_NULL
4000

Fig. 40

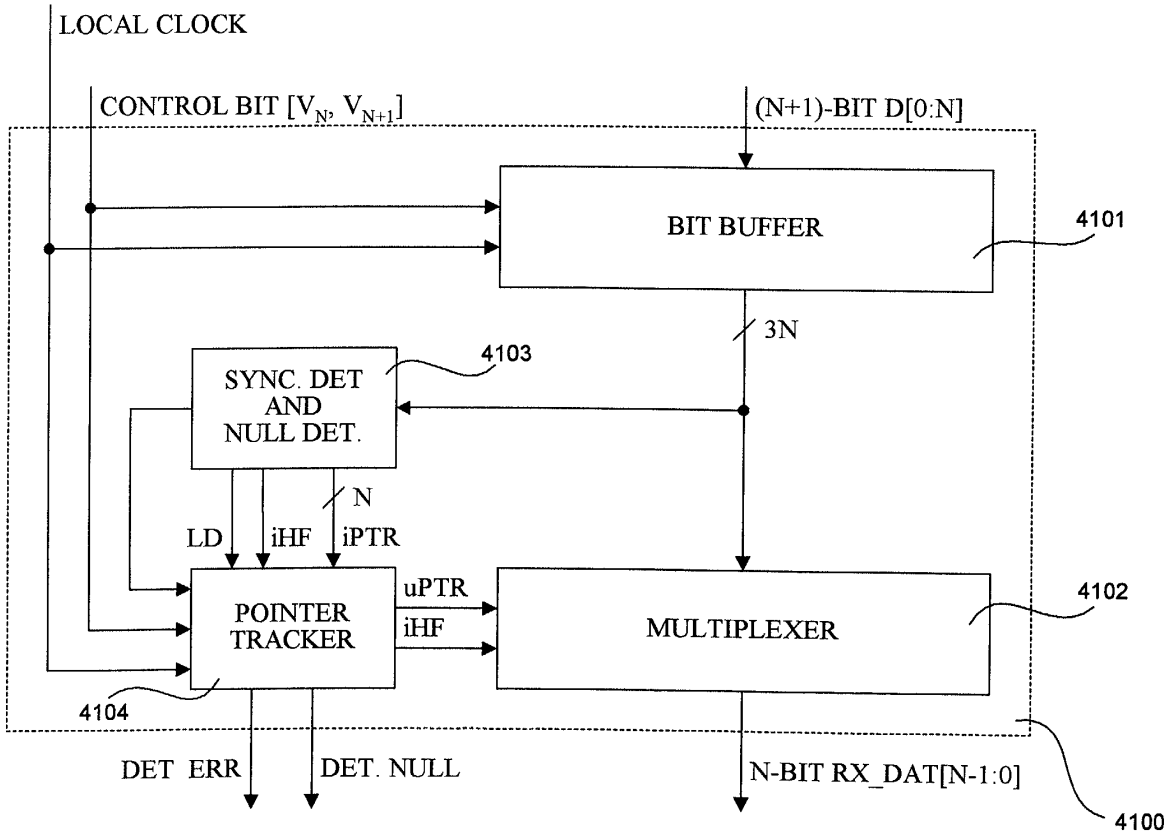

LOCAL CLOCK
CONTROL BIT [V_N, V_{N+1}]
(N+1)-BIT D[0:N]
BIT BUFFER
4101
3N
SYNC. DET AND NULL DET.
4103
N
LD
iHF
iPTR
POINTER TRACKER
uPTR
iHF
MULTIPLEXER
4102
4104
DET ERR
DET. NULL
N-BIT RX_DAT[N-1:0]
4100

Fig. 41

20090 "TT6SE00T

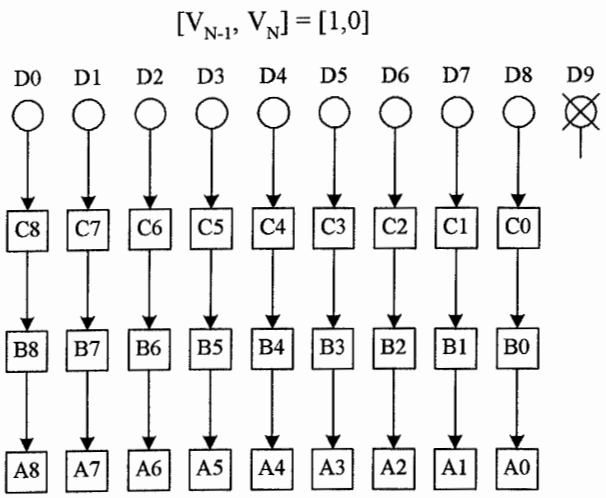


Fig. 42A

20S090 TT6SE00T

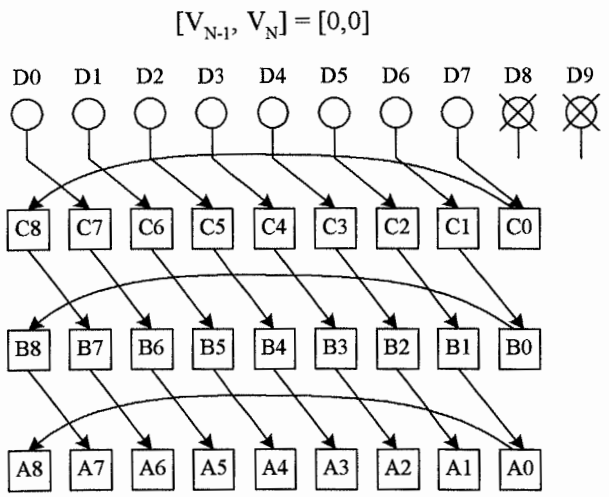


Fig. 42B

20S090" TT6SE00T

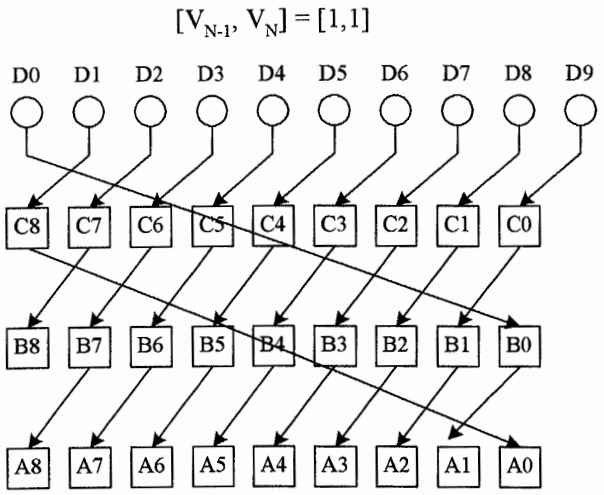


Fig. 42C

2050301 TT6SE00T

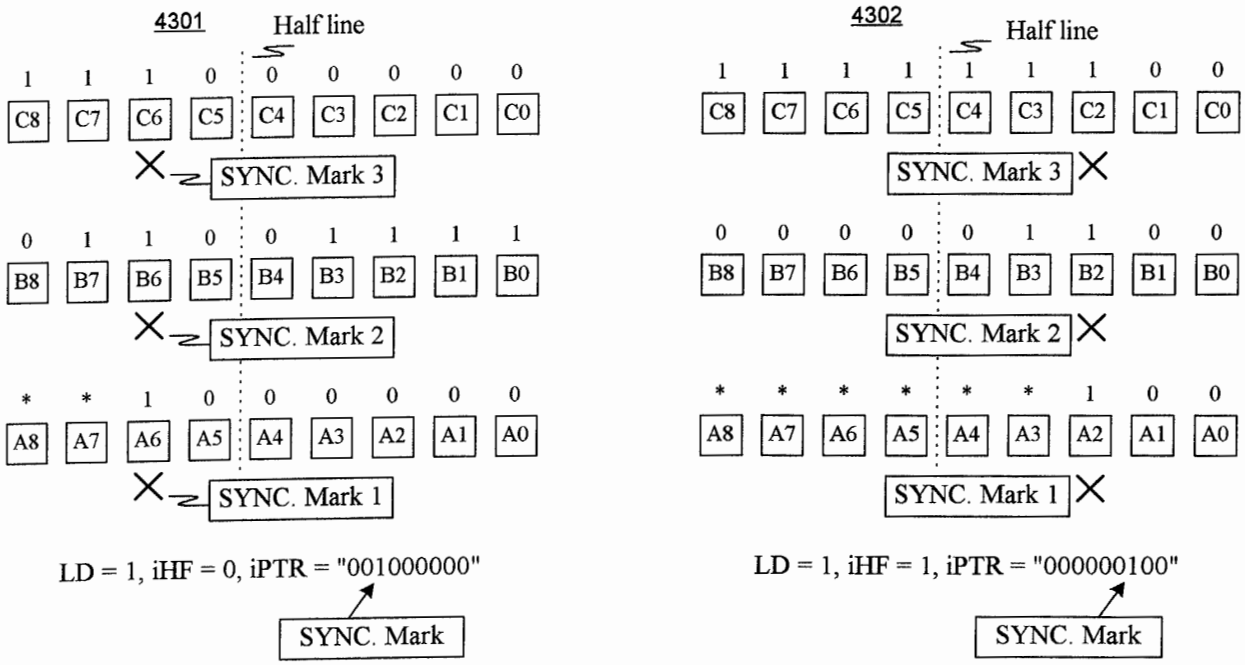
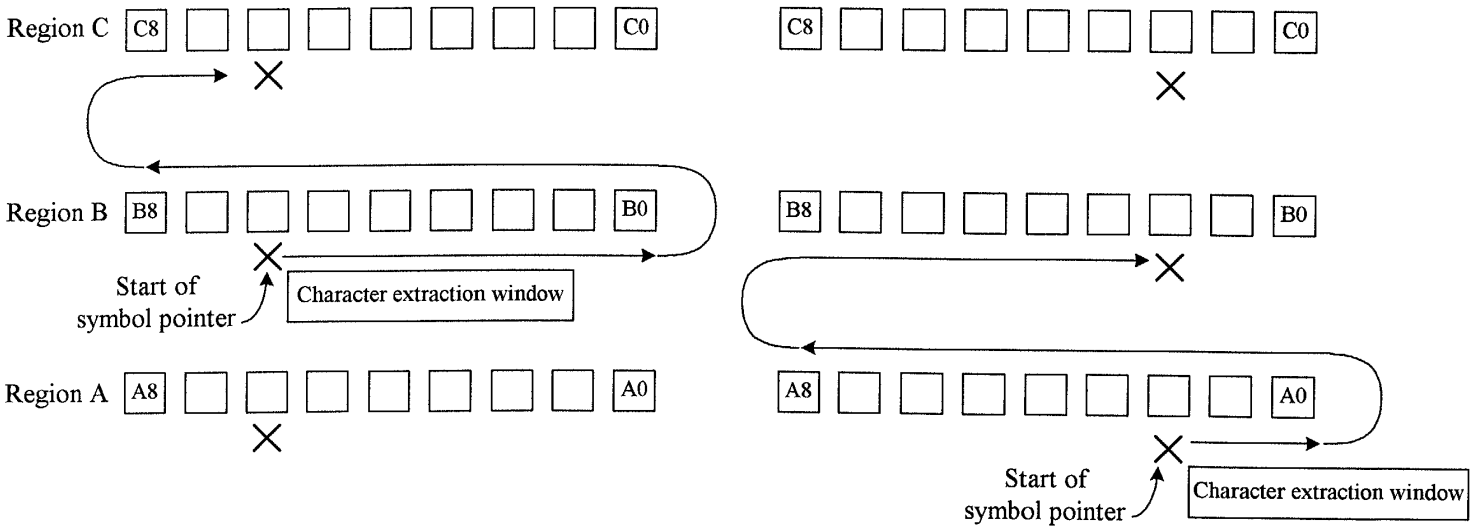


Fig. 43



LD = 1, iHF = 0, iPTR = "001000000"

LD = 1, iHF = 1, iPTR = "000000100"

Fig. 44

205090" T765E00T

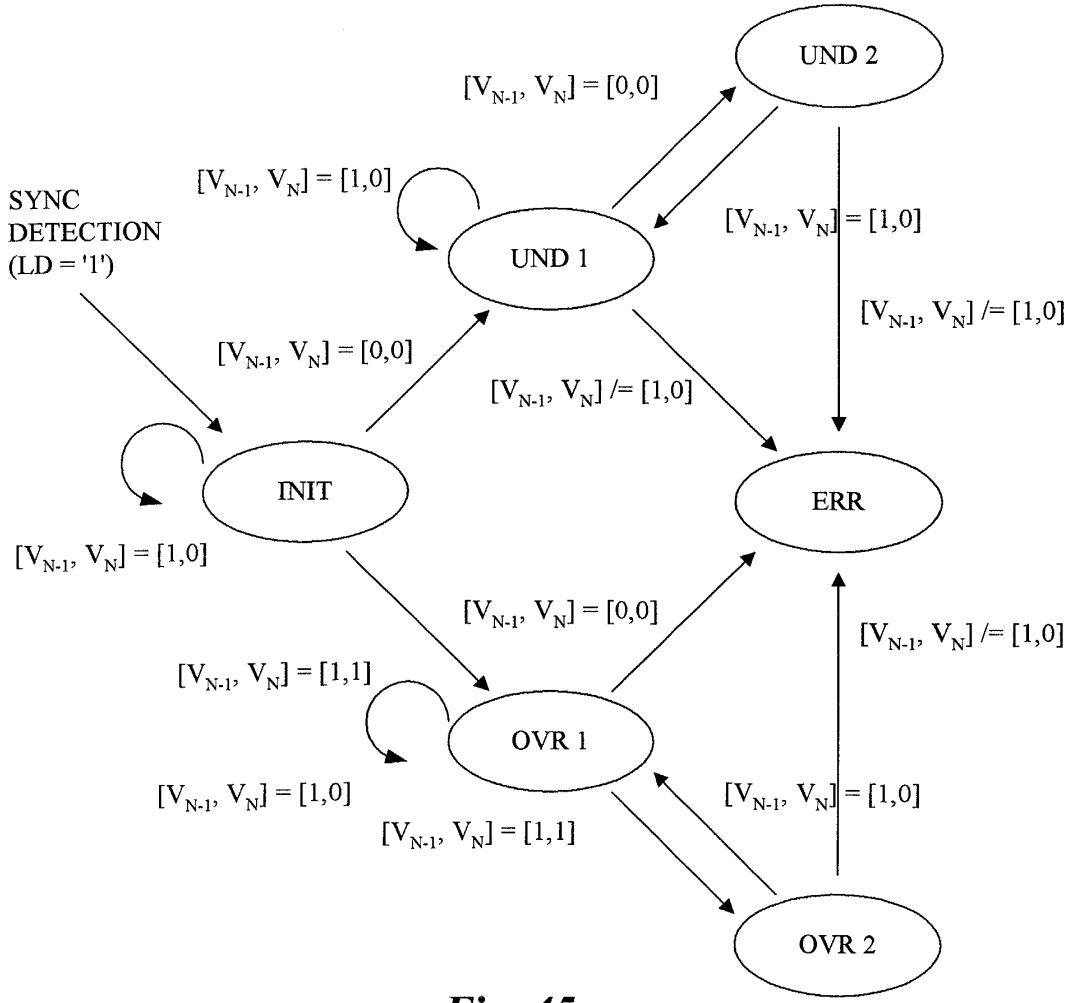


Fig. 45

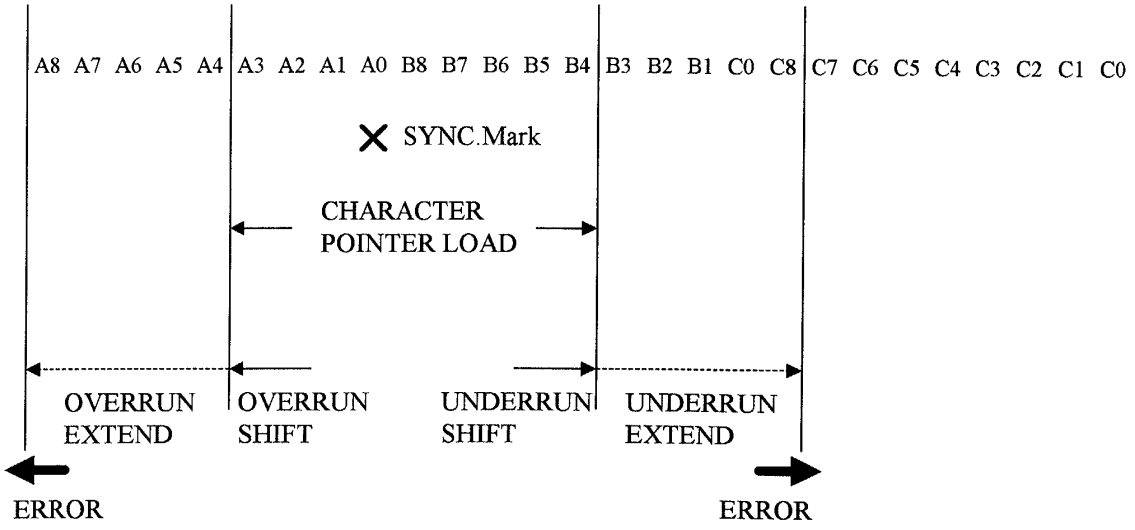

A8 A7 A6 A5 A4 A3 A2 A1 A0 B8 B7 B6 B5 B4 B3 B2 B1 C0 C8 C7 C6 C5 C4 C3 C2 C1 C0
SYNC.Mark
CHARACTER
POINTER LOAD
OVERRUN
EXTEND
OVERRUN
SHIFT
UNDERRUN
SHIFT
UNDERRUN
EXTEND
ERROR
ERROR

Fig. 46

20S090 TT6SE00T

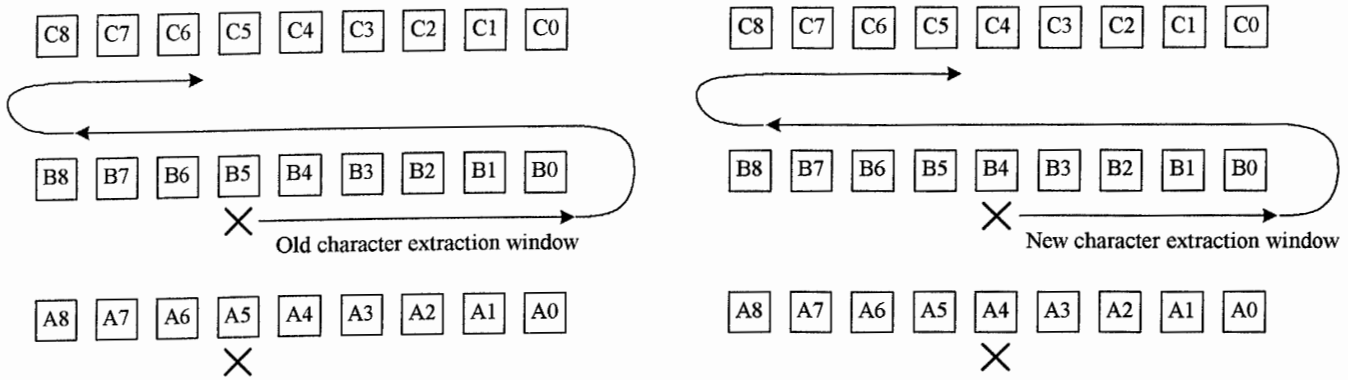


Fig. 47A

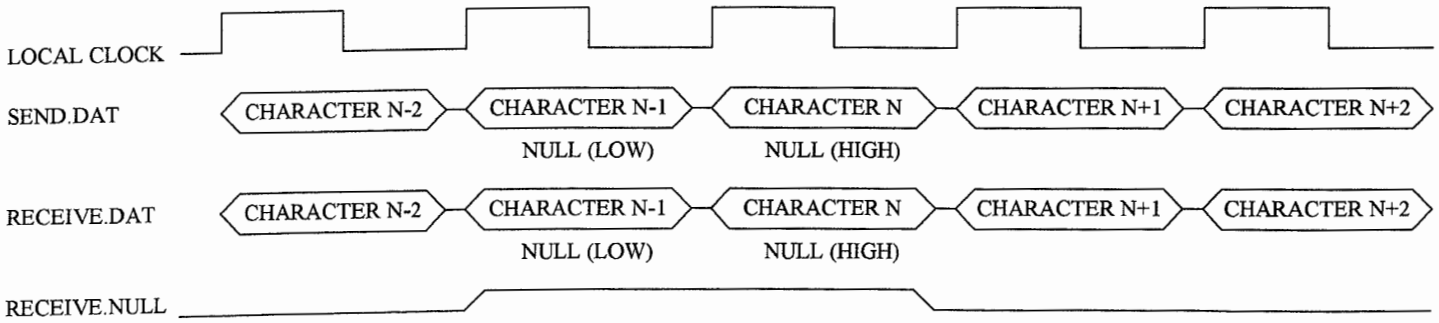


Fig. 47B

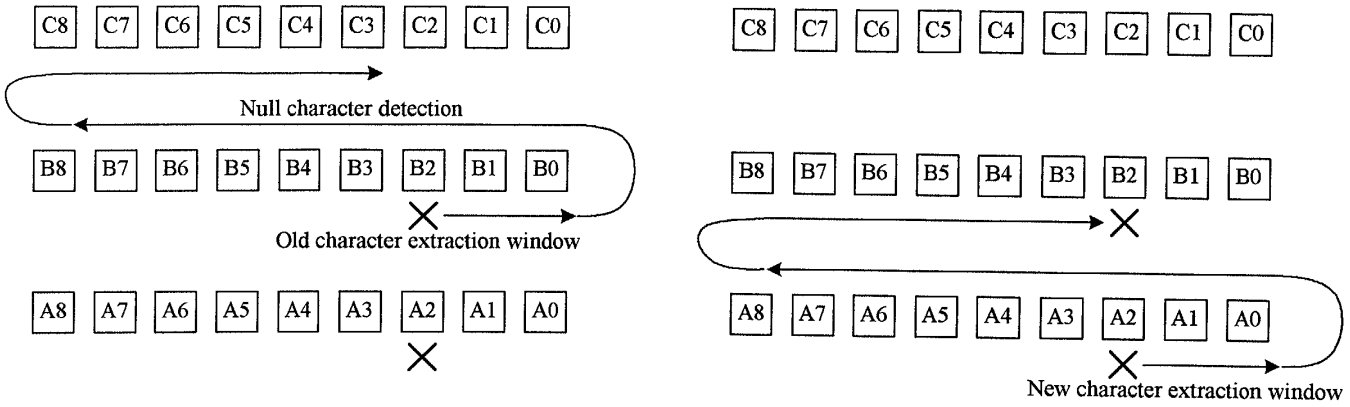


Fig. 48A

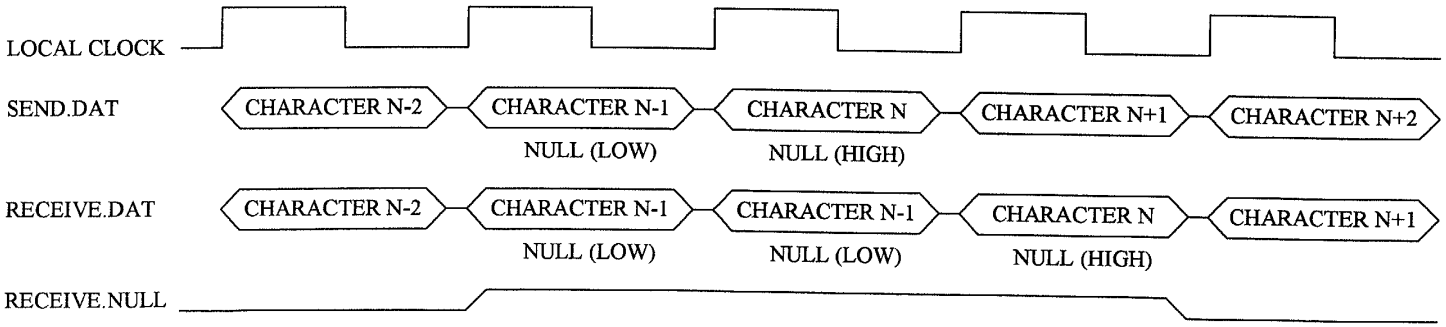


Fig. 48B

20S090 TT6SE00T

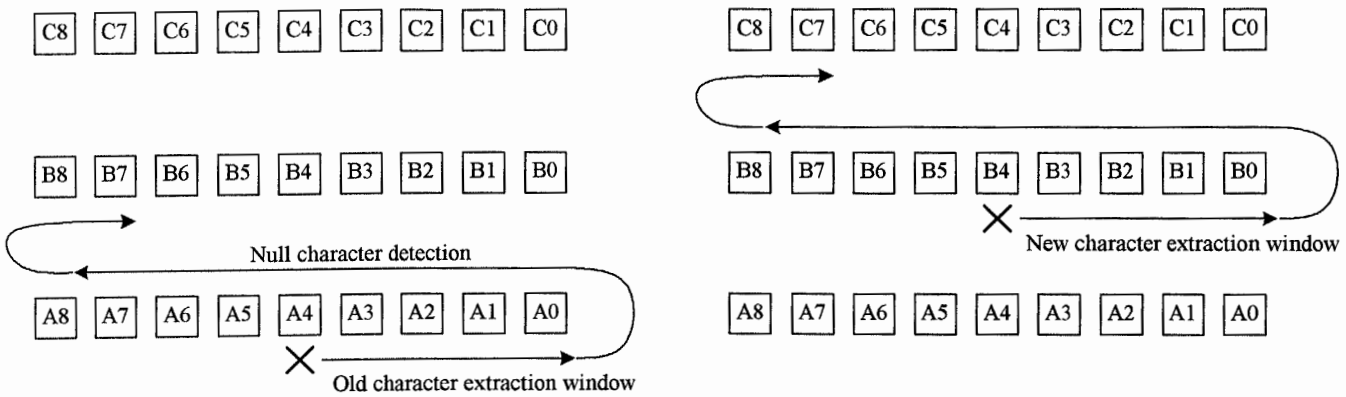


Fig. 49A

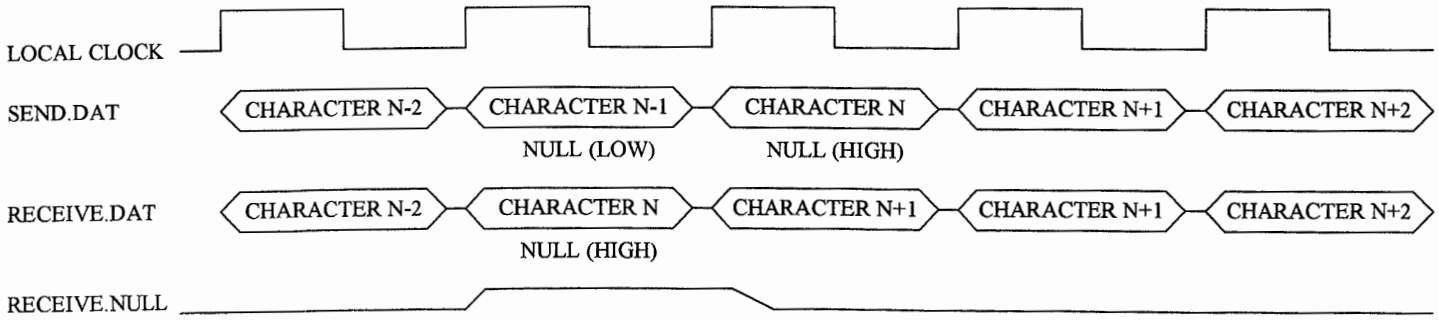


Fig. 49B

O I P E JC41 JUN 05 2002 PATENT & TRADEMARK OFFICE

DECLARATION

As the below-named inventors, we declare that:

Our residences, post office addresses, and citizenships are as stated below under our names.

We believe we are the original, first, and joint inventors of the subject matter claimed and for which a patent is sought on the invention entitled "METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS" in the foregoing specification and claims, which claims the benefit of U.S. Provisional Application No. 60/252,724, filed November 22, 2000.

We have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment specifically referred to above.

We acknowledge our duty to disclose information which is material to the patentability of this application in accordance with 37 C.F.R. § 1.56(a).

We further declare that all statements made herein of our 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 the making of willfully false statements and the like is punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and may jeopardize the validity of any patent issuing from this patent application.

EXPRESS MAIL NO. EL 955 934 705 US

Yeshik Shin

Date Oct -29th_2001

Residence : City of Sunnyvale
State of California

Citizenship : Korea

P.O. Address : 1035 Aster Avenue, #1116
Sunnyvale, California 94086

20050502.P0054

IPR2024-001428
HP – EX1002, Page 348 of 487

David D. Lee

Date 10-29-01

Residence : City of Palo Alto
State of California

Citizenship : United States of America

P.O. Address : ~~3715 Redwood Circle~~ 4180 Wallis Court PC
Palo Alto, California 94306

20359.1-060502

IPR2024-001428
HP – EX1002, Page 349 of 487

Deog-Kyoon Jeong

Date Oct. 31, 2001

Residence : City of Seoul
Country of Korea

Citizenship : Korea

P.O. Address : San 56-1 Shinlim-Dong, Kwanak-Gu
Seoul, Korea 15174-2

20050502.1750.0950

IPR2024-001428
HP – EX1002, Page 350 of 487

Shing Kong

Date 10/29/2001

Residence : City of Brisbane
State of California

Citizenship : United States of America

P.O. Address : 112 Rock Wren Lane
Brisbane, California 94005

Express Mail Label No. EL 705 447 205 US

IPR2024-001428
HP – EX1002, Page 351 of 487

JUN 05 2002

COPY OF PAPERS
ORIGINALLY FILED

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Yeshik Shin et al.
Application No.	:	10/035,911 Confirmation No.: 2160
Filed	:	November 7, 2001
For	:	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

Docket No.	:	371798019US
Date	:	May 24, 2002

Commissioner for Patents
Washington, DC 20231

AUTHORIZATION FOR EXTENSIONS OF TIME UNDER 37 C.F.R. § 1.136(A)(3)

Sir:

With respect to the above-identified application, the Commissioner is authorized to treat any concurrent or future reply requiring a petition for an extension of time under 37 C.F.R. § 1.136(a)(3) for its timely submission as incorporating a petition therefor for the appropriate length of time. The Commissioner is also authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account No. 50-0665.

Date May 24, 2002

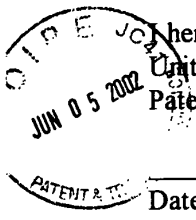
Maurice J. Pirio
Registration No. 33,273

PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888
FAX: (206) 583-8500

Q:\Clients\Silicon Image (37179)\8019 (Nesting)\US00\extension auth.doc

#3

PATENT



I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Commissioner for Patents, Washington, DC 20231.

5/24/02 — Jeanne Connelly

Date

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Yeshik Shin et al.
Application No.	:	10/035,911 Confirmation No.: 2160
Filed	:	November 7, 2001
For	:	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

Docket No.	:	371798019US
Date	:	May 24, 2002

Commissioner for Patents
Washington, DC 20231

TRANSMITTAL OF FORMAL DRAWINGS

Sir:

Enclosed please find 59 sheets of formal drawings (Figures 1-49B) for the above-identified application.

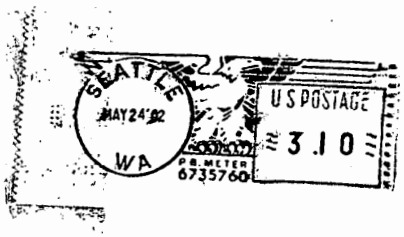
Respectfully submitted,
Perkins Coie LLP

Maurice J. Pirio
Registration No. 33,273

MJP:jc
Enclosures:
59 Sheets of Drawings

PERKINS COIE LLP
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888
FAX: (206) 583-8500

Q:\Clients\Silicon Image (37179)\8019 (Nesting)\US00\formal drawings.doc



First Class Mail

First Class Mail

RECEIVED
JUN 06 2002
USPTO MAIL CENTER

PERKINS COIE LLP
1201 THIRD AVENUE, SUITE 4800
SEATTLE, WASHINGTON 98101-3099

OIPE
JUN 05 2002
PATENT & TRADEMARK OFFICE

Box Missing Parts
Commissioner for Patents
Washington, DC 20231

COPY OF PAPERS
ORIGINALLY FILED

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Commissioner for Patents, Washington, DC 20231.

5/24/02 — Jeanne Connelly
Date

JC41 IPE JUN 05 2002 PATENT & TM

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Yeshik Shin et al.
Application No.	:	10/035,911 Confirmation No.: 2160
Filed	:	November 7, 2001
For	:	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

Docket No.	:	371798019US
Date	:	May 24,2002

Commissioner for Patents
Washington, DC 20231

PETITION FOR EXTENSION OF TIME UNDER 37 CFR. § 1.136(a)

Sir:

Applicants herewith petition the Commissioner of Patents under 37 CFR § 1.136(a) for a two-month extension of time for filing the response to the Notice to File Missing Parts dated February 7, 2002, from April 7, 2002 to June 7, 2002. A check is enclosed which includes the $400.00 fee. Any deficiency or overpayment should be charged or credited to Deposit Account No. 50-0665. This petition is being submitted in triplicate.

Respectfully submitted,
Perkins Coie LLP

Maurice J. Pirio
Registration No. 33,273

P.O. Box 1247
Seattle, Washington 98111-1247
(206) 583-8888
Fax: (206) 583-8500

IPR2024-001428
HP – EX1002, Page 355 of 487

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
10/035,911	11/07/2001	Yeshik Shin	371798019US

CONFIRMATION NO. 2160

25096
PERKINS COIE LLP
PATENT-SEA
P.O. BOX 1247
SEATTLE, WA 98111-1247

FORMALITIES LETTER

OC000000007439531

Date Mailed: 02/07/2002

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The statutory basic filing fee is missing.
  Applicant must submit $ **740** *to complete the basic filing fee for a non-small entity. If appropriate, applicant may make a written assertion of entitlement to small entity status and pay the small entity filing fee (37 CFR 1.27).*
- Total additional claim fee(s) for this application is $264.
  - **$180** for **10** total claims over 20.
  - **$84** for **1** independent claims over 3 .
- The oath or declaration is missing.
  A properly signed oath or declaration in compliance with 37 CFR 1.63, identifying the application by the above Application Number and Filing Date, is required.
- To avoid abandonment, a late filing fee or oath or declaration surcharge as set forth in 37 CFR 1.16(l) of $130 for a non-small entity, must be submitted with the missing items identified in this letter.
- **The balance due by applicant is $ 1134.**

A copy of this notice MUST be returned with the reply.

Tram Chau

Customer Service Center
Initial Patent Examination Division (703) 308-1202

PART 3 - OFFICE COPY

Please type a plus sign (+) inside this box → [+]

01-04-02

PTO/SB/05 (03-01)
Approved for use through 10/31/2002. 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

JC691 U.S. PTO 11/07/01

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 C.F.R. 1.53(b))

Attorney Docket No.	371798019US
First Inventor	Yeshik Shin
Title	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS
Express Mail Label No.	EL669108842US

J1017 U.S. PTO 10/035911 11/07/01

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Box Patent Application, Commissioner for Patents, Washington, DC 20231

1. ☐ Fee Transmittal Form (e.g., PTO/SB/17) *(Submit an original and a duplicate for fee processing)*
2. ☐ Applicant claims small entity status. See 37 CFR 1.27.
3. ☒ Specification [*Total Pages* 69]
   (preferred arrangement set forth below)
   - Descriptive title of the Invention
   - Cross Reference to Related Applications
   - Statement Regarding Fed sponsored R & D
   - Reference to sequence listing, a table, or a computer program listing appendix
   - Background of the Invention
   - Brief Summary of the Invention
   - Brief Description of the Drawings *(if filed)*
   - Detailed Description
   - Claim(s)
   - Abstract of the Disclosure
4. ☒ Drawing(s) (*35 U.S.C.113*) [*Total Sheets* 58]
5. Oath or Declaration [*Total Pages*]
   a. ☐ Newly executed (original or copy)
   b. ☐ Copy from a prior application (37 CFR 1.63 (d)) *(for a continuation/divisional with Box 18 completed)*
      i. ☐ DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
6. ☐ Application Data Sheet. See 37 CFR 1.76
7. ☐ CD-ROM or CD-R in duplicate, large table or Computer Program *(Appendix)*
8. Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. ☐ Computer Readable Form (CRF)
   b. Specification Sequence Listing on:
      i ☐ CD-ROM or CD-R (2 copies); or
      ii. ☐ paper
   c. ☐ Statements verifying identity of above copies

ACCOMPANYING APPLICATIONS PARTS

9. ☐ Assignment Papers (cover sheet & document(s))
10. ☐ 37 C.F.R.§3.73(b) Statement *(when there is an assignee)* ☐ Power of Attorney
11. ☐ English Translation Document *(if applicable)*
12. ☐ Information Disclosure Statement (IDS)/PTO-1449 ☐ Copies of IDS Citations
13. ☐ Preliminary Amendment
14. ☒ Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
15. ☐ Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. ☒ Request and Certification under 35 U.S.C. 122 (b)(2)(B)(i). Applicant must attach form PTO/SB/35 or its equivalent
17. ☐ Other:

18. If a CONTINUING APPLICATION, *check appropriate box, and supply the requisite information below and in a preliminary amendment, or in an Application Data Sheet under 37 CFR 1.76:*

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: _____ / _____

Prior application information: *Examiner* _____ *Group / Art Unit:* _____

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 5b, is considered a part of the disclosure of the accompanying or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

☒ Claims the benefit of Provisional Application No. 60/252,724 filed November 22, 2000

17. CORRESPONDENCE ADDRESS

☒ *Customer Number or Bar Code Label* 25096 PATENT TRADEMARK OFFICE or ☐ *Correspondence address below*

Name (Print/Type)	Maurice J. Pirio	*Registration No. (Attorney/Agent)*	33,273
Signature	Maurice J. Pirio	*Date*	November 7, 2001

Burden Hour Statement: This form is estimated to take 0 2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, Box Patent Application, Washington, DC 20231.

IPR2024-001428
HP – EX1002, Page 357 of 487

EXPRESS MAIL NO. EL669108842US
PTO/SB/35 (11-00)
Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office; U. S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

NONPUBLICATION REQUEST UNDER 35 U.S.C. 122(b)(2)(B)(i)		
	First Named Inventor	Yeshik Shin
	Title	METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS
	Atty Docket Number	371798019US

I hereby 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 agreement, that requires publication at eighteen months after filing.

I hereby request that the attached application not be published under 35 U.S.C. 122(b).

November 7, 2001
Date

Signature

Maurice J. Pirio
Typed or printed name

This request must be signed in compliance with 37 CFR 1.33(b) and submitted with the application **upon filing.**

Applicant may rescind this nonpublication request at any time. If applicant rescinds a request that an application not be published under 35 U.S.C. 122(b), the application will be scheduled for publication at eighteen months from the earliest claimed filing date for which a benefit is claimed.

If applicant subsequently files an application directed to the invention disclosed in the attached application in another country, or under a multilateral international agreement, that requires publication of applications eighteen months after filing, the applicant **must** notify the United States Patent and Trademark Office of such filing within forty-five (45) days after the date of the filing of such foreign or international application. **Failure to do so will result in abandonment of this application (35 U.S.C. 122(b)(2)(B)(iii)).**

Burden Hour Statement: This collection of information is required by 37 CFR 1.213(a). The information is used by the public to request that an application not be published under 35 U.S.C. 122(b) (and the PTO to process that request). Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This form is estimated to take 6 minutes to complete. This time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, Washington, DC 20231.

Q:\Clients\Silicon Image (37179)\8019 (Nesting)\PTO.doc

EXPRESS MAIL NO. EL669108842US

METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application No. 60/252,724 entitled "METHOD AND APPARATUS FOR STORAGE I/O WITH FULL-DUPLEX ONE-TIME BLOCK I/O TRANSFER AND ADAPTIVE PAYLOAD SIZING," filed November 22, 2000, and is related to U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR PLESIOSYNCHRONOUS COMMUNICATIONS WITH NULL INSERTION AND REMOVAL" (Attorney Docket No. 371798002US); U.S. Patent Application No. __, entitled "METHOD AND SYSTEM FOR TRANSITION-CONTROLLED SELECTIVE BLOCK INVERSION COMMUNICATIONS" (Attorney Docket No. 371798007US); U.S. Patent Application No. _____ entitled "COMMUNICATIONS ARCHITECTURE FOR STORAGE-BASED DEVICES" (Attorney Docket No. 371798008US1); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR PACKET ORDERING BASED ON PACKET TYPE" (Attorney Docket No. 371798013US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR HOST HANDLING OF COMMUNICATIONS ERRORS" (Attorney Docket No. 371798014US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR DYNAMIC SEGMENTATION OF COMMUNICATIONS PACKETS" (Attorney Docket No. 371798015US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR ASYMMETRIC PACKET ORDERING BETWEEN COMMUNICATIONS DEVICES" (Attorney Docket No. 371798016US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR COMMUNICATING CONTROL INFORMATION VIA OUT-OF-BAND SYMBOLS" (Attorney Docket No. 371798017US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR INTEGRATING PACKET TYPE

EXPRESS MAIL NO. EL669108842US

IPR2024-001428
HP – EX1002, Page 359 of 487

INFORMATION WITH SYNCHRONIZATION SYMBOLS" (Attorney Docket No. 371798018US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS" (Attorney Docket No. 371798019US); U.S. Patent Application No. _____ entitled "COMMUNICATIONS ARCHITECTURE FOR MEMORY-BASED DEVICES" (Attorney Docket No. 371798012US); U.S. Patent Application No. _____ entitled "METHOD AND SYSTEM FOR DC-BALANCING AT THE PHYSICAL LAYER" (Attorney Docket No. 371798020US); and U.S. Patent Application No. _____ entitled "MULTISECTION MEMORY BANK SYSTEM" (Attorney Docket No. 371798021US, which are all hereby incorporated by reference in their entirety.

BACKGROUND

[0002] The described technology relates generally to communications techniques and particularly to communications between hosts and data store devices.

[0003] The speed and capacity of the data store devices, such as disk drives and memories, have increased significantly over the past several years. As a result of their improved performance, these data store devices are being used in many new applications, such as database servers, Web servers, personal video recorders, and digital displays. These applications often require large amounts of data to be communicated between data store accessing devices ("hosts") and data store devices. (Hosts may include computers, CPUs, or any logic for accessing a data store device.) Moreover, as host speed increases, the speed in communicating between hosts and data store devices can have a significant impact on the overall performance of the application. In particular, even though the speed of hosts and data store devices has increased significantly, the speed of communications between hosts and data store devices has not increased as significantly, especially for communications over long distances (e.g., greater than one meter). Thus, the communications speed presents a bottleneck in many new applications.

[0004] Current communications techniques typically communicate between hosts and certain types of data store devices, such as disk drives, using a bus with

Express Mail No. EV009615954US

IPR2024-001428
HP – EX1002, Page 360 of 487

many parallel lines or using a single serial communications link. The Integrated Disk Electronics ("IDE") bus and the Small Computer Systems Interface ("SCSI") bus are examples of bus-based parallel communications techniques. These communications techniques, however, present many problems. Performance of bus-based communications techniques is generally improved by increasing the number of lines in the bus, which may significantly increase the cost of such techniques. In addition, bus-based communications techniques generally provide arbitration so that multiple hosts and data store devices can share the same bus. The use of arbitration can significantly increase the cost of such a bus. The cost of such bus-based communication techniques is further increased because their design needs to address additional problems such as cross-talk and clock skew. In particular, as the communications speed increases, the solution to cross-talk and clock skew become much more complex.

[0005] Some serial communications techniques have been developed to address some of the problems of bus-based communications techniques. Current serial communications techniques, however, have problems of their own. Serial AT, Attachment, which is intended to replace IDE, does not scale well and only operates in a half duplex mode. Fibre Channel, currently used to support storage area networks ("SANs"), is very generic and therefore, not optimized for any particular application. In particular, Fibre Channel has a relatively small packet size with a large header. As a result, use of Fibre Channel often results in an unacceptably large overhead. For example, data transmitted to disk drives is typically sent in very large blocks (e.g., 216 bytes). With Fibre Channel, such large blocks need to be divided in many (e.g., 32) packets, which results in a high overhead in the amount of redundant header information and in the redundant processing performed as a packet is routed to its destination. Thus, Fibre Channel may not be appropriate for many applications.

[0006] Current memory devices, such as SDRAM and RDRAM, are typically designed to be synchronous with the accessing processing unit. The hosts and the memory devices are synchronous in that they share the same clock signal.

IPR2024-001428
HP – EX1002, Page 361 of 487

These memory devices are typically optimized for access patterns that are both temporally and spatially related. In particular, these memory devices are optimized to read and write arrays (or streams) of data. There is a setup overhead (e.g., 5 clock cycles) when accessing the first word of an array in memory, but access of subsequent words in the array occurs at the synchronized clock rate (e.g., 1 access per clock cycle). Since the access patterns of central processing units and graphics processors are typically temporally and spatially related, they can access such memory devices efficiently.

[0007] Existing memory devices that are designed to support access patterns with a high temporal and spatial relationship may not be appropriate for uses having access patterns with a lower spatial relationship. The setup overhead for each access may be too high. There are, indeed, many uses for memory devices with access patterns that are not as spatially or temporally related as those of a central processing unit or a graphics processor. For example, a switch may have a memory device in which packets of data received via an input port are stored before they are transmitted via an output port. Traditionally, switches used crossbars to provide the switching function and FIFOs to provide a buffering function. When a memory device is used on a switch in place of a crossbar, then all the input and output ports need access to the memory device. The accesses by the different ports are, however, not particularly spatially related. Moreover, when the packet size is small (e.g., 53 bytes in the case of an ATM switch), the spatial relationship of accesses by a single port may not be significant. Other uses in which there may not be a significant spatial relationship of accesses include network processors and caches for storage area networks. In such uses, the data is received from disparate sources at disparate times and may not be spatially related.

[0008] Many existing memory devices are not particularly suitable for many uses because the memory devices typically allow access by only one accessing device at a time and because the memory devices typically operate at different clock rates than the accessing devices. Because such memory devices can only be

10/25/01 15:45:05

IPR2024-001428
HP – EX1002, Page 362 of 487

accessed by one device at a time, the accessing devices may need to enter a wait state because the memory device is busy or a memory controller may need to have a buffering component. Of course, the use of a wait state may result in unacceptable performance. Also, the addition of a buffering component may increase complexity and cost. In addition, when multiple accessing devices access the same memory device through a single bus (e.g., one writing to the memory device and the other reading from the memory device), then all the devices that access the memory device need to be synchronized with the memory device. Because the accessing devices may have different underlying clock rates, complex and costly logic is needed to support the mapping to the bus clock rate.

[0009] Existing communications protocols, such as Fibre Channel, may have an unacceptable overhead for communicating with memory devices. The communications from a host to a memory device may occur in relatively short blocks (e.g., 32 bytes). Each block needs to be transmitted in a separate packet with a relatively large header. In some packets, the header may be larger than the data itself, which can significantly reduce the overall bandwidth and speed of transmission. More generally, communications between devices typically occurs in a synchronous or an asynchronous mode. In a synchronous mode, the transmitting and receiving devices use the same clock signal. The transmitting device can send the clock signal to the receiving device either as a separate signal or as a signal that can be derived from the data signals. When the clock is sent as a separate signal, problems arise resulting from the different delays in the data signals and the clock signal. These delays and resulting problems are increased as the transmission speed and distance are increased. It is very difficult and costly to account for these delays. In addition, the receiving device will have an asynchronous clock boundary. That is, a portion of the receiving device will operate at the clock frequency based on the transmitting device's clock frequency (i.e., the transmitter's clock domain) and another portion will operate at the receiving device's local clock frequency (i.e., the receiver's clock domain). As a result of the asynchronous boundary, the receiving device typically needs to

Express Mail Label No. EV004954705US

IPR2024-001428
HP – EX1002, Page 363 of 487

buffer control and data signals sent between the clock domains using elastic buffers, which adds to the complexity and cost of the receiving devices. These elastic buffers require substantial space (e.g., chip area), and when a single chip has multiple communications ports, the design is complicated because each port needs its own elastic buffer. When the clock is derived from the data signal, the problems of the delay are reduced somewhat, but there are still the problems associated with an asynchronous clock boundary.

[0010] A plesiosynchronous clocking technique can be used to avoid the need to transmit a separate clock signal or derive the clock signal from the data signal. With plesiosynchronous clocking (also known as "plesiochronous" clocking), the transmitting and receiving devices have clocks with nominally the same clock frequency. If the clock frequencies were exactly the same, then transmitting and receiving devices would be synchronized and the receiving device could accurately identify the transmitted data (in the case of serial transmission). Also, since the receiving device operates only at its local clock frequency, there is no asynchronous clock boundary. In practice, however, clock frequencies are not exactly the same but vary, for example, by 100 ppm. The receiving device can use techniques as described in U.S. Patent No. 6,229,859, entitled "System and Method for High-Speed, Synchronized Data Communication," which is hereby incorporated by reference, to account for clock variations. Those techniques use an oversampling of the data by the receiving device to detect edge boundaries of the transmitted data. The receiving device can vary the number of bits of data detected during an interval to compensate for the variations in frequency.

[0011] It would be desirable to have a communications architecture that provides high-performance for applications (e.g., data storage-based applications and memory-based applications) at a low cost. Such a communications architecture would allow for communications techniques to be tailored to particular applications.

IPR2024-001428
HP – EX1002, Page 364 of 487

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] Figure 1 is a block diagram illustrating components of the Storage Link architecture in one embodiment.

[0013] Figure 2 is a block diagram illustrating components of the transport layer, link layer, and physical layer in one embodiment.

[0014] Figure 3 is a block diagram illustrating components of the physical layer in one embodiment.

[0015] Figure 4 is a block diagram illustrating the format of a packet in one embodiment.

[0016] Figure 5 is a block diagram illustrating the organization of a data packet and a segmented data packet in one embodiment.

[0017] Figure 6 is a block diagram illustrating a receive packet memory before and after merging of a received packet.

[0018] Figure 7 is a logic diagram illustrating the segmentation of a data packet by the transport layer in one embodiment.

[0019] Figure 8 is a logic diagram illustrating the merging of data packets by the transport layer in one embodiment.

[0020] Figure 9A is a diagram illustrating the use of a synchronization primitive encoded with the packet type in one embodiment.

[0021] Figure 9B illustrates the identification of synchronization symbols in one embodiment.

[0022] Figure 9C is a block diagram illustrating a component of a link layer for inserting a synchronization symbol into a stream of symbols.

[0023] Figure 10 is a block diagram illustrating an organization of a packet memory in one embodiment.

[0024] Figure 11 is a logic diagram illustrating the process of storing packets in a transmit packet memory in one embodiment.

[0025] Figure 12 is a logic diagram illustrating the transport layer processing of packets stored in the packet memory in one embodiment.

IPR2024-001428
HP – EX1002, Page 365 of 487

[0026] Figure 13 is a diagram illustrating the preemption of a packet in one embodiment.

[0027] Figure 14 is a logic diagram illustrating the processing of packet preemption by a transmitting communications node in one embodiment.

[0028] Figure 15 is a logic diagram illustrating the processing of packet preemption by a receiving communications node in one embodiment.

[0029] Figure 16 is a block diagram illustrating asymmetric packet ordering in one embodiment.

[0030] Figure 17 is a diagram illustrating transaction-based asymmetric packet ordering in one embodiment.

[0031] Figure 18 is a logic diagram illustrating processing by a switch that ensures packet ordering within a transaction in one embodiment.

[0032] Figure 19A is a block diagram illustrating error handling during transmission from a host to a data store device in one embodiment.

[0033] Figure 19B is a block diagram illustrating the handling of errors during the transmission from a data store device to a host in one embodiment.

[0034] Figure 19C is a logic diagram illustrating the processing of a communications node detecting an error in one embodiment.

[0035] Figure 20 illustrates an alternate bit inversion encoding technique in one embodiment.

[0036] Figure 21A is a diagram illustrating the selective block inversion encoding technique in one embodiment.

[0037] Figure 21B is a block diagram illustrating selective block inversion with a programmable block size in one embodiment.

[0038] Figure 21C is a block diagram illustrating transition inversion and polarity inversion in one embodiment.

[0039] Figure 22 is a diagram illustrating the format of a primitive in one embodiment.

[0040] Figure 23 is a logic diagram illustrating transition control and selective block inversion encoding in one embodiment.

Docket No. 37179-8019

IPR2024-001428
HP – EX1002, Page 366 of 487

[0041] Figure 24 is a logic diagram illustrating transition control optimization in one embodiment.

[0042] Figure 25 is a logic diagram illustrating block inversion in one embodiment.

[0043] Figure 26 is a logic diagram illustrating transition control and selective block inversion decoding in one embodiment.

[0044] Figure 27 is a logic diagram illustrating the undoing of block inversion in one embodiment.

[0045] Figure 28 is a logic diagram illustrating the undoing of transition control optimization in one embodiment.

[0046] Figure 29 is a logic diagram illustrating the processing of a primitive in one embodiment.

[0047] Figure 30 is a block diagram illustrating the overall architecture of a multiport memory device in one embodiment.

[0048] Figure 31 is a block diagram illustrating the use of a single phase lock loop for the physical layer of a multiport memory device in one embodiment.

[0049] Figure 32 is a block diagram illustrating contents of a bank cache in one embodiment.

[0050] Figure 33 is a logic diagram illustrating processing of the access layer when it receives data from a host in one embodiment.

[0051] Figure 34 is a logic diagram illustrating the processing of a write command by the access layer in one embodiment.

[0052] Figure 35 is a logic diagram illustrating processing of the access layer when it is to provide data to the transport layer in one embodiment.

[0053] Figure 36 is a block diagram of a multisection memory bank in one embodiment.

[0054] Figure 37A is a block diagram illustrating a variable strength line driver in one embodiment.

[0055] Figure 37B is a block diagram illustrating the circuit of a variable-strength line driver in one embodiment.

IPR2024-001428
HP – EX1002, Page 367 of 487

[0056] Figure 38A is a block diagram of a plesiosynchronous communications system in one embodiment.

[0057] Figure 38B is a block diagram illustrating a switch supporting multiple data store devices in one embodiment.

[0058] Figure 39A is a block diagram illustrating a prior art synchronous clocking system.

[0059] Figure 39B is a block diagram illustrating resolution of an asynchronous clock boundary in the physical layer in one embodiment.

[0060] Figure 40 is a block diagram illustrating NULL insertion and deletion at the physical layer in one embodiment.

[0061] Figure 41 is a block diagram illustrating the frame aligner in one embodiment.

[0062] Figures 42A, 42B, and 42C illustrate the loading of the bit buffer in one embodiment.

[0063] Figure 43 is a diagram illustrating the detection of a synchronization primitive by the sync and null detector in one embodiment.

[0064] Figure 44 is a diagram illustrating the portion of the bit buffer that contains the next symbol to be extracted in one embodiment.

[0065] Figure 45 is a state diagram of the symbol pointer tracker in one embodiment.

[0066] Figure 46 is a diagram illustrating movement of the start-of-symbol pointer due to overrun or underrun in one embodiment.

[0067] Figures 47A and 47B are diagrams illustrating NULL detection while the start-of-symbol pointer is not within the overrun or underrun extend regions in one embodiment.

[0068] Figures 48A and 48B are diagrams illustrating NULL detection while the start-of-symbol pointer is in the underrun extend region.

[0069] Figures 49A and 49B are diagrams illustrating NULL detection while the start-of-symbol pointer is in the overrun extend region.

IPR2024-001428
HP – EX1002, Page 368 of 487

DETAILED DESCRIPTION

[0070] A serial communications architecture for communicating between hosts and data store devices is provided. Aspects of the serial communications architecture, referred to as "Storage Link," are specially adapted for communications with storage devices, such as disk drives. Other aspects of the serial communications architecture, referred to as "Memory Link," are specially adapted for communications with memory devices, such as RAM devices. In addition, Storage Link and Memory Link may use an improved plesiosynchronous technique when receiving data transmitted via a serial communications link.

[0071] In one embodiment, the Storage Link architecture is specially adapted to support communications between multiple hosts and storage devices via a switching network, such as a storage area network. The Storage Link architecture specifies various communications techniques that can be combined to reduce the overall cost and increase the overall performance of communications. The Storage Link architecture defines a hierarchy of transport, link, and physical layers such that each layer in the hierarchy is aware of and can take advantage of characteristics of lower layers in the hierarchy. For example, the transport layer, which is responsible for handling packets, may receive signals from the link layer that indicate when an end of packet is received. Traditional communications architectures prohibit a lower layer from being aware of the functions of a higher layer. The Storage Link architecture may provide packet ordering based on packet type, dynamic segmentation of packets, asymmetric packet ordering, packet nesting, variable-sized packet headers, and use of out-of-band symbols to transmit control information as described below in more detail. The Storage Link architecture may also specify encoding techniques to optimize transitions and to ensure DC-balance. The Storage Link architecture may also use the NULL insertion and removal techniques with plesiosynchronous clocking as described below in detail. The Storage Link architecture thus provides an improved way to access storage devices that reduces cost and increases communications speed.

IPR2024-001428
HP – EX1002, Page 369 of 487

[0072] In one embodiment, the Memory Link architecture also provides various communications techniques that can be combined to reduce the overall cost and increase the overall performance of communications between a host and a memory device. Like the Storage Link architecture, the Memory Link architecture uses a hierarchy of layers in which each layer can take advantage of the characteristics of a lower layer in the hierarchy. The Memory Link architecture in one embodiment provides a multiport memory device for serial communications. Each port may use a plesiosynchronous technique when receiving data and may share the same phase lock loop to control sampling of the received data. The Memory Link architecture may also use a physical layer D-C balancing technique that provides an additional driving of a communications link to offset the running disparity. The Memory Link architecture may also use an improved plesiosynchronous clock technique that inserts and removes special symbols from the received symbols to compensate for variations in clock frequencies between a transmitting and receiving device. Various combinations of the Memory Link architecture allow for the cost of designing, developing, and manufacturing memory-based systems to be reduced and the speed of memory access to be increased.

[0073] In one embodiment, the plesiosynchronous clocking technique uses the insertion and removal of symbols by the physical layer of a receiving communications node to compensate for variations in clock frequency between the transmitter and receiver. The receiver maintains a small buffer (e.g., 27 bits of the received data) at the physical layer. When the receiver detects a synchronization symbol, it initializes a start-of-symbol pointer in the buffer. As subsequent bits of data are received, the start-of-symbol pointer is adjusted to account for the variation in clock frequencies of the transmitter and the receiver. In particular, if the transmitter's clock frequency is faster ("an overrun condition"), then the pointer is progressively moved backward in the buffer (i.e., toward the last received bits). Analogously, if the transmitter's clock frequency is slower ("an underrun condition"), then the pointer is progressively moved forwards in the

PDX/DFT-1656.001

IPR2024-001428
HP – EX1002, Page 370 of 487

buffer (i.e., toward the first received bits). When the receiver detects a special symbol of a certain type (e.g., a NULL symbol) and the pointer has moved to near the beginning or end of the buffer, then the receiver removes the special symbol from the buffer to account for the overrun condition or inserts a special symbol into the buffer to account for the underrun condition. In this way, the receiver stays synchronized with the transmitter even though their clock frequencies vary. Also, asynchronous clock boundaries are avoided in the receivers.

[0074] In the following, aspects of the serial communications architecture are described using block diagrams and logic diagrams. One skilled in the art will appreciate that the serial communications architecture can be implemented using different combinations of logic circuits and/or firmware. In particular, the logic diagrams illustrate processing that may be performed in parallel using duplicate logic circuits (e.g., one for each communications link) or may be performed in serial using a single logic circuit. The particular logic designs can be tailored to meet the cost and performance objectives of the implementation of the serial communications architecture. One skilled in the art will be able to readily design logic circuits based on the following descriptions.

1. STORAGE LINK ARCHITECTURE

[0075] Figure 1 is a block diagram illustrating components of the Storage Link architecture in one embodiment. The hosts 110 and data store devices 120 are interconnected to switching network 130 via serial communications links 140. The hosts may include computer systems that access the data store devices. The data store devices may include storage area network devices (e.g., disk drives), high-speed memory devices, and other devices for storing data. The Storage Link architecture, however, is designed to support block-oriented data store devices, such as disk drives. The switching network may include multiple switches that are interconnected so that communications paths, especially between hosts and data store devices, can be established.

IPR2024-001428
HP – EX1002, Page 371 of 487

[0076] Each host, data store device, and switch is a communications node that includes one or more communications interfaces 115, 125, and 135 with a transport layer, a link layer, and a physical layer. The hosts and data store devices have an upper layer that communicate to each other via transactions. When transmitting data, the upper layer provides the data for the transaction to the transport layer. The transport layer receives the data of the transaction and generates packets (i.e., packetizes the data) for transmission. The transport layer then provides each code (e.g., each byte) of the packets to the link layer for transmission to the destination. When receiving data, the transport layer receives the codes of transaction from the link layer and identifies the packets. The transport layer then combines the data of the packets of the transaction (i.e., depacketizes the data) and provides the data of the transaction to the upper layer. Each host and data store device may include an application upper layer that provides the data of a transaction to the transport layer to be transmitted to a destination and receives the data of a transaction from the transport layer that was transmitted by a source. Each packet includes a header section and a payload section. The header section identifies the source and destination and a packet type (e.g., data or command). Each switch may include a communications interface for each port of the switch. Each port may be connected to a serial communications link comprising a receive and transmit link. The switches and their interconnecting communications links form the switching network. The transport layer of a switch may not packetize and depacketize the data, rather it may direct packets received via one port to be transmitted via another port to affect the routing of the packets from the source to the destination as indicated in the header section.

[0077] The link layer encodes the codes to be transmitted via its communications link into symbols and decodes symbols received via its communications link into codes. The link layer receives codes to be transmitted from the transport layer, encodes those codes into symbols, and provides those symbols to the physical layer. The link layer also receives symbols from the physical layer, decodes those

IPR2024-001428
HP – EX1002, Page 372 of 487

symbols into codes, and provides those codes to the transport layer. The encoding may include mapping the codes to symbols to optimize bit transitions and to ensure DC-balance. The physical layer receives the symbols from the link layer, serializes the symbols, and transmits the serialized symbols via the communications link. The physical layer receives serialized symbols via the communications link, deserializes the symbols, and provides the symbols to the link layer.

[0078] Figure 2 is a block diagram illustrating components of the transport layer, link layer, and physical layer in one embodiment. Each layer includes transmit components 201 and receive components 202. The transport layer includes transmit components, such as a transmit memory 211, a packetizer 212, and a transmit controller 213, and receive components, such as a receive memory 215, a depacketizer 216, and a receive controller 217. The data of the transaction to be transmitted is stored in the transmit memory. The transmit controller controls the packetizer to retrieve data of a transaction from the transmit memory, packetize the data, and provide each code of the packets to the link layer. The receive controller controls the depacketizer to receive codes provided by the link layer, identify the packets, and store the codes as packets in the receive memory.

[0079] The link layer includes transmit components, such as an encoder 221, a data selector 222, and a control symbol generator 223, and receive components, such as a decoder 225 and a control symbol identifier 226. The link layer may be responsible for generating and transmitting and for receiving and identifying, control symbols, synchronization symbols, timing symbols, and so on, as described below in more detail. In one embodiment, however, the physical layer is responsible for detecting control symbols and synchronization symbols. The transport layer may indicate to the link layer when to transmit control symbols, and the link layer may indicate to the transport layer when control symbols are received. For example, the transmit layer may notify the link layer when the receive memory is full. In such a case, the link layer may transmit a control symbol (e.g., XOFF) notifying the other end of the communications link not to

IPR2024-001428
HP – EX1002, Page 373 of 487

transmit any more data. When such transmitted control symbol is received, the link layer of the other end may indicate to its transport layer to stop transmitting data. The data selector selects symbols encoded from codes provided by the transport layer or symbols generated by the control symbol generator. The data selector effectively inserts control symbols and synchronization symbols into the sequence of symbols that are to be transmitted. Various possible encoding techniques for the codes are described below in detail. The data decoder receives symbols from the physical layer and decodes them into codes that are provided to the transport layer. The control symbol identifier identifies control symbols and signals the transport layer as appropriate.

[0080] The physical layer includes a transmitter 231 and receiver 235. The transmitter serializes the symbols provided by the link layer and transmits the serialized symbols onto the communications link. The receiver receives the serialized symbols via the communications link, deserializes the symbols, and provides the symbols to the link layer.

[0081] The transmit components and the receive components can transmit and receive packets in full duplex mode. That is, a packet received by the transport layer can be transmitted by the transmit components simultaneously with a packet being received by the receive components. As discussed below in detail, the encoding techniques enable transmitting of control symbols by the link layer for link control (e.g., flow control) even while packets are being transmitted in full duplex mode. That is, link control symbols can be inserted into a sequence of symbols generated from codes provided by the transport layer. Thus, a communications link can be simultaneously transmitting and receiving data symbols, and control symbols can be inserted for link control. Prior transmission techniques typically transmitted in half-duplex mode because the receiving link was reserved for transmission of control information. Thus, the Storage Link architecture has a significant bandwidth advantage because of its full-duplex mode.

IPR2024-001428
HP – EX1002, Page 374 of 487

[0082] Figure 3 is a block diagram illustrating components of the physical layer in one embodiment. The physical layer uses a plesiosynchronous timing mechanism to control the transmission and reception of the symbols. A plesiosynchronous-based physical layer is described in U.S. Patent Number 6,229,859, entitled "System and Method for High-Speed Synchronized Data Communication," which is hereby incorporated by reference.

[0083] Improvements to the plesiosynchronous timing mechanism are described below in detail. For example, one improvement is the insertion and removal of NULL symbols, which renders the use of elastic buffers unnecessary. Another improvement is the sharing of a single phase lock loop by multiple ports, which avoids the complexity and cost of having one phase lock loop for each port. The physical layer 230 includes a transmitter 231, a receiver 235, a phase lock loop 301, and a clock 302. The phase lock loop provides a timing signal to the transmitter for serial transmission of the symbols and provides multiple timing signals with different phases to the receiver for receiving the serially transmitted symbols. One skilled in the art will appreciate that a delay lock loop, or more generally any multiphase clock generating device, can be used in place of the phase lock loop. The receiver may include an oversampler 303, a digital phase lock loop 304, and a frame aligner 305. The digital phase lock loop selects the best sample by a process known as majority voting. Majority voting counts transitions from one oversampled bit to another. In one embodiment, each bit is oversampled three times and the position with the highest number of transitions is selected. The digital phase lock loop thus identifies the bit values of the received symbols, and the frame aligner aligns the bits into symbols and provides the deserialized symbols.

[0084] Figure 4 is a block diagram illustrating the format of a packet in one embodiment. The packet 400 includes a header section 410 and a payload section 420. The header section includes reserved fields 411 and 412 (e.g., 16 bits each), a destination address field 413 (e.g., 16 bits), a source address field 414 (e.g., 16 bits), a type field 415 (e.g., 8 bits), a protocol field 416 (e.g., 8 bits), a

IPR2024-001428
HP – EX1002, Page 375 of 487

class specific control field ("CSF") 417 (e.g., 8 bits), and a transaction (or queue) index field 418 (e.g., 8 bits). One skilled in the art will appreciate that the actual size of the fields can vary depending on the desired performance characteristics. For example, the destination and source address fields can be limited to 4 bits when there are at most 16 hosts or data store devices. The class-specific control field contains the control field of the interface class (e.g., SCSI, ATA), which may be unused in some interface classes. The protocol field identifies the appropriate protocol for the upper layer and identifies the protocol of the storage media (e.g., SCSI, ATA, or Storage Link). The destination address field identifies the host or data store device to which the packet is to be transmitted. The source address field identifies the host or data store device that originated the transmission of the packet. In one embodiment, the reserved fields may be used in conjunction with the source address field and the destination address field to permit 32-bit addresses. The type field identifies the type of the packet. The types of packets may include a data packet, a command packet, a status packet, and a message packet. The command packet, status packet, and message packet are "control" packets. Control packets are generally much smaller than data packets. The type field further specifies the direction of the packet transmission (i.e., whether it is being transmitted from a host to a data store device, or vice versa) and specifies whether the packet is intended for the physical layer, link layer, transport layer, or application layer. For example, an application layer packet may be a command packet sent from a host device to a data store device requesting that data be retrieved from the data store device. The payload section contains type-specific information. For example, the payload section of a command packet for retrieving data from a data store device may include the starting address of the data to be retrieved and the number of bytes of data to be retrieved. The payload section of a data packet transmitted in response to receiving a command packet may contain the retrieved data in the payload section. The control field contains type-specific information that may be used to control the data store device. The transaction index field identifies a series of packets that correspond to the same

IPR2024-001428
HP – EX1002, Page 376 of 487

transaction. For example, a transaction may be the writing of data to a data store device. A host may transmit the data to be written in multiple packets that form a single transaction. Each packet within the transaction contains the same transaction index so that the data store device can identify that the packets belong to the same transaction. Although not shown, a packet may also include a cyclic redundancy check (CRC) field after the payload section.

[0085] In various embodiments, the size of the header portion can vary based on the number of hosts and data store devices that are addressable. For example, if a single host is directly connected to a single data store device, then a small header including the type field, control field, protocol field, a class specification control field and transaction index field would be sufficient. The destination and source address fields and reserved fields would not be needed. When there are either multiple hosts or multiple data store devices, then the destination and source address fields would be included in the header. If more hosts and data store devices are addressable than can be addressed by 16 bits, then the reserved fields can be included in the header. Thus, the size of the header can vary depending on the number of possible sources and destinations. In one embodiment, the transport layer may be configurable to include a small, medium, or large header depending on the number of possible sources and destinations.

[0086] A packet that is transmitted on a serial communications link typically comprises in-band symbols with certain desirable characteristics, such as with its number of bit transitions optimized, as described below in more detail. The physical layer continually transmits symbols onto the serial communications link. If a packet is not available to be transmitted, then the link layer provides IDLE symbols to the physical layer for transmission. In one embodiment, an IDLE symbol is an out-of-band symbol, which is a symbol that does not normally appear within a packet because it may not have the same desirable characteristic as an in-band symbol (e.g., bit transitions not optimized). A receiving link layer identifies the start of a packet by a transition from receiving IDLE symbols to receiving in-band symbols. Similarly, a receiving link layer identifies the end of a packet by a

IPR2024-001428
HP – EX1002, Page 377 of 487

transition from receiving in-band symbols to receiving IDLE symbols. In one embodiment, a IDLE symbol is actually a sequence of two out-of-band symbols for reasons described below in detail. More generally, a sequence of two out-of-band symbols is a primitive that is used to exchange control information (e.g., XON and XOFF) between link layers as described below in detail. Because control information is encoded as out-of-band symbols, the control symbols can be inserted into a sequence of in-band symbols and readily detected as control symbols when received. The control symbols can be interspersed with the transmission of symbols of a packet so that one end of a communications link can provide link control information to the other end even while packets are being transmitted in both directions (i.e., full-duplex mode)

EXPRESS MAIL NO. EL 954 074 204 US

1.1 Packet Segmentation

[0087] In one embodiment, a packet (e.g., long packets, such as data packets) may be dynamically divided or segmented into multiple packets by a communications node (e.g., a host, data store device, or switch) as the packet is being transmitted from a source to a destination. The dynamic segmentation of packets allows for the efficient transmission of large packets via a single path when the switching network is not congested and allows for the efficient transmission of segmented packets on different paths when congestion prevents use of a single path. In addition, dynamic segmentation may avoid the resending of portions of the payload when segmenting a packet.

[0088] A packet may be segmented by a communications node when, for example, the next communications node in the path indicates that it cannot receive any more symbols of a packet. The transmitting communications node may receive an XOFF signal from the receiving communications node (e.g., because of a buffer overflow at the receiving communications node) in the middle of transmitting a packet. The transmitting communications node stops transmitting the packet until an XON signal is received from the receiving communications node or unless it can start transmitting the remainder of the packet on a different communications link (i.e., through a different port). In either

IPR2024-001428
HP – EX1002, Page 378 of 487

case, the transmitting communications node transmits a new header with the remainder of the packet, which results in the segmentation of the packet into a first packet with a first header and a second packet with a second header. The packet segmentation is performed by the transport layer based on signals provided by the link layer (e.g., in response to receiving an XOFF signal).

[0089] Figure 5 is a block diagram illustrating the organization of a data packet and a segmented data packet in one embodiment. The payload section of a data packet may include a target address and blocks of data to be stored at the target address. For example, if the data store device is a disk drive, then the target address may be a sector address of the disk drive and the blocks of data may each correspond to a sector of data. In one embodiment, the blocks have a fixed size (e.g., the sector size). Packet 510 includes a header section 500 and a payload section 511 that includes an address 500a and blocks 501-506. When a communications node is transmitting packet 510, it may transmit the header section and address 500a and blocks 501-503 of the payload section. During transmission of block 504, the transmitting communications node may receive an XOFF signal from the receiving communications node. (The receiving communications node may transmit an XOFF control primitive.) The transmitting communications node completes the transmission of block 504 and then stops transmitting the packet on that serial communications link. (The link layer then starts transmitting an IDLE symbol on the communications link.) When the transmitting communications node eventually receives an XON signal from the receiving communications node, it transmits a new header section 531 plus a payload section indicating a new address 531a and blocks 505 and 506 to complete the transmitting of the packet. The new address may be represented by an offset that is added to a start address to generate the new address. The new header may contain the same header information as the previously transmitted header, but the address (e.g., as an offset) of the payload section is updated to reflect the starting address at which the remaining blocks of the packet are to be stored. For example, if packet 510 is segmented after block 504 and the starting

IPR2024-001428
HP – EX1002, Page 379 of 487

address for that packet is address 211, then the starting address for the packet containing blocks 505 and 506 is 215 (i.e., 211+4). Packet 510 is thus segmented into packets 520 and 530. When a packet is segmented, the transport layer may route the remaining packet on a different communications link to the destination. The use of different communications links helps to optimize the use of available bandwidth.

[0090] In addition to segmenting packets, the transport layer may merge packets that were previously segmented. For example, if packet 520 is currently stored in a receive memory of a switch awaiting transmission and packet 530 is received at the switch, then the transport layer may recognize that the packets are part of the same transaction (e.g., based on source addresses, destination address, and transaction index) and that their blocks are contiguous as indicated by their target addresses. The transport layer can merge packets 520 and 530 to form packet 510. One skilled in the art will appreciate that packets 520 and 530 can be merged regardless of the order in which they were received at the switch. Also, one skilled in the art will appreciate that if a switch is in the process of transmitting packet 520 and packet 530 is received, the packets can be merged by transmitting blocks 505 and 506 immediately after block 504. One skilled in the art will also appreciate that a transport layer can merge packets stored in a transmit memory. In this way, packets received via different ports of a switch can be merged. The merging of packets allows data of the same transaction to be more efficiently transmitted when traffic on the switch network allows.

[0091] Figure 6 is a block diagram illustrating a receive packet memory before and after merging of a received packet. Packet 600 forming transaction T1 initially contains 8 blocks. During transmission, it is segmented into packets 614, 630, and 612. Packets 614 and 612 are currently buffered at a switch which then receives packet 630. Because the switch has all three packets, they can be merged into a single packet. Before the merge, packet memory 610 contains 4 packets. Packet 611 is part of transaction T3 and contains 5 blocks of data to be stored starting at address 25. Packet 612 is part of transaction T1 and contains 3

IPR2024-001428
HP – EX1002, Page 380 of 487

blocks of data to be stored starting at address 4. Packet 613 is part of transaction T2 and contains 10 blocks of data to be stored starting at address 50. Packet 614 is part of transaction T1 and contains 2 blocks of data to be stored starting at address 0. Packets 612 and 614 are both part of transaction T1. These packets, however, are not contiguous packets within transaction T1 and thus cannot be merged together. Packet 630 represents a packet just received that is part of transaction T1 and contains 2 blocks to be stored starting at address 2. Packet 630 is contiguous to both packets 614 and 612. That is, packet 630 can be merged onto the end of packet 614 and packet 612 can be merged onto the end of packet 630. Packet memory 620 contains the packets after the merge. Packets 611 and 613 have been left unchanged. Packet 614, however, has been modified to add the blocks from packet 630 and from packet 612. Packet 614 indicates that is the part of transaction T1 and contains 7 blocks of data to be stored starting at address 0. Packet 612 has been deleted from the packet memory because it has been merged with packet 614, and packet 630 has not been separately added to the packet memory because it has been merged with packet 614.

[0092] Figure 7 is a logic diagram illustrating the segmentation of a data packet by the transport layer in one embodiment. In block 701, a component of the transport layer waits for a block of data. In decision block 702, if an XOFF signal has been received, then the component waits for an XON signal in block 703, else the component continues at block 704. In block 704, the component sends the header to the link layer for transmission. In blocks 705-707, the component transmits successive blocks of the packet and segments the packet when an XOFF signal is received from the link layer. In block 705, the component transmits the next block of the packet. In decision block 706, if an XOFF signal is pending as indicated by the link layer, then the component continues to segment the packet at block 708, else the component continues at block 707. In decision block 707, if the next block is ready, then the component loops to block 705 to transmit the next block, else the component segments (or ends) the packet at

IPR2024-001428
HP – EX1002, Page 381 of 487

block 708. In block 708, the component transmits the CRC to end the packet and waits for the next block in 701.

[0093] Figure 8 is a logic diagram illustrating the merging of data packets by the transport layer in one embodiment. In block 801, a component of the transport layer receives a data packet from the link layer. In blocks 802-807, the component checks to see if the data packet can be merged with a previously received data packet that is stored in the receive memory. In block 802, the component selects the next data packet in the receive memory. In decision block 803, if all the data packets of the receive memory have already been selected, then the component continues at block 808, else the component continues at block 804. In decision block 804, if the selected and received data packets are contiguous, then the component continues at block 805, else the component selects the next data packet in the receive memory. In block 806, the component adds the blocks of the selected packet to the receive packet to affect the merging of the blocks. In block 807, the component removes the selected packet from the receive memory. The component then selects the next data packet because there may be another data packet that can be merged at the beginning or end of the received data packet. In block 808, the component adds the received data packet to the receive memory and then completes.

FOLEY-EPSON1

1.2 Synchronization Primitive and Packet Type

[0094] In one embodiment, the serial communications architecture defines multiple primitives to operate as synchronization signals. A synchronization signal is typically used by the physical layer to help ensure correct alignment of symbols. A transmitting communications node periodically transmits synchronization primitives so that the receiving communications node can properly align with that synchronization primitive. When a communications node receives a synchronization primitive, it knows that the primitive is correctly aligned on a symbol boundary. The use of multiple synchronization primitives allows for encoding packet type within a synchronization primitive. In particular, the transport layer provides the packet type to the link layer. The link layer can then

IPR2024-001428
HP – EX1002, Page 382 of 487

transmit a synchronization primitive that indicates packet type immediately before the packet itself. The receiving communications node can take the appropriate action based on the packet type encoded in the synchronization primitive without having to parse the packet header. Indeed, the packet header may not need to separately include the packet type. The use of a synchronization primitive with encoded packet type may be well adapted to use in an environment where the transmission time of a packet through the switching network may be long relative to the time needed to store the packet at the data store device. For example, the time to transmit a packet through the switching network may be longer than the time needed to store the data of the packet in a memory-based data store device. The rapid detection of packet type through a synchronization primitive, rather than through a packet header, can reduce the processing time of a switch. Also, when a synchronization primitive encodes the packet type, then the link layer, rather than the transport layer, can detect packet type, which can enhance the overall processing speed.

[0095] Figure 9A is a diagram illustrating the use of a synchronization primitive encoded with the packet type in one embodiment. Serial link content diagram 900 indicates the information transmitted on a serial link over a period of time. Initially, an IDLE primitive is transmitted onto the communications link. (An IDLE primitive may be a synchronization primitive that does not encode any packet type.) Immediately prior to transmitting control packet 902 (e.g., command, status, or message packets), the transport layer of the transmitting communications node signals the link layer that a control packet is to be transmitted next. The link layer then transmits a synchronization primitive 901 encoded with a packet type of control. After the packet has been transmitted, the link layer of the transmitting communications node starts transmitting the idle primitive. Immediately before the transport layer transmits data packet 904, it notifies the link layer that a data packet is to be transmitted. The link layer transmits a synchronization primitive 903 encoded with a packet type of data. After the data packet is transmitted, the link layer starts transmitting the IDLE primitive. The receiving communications

IPR2024-001428
HP – EX1002, Page 383 of 487

node uses the synchronization primitive encoded with packet type to align its symbol boundaries and to rapidly identify the packet type. This alignment and identification can occur in the physical and link layers.

[0096] As described above, the primitives are pairs of out-of-band symbols. (In the following, a symbol is assumed to have nine bits.) In one embodiment, when the in-band symbols are transition maximized, then the symbols of the control primitives have three transitions each and the symbols of the synchronization primitives have two transitions each. When the in-band symbols are transition minimized, then the symbols of the control primitives have five transitions each and the synchronization primitives have one symbol with five transitions and another symbol with seven transitions.

[0097] Table 1 illustrates control primitives in one embodiment when the in-band symbols are transition maximized. Each control primitive comprises two symbols in which the first symbol is a polarity inversion of the second symbol. As a result, the disparity (described below) of each control primitive is 0. The control primitives such as XON, XOFF, preempt, and continue are selected from this table of control primitives.

Table 1: Control primitives in transition maximization mode

0.	000010111 – 111101000
1.	000011011 – 111100100
2.	000011101 – 111100010
3.	000100111 – 111011000
4.	000110011 – 111001100
5.	000111001 – 111000110
6.	001000111 – 110111000
7.	001100011 – 110011100
8.	001110001 – 110001110
9.	010000111 – 101111000
10.	011000011 – 100111100
11.	011100001 – 100011110
12.	100001110 – 011110001
13.	100011100 – 011100011
14.	100111000 – 011000111
15.	101110000 – 010001111
16.	110000110 – 001111001
17.	110001100 – 001110011
18.	110011000 – 001100111

IPR2024-001428
HP – EX1002, Page 384 of 487

19.	110110000 – 001001111
20.	111000010 – 000111101
21.	111000100 – 000111011
22.	111001000 – 000110111
23.	111010000 – 000101111

[0098] Table 2 illustrates control primitives in one embodiment when the in-band symbols are transition minimized.

Table 2 - Control primitives in transition minimization mode

0.	000010101 – 111101010
1.	000100101 – 111011010
2.	000101001 – 111010110
3.	000101011 – 111010100
4.	000101101 – 111010010
5.	000110101 – 111001010
6.	001000101 – 110111010
7.	001001001 – 110110110
8.	001001011 – 110110100
9.	001001101 – 110110010
10.	001010001 – 110101110
11.	001010011 – 110101100
12.	001011001 – 110100110
13.	001100101 – 110011010
14.	001101001 – 110010110
15.	010000101 – 101111010
16.	010001001 – 101110110
17.	010001011 – 101110100
18	010001101 – 101110010
19.	010010001 – 101101110
20.	010010011 – 101101100
21.	010011001 – 101100110
22.	010100001 – 101011110
23.	010100011 – 101011100
24.	010110001 – 101001110
25.	011000101 – 100111010
26.	011001001 – 100110110
27.	011010001 – 100101110
28.	100001010 – 011110101
29.	100010010 – 011101101
30.	100010100 – 011101011
31.	100010110 – 011101001
32.	100011010 – 011100101
33.	100100010 – 011011101
34.	100100100 – 011011011
35.	100100110 – 011011001
36.	100101000 – 011010111

PD/07/47/6SC004

IPR2024-001428
HP – EX1002, Page 385 of 487

37.	100101100 – 011010011
38.	100110010 – 011001101
39.	100110100 – 011001011
40.	101000010 – 010111101
41.	101000100 – 010111011
42.	101000110 – 010111001
43.	101001000 – 010110111
44.	101001100 – 010110011
45.	101010000 – 010101111
46.	101011000 – 010100111
47.	101100010 – 010011101
48.	101100100 – 010011011
49.	101101000 – 010010111
50.	110001010 – 001110101
51.	110010010 – 001101101
52.	110010100 – 001101011
53.	110100010 – 001011101
54.	110100100 – 001011011
55.	110101000 – 001010111

FOUNDRY-1650.PDF

[0099] The synchronization primitives are distinct from the control primitives. The synchronization primitives are selected to have the characteristic that when two synchronization primitives are transmitted sequentially the receiving device can unambiguously identify the synchronization primitives regardless of what is transmitted before or after the synchronization primitives. The synchronization primitives have a unique number of transitions that is not found in any control primitive or in-band symbol. In one embodiment, the synchronization primitive can be detected by looking for a sequence of symbols with that number of transitions. In one embodiment, the synchronization primitives are selected to start with a falling edge (i.e., "10"), a rising edge (i.e., "01"), a high level (i.e., "11"), or a low level (i.e., "00"). The selection of the start of the synchronization primitives is referred to as the "synchronization detection mode." Table 3 contains synchronization primitives in transition maximization mode that start with a falling edge and a rising edge. Each symbol in Table 3 contains two transitions resulting in synchronization primitives with four transitions. Table 4 contains synchronization primitives in transition minimization mode that start with a low level and a high level. Each synchronization primitive in Table 4 has

IPR2024-001428
HP – EX1002, Page 386 of 487

13 transitions. The high-level and low-level synchronization primitives have five transitions in one symbol, seven transitions in the other symbol, and one transition from their first to second symbol.

Table 3

	Falling Edge	Rising Edge
0	100000001 – 100111111	011000000 – 011111110
1	100000011 – 100011111	011100000 – 011111100
2	100000111 – 100001111	011110000 – 011111000
3	100001111 – 100000111	011111000 – 011110000
4	100011111 – 100000011	011111100 – 011100000
5	100111111 – 100000001	011111110 – 011000000

Table 4

	High Level	Low Level
0	110101000 – 110101010	001010101 – 001010111
1	110100010 – 110101010	001010101 – 001011101
2	110001010 – 110101010	001010101 – 001110101
3	110100100 – 110101010	001010101 – 001011011
4	110010100 – 110101010	001010101 – 001101011
5	110101010 – 110101000	001010111 – 001010101
6	110101010 – 110100010	001011101 – 001010101
7	110101010 – 110001010	001110101 – 001010101
8	110101010 – 110100100	001011011 – 001010101
9	110101010 – 110010100	001101011 – 001010101

[00100] Table 5 lists the transition distribution of the low and high polarity symbols of the synchronization primitives.

IPR2024-001428
HP – EX1002, Page 387 of 487

Table 5

Sync Detect Mode	Transition Distribution in Low Polarity	Transition Distribution in High Polarity	Total # of Transitions in a Sync Primitive
Falling edge ("10")	one "10" and one "01"	one "10" and one "01"	4
Rising edge ("01")	one "01" and one "10"	one "01" and one "10"	4
High level ("11")	one "11" and two "00"	one "11" and no "00"	13
Low level ("00")	one "00" and no "11"	one "00" and two "11"	13

[00101] In one embodiment, the synchronization primitives are detected by buffering three symbols (e.g., 27 bits). The synchronization primitives are selected so that any time the buffer has exactly three of the synchronization detection mode bits (e.g., "10" for falling edge) that are each 9 bits apart then the bits are at the beginning of a synchronization symbol. The bit buffer contains 27 bits, A[8:0], B[8:0], and C[8:0], in which bit A[8] was received first and bit C[0] was received last. In this example, the content of bits A[6]:B[7] represent synchronization symbol "100000001" and B[6]:C[7] represent synchronization symbol "100111111." The starting points of the synchronization symbols ("synchronization markers") are A[6], B[6], and C[6].

[00102] Figure 9C is a block diagram illustrating a component of a link layer for inserting a synchronization symbol into a stream of symbols. The link layer 910 is provided with the bits of the code ("Data") to be transmitted and with an indication of a control primitive ("CTL-S") to transmit. The idle inserter 910 of the link layer automatically inserts an IDLE primitive (e.g., synchronization primitive) to ensure that not too may data symbols have been sent without sending a synchronization primitive. In one embodiment, the transmitter may send an IDLE primitive after every block of a packet. If packets are short, the inter-packet gap ("IPG") is sufficient for clock compensation. That is, the receiver will receive synchronization primitives often enough to maintain proper synchronization. If a packet is long, however, the link layer can periodically insert a synchronization primitive to ensure

FOLOFF-PF6SE0OF

IPR2024-001428
HP – EX1002, Page 388 of 487

that the receiver maintains proper synchronization. The multiplexer of the link layer selects the appropriate symbol to send to the physical layer.

1.3 Transmission Order Based on Packet Type

[00103] In one embodiment, a communications node transmits packets of certain types before packets of other types regardless of the order in which the packets are received or generated by the communications node. For example, a switch may transmit control packets (e.g., command packets, status packets, and message packets) before transmitting data packets. The transmitting of control packets before data packets may help ensure that the control packets are not unnecessarily delayed by data packets, which may have a tendency to be quite large. For example, a command packet requesting a read from a data store device may be transmitted before a data packet even through the control packet was received after the data packet. The data store device can thus start processing the read request without having to wait until the data packet is transmitted. A communications node may store control packets and data packets in separate areas of the packet memory and only select packets from the data packet area when the control packet area is empty. Alternatively, the communications node may use various algorithms to select control or data packets to be transmitted. For example, the algorithm may select a data packet that has been in the packet memory for certain amount of time even though there may be a control packet in the control area. In this way, a sequence of many control packets will not delay the data packet more than that certain amount of time. One skilled in the art will appreciate that there may be multiple packet types with different transmission orders. For example, a status packet might be transmitted before a command packet, and a command packet might be transmitted before a data packet.

[00104] Figure 10 is a block diagram illustrating an organization of a packet memory in one embodiment. The transmit packet memory 211 may include control queue 1010 and data queue 1020. The control queue contains the control packets 1011, and the data queue contains the data packets 1021. Packets to be

10/26/01, 16:26:00

IPR2024-001428
HP – EX1002, Page 389 of 487

transmitted are stored either in the control queue or the data queue based on packet type (which may be encoded in a synchronization primitive). When a transmitting communications node is ready to transmit the next packet, it retrieves the next packet from the control queue. If the control queue is empty, then the transmitting communications node retrieves the next packet from the data queue.

[00105] Figure 11 is a logic diagram illustrating a process of storing packets in a transmit memory in one embodiment. Similar logic can be used to store packets in the receive memory of a communications node. In block 1101, a component of the transport layer receives the next packet from an upper layer. In decision block 1102, if the received packet is a control packet, then the component continues at block 1103, else the component continues at block 1104. The component may determine the type of the packet either from the type field of the packet header or, when the packet is received from another communications node, from a synchronization primitive encoded with the packet type. When the link layer detects a synchronization primitive designating packet type, it signals the transport layer so that the type of packet can be used when the packet is stored in the receive memory, retrieved from the receive memory, stored in the transmit memory, and retrieved from the transmit memory. In block 1103, the component stores the packet in the control queue. In block 1104, the component stores the packet in the data queue. The component then retrieves the next packet.

[00106] Figure 12 is a logic diagram illustrating the transport layer processing of packets stored in the packet memory in one embodiment. In decision block 1201, if the control queue is empty, then the component continues to block 1203, else the component retrieves a control packet from the control queue in block 1202 and continues at block 1205. In decision block 1203, if the data queue is empty, then the component again checks for a packet in the control queue, else the component retrieves a packet from the data queue in block 1204 and continues at block 1205. In blocks 1205-1207, the component selects the codes of the retrieved packet. In block 1205, the component selects the next code of the retrieved packet. In decision block 1206, if all codes have already been selected,

IPR2024-001428
HP – EX1002, Page 390 of 487

then the component continues at block 1201 to check for the next packet, else the component continues at block 1207. In block 1207, the component provides the selected code to the link layer and then continues at block 1205 to select the next code of the packet.

1.4 Packet Preemption

[00107] In one embodiment, a communications node may preempt or interrupt the transmission of a packet when a packet with a higher priority is to be transmitted on the same communications link. Packet preemption helps ensure that data store devices receive control packets as soon as possible without being slowed by network congestion as a result of data packets. In particular, the transmission of a data packet, which may be large, may be preempted so that a control packet, which is typically small, may be transmitted. When a data packet is to be preempted, the transport layer of the communications node stops providing the codes of the data packet to the link layer, signals the link layer to transmit a preempt primitive, and then starts providing the code of the control packet to the link layer for transmission as a preempting packet. When the preempting packet has been completely transmitted, the transport layer of the communications node signals the link layer to transmit a continue primitive and then resumes providing the remainder of the codes of the data packet to the link layer. The link layer of the receiving communications node detects the preempt primitive while it is receiving the codes of the data packet and signals its transport layer. The transport layer then starts storing the subsequent codes as a control packet. When the link layer detects the continue primitive, it signals the transport layer, which continues receiving the preempted data packet. The preempt and continue primitives, which are control primitives, serve to delimit transmission of a preempting packet. One skilled in the art will appreciate that packet preemption could be nested to any level. For example, a status packet might preempt a command packet that already preempted a data packet.

[00108] Figure 13 is a diagram illustrating the preemption of a packet in one embodiment. Serial link content diagram 1300 illustrates a data packet that is

FOLGTF-T5E00T

IPR2024-001428
HP – EX1002, Page 391 of 487

preempted by a control packet. Initially, the transmitting communications node transmits the first portion of data packet 1301. When the packet is to be preempted, the communications node then transmits the preempt primitive 1302. After transmitting the preempt primitive, the transmitting communications node transmits the preempting control packet 1303. After the control packet is transmitted, the communications node transmits the continue primitive 1304, and then resumes transmitting the remainder of the preempted data packet 1305.

[00109] Figure 14 is a logic diagram illustrating the processing of packet preemption by a transmitting communications node in one embodiment. In block 1401, the component of the transport layer retrieves the next packet from the transmit memory. In block 1402, the component retrieves the next code of the retrieved packet. In decision block 1403, if all the codes have already been retrieved from the packet, then the component continues at block 1401 to retrieve the next packet, else the component continues at block 1404. In block 1404, the component provides the retrieved code to the link layer. In decision block 1405, if a preempt signal has been raised (e.g., as a result of being provided a control packet by an upper layer), then the component continues at block 1406, else the component continues at block 1402 to retrieve the next code of the packet. In block 1406, the component notifies the link layer to transmit a preempt primitive. In blocks 1407-1409, the component provides each code of the preempting packet to the link layer. In block 1407, the component selects the next code of the preempting packet. In decision block 1408, if all the codes of the preempting packet have already been selected, then the component continues at block 1410, else the component continues at block 1409. In block 1409, the component provides the selected code to the link layer and then continues at block 1407 to select the next code of the preempting packet. In block 1410, the component signals the link layer to transmit a continue primitive and then continues processing the preempted packet at block 1402.

[00110] Figure 15 is a logic diagram illustrating processing of packet preemption by a receiving communications node in one embodiment. In block 1501, the

IPR2024-001428
HP – EX1002, Page 392 of 487

component of the transport layer retrieves the next code from the link layer. In decision block 1502, if the link layer signals that a preempt primitive has been received, then the component continues at block 1504, else the component continues at block 1503. In block 1503, the component adds the code to the current packet and retrieves the next code in block 1501. In block 1504, the component saves current packet information so that it can continue storing codes in the packet after the preemption is complete. In block 1505, the component processes the preempting packet. The processing includes receiving each code and storing the codes in a preempting packet until the link layer indicates that it received a continue primitive. In block 1505, the component restores the saved packet information so that subsequent received codes can be stored in the preempted packet. The component then selects the next code of the preempted packet in block 1501.

1.5 Asymmetric Packet Ordering

[00111] In one embodiment, the serial communications architecture may guarantee that packets (in particular packets of the same transaction) transmitted from a host to a data store device are received at the data store device in the order in which they were transmitted from the host. In contrast, the serial communications architecture might not guarantee that packets transmitted from a data store device to a host are received at the host in the same order that they were transmitted from the data store device. The guarantee of order in one direction, but not in the other direction, is referred to as asymmetric packet ordering. Because the packets are received at the data store device in order, the data store device does not need the hardware or software necessary to reorder packets that are received out-of-order. As a result, the overall cost of the data store device can be reduced. The guarantee of order may be provided by requiring that all packets of the same transaction transmitted from a host to a data store device travel on the same path. In contrast, packets of the same transaction transmitted from a data store device to a host may each travel on different paths. Because the packets transmitted from a host to a data store device travel along the same path, the actual time to

IPR2024-001428
HP – EX1002, Page 393 of 487

transmit the packets may take longer than if alternate paths could be used. For example, one communications link in the path may become congested with no alternative but to wait for the congestion to end, whereas if an alternate path was used, the remaining packets could be routed to avoid the congestion; thus, the packets would arrive sooner, but possibly out-of-order. In one embodiment, the serial communications architecture may only guarantee the order of packets transmitted from a host to a data store device within each transaction. Packets within a transaction have a sequential order. The guarantee of order within a transaction may be provided by requiring that all the packets of each transaction to be transmitted from a host to a data store device along the same path. The packets of different transactions can, however, be transmitted along different paths. In this way, a packet within a later transmitted transaction may arrive at the data store device before a packet of an earlier transmitted transaction.

[00112] A host may be a computer system running application programs, and the data store device may be a disk drive. In such an environment, the speed at which data is transmitted to a disk drive may not be as important as the speed at which the data is transmitted to the computer system. In particular, when an application program writes data to a disk drive, the data is normally cached by the computer system before transmission to the disk drive so that the application program can continue its processing without having to suspend its execution. In contrast, when an application program reads data from a disk drive, the application normally suspends its execution until the data is received. Thus, the speed of transmitting data from the disk drive is more important than the speed of transmitting data to a disk drive because it can reduce the time a program is suspended. The asymmetric packet ordering takes advantage of this variation in the need for speed.

[00113] Figure 16 is a block diagram illustrating asymmetric packet ordering in one embodiment. Packets of the same transaction, transmitted from a host 1610 to a data store device 1620 all travel along the same path: switch 1631, switch 1634, and switch 1636. Since the data packets are processed in a first-in-first-out

IPR2024-001428
HP – EX1002, Page 394 of 487

manner by each switch, packet ordering is preserved in transit. In contrast, packets of the same transaction transmitted from a data store device 1620 to a host 1610 may travel along two different paths: (1) a path including switch 1636, switch 1633, and switch 1631 and (2) path including switch 1636, switch 1635, switch 1632, and switch 1631. Packets traveling on different paths may arrive out-of-order at the host.

[00114] Figure 17 is a diagram illustrating transaction-based asymmetric packet ordering in one embodiment. Illustration 1701 illustrates preserving packet ordering within a transaction transmitted from a host to a data store device. In this example, transaction 1 includes packets 1 and 2 and transaction 2 includes packets 1, 2, 3, and 4. Transaction 1 is transmitted from the host before transaction 2. The data store device received the packets of each transaction in order within their transaction, but some of the packets of transaction 2 arrived before the some of the packets of transaction 1. For example, packet 1 of transaction 2 arrived before packet 1 of transaction 1, but after packet 1 of transaction 2. Illustration 1702 illustrates transmitting without preserving packet ordering within a transaction. In this example, transactions 1 and 2 are transmitted from the data store device to the host. The host receives the packets of the transactions out of order. For example, packet 2 of transaction 2 arrived before packet 1 of transaction 2.

[00115] Figure 18 is a logic diagram illustrating processing by a switch that ensures packet ordering within a transaction in one embodiment. In block 1801, a component of the switching layer retrieves the next data packet to be routed. In decision block 1802, if another packet of the same transaction has been transmitted by the switch, then the component identifies the same communications link in block 1803, else the component identifies a communications link for the first packet of that transaction.

1.6 Host Error Handling

[00116] In one embodiment, the communications nodes detect errors but do not attempt to correct or handle the errors. Rather, communications nodes forward

IPR2024-001428
HP – EX1002, Page 395 of 487

error messages to the communications node that initiated the transaction associated with the error. For example, a host may request a data store device, such as a disk drive, to provide data stored at the data store device. If an error occurs (e.g., a CRC error) while the data is being transmitted from the data store device to the host, the communications node (e.g., a switch) that detects the error transmits an error message to the initiating host. The initiating host can then handle to the error as appropriate. For example, the initiating host may retransmit the request to retrieve the data from the data store device. Because the communications node (e.g., data store device or switch) does not attempt to handle errors, the complexity of these communications nodes is reduced. Although the handling of the error by the initiating host may be slow, the slowness may be acceptable because errors are expected to be relatively uncommon.

[00117] Figure 19A is a block diagram illustrating the handling of errors during transmission from a host to a data store device in one embodiment. The host 1901 transmits a packet 1904 to switch 1902. The switch detects an error, transmits an error message packet 1905 to the host reporting the error, and suppresses further transmission of the packet to the data store device 1903. When the host receives the error message packet, it forwards an error message 1906 to the upper layer for processing. Similarly, when the data store device detects an error in the transmission of packet 1907 from the switch, it transmits an error message packet 1908 reporting the error to the switch, which is then transmitted as an error message packet 1909 to the host, and the error message 1910 is eventually reported to the upper layer of the host.

[00118] Figure 19B is a block diagram illustrating the handling of errors during the transmission from a data store device to a host in one embodiment. The switch 1902 detects an error in the transmission of packet 1911 from the data store device 1903. Rather than transmitting an error message packet 1912 to the data store device, the switch transmits an error message packet 1913 to the host 1901 that initiated the transaction of the packet 1911 for which the error was detected. Upon receiving the error message packet, the host forwards an error message

IPR2024-001428
HP – EX1002, Page 396 of 487

1914 to an upper layer for handling. Similarly, when the host detects an error in transmission of packet 1915, it does not transmit error message packet 1916 to the switch 1902, rather it forwards an error message 1917 to the upper layer. As can be seen from Figures 19A and 19B, the switches and data store devices merely generate and forward error message packet, and do not perform error handling to correct the error.

[00119] Figure 19C is a logic diagram illustrating the processing of a communications node detecting an error in one embodiment. In block 1921, a component of a transport layer receives an error signal from the link layer such as from a line indicating a CRC error. In block 1922, the component retrieves packet identifier information associated with the error. The packet identifier information includes the address of the initiating host. In block 1923, the component generates an error packet to transmit to the initiating host. In block 1924, the component transmits the error packet to the initiating host.

Attorney Docket No. 37179-8019

1.7 Transition Control & Selective Block Inversion

[00120] In one embodiment, the link layer of a communications node encodes codes into symbols in order to optimize transition control and to ensure DC-balance. Transition control may be optimized to either maximize or minimize the number of transitions in symbols that are to be transmitted. A transition is defined as a bit value of 1 followed by a bit value of 0 or a bit value of 0 followed by a bit value of 1. When the number of transitions is maximized, clock information may be more accurately derived from the transmitted symbols. When the number of transitions is minimized, overall electromagnetic interference may be minimized. One skilled in the art will appreciate that the link layer may be designed to minimize or maximize transitions depending on the overall design objectives. In the following, the described encoding technique attempts to maximize the number of transitions in symbols to be transmitted. In an alternate embodiment, the transitions can be maximized at times and minimized at other times to balance the objectives of deriving a clock signal and reducing electromagnetic interference.

IPR2024-001428
HP – EX1002, Page 397 of 487

[00121] The encoding technique in one embodiment encodes an 8-bit code based on the number of transitions in that code. The maximum possible number of transitions in an 8-bit code is seven and occurs when the bit values alternate between 0 and 1 (e.g., "10101010"). The minimum possible number of transitions in an 8-bit code is zero and occurs when all the bit values are the same (e.g., "11111111"). To maximize the number of transitions, the encoding technique inverts the bit value of alternate bits in each code with less than four transitions. This alternate bit inversion, also referred to as transition inversion, results in a code that previously had "n" transitions to now have |n-7| transitions. For example, the result of alternate bit inversion of a code with zero transitions (e.g., "11111111") is a code with seven transitions (e.g., "01010101"), and the result of alternate bit inversion of the code with three transitions (e.g., "00010111") is a code with four transitions (e.g., "10111101"). The encoding technique adds a ninth bit (i.e., a transition inversion indicating bit) to the 8-bit code to indicate whether the alternate bits of the code were inverted, which results in a 9-bit symbol. For example, a bit value of 0 in the ninth bit indicates that the code did not have its alternate bits inverted, and a bit value of 1 in the ninth bit indicates that the code did have its alternate bits inverted. The symbol "010111101" has a bit value of 0 in its ninth bit, which indicates that alternate bits were not inverted. In contrast, the symbol "110111101" has a bit value of 1 in its ninth bit, which indicates that alternate bits were inverted. One skilled in the art will appreciate that the ninth bit can be stored at any bit position within the symbol. In this description, the ninth bit is represented as the most significant bit of the symbols.

[00122] When a symbol is received at a communications node, the ninth bit is used to determine whether the symbol needs to have its alternate bits inverted to decode the symbol. Since the symbol "010111101" has a bit value of 0 in its ninth bit, its alternate bits were not inverted and the resulting decoded code is "10111101." Conversely, since the symbol "110111101" has a bit value of 1 in its ninth bit, its alternate bits were inverted and the resulting decoded code is "00010111." All the symbols generated by alternate bit inversion encoding are

IPR2024-001428
HP – EX1002, Page 398 of 487

referred to as "in-band symbols" because they are 9-bit symbols with at least four transitions. All 9-bit symbols with three or less transitions are referred to as "out-of-band symbols." This alternate bit inversion encoding technique generates only in-band symbols.

[00123] Figure 20 illustrates an alternate bit inversion encoding technique in one embodiment. The left column represents the 8-bit codes, and the right column represents the 9-bit symbols. The alternate bit inversion encoding technique is referred to as an 8b/9b encoding, which is a more efficient encoding than standard 8b/10b encodings. Transition control encoding is described in U.S. Patent No. 5,999,571, entitled "Transition-Controlled Digital Encoding and Signal Transmission System," which is hereby incorporated by reference.

[00124] The encoding technique may also help ensure DC-balance by using selective block inversion encoding on symbols that have been transition optimized. Selective block inversion refers to an encoding technique that inverts certain bit values in a block of symbols to help ensure DC-balance is maintained. In one embodiment, selective block inversion may perform polarity inversion (i.e., inverting each bit of a symbol) on all but one symbol of a block to help ensure DC-balance and transition inversion on that one symbol in the block to indicate that the other symbols of the block have been polarity inverted. Although the encoding technique performs selective block inversion on blocks of four symbols, one skilled in the art will appreciate that a block can contain any number of symbols. Conventional encoding techniques, in contrast, typically process only one symbol at a time when attempting to ensure DC-balance. Because the selective block inversion encoding technique evaluates DC-balance over multiple symbols, DC-balance is more likely assured and there is essentially no bandwidth used to indicate whether a symbol has been inverted.

[00125] The selective block inversion encoding technique tracks the running disparity on symbol boundaries of the symbols that have been transmitted so far. The running disparity represents the difference in the overall number of 1-bit values and 0-bit values of the symbols that have been transmitted on the

IPR2024-001428
HP – EX1002, Page 399 of 487

communications link. For example, if 452 1-bit values and 448 0-bit values have been transmitted (e.g., 100 symbols), then the running disparity is +4 (e.g., in the 1-direction or positive). If 448 1-bit values and 452 0-bit values have been transmitted, then the running disparity is -4 (e.g., in the 0-direction or negative). If the number of 1-bit values and 0-bit values are equal, then the running disparity is 0 (e.g., neutral). To determine whether to perform selective block inversion on a block of symbols, the encoding technique compares the running disparity of the symbols transmitted so far to the block disparity of that block. Block disparity is the difference between number of 1-bit values and 0-bit values of the symbols of a block. If the running disparity and the block disparity are both positive or both negative, then the encoding technique performs block inversion on that block.

[00126] The encoding technique inverts a block by inverting alternate bits of the first symbol in the block (i.e., transition inversion) and inverting all the bits of the other symbols in the block (i.e., polarity inversion). The inverting of the alternate bits of the first symbol results in an out-of-band symbol that is used to indicate whether the following block has been block inverted. When a block of symbols is received, the receiving communications node can determine whether the block has been block inverted by checking the first symbol of the block. If the first symbol of the block is an in-band symbol, then the block has not been block inverted. If the first symbol of the block is an out-of-band symbol, then the block has been block inverted. If the block has been block inverted, then the decoding technique undoes the alternate bit inversion of the first symbol by inverting alternate bits of the first symbol and undoes the bit inversion of the other symbols by inverting all of the bits of the other symbols. One skilled in the art will appreciate that the symbol used to indicate whether the block is block inverted need not be the first symbol of the block, but rather can be any symbol of the block so long as the receiving communications node can identify which symbol indicates selective block inversion.

[00127] Figure 21A is a diagram illustrating the selective block inversion encoding technique in one embodiment. In this example, a block is defined as having four

IPR2024-001428
HP – EX1002, Page 400 of 487

symbols. One skilled in the art will appreciate that a block can have any number of two or more symbols. In this example, the block disparity is +6, because there are 21 1-bit values and 15 0-bit values in the block. If the running disparity is positive, then the encoding technique performs block inversion. The first symbol of the block has its alternate bits inverted, and the remaining symbols of the block have all their bits inverted. The resulting symbols have a block disparity of -4. Thus, the DC-balance would be improved by transmitting the inverted block, rather than the noninverted block. The encoding technique may alternatively only perform block inversion when the running disparity is outside a certain range. For example, if the running disparity is between –100 and +100, then no selective block inversion is performed. If outside the range, then selective block inversion is performed. One skilled in the art will appreciate that this range can be selected based on various characteristics of the communications link such as transmission rate or maximum transmission distance.

[00128] Figure 21B is a block diagram illustrating selective block inversion with a programmable block size in one embodiment. In this example, the block size is represented by the variable "k." The selective block inversion encoding technique processes k sequential symbols at a time as indicated by blocks 2101 and 2102. The transition from block 2101 to 2102 illustrates selective block inversion for transmission, and the transition from block 2102 to 2101 illustrates the undoing of selective block inversion upon reception. To perform block inversion, the block header symbol Ci is transition inverted (TI) resulting in symbol Bi and the symbol Ci+1 to symbol Ci+k-1 are polarity inverted (SI) resulting in symbol $\overline{C_{i+1}}$ to symbol $\overline{C_{i+k-1}}$. To undo block inversion, the block header symbol Bi is transition inverted resulting in symbol Ci and the symbols $\overline{C_{i+1}}$ to symbol $\overline{C_{i+k-1}}$ are polarity inverted resulting in symbol Ci+1 and symbol Ci+k+1.

[00129] Figure 21C is a block diagram illustrating transition inversion and polarity inversion in one embodiment. In this example, 8-bit codes are inverted. The code 2110 may be transition inverted in two ways to generate codes 2111 and 2112. Code 2111 results from inverting alternate bits starting with the first bit, and code

IPR2024-001428
HP – EX1002, Page 401 of 487

2112 results from inverting alternate bits starting with the second bit. The code 2110 is polarity inverted by inverting each bit to generate code 2113. The same process is applied to codes 2111, 2112, and 2113 to generate code 2110.

[00130] As discussed above, a primitive is a sequence of two out-of-band symbols. Thus, a receiving communications node can distinguish between a primitive and the first symbol of an inverted block because a primitive is a sequence of two out-of-band symbols and one out-of-band symbol by itself represents the first symbol of an inverted block.

[00131] Figure 22 is a diagram illustrating the format of a primitive in one embodiment. A primitive is defined as a sequence of two out-of-band symbols. The out-of-band symbols are preferably selected to have minimal impact on DC-balance. In particular, the disparity of a primitive is selected to be as close to 0 as possible. To achieve this disparity, one symbol of the primitive may have a negative disparity and the other symbol of the primitive may have any positive disparity. (A symbol with an odd number of bits cannot have a 0 disparity.) When the disparity of a primitive is 0, then the primitive can be transmitted at any time without affecting the running disparity and thus with minimal impact on the DC-balance. In one embodiment, the first symbol of a primitive has the negative disparity (i.e., negative polarity) and the second symbol has the positive disparity (i.e., positive polarity), or vice versa. The receiving communications node can detect a transmission error when a sequence of two out-of-band symbols has an improper ordering for their disparity or both symbols have either a positive or a negative disparity. When a primitive has minimal or no effect on DC-balance, then the complexity of the transmitting communications node may be reduced because the logic to calculate the running disparity need not include primitives in the calculation.

[00132] Figures 23-28 are logic diagrams illustrating encoding and decoding techniques in one embodiment. Figure 23 is a logic diagram illustrating transition control and selective block inversion encoding in one embodiment. A component of the link layer receives codes to be transmitted one code at a time from the

IPR2024-001428
HP – EX1002, Page 402 of 487

transport layer. In block 2301, the component performs transition control optimization to generate a symbol from the code. In block 2302, the component buffers the symbols until a complete block of symbols has been buffered. In decision block 2303, if a complete block of symbols has been buffered, then the component continues at a block 2304, else the component waits to receive the next code. In decision block 2304, if the running disparity is equal to the block disparity, then the component continues at block 2305, else the component continues at block 2306. In block 2305, the component performs block inversion on the symbols of the block. In block 2306, the component provides the symbols of the buffer to the physical layer and then waits to receive the next code from the transport layer.

[00133] Figure 24 is a logic diagram illustrating transition control optimization in one embodiment. In block 2401, the component receives a code for transition optimization. In block 2402, the component counts the number of transitions in the received code. In block 2403, if the number of transitions is less than four (assuming an 8-bit code), then the component continues at block 2405, else the component continues at block 2404. In block 2404, the component sets the ninth bit to a 0-bit value. In block 2405, the component performs alternate bit inversion on the received code by inverting every other bit. In block 2406, the component sets the ninth bit to a 1-bit value. In block 2407, the component generates the symbol as the ninth bit value plus the code.

[00134] Figure 25 is a logic diagram illustrating block inversion in one embodiment. In block 2501, the component selects the first symbol of the block. In block 2502, the component performs alternate bit inversion on the selected symbol by inverting alternate bits. In block 2503, the component selects the next symbol of the block. In decision block 2504, if all the symbols have already been selected, then the component completes, else the component continues at block 2505. In block 2505, the component inverts the selected symbol by inverting the value of each bit in the selected symbol. The component then selects the next symbol of the block.

IPR2024-001428
HP – EX1002, Page 403 of 487

[00135] Figure 26 is a logic diagram illustrating transition control and selective block inversion decoding in one embodiment. The component of the link layer is provided with one symbol at a time by the physical layer. In block 2601, the component stores the symbol in a buffer. In decision block 2602, if the buffer contains a primitive, then the component continues at block 2603 to process the primitive, else the component continues at block 2604. In block 2604, if the buffer contains a complete block, then the component continues at block 2605, else the component waits for the next symbol. In decision block 2605, if the first symbol in the buffer is an out-of-band symbol, then the transmitting communications node inverted the block and the component continues at block 2606, else the component continues at block 2607. In block 2606, the component undoes the block inversion of the block. In block 2607, the component selects the next symbol of the block. In decision block 2608, if all the symbols have already been selected, then the component waits for the next symbol, else the component continues at block 2609. In block 2609, the component undoes the optimization of the transitions of the selected symbol to generate the corresponding code. In block 2610, the component provides the code to the transport layer and then processes the next symbol of the block.

[00136] Figure 27 is a logic diagram illustrating the undoing of block inversion in one embodiment. In block 2701, the component selects the first symbol of the block. In block 2702, the component inverts alternate bits of the selected symbol. In block 2703, the component selects the next symbol of the block. In decision block 2704, if all the symbols have already been selected, then the component completes, else the component continues at block 2705. In block 2705, the component inverts all the bits of the selected symbol and then processes the next symbol of the block.

[00137] Figure 28 is a logic diagram illustrating the undoing of transition control optimization in one embodiment. In decision block 2801, if the ninth bit of the symbol is one, then the component inverts alternate bits of the symbol in block 2802. The component then outputs the resulting code.

IPR2024-001428
HP – EX1002, Page 404 of 487

[00138] Figure 29 is a logic diagram illustrating the processing of a primitive in one embodiment. Although not shown, the component may ensure that the disparity of the symbols of the primitive are in the correct order. In decision blocks 2901-2903, the component decodes the primitive. In blocks 2904-2906, the component raises a signal indicating the primitive that has been received. In one embodiment, the signals may be provided to the transport layer to control the packetizing of the received codes.

2. MEMORY LINK ARCHITECTURE

[00139] In one embodiment, various transport layer, link layer, and physical layer techniques can be used to provide a serial communications architecture that is well suited to support memory-based data store devices, especially multiport memory devices with random access memory. This architecture that is adapted to memory devices is referred to as "Memory Link." A multiport memory device provides multiple serial communications ports that can be connected to different hosts (e.g., any memory accessing device, such as a memory controller). Each host can independently access the memory of the memory device to read and write data. The memory may be organized into memory banks that can be simultaneously accessed by the same or different hosts. Many applications can advantageously use a multiport memory device. For example, an asynchronous transfer mode ("ATM") switch can use a multiport memory device to replace a crossbar switch. The ATM switch may have multiple input ports that store received data in the multiport memory device and multiple output ports that retrieve data from the multiport memory device. A multiport memory device may include a switch for dynamically interconnecting the memory to the ports so that each port can selectively access each memory bank. Alternatively, the ports may access memory using time-division multiplexing techniques.

[00140] Because of the way memory is typically accessed, the physical layer, link layer, and transport layer might be simplified as compared to the layers needed in a more general data store device, such as a disk drive. Each port of a multiport

IPR2024-001428
HP – EX1002, Page 405 of 487

memory device is typically directly connected to a host. In such a case, packets do not need to specify a source and a destination and the headers are thus simplified. In addition, packet types can be encoded in synchronization primitives to further simplify the packet header. The transport layer of a port in a multiport memory device can be simplified or eliminated because (1) routing to another device is not needed, (2) minimal or no parsing of the header may be needed, and (3) preemption and priority ordering of packets may not be needed. In addition, the link layer may be simplified by the use of physical layer DC-balancing, rather than the use of the selective block inversion to achieve DC-balance. Physical layer DC-balancing can be used with a multiport memory device because the distance between the host and the multiport memory device is typically relatively short.

[00141] The physical layer may also be simplified because a single multiphase clock generating device, such as an analog phase lock loop ("PLL"), or a delay lock loop, can be used to provide clocking for the transmitter and receiver for each port of the multiport memory device. The use of a single multiphase clock generating device can greatly reduce the area of a multiport memory chip used by the physical layer. A single-chip, multiport memory device has traditionally been infeasible because of the need to have multiple analog phase lock loops. The physical layer may also use plesiosynchronous null insertion techniques as described below to compensate for variations in the frequency between clocks of the hosts and the clock of the multiport memory device. The multiport memory device may transmit control information using out-of-band symbols as described above. Also, commands for accessing the memory device may be encoded in synchronization symbols.

[00142] Figure 30 is a block diagram illustrating the overall architecture of a multiport memory device in one embodiment. The multiport memory device 3000 includes multiple ports 3010-3019. Each port includes a physical layer, a link layer, a transport layer (optional), and an access layer. The access layers are connected to switch 3050. The multiport memory device includes memory with

IPR2024-001428
HP – EX1002, Page 406 of 487

memory banks 3320-3027. Each memory bank has associated bank cache 3030-3037. The bank caches are connected to the switch. A switch arbiter (not shown) is responsible for connecting the ports to the bank caches as needed to process the memory access requests. Each port can be independently and simultaneously connected to memory banks. Alternatively, access of the ports to the memory banks can be time-division multiplexed. With time-division multiplexing, each port has its allotted time period for communicating with the memory bank. A clock signal with a different phase for each port can be used to coordinate the access. The physical layer, link layer, and transport layer may be implemented to provide various combinations of the functions described above (e.g., transition control optimization and synchronization primitives encoded with packet types). The access layer receives commands and data from the transport layer, directs the arbiter to connect the port to the appropriate memory bank, and transmits commands and data to the memory bank. The access layer also receives data from the memory banks and provides the data to the transport layer for transmission to the host.

[00143] Figure 31 is a block diagram illustrating the use of a single phase lock loop for the physical layer of a multiport memory device in one embodiment. The physical layer 3100 of the multiport memory device includes multiple ports 3110-3119. Each port includes a serial communications line connected to a transmitter and another serial communications line connected to a receiver that comprise the serial communications link. Each port also includes a serializer and deserializer. The phase lock loop 3120 receives a clock signal from clock 3121. The phase lock loop outputs a clock signal 3122 and phase shifted clock signals 3123. The clock signal 3122 provides clocking information for the serializer. The phase shifted clock signals 3123 are provided to the deserializers to control the sampling of the input data.

[00144] Figure 32 is a block diagram illustrating contents of a bank cache in one embodiment. Each bank cache includes an input queue 3201 and an output queue 3202. Each entry of the input queue includes a port field, a read/write field,

IPR2024-001428
HP – EX1002, Page 407 of 487

an address field, and a data field. The port field identifies the port (e.g., 3) that provided the command. The read/write field identifies whether the access is a read or a write (i.e., R or W). The address field identifies the memory address to be accessed (e.g., 1000). The address field may be latched from the address bus. The data field includes the data to be written to memory. Each entry of the output queue includes a valid field, a port field, and a data field. The valid field indicates whether the entry contains valid data (e.g., 0 or 1). If a memory bank has multiple sections as described below, then the valid field may have a bit indicating whether the data of each section is valid. The port field identifies the port (e.g., 3) to which the data is to be transmitted. The data field contains the data retrieved from memory. Each memory bank successively retrieves entries from the input queue and then performs the requested access. When the access is to read data from memory, the memory bank also adds entries to the output queue. The memory bank adds an entry to the output queue on every possible read cycle. The valid field is used to indicate whether the entry contains valid data.

[00145] Figure 33 is a logic diagram illustrating processing of the access layer when it receives data from a host in one embodiment. In block 3301, a component of the access layer receives the next command from the transport layer. In decision block 3302-3304, the access layer parses the command. In block 3305-3307, the component performs the processing associated with that command.

[00146] Figure 34 is a logic diagram illustrating the processing of a write command by the access layer in one embodiment. In block 3401, the component retrieves the target address for the write command. In block 3402, the component requests the arbiter to configure the switch to connect the port with the appropriate memory bank. In block 3403, the component retrieves the next byte (i.e., code) from the transport layer. In decision block 3404, if all the bytes have already been retrieved, then the write is complete, else the component continues at block 3405. In block 3405, the component stores an entry into the input queue of the memory bank. The component may need to buffer multiple bytes depending on the width

IPR2024-001428
HP – EX1002, Page 408 of 487

of the word. In block 3406, the component increments the address and then reads the next byte. One skilled in the art will appreciate that when time-division multiplexing is used, the component supplies the data to the input queue during its allotted time period.

[00147] Figure 35 is a logic diagram illustrating processing of the access layer when it is to provide data to the transport layer in one embodiment. In block 3501, the component receives the next byte from the switch. In decision block 3502, if all the bytes have already been received, then the component continues at block 3504, else the component continues at block 3503. In block 3503, the component provides the byte to the transport layer and then proceeds to receive the next byte. In block 3504, the access layer signals the end of the read to the transport layer.

4005345.4200704

2.1 Multisection Memory Banks

[00148] Figure 36 is a block diagram of a multisection memory bank in one embodiment. A multisection memory bank can significantly reduce power consumption of a memory device. A multisection memory bank subdivides each row (or word) of data in a memory bank into multiple sections. The enabling of sections can be configured on a port-by-port basis. When a section is disabled, then the amount of power consumed by the memory bank during an access is much less than if the section was enabled. In this way, each port can specify only the needed sections which can result in a significant reduction in the power consumption because only the needed sections are enabled during access. The data for all the rows for each subdivision of a memory bank is stored in a section (or a page). For example, a memory bank may have a word width of 64 bits. Such a multisection memory bank may be divided into two sections of 32 bits each that can be accessed independently and simultaneously based on configurable parameters. If a host only needs a 32-bit word, then only one section needs to be enabled whenever that host accesses the memory bank.

[00149] As shown in Figure 36, the multisection memory bank includes section 3600 and section 3610 each of which is 32 bits wide. Each wordline of each

IPR2024-001428
HP – EX1002, Page 409 of 487

section includes an AND gate 3601, 3611 that is connected to a corresponding line of the row decoder 3630 and to a section enable line from configuration register 3650. The row decoder decodes the row portion of the address and enables the appropriate wordline when the corresponding section is enabled. For example, if the row portion of the address is eight bits, then the row decoder enables one of 256 wordlines. The AND gate enables the wordline for its section only when that section is enabled for the port. If a port only accesses section 3600, then whenever an access for the port is processed, section 3600 is enabled and section 3610 is disabled. Each wordline of each section may also include a sequence of latches 3602, 3612 for buffering signals. The number of latches needed depends on row access latency, column access latency, and desired level of simultaneous access. Each section also includes a column decoder and column cache 3603, 3613. The column decoder latches data into the column cache based on the column address provided by the sequence of latches 3104, 3114 for buffering column address. Because the wordlines are only enabled on a section-by-section basis and because retrieved data is only latched on a section-by-section basis, the power requirements of a multisection memory bank is lower than a memory bank that does not have multiple sections. When data is retrieved from a multisection memory bank, the valid field of the output queue indicates the divisions of the word that are valid. For example, if data is retrieved from only section 0, then the valid field may contain "10." If data is retrieved from both sections 0 and 1, then the valid field may contain "11." One skilled in the art will appreciate that a memory bank may be subdivided into a number of sections. Also, a multisection memory bank may simultaneously perform multiple access commands so long as the commands do not access the same wordline of the same section. Thus, the processing of a command to access wordline 10 of section 0 may overlap with the processing of a command to access wordline 11 of section 0. Also, the processing of a command to access wordline 10 of section 0 may be overlapped with the processing of a command to access wordline 10 of

IPR2024-001428
HP – EX1002, Page 410 of 487

section 1. Thus, increasing the number of sections of a memory bank can increase the amount of parallel processing that can be performed.

2.2 Physical layer DC-balancing

[00150] In one embodiment, DC-balancing of the communications link is handled at the physical layer in memory link. The performing of DC-balancing at the physical layer can help reduce the cost and complexity of the link layer. To help ensure DC-balance, the physical layer uses a variable strength line driver that can drive a standard strength or increased strength 0-bit value or 1-bit value onto the communications link. When the running disparity is zero or positive (i.e., in the 1-direction) and a 1-bit value is to be transmitted, then a standard strength 1-bit value is driven. Similarly, when the running disparity is a zero or negative (i.e., in the 0-direction) and a 0-bit value is to be transmitted, then a standard strength 0-bit value is driven. In contrast, when the running disparity is negative and a 1-bit value is to be transmitted, then an increased strength 1-bit value (i.e., a 1-bit voltage value) is driven. This increased strength 1-bit value tends to offset the effects of a negative running disparity. Similarly, when the running disparity is positive and a 0-bit value is to be transmitted, then an increased strength 0-bit value (i.e., a 0-bit voltage value) is driven. This increased strength 0-bit value tends to offset the effects of a positive running disparity. The use of a variable-strength line driver is especially useful in controlling DC-balance over short transmission distances. In such a case, physical layer DC-balancing can be used in lieu of link layer DC-balancing.

[00151] Figure 37A is a block diagram illustrating a variable strength line driver in one embodiment. A variable strength line driver 3700 includes a fixed-strength component (or portion) 3701 and a variable-strength component 3702. The variable-strength line driver has 4 inputs: data in 3703, negative running disparity signal (RD-) 3705, positive running disparity signal (RD+) 3706, and enable signal 3707. The variable-strength line driver has one output: data out 3704. The fixed-strength component outputs a standard strength 0-bit value or 1-bit value depending on data in. The variable-strength component pulls up, pulls down, or

IPR2024-001428
HP – EX1002, Page 411 of 487

has no effect on data out, depending on the running disparity and data in. If the running disparity is positive and data in is a 0-bit value, then the variable-strength component pulls down data out. If the running disparity is negative and data in is a 1-bit value, then the variable-strength component pulls up data out. More generally, the running disparity signals may indicate when the running disparity is sufficiently negative or sufficiently positive to activate the variable-strength component.

[00152] Figure 37B is a block diagram illustrating the circuit of a variable-strength line driver in one embodiment. The fixed-strength component 3701 and the variable-strength component 3702 employ the same logic to activate their pull-up and pull-down transistors. One skilled in the art will appreciate that the transistors may be selected to achieve the overall desired physical layer characteristics of the communications link. NAND gate 3710 is used to control pull-up transistor 3711, and NOR gate 3712 is used to control pull-down transistor 3713. When the data in is a 1-bit value and the driver is enabled, NAND gate 3710 outputs a 0, which turns on pull-up transistor 3711. When the data in is a 0-bit value and the driver is enabled, NOR gate 3712 outputs a 1, which turns on pull-down transistor 3713. The variable-strength component has similar control logic, except that pull-up transistor 3721 is activated only when the running disparity is negative and except that the pull-down transistor 3723 is activated only when the running disparity is positive.

3. PLESIOSYNCHRONOUS CLOCKING WITH NULL INSERTION/REMOVAL

[00153] In one embodiment, the serial communications architecture specifies that NULL symbols or NULL primitives are inserted into or removed from a stream of symbols to account for variations in clock frequency between the transmitting and receiving communications nodes when using plesiosynchronous clocking. If a transmitting communications node transmits at a slightly higher frequency than the receiving communications node receives, then the receiving communications node

PALM-3529.004

IPR2024-001428
HP – EX1002, Page 412 of 487

traditionally needs to buffer the received symbols to handle such an overrun. For example, if both communications nodes use clocks with a frequency of 100 Hz with a maximum variation of 1%, then it is possible that the transmitting communications node transmits at 101 Hz and the receiving communications node receives at 99 Hz. In such a case, the receiving communications node will receive an extra 2 bits approximately every second, assuming 1 bit is transmitted every clock cycle. Receiving communications nodes typically have an elastic buffer for storing and aligning these extra bits on symbol boundaries. Similarly, an underrun occurs when a transmitting communications node transmits data at a slightly lower rate than the receiving communications node receivers. The handling of an underrun can also be costly and complex. These underrun and overrun conditions were typically handled at the link layer. These complexities and costs may, however, be avoided by the physical layer of a receiving communications node periodically inserting or removing NULL symbols or NULL primitives from the received stream of symbols at a rate sufficient to compensate for variations in clock frequencies.

[00154] Figure 38A is a block diagram of a plesiosynchronous communications system in one embodiment. Communications node 3801 and communications node 3811 are connected via a bi-directional serial link. The transport layer and the link layer 3802, 3812 may be implemented in a single digital ASIC chip with other interface logic such as PCI bus interface logic. The physical layer 3803 and 3813 may be built in a separated custom analog chip.

[00155] The link layer provides symbols (i.e., N-bit encoded data) to be transmitted to the physical layer via TX_DAT lines. The link layer is provided symbols received by the physical layer via the RX_DAT lines. In one embodiment, the TX_DAT and RX_DAT lines are both synchronized to the local clock, and the setup/hold time relation between TX_DAT and RX_DAT are the same to setup/hold time relationship to the local clock. Both the transmitter 3804, 3814 and receiver 3805, 3815 of the physical layer share the same phase lock loop ("PLL") 3806, 3816. This overall architecture is described in U.S. Patent

IPR2024-001428
HP – EX1002, Page 413 of 487

Application No. 6,229,859, entitled "System and Method for High-Speed, Synchronized Data Communication."

[00156] Figure 38B is a block diagram illustrating a switch supporting multiple data store devices in one embodiment. The switch 3820 has 5 ports 3821-3825. Ports 3821-3824 are shown connected to disk drives 3831-3834. The switch has one phase lock loop 3826 that is shared by all the ports. Since all the ports of the switch are synchronized to the same clock, the data can be freely (without buffering) transferred between ports. (See discussion below on asynchronous clock boundaries.)

[00157] Figure 39A is a block diagram illustrating a prior art synchronous clocking system. In this synchronous clocking system, the RX_DAT lines provided to the link layer 3920 are synchronized not to the local clock, but to the RX_CLK line provided by the physical layer. The RX_CLK line provides a clock that has been recovered from the incoming bit stream, which is synchronized to the transmitter's clock. The receiver may recover the clock from a clock signal on a separate line or from the stream of data bits transmitted by the transmitter. In the link layer, there are two clock domains, the TX clock domain 3921 that is the same as the local clock domain, and the RX clock domain 3922 that is the same as the transmitter's clock domain. Consequently, an asynchronous clock boundary 3923 is formed between the TX clock domain and the RX clock domain. When information is transferred from the TX clock domain to the RX clock domain, the FIFO 3924 is used for the data and flip-flop synchronizer 3925 is used for the control information. When information is transferred from the RX clock domain to the TX clock domain, the FIFO 3926 is used for the data and the flip-flop synchronizer 3927 is used for the control information.

[00158] Figure 39B is a block diagram illustrating resolution of an asynchronous clock boundary in the physical layer in one embodiment. The clock and data recover ("CDR") processing of the physical layer 3950 provides to the link layer RX_DAT lines that are synchronized to the local clock. Thus, the asynchronous boundary at the link layer along with the needed logic to resolve the boundary is

IPR2024-001428
HP – EX1002, Page 414 of 487

avoided in the link layer. The CDR processing also provides a 1-bit control signal (DET_NULL) that indicates a NULL symbol is provided in RX_DAT (i.e., to inform the link layer that a primitive has been received) and a 1-bit control signal (DET_ERR) that indicates an error in the physical layer data recovery.

[00159] Figure 40 is a block diagram illustrating NULL insertion and deletion at the physical layer in one embodiment. Sampler 4010 in one embodiment corresponds to the over sampler 4011, digital phase lock loop 4012, and phase selector 4013 described in U.S. Patent No. 6,229,859, entitled "System and Method for High-Speed, Synchronized Data Communication." The sampler outputs N+1 bits (where N is the number of bits in a symbol) of data lines D[0:N] and two valid bits V[N-1] and V[N]. The sampler identifies a stream of data bits that are received and provides those data bits in data lines D[0:N]. The valid bits indicate whether D[N-1] and D[N] contain valid data (e.g., V[N-1] = 1 indicates that D[N-1] is valid). The data bits provided by the sampler are not, however, frame (i.e., symbol) aligned; that is, D[0] is not necessarily (and more often than not would not be) the first bit in a symbol. The valid bits indicate whether an overrun or underrun condition has occurred. An overrun occurs as a result of the transmitter's clock frequency being faster than the receiver's, and an underrun occurs as a result of the transmitter's clock frequency being slower than the receiver's. Nominally, N data bits are provided by the sampler every sample interval. In such a case D[N-1] is valid, but D[N] is invalid. If, however, there is an overrun, then the sampler provides N+1 data bits; that is, D[N-1] and D[N] are both valid. If there is an underrun, then the sampler provides N-1 data bits; that is, D[N-1] and D[N] are both invalid. The data bits and the valid bits are supplied to a frame aligner 4020. The frame aligner not only aligns the frames, but also inserts and removes NULL symbols to adjust for variations in the transmitting and receiving clock frequencies.

[00160] Figure 41 is a block diagram illustrating the frame aligner in one embodiment. The frame aligner 4100 includes bit buffer 4101, sync and null detector 4103, symbol pointer tracker 4104, and multiplexer 4102. The bit buffer

IPR2024-001428
HP – EX1002, Page 415 of 487

stores the last 27 bits (e.g., 3 X N) received as provided by D[0:9], assuming that a symbol includes 9 bits. The bit buffer is organized into three buffers or regions (i.e., A, B, and C) of 9 bits each. The sync and null detector scans the bit buffer to detect whether it contains a NULL primitive and, in particular, a synchronization primitive. If a synchronization primitive is detected, it signals the symbol pointer tracker to align itself on the synchronization primitive. The sync and null detector also signals the symbol pointer tracker whether a NULL symbol should be inserted or removed from the symbol stream. The symbol tracker pointer tracks the symbol alignment over time.

[00161] The sync and null detector provides a 1-bit load data signal ("LD"), a 1-bit last half signal ("iHF"), and a 9-bit synchronization symbol pointer ("iPTR") to the symbol pointer tracker. The LD signal indicates that a synchronization symbol has been detected. The iHF signal indicates whether the synchronization symbol starts in the first or last half of one of the 9-bit buffers of the bit buffer. The iPTR signal indicates a bit position of the start of the synchronization primitive in the bit buffer.

[00162] The symbol pointer tracker provides 1-bit null selection signal ("uHF") and 9-bit start-of-symbol pointer ("uPTR") to the multiplexer. The uHF signal indicates whether to insert a NULL symbol into the stream of symbols, and the uPTR signal indicates the start bit position of a symbol. The symbol pointer tracker also provides 1-bit error signal ("DET.ERR") and 1-bit null signal ("DET.NULL") to the link layer. The multiplexer provides the 9-bit RX_DAT to the link layer.

[00163] Figures 42A, 42B, and 42C illustrate the loading of the bit buffer in one embodiment. The bit buffer contains three 9-bit buffers A[8:0], B[8:0], and C[8:0]. When the valid bits V[8:9] indicate that only D[8] is valid (i.e., V[8:9] = "10" a normal condition), then all the data in the bit buffer is rotated 9 bits to the left. In other words, the data from B[8:0] is transferred to A[8:0], the data from C[8:0] is transferred to B[8:0], and the data from D[0:8] is transferred to C[8:0] as indicated by Figure 42A. When the valid bits V[8:9] indicate that neither D[8] or D[9] is valid (i.e., V[8:9]="00" an underrun condition), then the data in the bit buffer is rotated

IPR2024-001428
HP – EX1002, Page 416 of 487

8 bits to the left. In other words, A[0] is transferred to A[8], B[8:1] is transferred to A[7:0], B[0] is transferred to B[8], C[8:1] is transferred to B[7:0], C[0] is transferred to C[8], and D[0:7] is transferred to C[7:0] as illustrated by Figure 42B. When the valid data V[8:9] indicates that both D[8] and D[9] are valid (i.e., V[8:9]="11" an overrun condition), then the data in the bit buffer is rotated 10 bits to the left. In other words, B[7:0] is transferred to A[8:1], C[8] is transferred to A[0], C[7:0] is transferred B[8:1], D[0] is transferred to B[0], and D[1:9] is transferred to C[8:0] as illustrated by Figure 42C.

[00164] Figure 43 is a diagram illustrating the detection of a synchronization primitive by the sync and null detector in one embodiment. Assuming that the synchronization detection mode is falling edge, the detector determines whether there are only three falling edges in the bit buffer and whether they are 9 bits apart. If so, the detector sets the LD signal to 1, else the detector sets the LD signal to 0. If the falling edges are in the first half of the bit buffer (e.g., A[8:5]), then the detector sets the iHF signal to 0, else the detector sets the iHF signal to 1. The detector also sets the iPTR signals to indicate the position of the falling edge that starts the synchronization symbols, which indicate the start bit position of the synchronization symbol. In this example, the bit buffer 4301 contains the synchronization primitive "100000001-100111111" starting at A[6] and a portion of that synchronization primitive starts at C[6]. If A[8:7] contained a falling edge from a non-synchronization symbol, then when the next symbol of the second synchronization primitive is received from the sampler, the bit buffer would be guaranteed to contain only three falling edges. The bit buffer 4302 contains the same synchronization primitives.

[00165] Figure 44 is a diagram illustrating the portion of the bit buffer that contains the next symbol to be extracted in one embodiment. If the start of a symbol is in the first half of a region of the bit buffer (i.e., A[8:5]), then the symbol is extracted starting in the B region of the bit buffer and proceeding to the C region. If the start of a symbol is in the last half of a region of the bit buffer (i.e., A[4:0]), then the

IPR2024-001428
HP – EX1002, Page 417 of 487

symbol is extracted starting in the A region of the bit buffer and proceeding to the B region.

[00166] Figure 45 is a state diagram of the symbol pointer tracker in one embodiment. The symbol pointer tracker tracks the start position of a symbol (e.g., frame) based on the start position of a synchronization symbol and the valid bits (e.g., V[8:9]) provided by the sampler. The symbol pointer tracker enters an initial state ("INIT") whenever the LD signal of the sync and null detector indicates that a synchronization symbol has been detected. The tracker stays in that initial state so long as the frame aligner is provided with 9 valid bits (i.e., N valid bits) at a time (i.e., V[8:9]="10" a normal condition). If the frame aligner is provided with 10 valid bits (i.e., V[8:9]="11" an overrun condition), then the tracker enters a first overrun state ("OVR1"), which means that the transmitter's clock frequency is faster than the receiver's clock frequency. The tracker stays in the first overrun state so long as the frame aligner is provided with 9 valid bits at a time. If the frame aligner is provided with 10 valid bits, then the tracker enters the second overrun state ("OVR2") from the first overrun state. If the frame aligner is provided with 8 valid bits, then an error has occurred (e.g., 10 bits valid followed by 8 bits valid) and enters an error state from the first overrun state. The tracker exits the second overrun state when the next bits are provided to the frame aligner. If the tracker is provided with 9 valid bits, then the tracker returns to the first overrun state, else the tracker enters an error state. Once an overrun is detected, the tracker thus expects to receive only 9 or 10 valid bits at a time and does not expect to receive two sets of 10 valid bits in a row. The receiving of two sets of 10 valid bits in a row would indicate that the transmitter's clock frequency is too high to be compensated for by the receiver. The receiving of 8 bits in an overrun state indicates that the transmitter's clock frequency now appears to be slower, which means some sort of error has occurred. The first underrun state ("UND1") and the second underrun state ("UND2") are entered in a manner that is analogous to entry into the overrun states.

EXHIBIT - 1007

[00167] When the tracker detects an LD signal of 1, it sets the start-of-symbol pointer to the start of the synchronization symbol as indicated by the iPTR lines and enters the initial state. It then sets the extraction region for the start-of-symbol pointer to A if the iHF signal is 1 and to B if the iHF signal is 0. The extraction region indicates the region that contains the first bit of the next symbol to be extracted. When the tracker detects an LD signal of 0, it moves the start-of-symbol pointer either backward or forward, depending on whether the tracker transitions to an overrun state as a result of receiving 10 bits or transitions to an underrun state as a result of receiving 8 bits.

[00168] Figure 46 is a diagram illustrating movement of the start-of-symbol pointer due to overrun or underrun in one embodiment. When a synchronization symbol is detected (i.e., LD=1), then the start-of-symbol pointer is somewhere within the range of A[3]:B[4]. If the transmitter's local clock frequency is slightly slower than the receiver's local clock frequency (i.e., a potential underrun condition), the tracker moves the pointer forward to the C region. When the pointer is moved into the underrun extend region (e.g., B[3]:C[8]) and a NULL symbol is detected, then the tracker moves the start-of-symbol pointer backward 9 bit positions (i.e., 1 symbol position) and inserts a NULL symbol at the start-of-symbol pointer. This insertion of the NULL symbol compensates for the underrun by adding an extra symbol to the received stream. In one embodiment, a NULL symbol can be any symbol of any primitive, that is, control or synchronization. Similarly, if the transmitter's local clock frequency is slightly faster than the receiver's local clock frequency (i.e., a potential overrun condition), the tracker moves the start-of-symbol pointer backward toward the A region. When the pointer is moved into the overrun extend region (i.e., A[8:4]) and a NULL symbol is detected, then the tracker moves the pointer forward 9 bits, which effectively removes the NULL symbol. The removal of the NULL symbol compensates for the overrun. One skilled in the art will appreciate that the size of the overrun and the underrun extend regions can be adjusted based on the maximum possible differences in

IPR2024-001428
HP – EX1002, Page 419 of 487

clock frequencies between the transmitter and receiver and the guarantee frequency at which a NULL primitive is to be transmitted.

[00169] Figures 47A and 47B are diagrams illustrating NULL detection while the start-of-symbol pointer is not within the overrun or underrun extend regions in one embodiment. In this example, the pointer is at B[4]. Only 8 valid bits are provided to the frame aligner (i.e., an underrun condition); as a result, the pointer is moved forward to B[4]. A NULL symbol is also detected starting at B[4]. However, since the pointer is not within the overrun or underrun extend regions, no NULL symbol is removed or inserted and the two symbols of the NULL primitive are extracted.

[00170] Figures 48A and 48B are diagrams illustrating NULL detection while the start-of-symbol pointer is in the underrun extend region. The pointer enters the underrun extend region as a result of the transmitter's clock frequency being slightly slower than the receiver's clock frequency. When the transmitter's clock frequency is slower, the receiver can receive more data than the transmitter transmits and the receiver will eventually run out of data as the start-of-symbol pointer is moved forward. When the pointer is in the underrun region, the tracker inserts a NULL symbol when another NULL symbol is detected to account for the underrun. In this example, the pointer is at B[2] when a NULL symbol is detected. The tracker moves the pointer backward to A[2] (i.e., 1 symbol position) and inserts a NULL symbol at A[2]:B[3]. The inserted NULL symbol and the two NULL symbols of the NULL primitive are then extracted. In one embodiment, the inserted NULL symbol is a duplicate of the detected NULL symbol.

[00171] Figures 49A and 49B are diagrams illustrating NULL detection, while the start-of-symbol pointer is in the overrun extend region. The pointer enters the overrun extend region as a result of the transmitter's clock frequency being slightly faster than the receiver's clock frequency. When the transmitter's local clock frequency is faster, the receiver cannot receive as much data as the transmitter transmits, and so the receiver would eventually lose some data. When the pointer is in the overrun region, the tracker removes a NULL symbol of a NULL primitive when a NULL primitive is detected to account for the overrun. In this example, the

EXPRESS MAIL NO. EL 703501754 US

IPR2024-001428
HP – EX1002, Page 420 of 487

pointer is at A[4] when a NULL symbol is detected. The tracker moves the pointer forward 1 symbol position, which effectively removes or skips over the NULL symbol at A[4]:B[5]. As a result, only the second NULL symbol of the NULL primitive is extracted. All primitives have the characteristic that each symbol of the primitive encodes the same information. Thus, one of the symbols of a primitive can be removed without loss of information.

[00172] From the foregoing, it will be appreciated that although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. Accordingly, the invention is not limited, except as by the appended claims.

Express Mail No. EV 035 625 745 US

IPR2024-001428
HP – EX1002, Page 421 of 487

CLAIMS

I/We claim:

[c1] 1. A method for transmitting packets, the method comprising:

receiving a first packet;

transmitting a portion of the received first packet;

receiving a second packet;

upon receiving the second packet,

stopping the transmitting of the first packet so that not all of the first packet has been transmitted;

transmitting a preempt indicator;

transmitting the second packet; and

upon completion of transmitting the second packet,

transmitting a continue indicator; and

transmitting the portion of the first packet that has not yet been transmitted.

[c2] 2. The method of claim 1 wherein the transmitted indicators are primitives.

[c3] 3. The method of claim 1 wherein the packets include in-band symbols and the indicators include one or more out-of-band symbols.

[c4] 4. The method of claim 3 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

[c5] 5. The method of claim 1 wherein the first packet is a data packet and the second packet is a control packet.

Express Mail No. EL 914 035 345 US

IPR2024-001428
HP – EX1002, Page 422 of 487

[c6] 6. The method of claim 1 wherein the first and second packets are transmitted through the same communications link.

[c7] 7. The method of claim 1 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

[c8] 8. The method of claim 1 wherein the first packet includes a header that is transmitted only once.

[c9] 9. A method for receiving packets, the method comprising:

receiving a first portion of symbols of a first packet;

receiving a preempt indicator indicating that a second packet of symbols is to be received;

receiving the second packet of symbols;

receiving a continue indicator indicating that a second portion of symbols of the first packet is to be received; and

receiving the second portion of symbols of the first packet.

[c10] 10. The method of claim 9 wherein the received indicators are primitives.

[c11] 11. The method of claim 9 wherein the symbols of the packets include in-band symbols and the indicators include one or more out-of-band symbols.

[c12] 12. The method of claim 11 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

[c13] 13. The method of claim 9 wherein the first packet is a data packet and the second packet is a control packet.

IPR2024-001428
HP – EX1002, Page 423 of 487

[c14] 14. The method of claim 9 wherein the first and second packets are received via the same communications link.

[c15] 15. The method of claim 9 wherein the receiving of the second packet is preempted so that a third packet can be received.

[c16] 16. The method of claim 9 wherein the first packet includes a header that is received only once.

[c17] 17. A method for transmitting packets via a communications link, the method comprising:

transmitting a first portion of a first packet;

transmitting a preempt indicator indicating that a second packet is to be transmitted;

transmitting the second packet;

transmitting a continue indicator indicating that a second portion of the first packet is to be transmitted; and

transmitting the second portion of the first packet

wherein the first and second packets and the preempt and continue indicators are transmitted via the same communications link.

[c18] 18. The method of claim 17 wherein the indicators are primitives.

[c19] 19. The method of claim 17 wherein the packets include in-band symbols and the indicators include one or more out-of-band symbols.

[c20] 20. The method of claim 19 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

Exhibit 1002 - Page 424 of 487

IPR2024-001428
HP – EX1002, Page 424 of 487

[c21] 21. The method of claim 17 wherein the first packet is a data packet and the second packet is a control packet.

[c22] 22. The method of claim 17 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

[c23] 23. The method of claim 17 wherein the first packet includes a header that is transmitted only once.

[c24] 24. A communications device for transmitting packets via a communications link, comprising

a transmission component that transmits a first packet; and

a preemption component that signals the transmission component to stop transmitting the first packet, transmits a preempt indicator indicating that a second packet is to be transmitted, transmits the second packet, and signals the transmission component to continue transmitting the first packet.

[c25] 25. The communications device of claim 24 wherein the indicators are primitives.

[c26] 26. The communications device of claim 24 wherein packets include in-band symbols and the indicators include one or more out-of-band symbols.

[c27] 27. The communications device of claim 26 wherein the in-band symbols are transition optimized and the out-of-band symbols are not transition optimized.

[c28] 28. The communications device of claim 24 wherein the first packet is a data packet and the second packet is a control packet.

Exhibit 1002 - Page 425 of 487

IPR2024-001428
HP – EX1002, Page 425 of 487

[c29] 29. The communications device of claim 24 wherein the transmitting of the second packet is preempted so that a third packet can be transmitted.

[c30] 30. The communications device of claim 24 wherein the first packet includes a header that is transmitted only once.

Express Mail No. EV 045 050 545 US

IPR2024-001428
HP – EX1002, Page 426 of 487

METHOD AND SYSTEM FOR NESTING OF COMMUNICATIONS PACKETS

ABSTRACT OF THE DISCLOSURE

A serial communications architecture for communicating between hosts and data store devices. The Storage Link architecture is specially adapted to support communications between multiple hosts and storage devices via a switching network, such as a storage area network. The Storage Link architecture specifies various communications techniques that can be combined to reduce the overall cost and increase the overall performance of communications. The Storage Link architecture may provide packet ordering based on packet type, dynamic segmentation of packets, asymmetric packet ordering, packet nesting, variable-sized packet headers, and use of out-of-band symbols to transmit control information as described below in more detail. The Storage Link architecture may also specify encoding techniques to optimize transitions and to ensure DC-balance.

EXPRESS MAIL NO. EL 909 135 245 US

IPR2024-001428
HP – EX1002, Page 427 of 487

10035911 . 110701

IPR2024-001428
HP – EX1002, Page 428 of 487

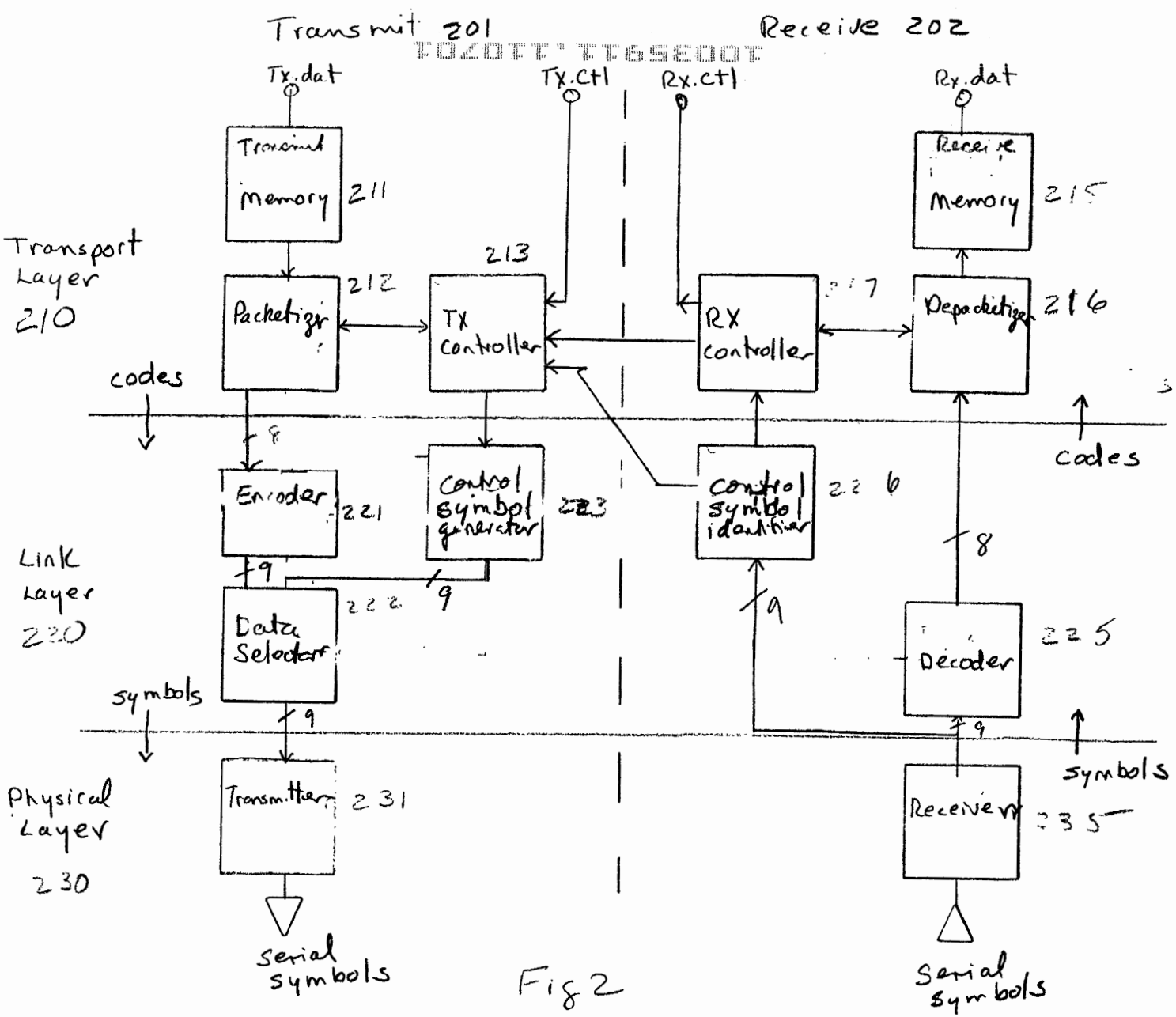

Transmit 201
Receive 202
Tx.dat
Tx.Ctl
Rx.Ctl
Rx.dat
Transmit memory 211
Receive Memory 215
Transport Layer 210
Packetizer 212
213
TX controller
RX controller
217
Depacketizer 216
codes
8
Encoder 221
control symbol generator 223
control symbol identifier 226
Link Layer 220
9
Data Selector 222
Decoder 225
symbols
Physical Layer 230
Transmitter 231
Receiver 235
serial symbols
Serial symbols

Fig 2

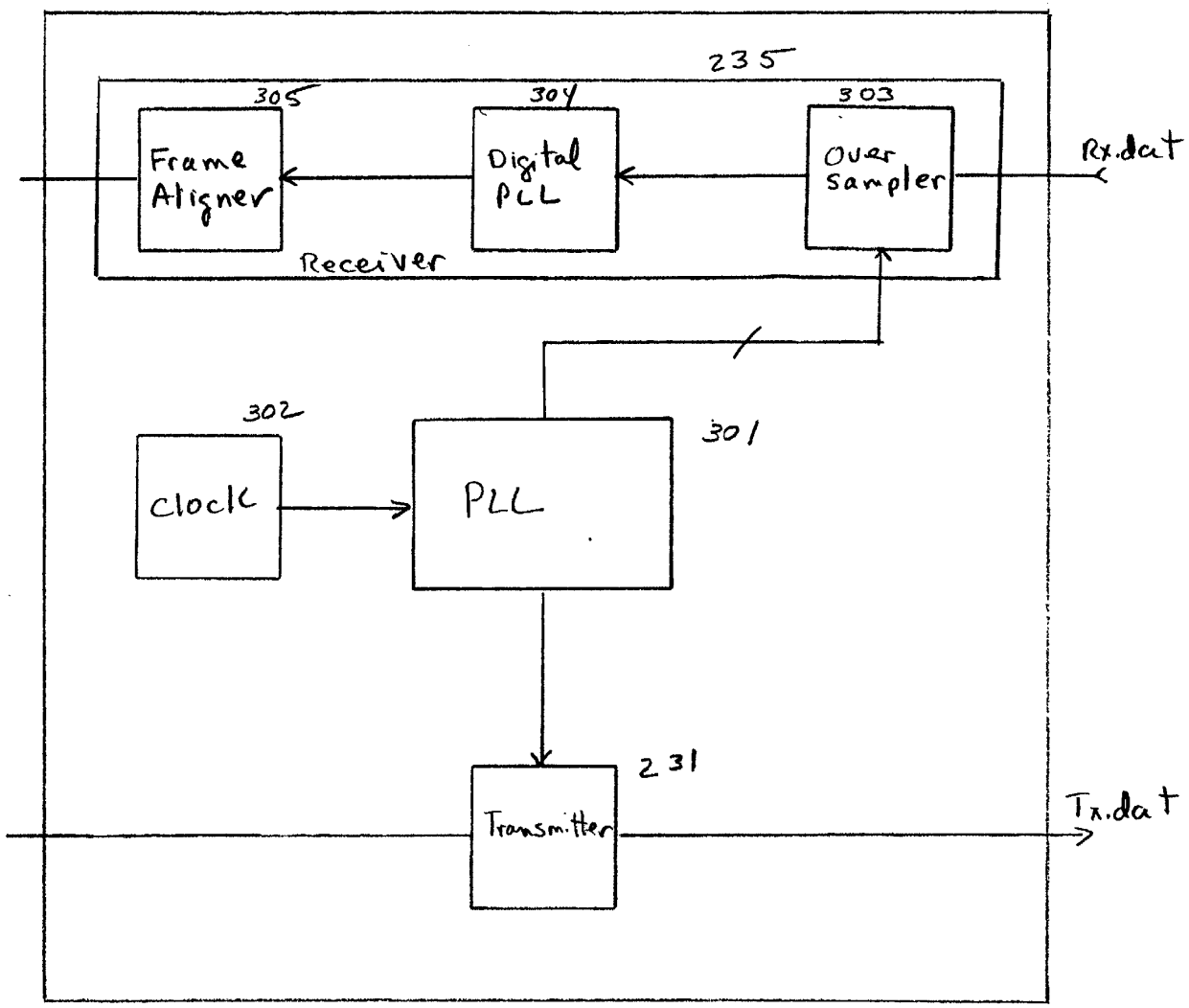


Fig 3

IPR2024-001428
HP – EX1002, Page 430 of 487

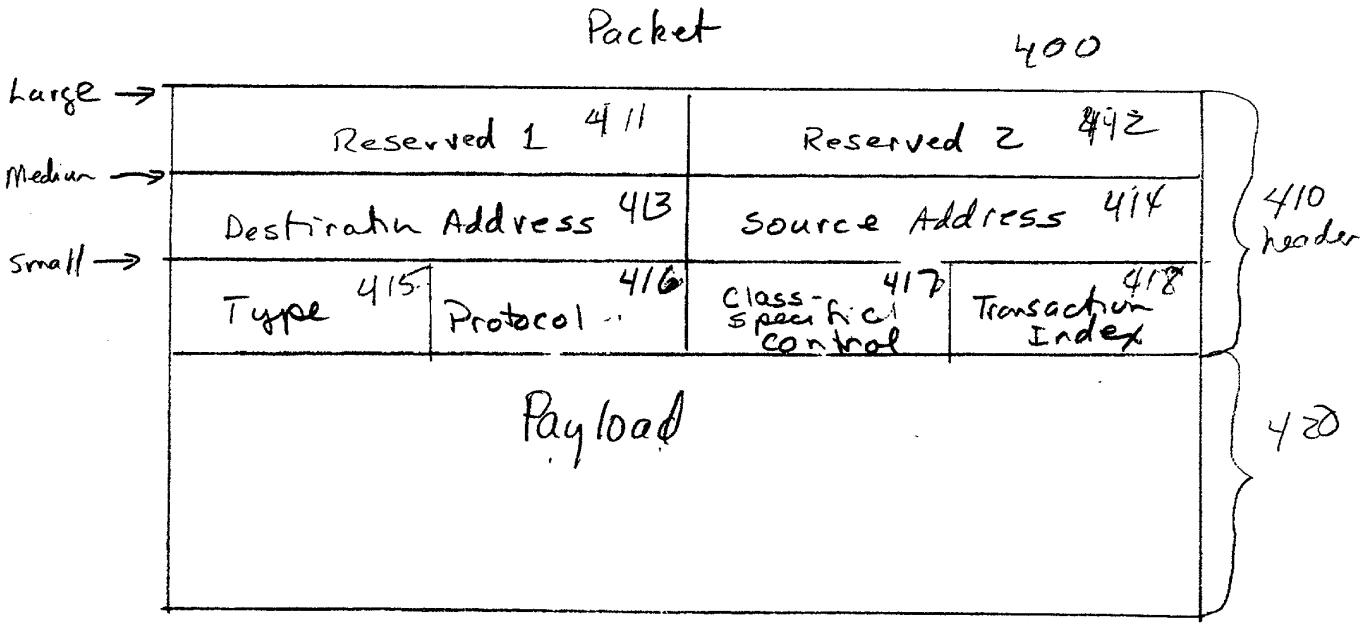
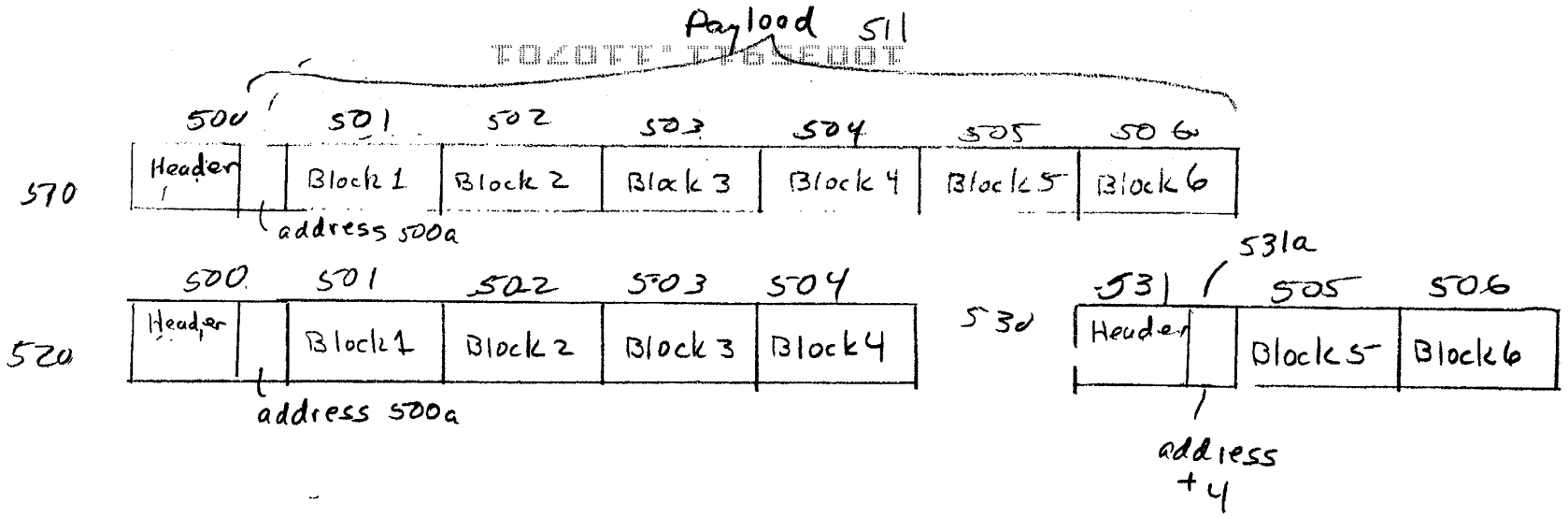


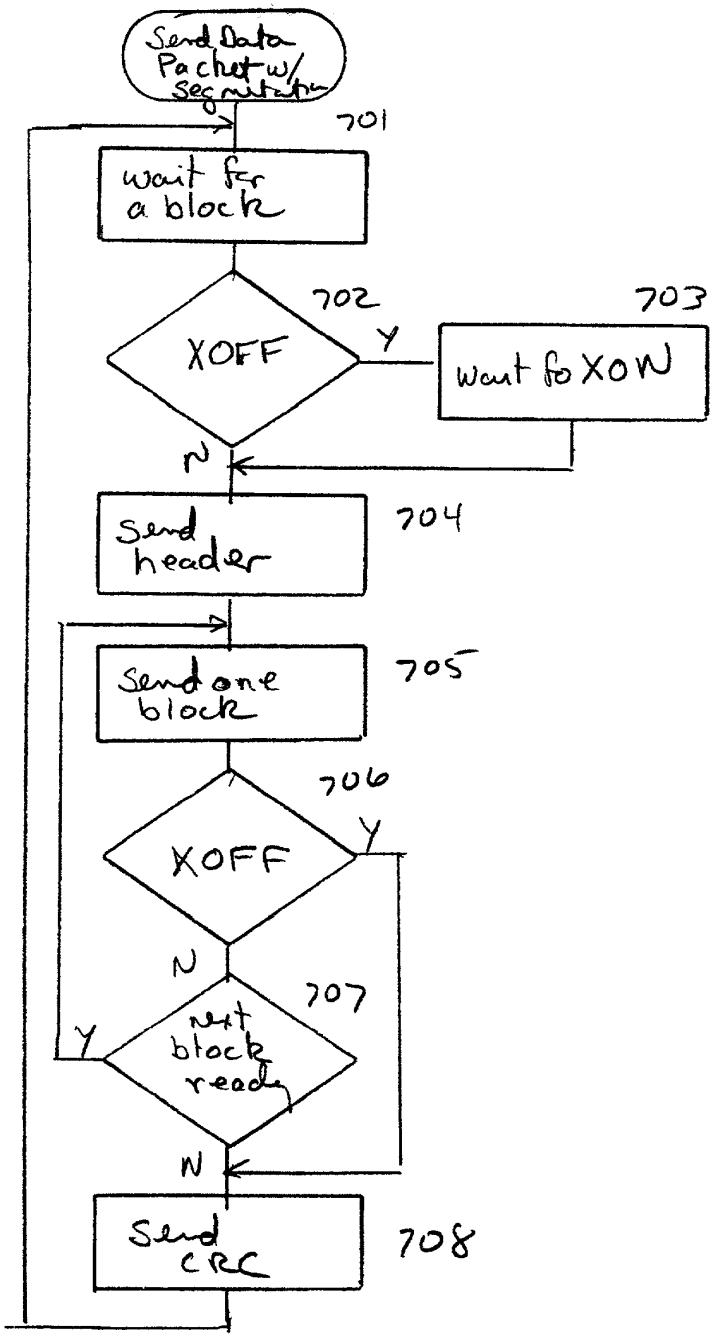
Fig 4

Docket No. 10015501-1


Payload 511
500
501
502
503
504
505
506
570
Header
Block 1
Block 2
Block 3
Block 4
Block 5
Block 6
address 500a
500
501
502
503
504
520
Header
Block 1
Block 2
Block 3
Block 4
address 500a
531a
531
505
506
530
Header
Block 5
Block 6
address
+4

F.g 5

Fig 6
600
T4
0 1 2 3 4 5 6 7 8
Packet Memory Before Merge
Packet Memory After Merge
T3
Add=25
Blocks=5
T1
Add=4
Blocks=3
T2
Add=50
Blocks=10
T1
Add=0
Blocks=2
T3
Add=25
Blocks=5
T2
Add=50
Blocks=10
T1
Add=0
Blocks=7
Received Packet
T1
Add=2
Blocks=2
610
611
612
613
614
620
630



F. 8 7

FOXCONN EX1002.008

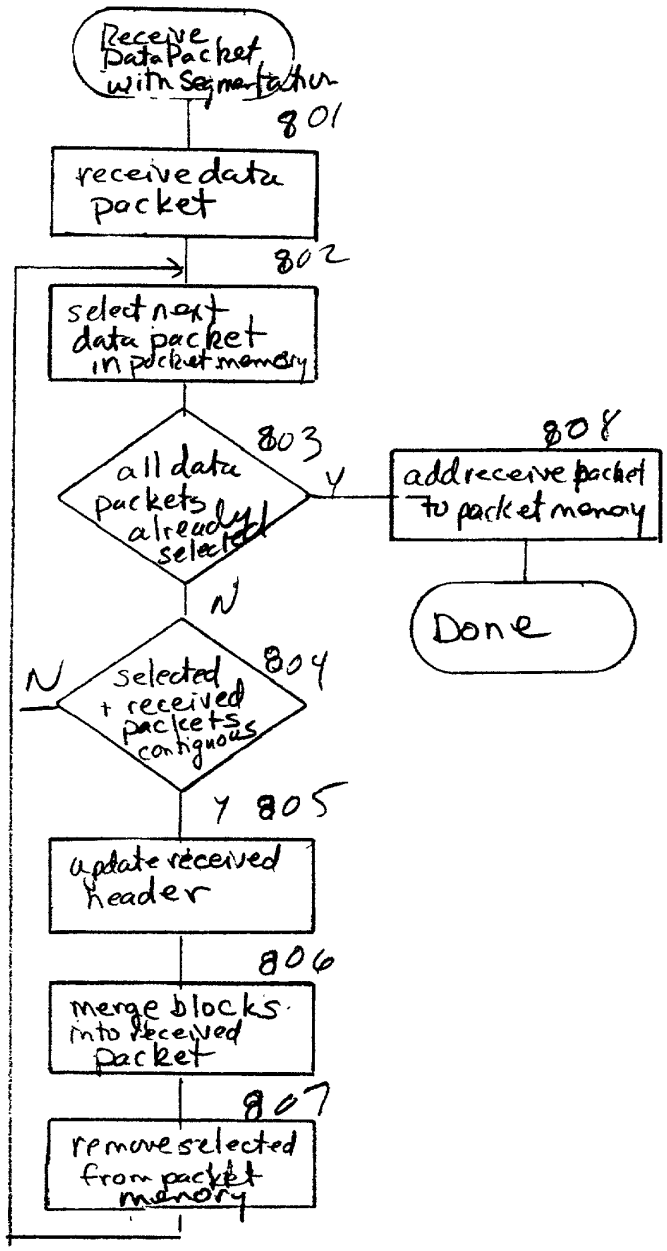
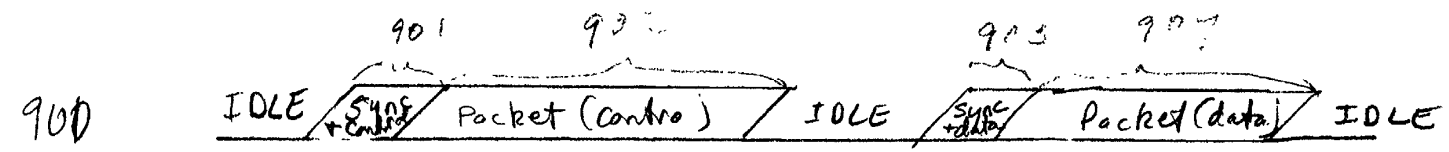


Fig. 8

EXHIBIT 1002, Page 435 of 487



Sync + packet type

Fig 9A

IPR2024-001428
HP – EX1002, Page 436 of 487

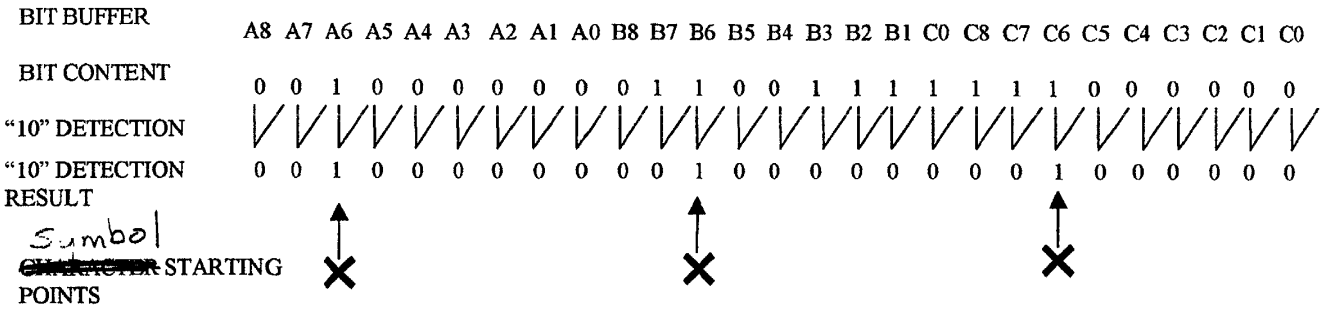

BIT BUFFER
A8 A7 A6 A5 A4 A3 A2 A1 A0 B8 B7 B6 B5 B4 B3 B2 B1 C0 C8 C7 C6 C5 C4 C3 C2 C1 C0
BIT CONTENT
0 0 1 0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0
"10" DETECTION
"10" DETECTION RESULT
0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
Symbol
STARTING POINTS

FIG.10

Fig 9B

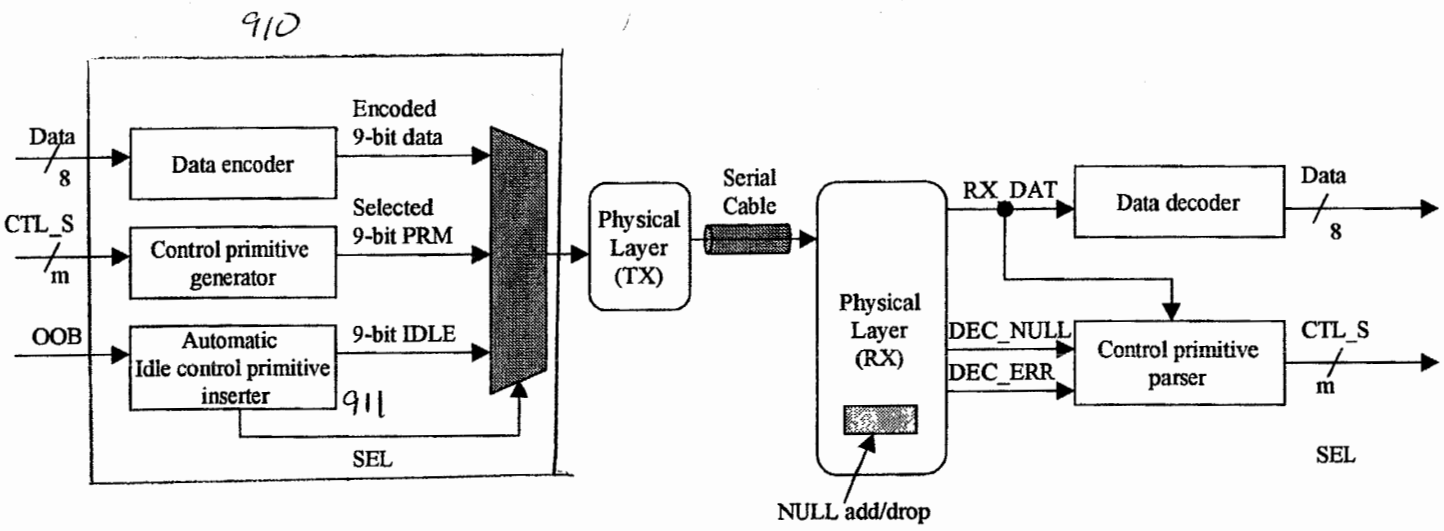

910
Data
8
CTL_S
m
OOB
Data encoder
Control primitive generator
Automatic Idle control primitive inserter
Encoded 9-bit data
Selected 9-bit PRM
9-bit IDLE
911
SEL
Physical Layer (TX)
Serial Cable
Physical Layer (RX)
NULL add/drop
RX_DAT
Data decoder
Data
8
DEC_NULL
DEC_ERR
Control primitive parser
CTL_S
m
SEL

Fig. 9C

T0200 T1656001

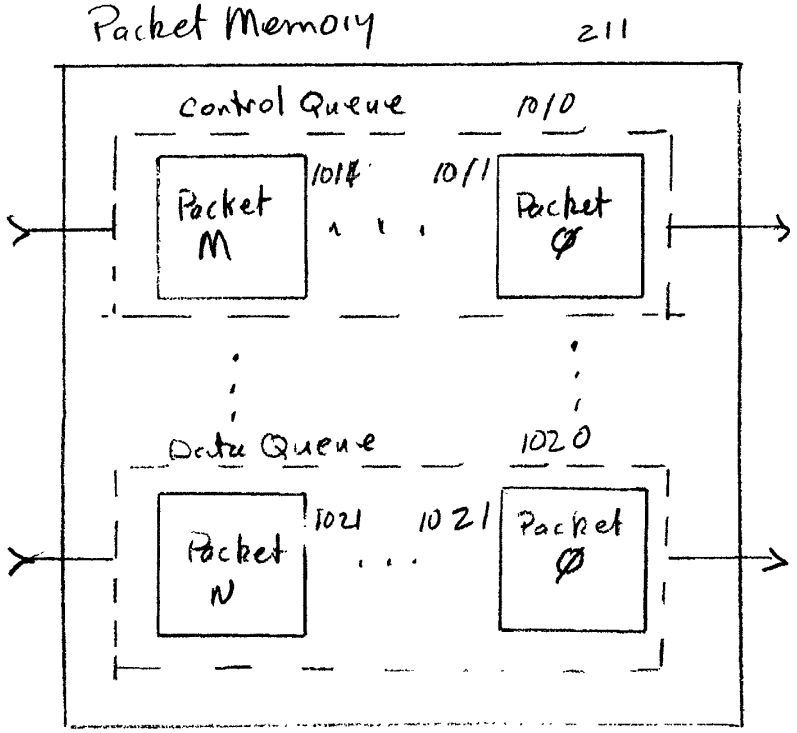


Fig 10

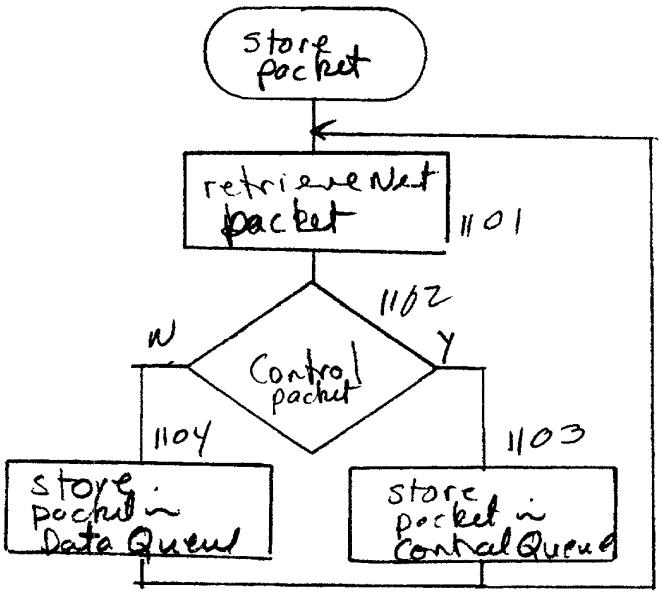


Fig 11

FOSTER-PEPPER5008

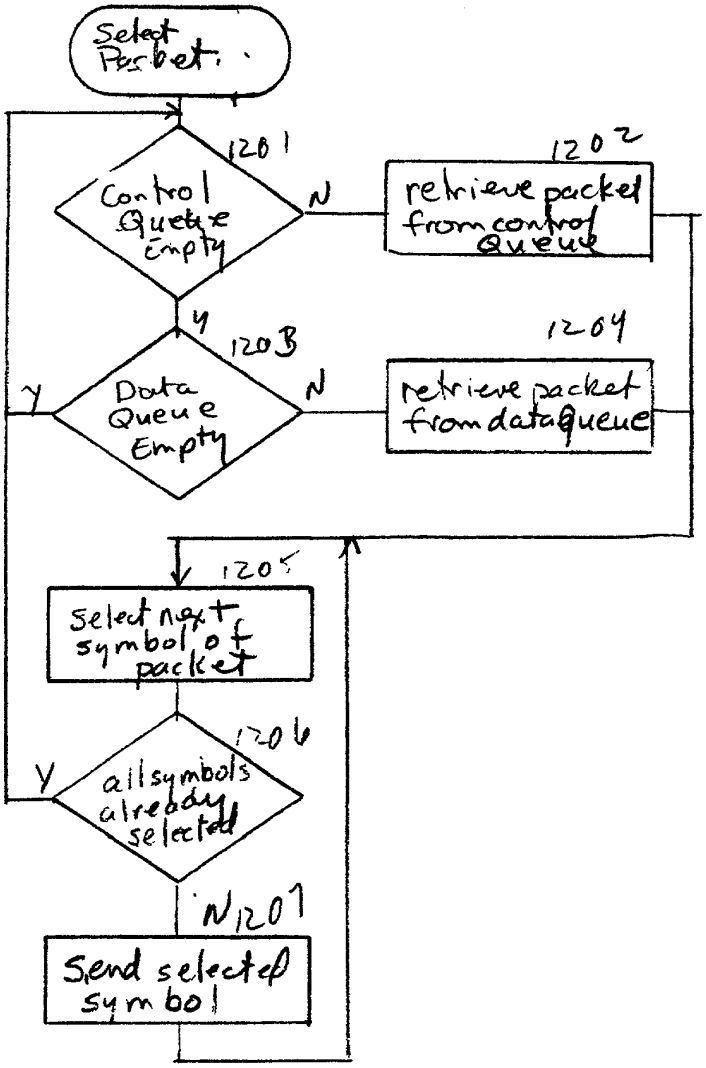


Fig 12

EX1002 p.441 of 487

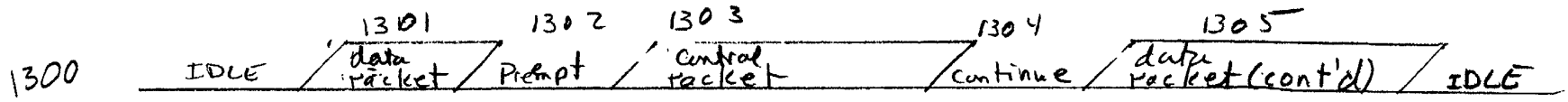


Fig 13

IPR2024-001428
HP – EX1002, Page 442 of 487

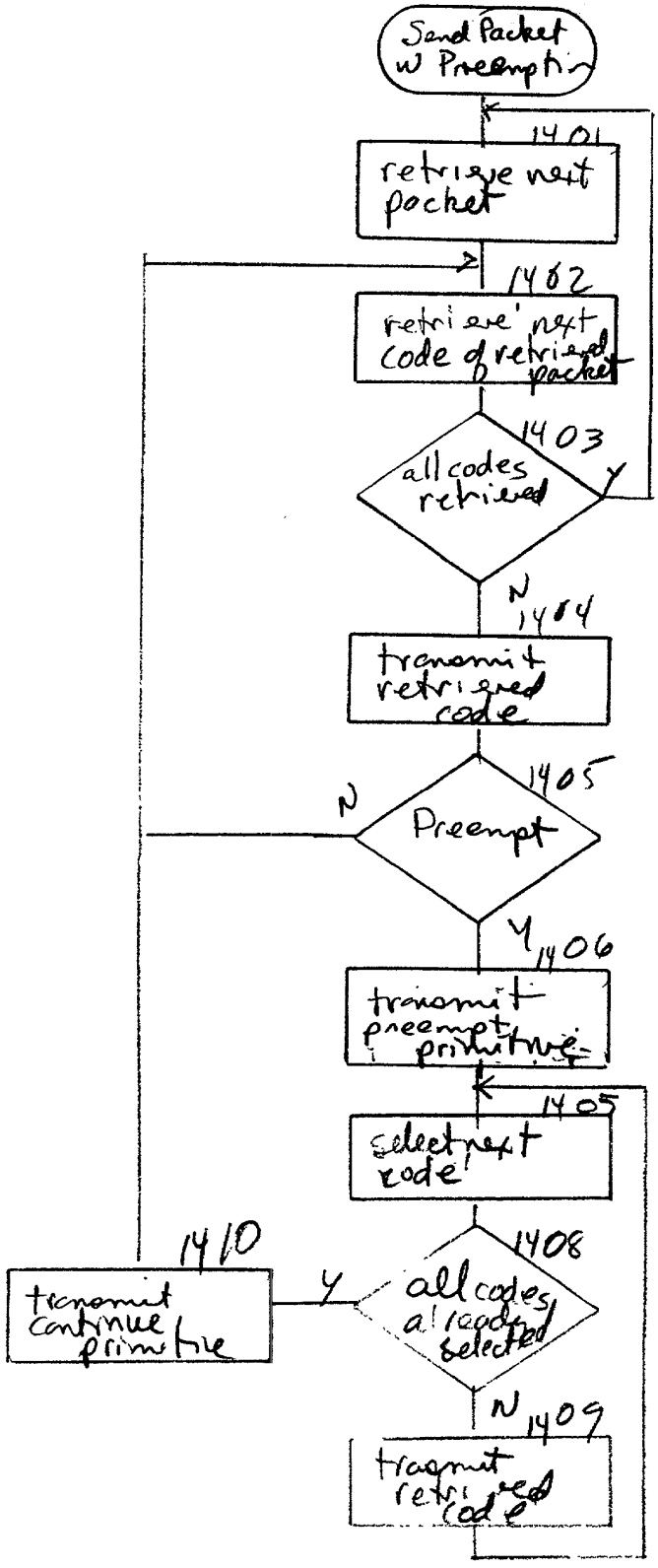


Fig 14

IPR2024-001428
HP – EX1002, Page 443 of 487

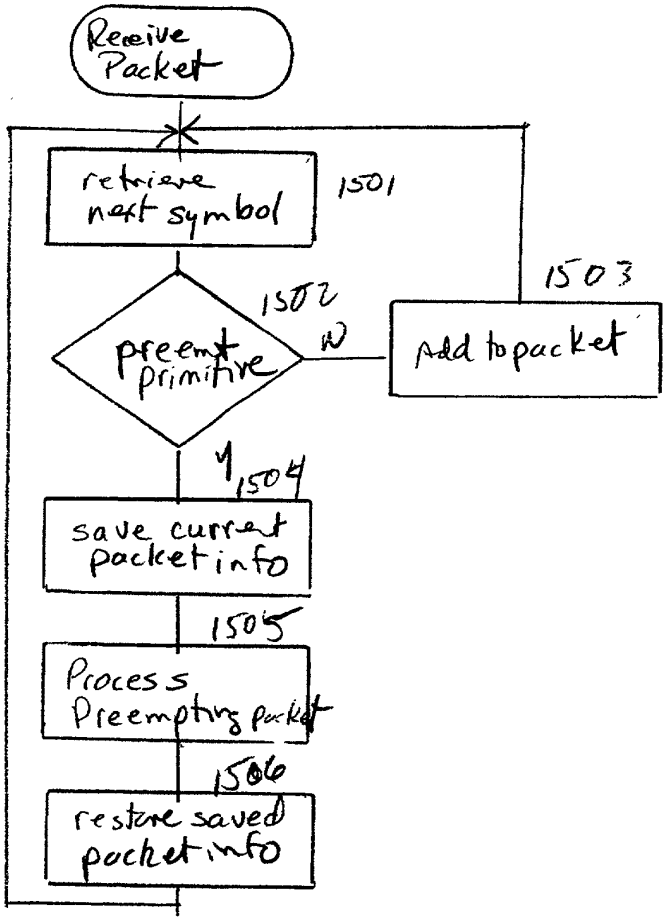


Fig 15

IPR2024-001428
HP – EX1002, Page 444 of 487

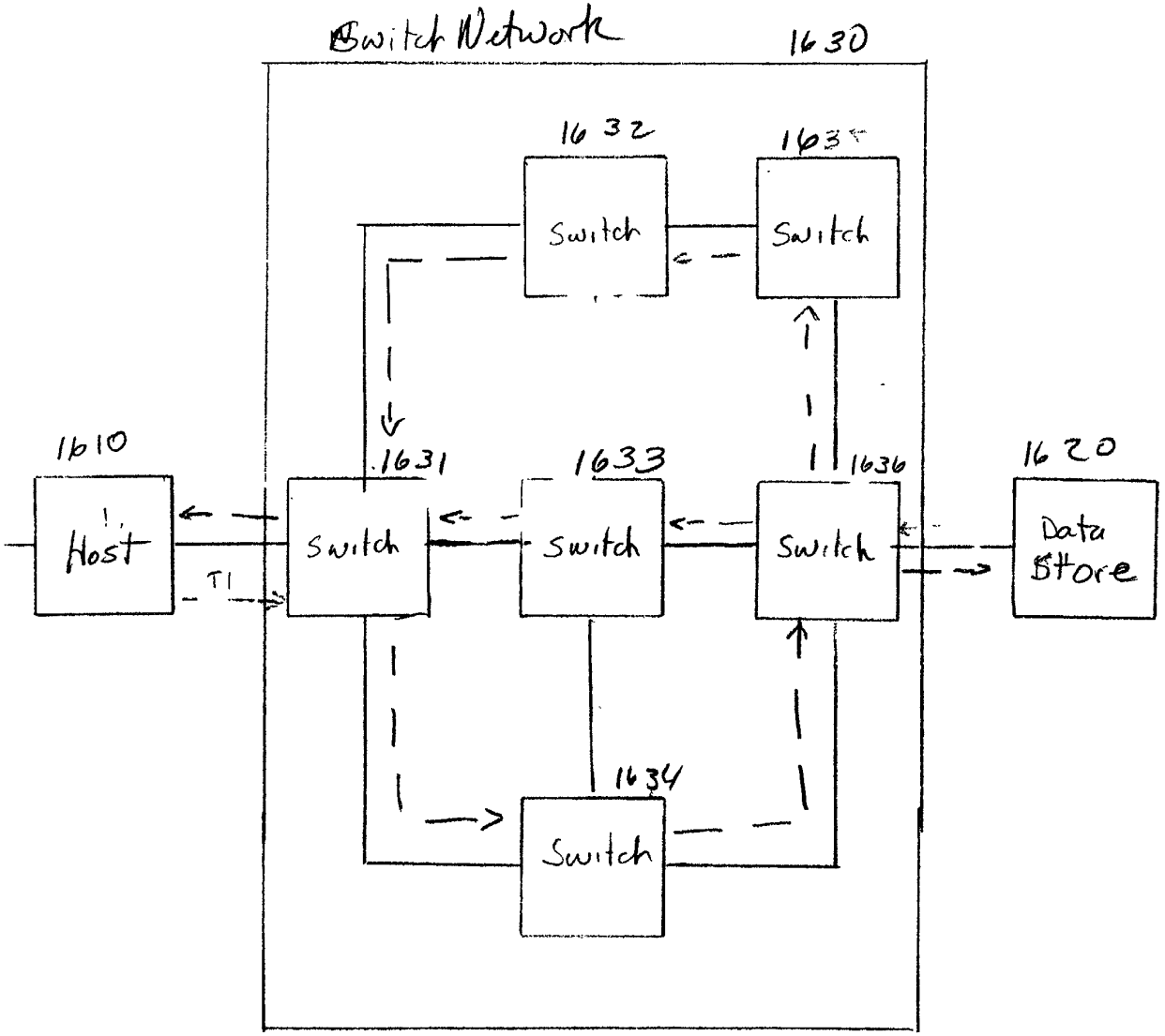


Fig 16

IPR2024-001428
HP – EX1002, Page 445 of 487

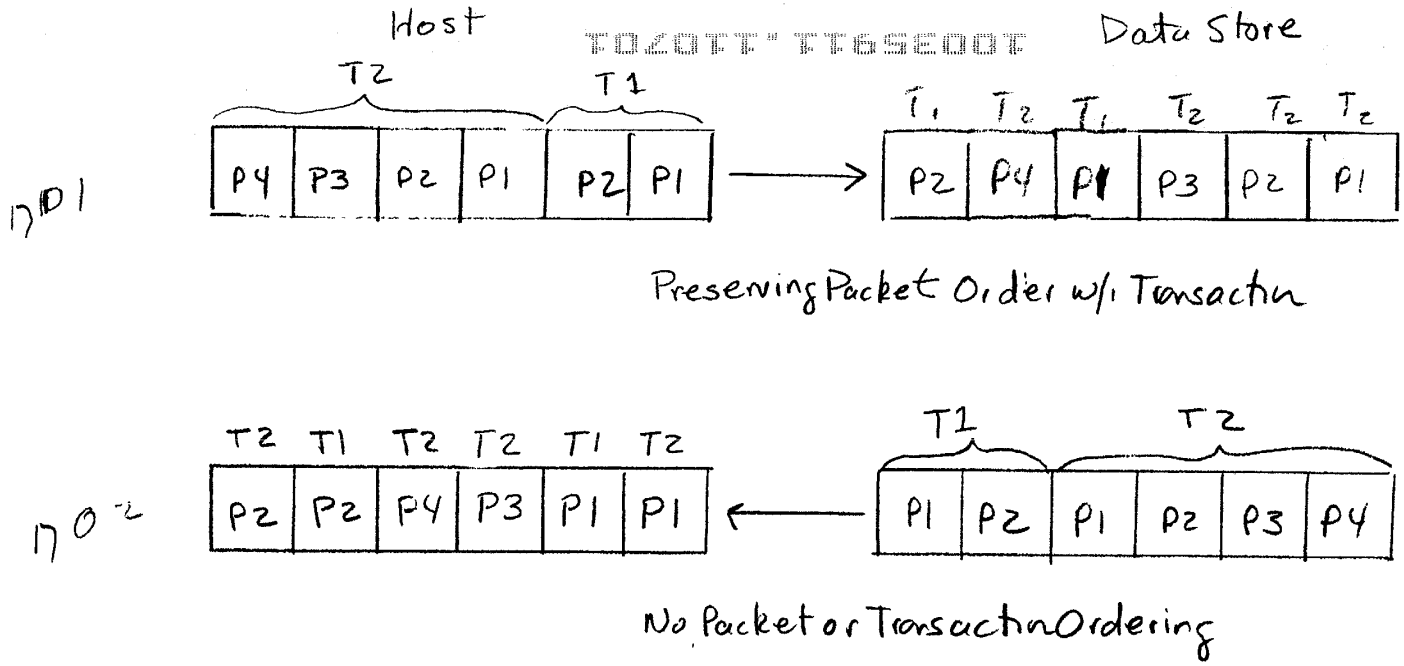

Host
Data Store
T2
T1
P4 P3 P2 P1 P2 P1
P2 P4 P1 P3 P2 P1
Preserving Packet Order w/ Transaction
T2 T1 T2 T2 T1 T2
P2 P2 P4 P3 P1 P1
T1
T2
P1 P2 P1 P2 P3 P4
No Packet or Transaction Ordering

Fig 17

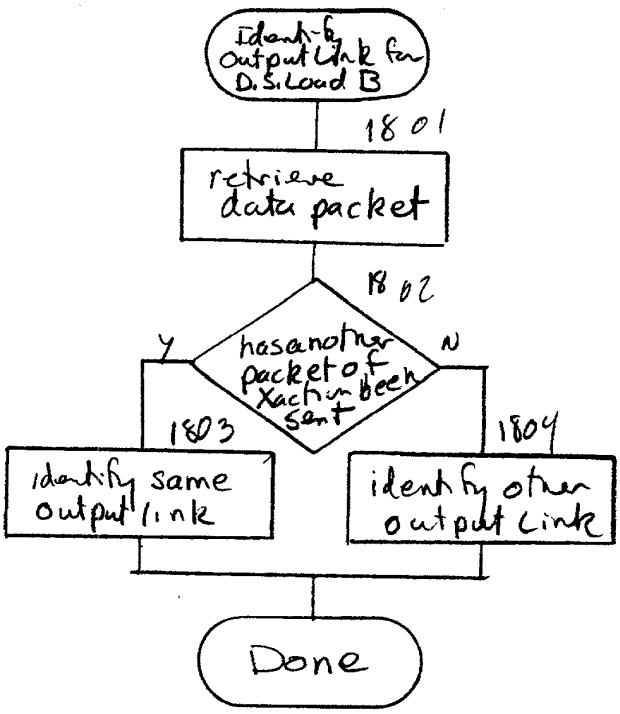


Fig 18

FIG 18 - FIG 18

FOZOFF TT6SE00F

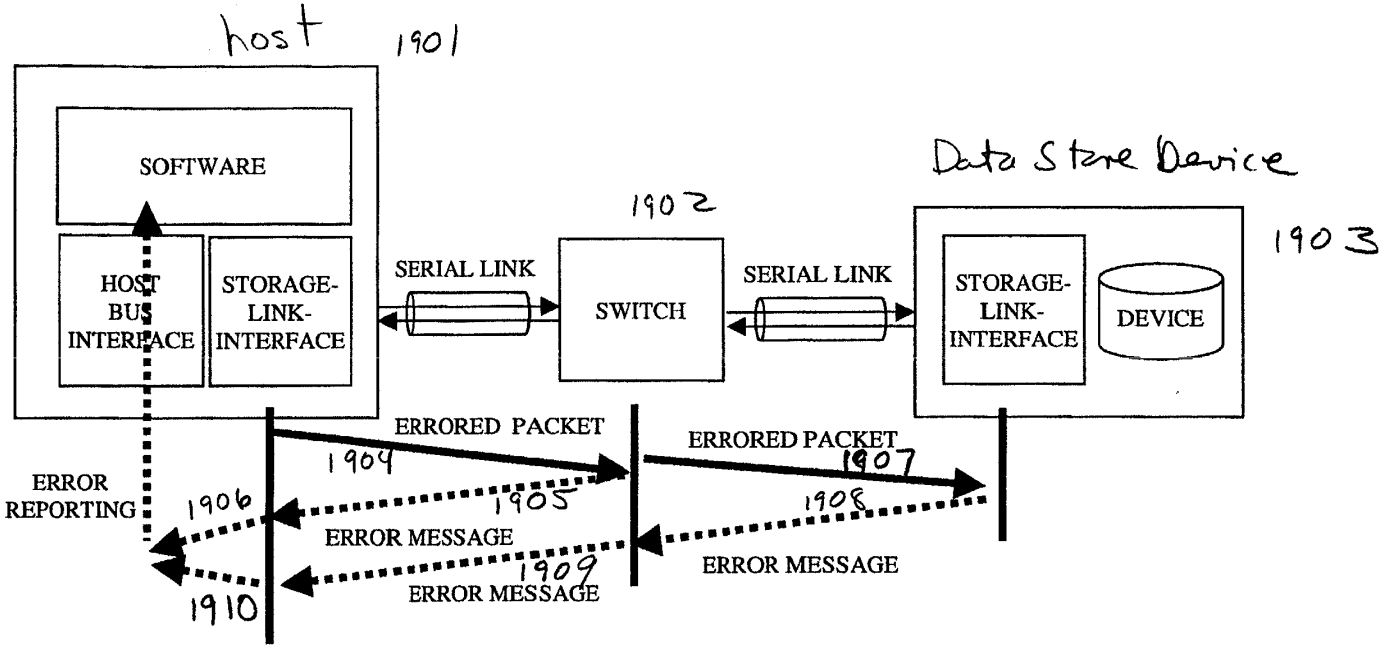


Fig 19A

TOZOTT" TT69E00T

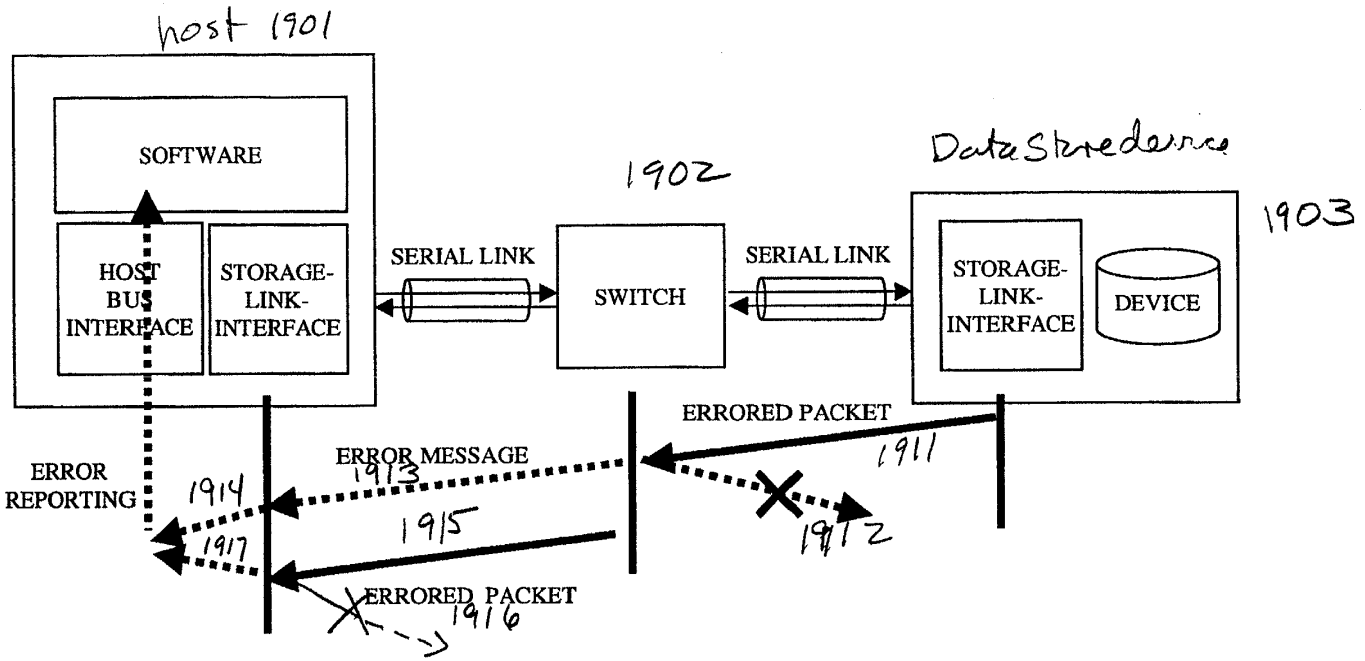


Fig 19B

19c

IPR2024-001428
HP – EX1002, Page 450 of 487

U.S. Patent Sep. 15, 1998

Fig 20

8b code	9bit symbol
0000 0000	101010101
0000 0001	101010100
0000 0010	101010011
⋮	⋮
0101 0101	010101010
⋮	⋮
1010 1010	001010101
⋮	⋮
1110 1110	011101101
1110 1111	100010010
⋮	⋮
1111 1111	010101010

Fig 21A

10035911.110701

TOZOFF TT6SE00T

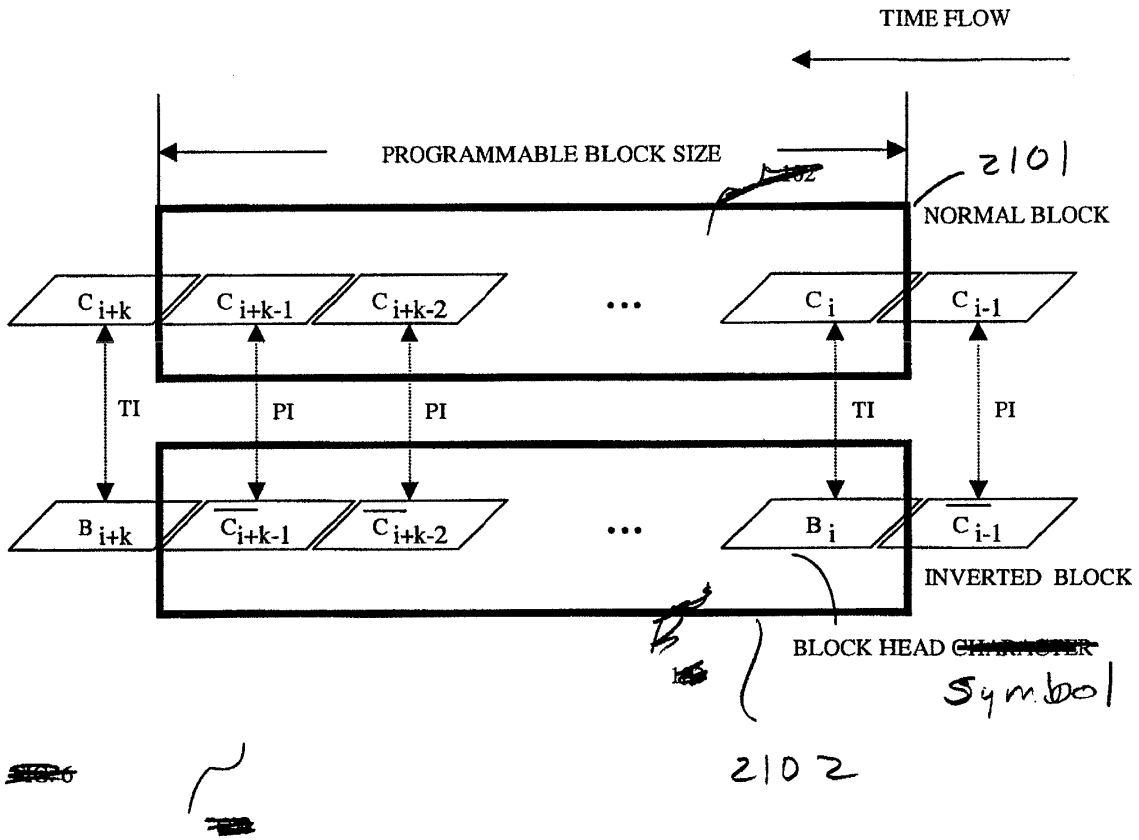


Fig 21B

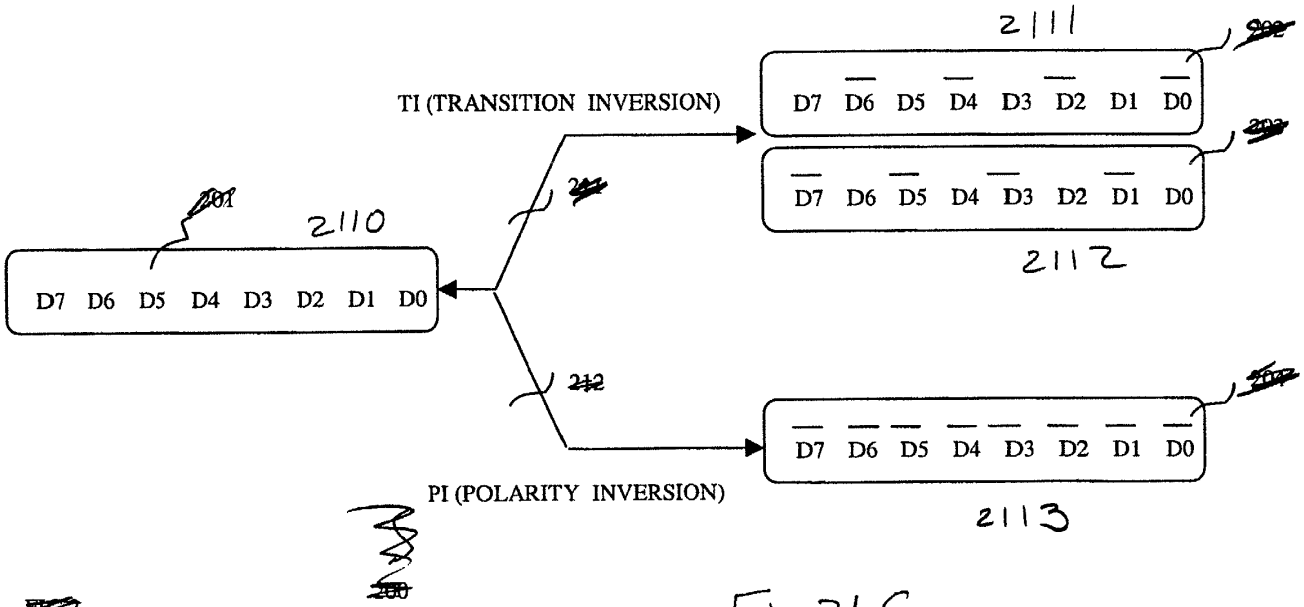

2111
TI (TRANSITION INVERSION)
D7 D6 D5 D4 D3 D2 D1 D0
D7 D6 D5 D4 D3 D2 D1 D0
2112
2110
D7 D6 D5 D4 D3 D2 D1 D0
PI (POLARITY INVERSION)
D7 D6 D5 D4 D3 D2 D1 D0
2113

Fig 21C

TOZOTT" TT6SZOOT

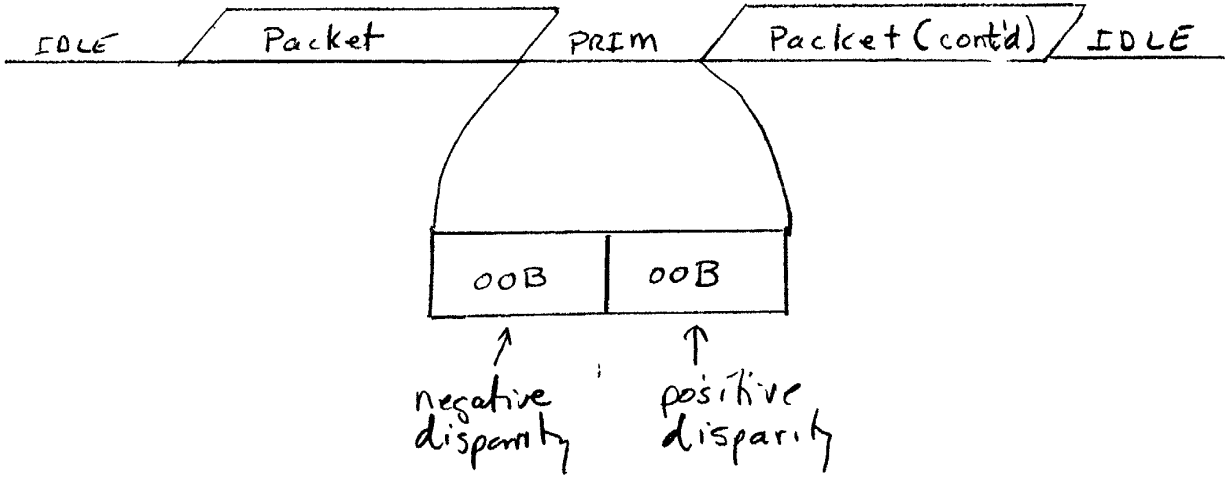


Fig 22

IPR2024-001428
HP – EX1002, Page 455 of 487

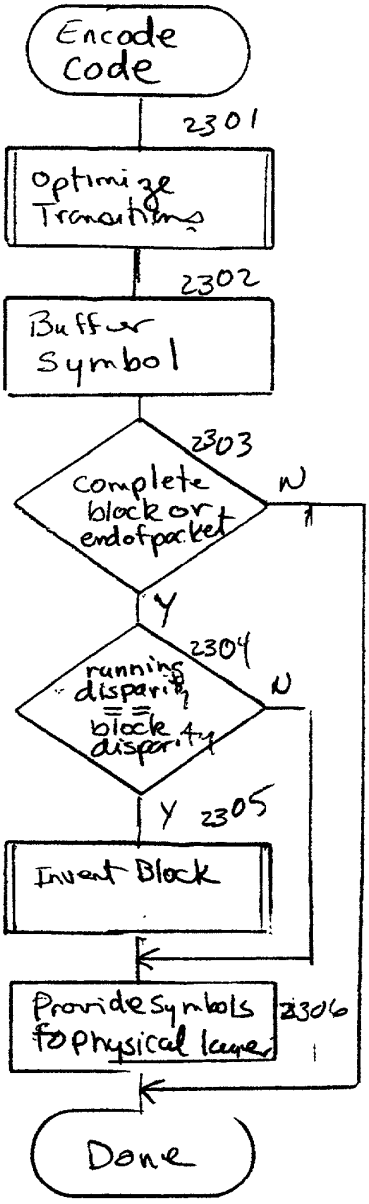


Fig 23

EXHIBIT 1002

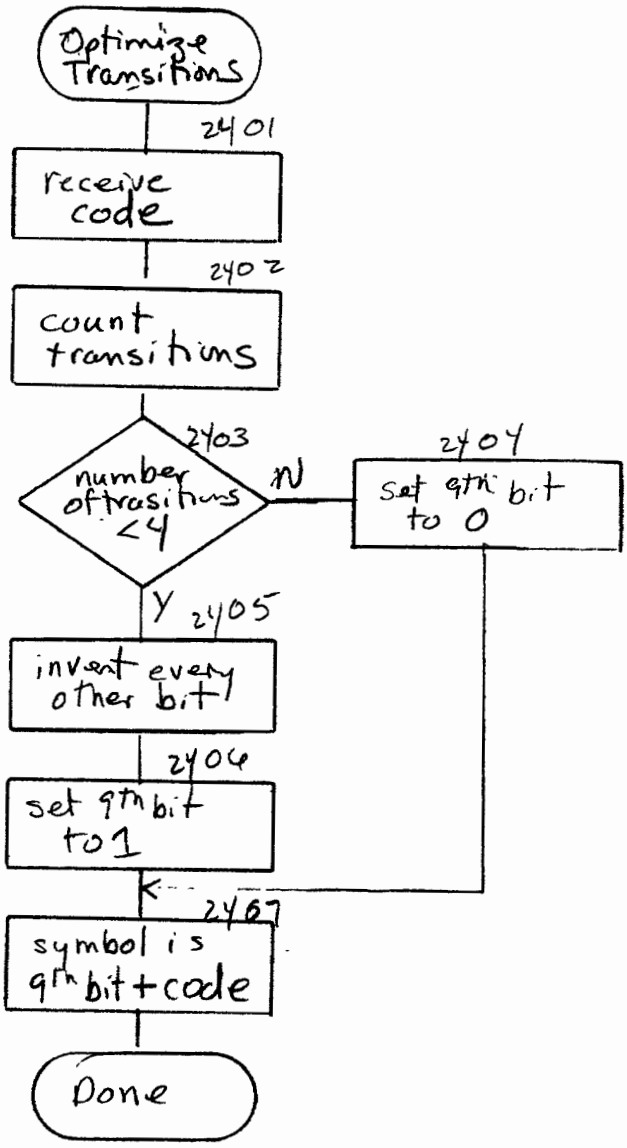


Fig 24

US 7,116,568 B2

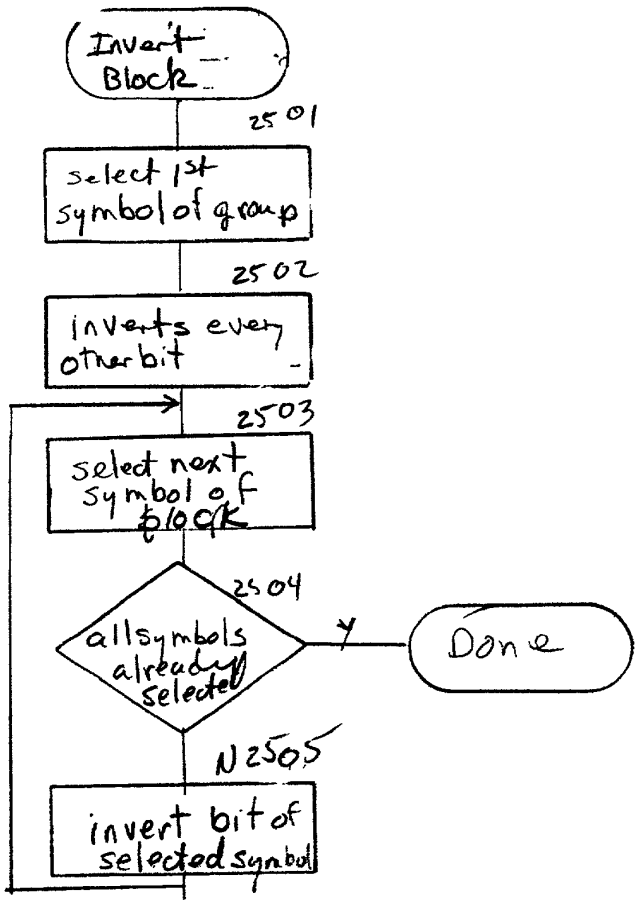


Fig 25

PROOF SHEET - PROOF SHEET

FOUO-FTS2001

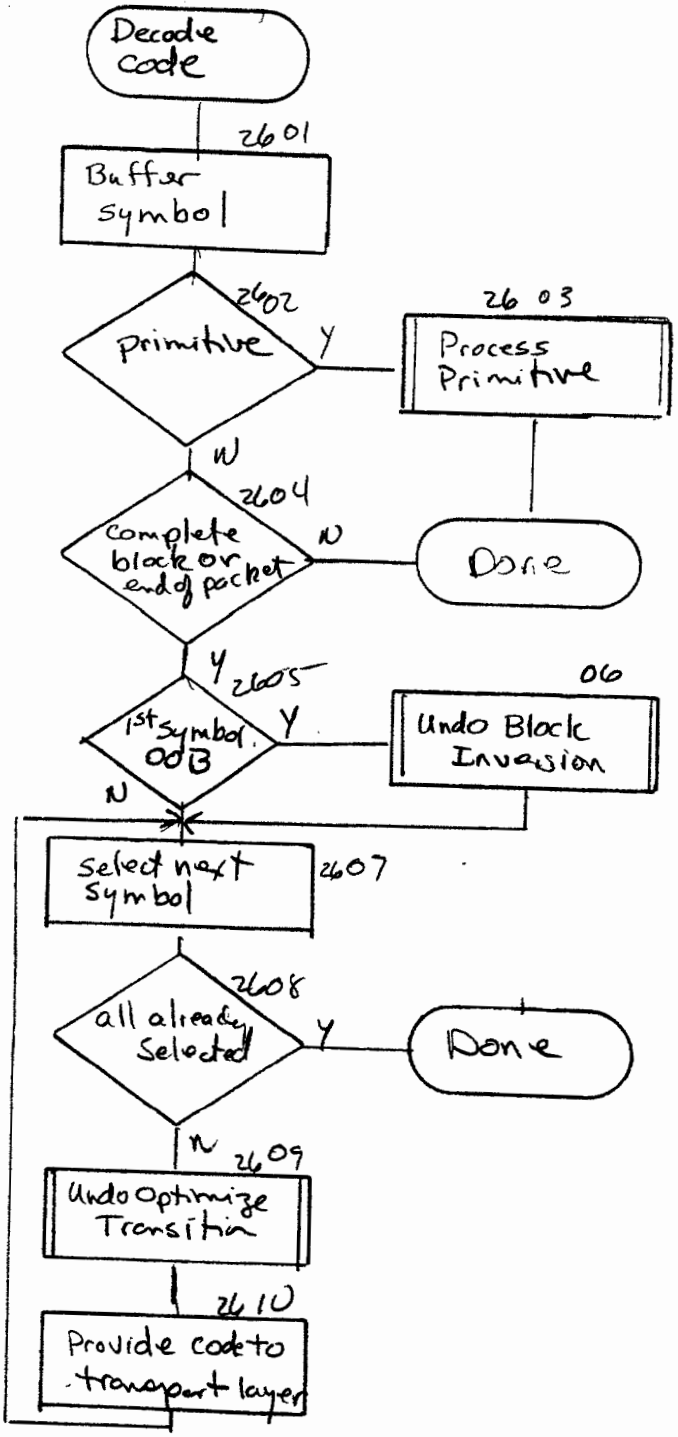


Fig 26

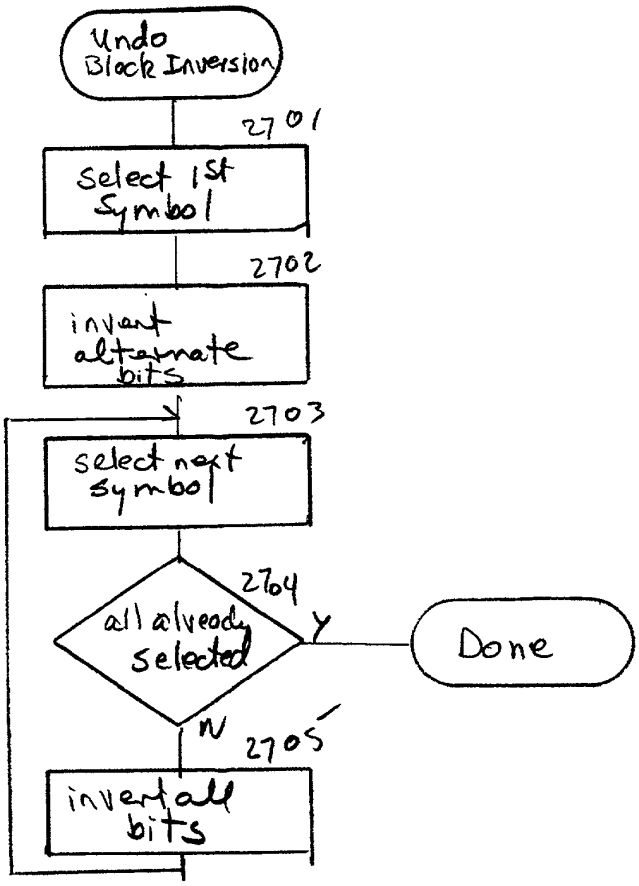


Fig 27

Attorney Docket No. 10030574-1

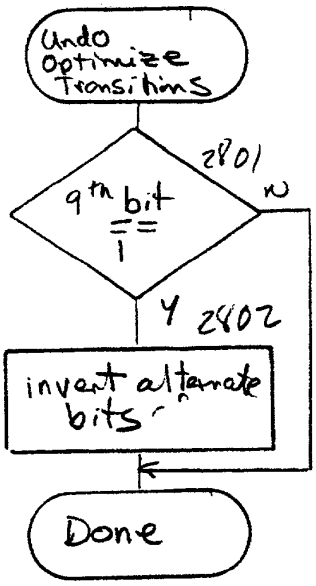


Fig 28

10/6/03, EAST Version: 1.03.0001

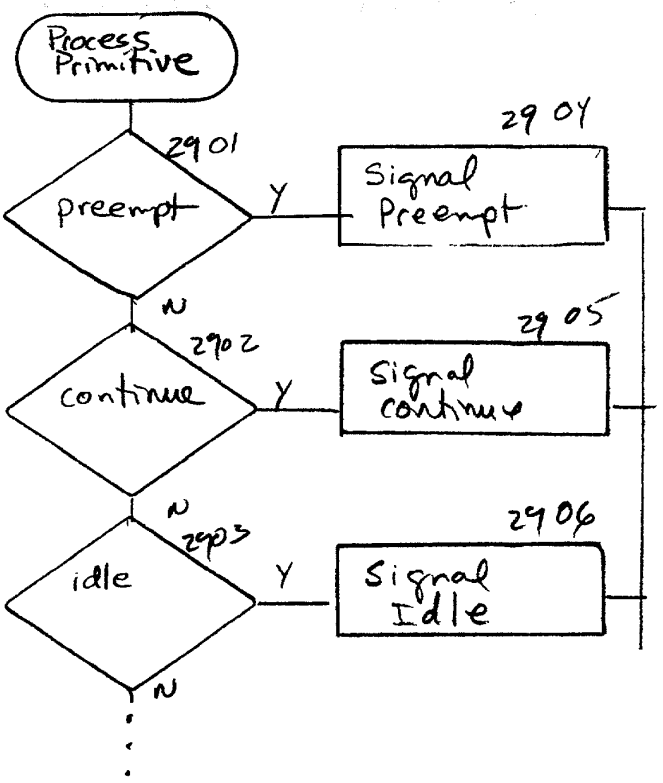


Fig 29

Petitioner's Exhibit 1002

TOZOTT" TTGSE00T

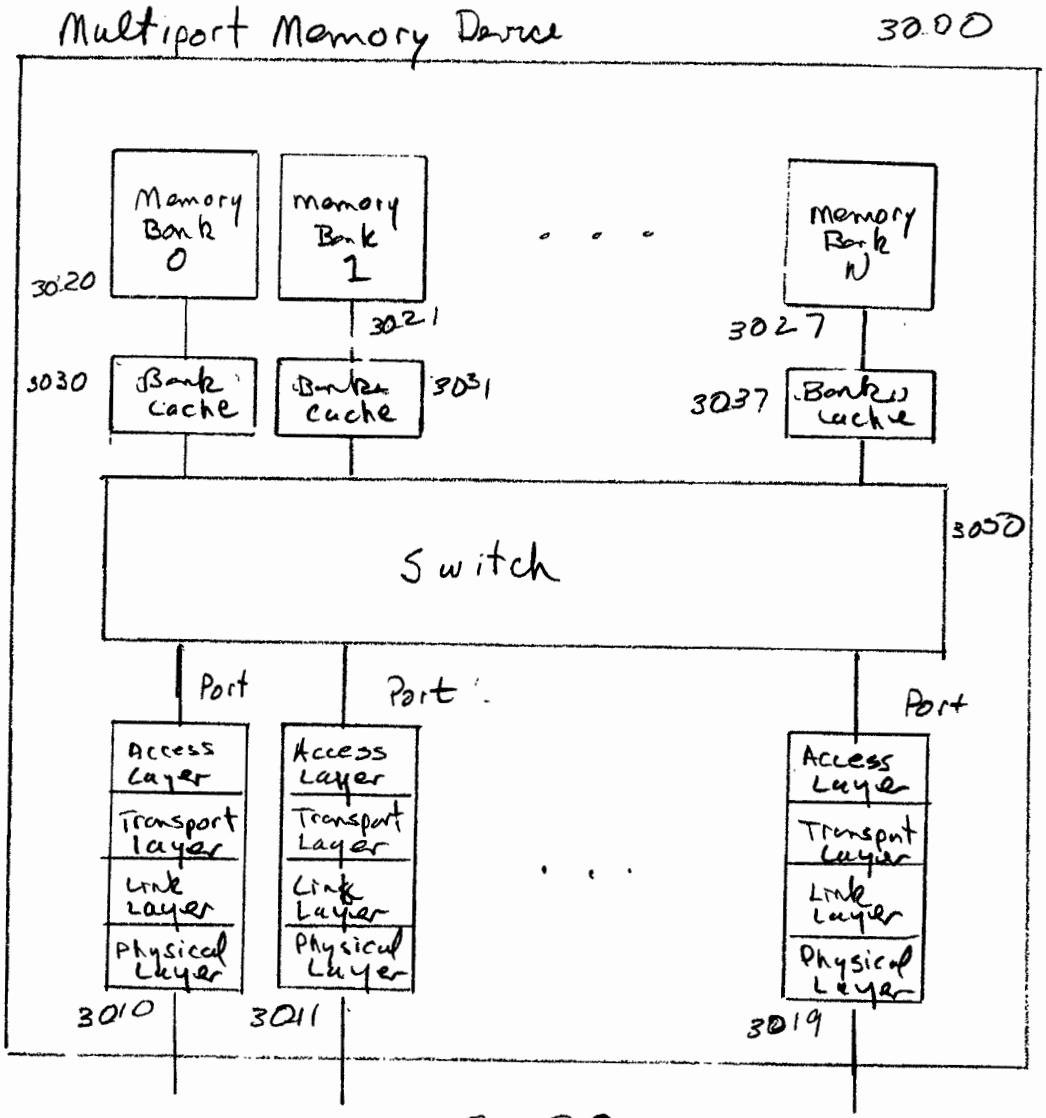


Fig 30

TOZOTT" TTGSEOOT

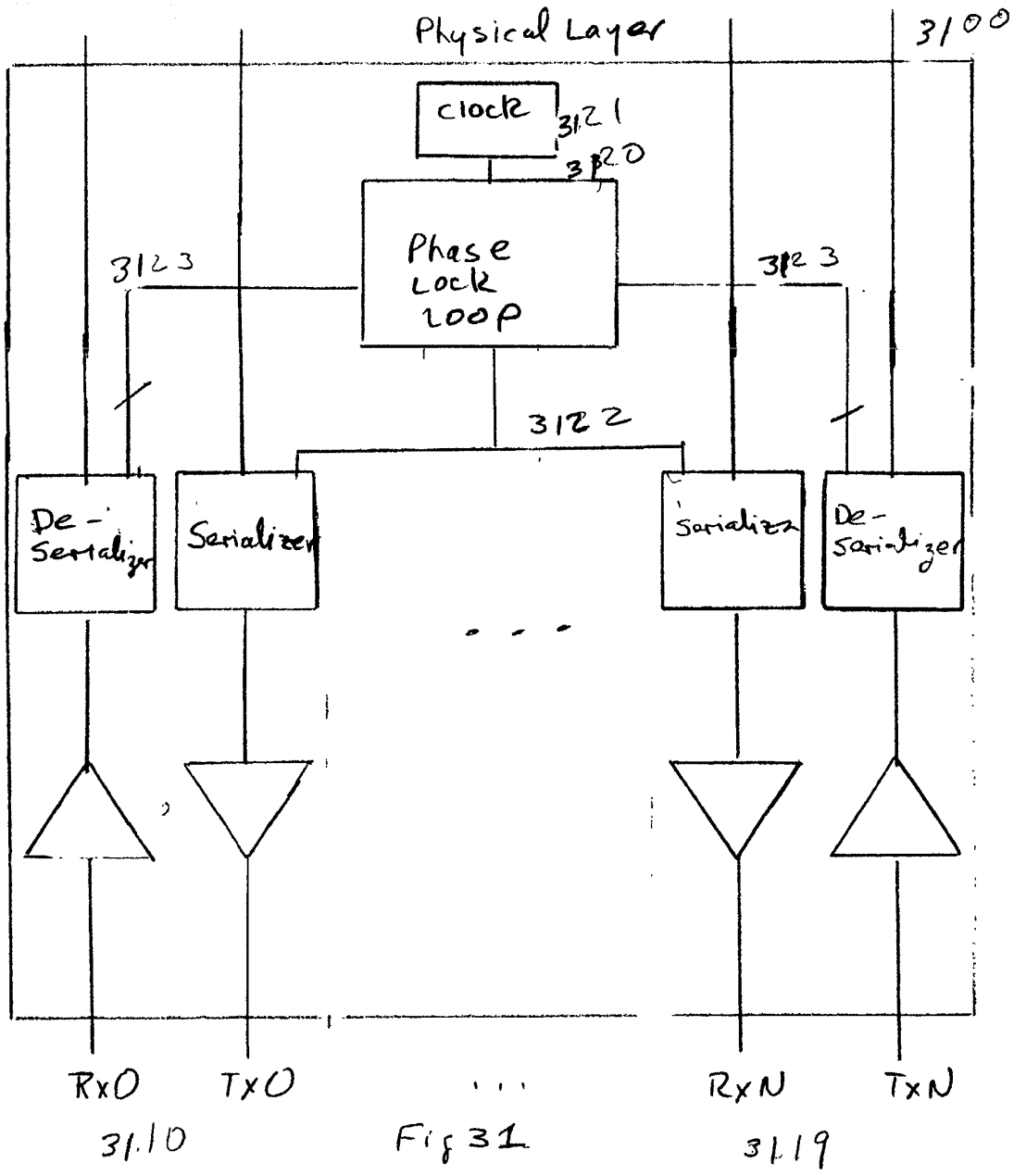


Fig 31

IPR2024-001428
HP – EX1002, Page 464 of 487

Input Queue 3201

Port	R/W	Address	Data
3	R	1000	
4	W	4000	10....1
3	W	1000	111...0
3	R	2000	

Output Queue 3202

Valid	Port	Data
1	3	11...0
0		
0		
1	3	101...1

Fig 32

IPR2024-001428
HP – EX1002, Page 465 of 487

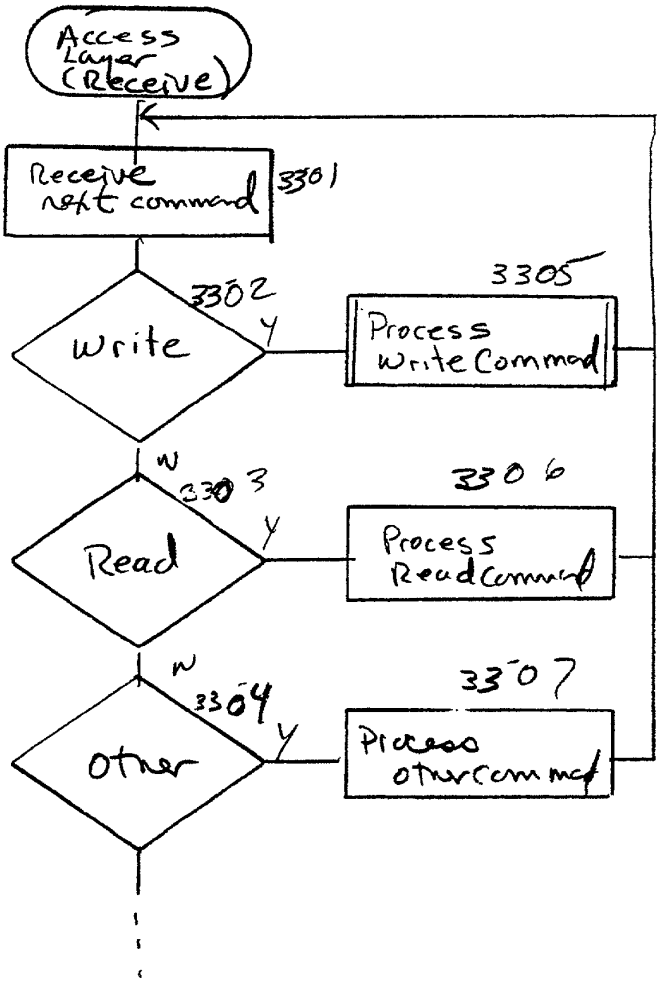


Fig 33

IPR2024-001428
HP – EX1002, Page 466 of 487

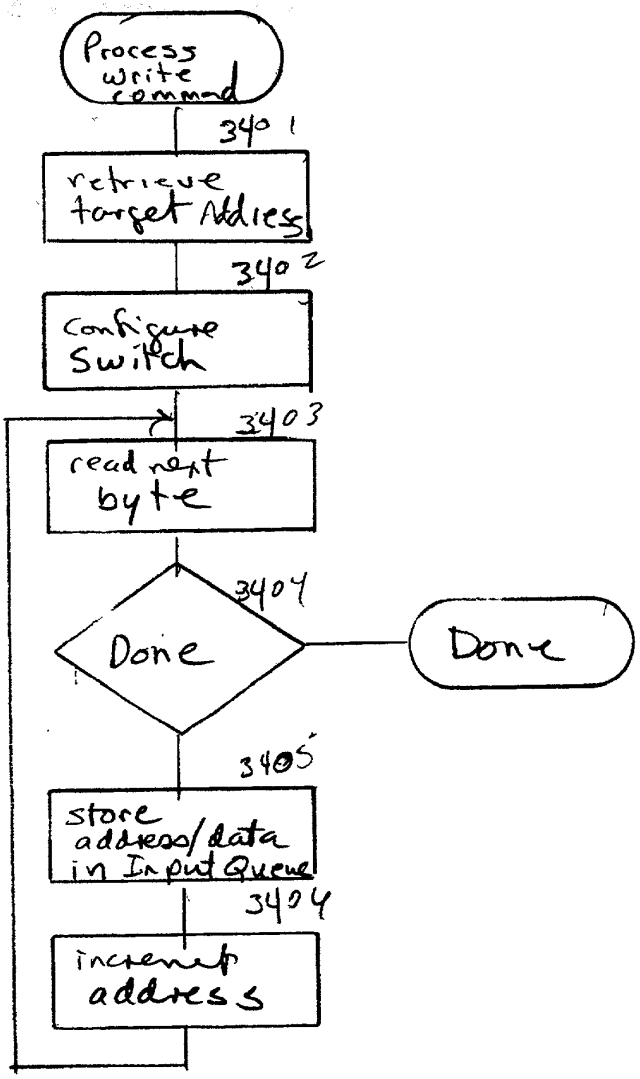


Fig 34

IPR2024-001428
HP – EX1002, Page 467 of 487

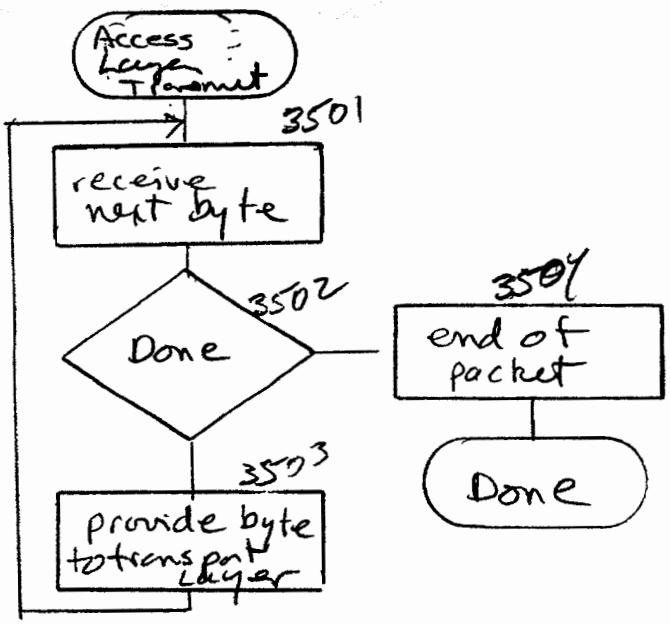


Fig 35

EXHIBIT PRESENTED

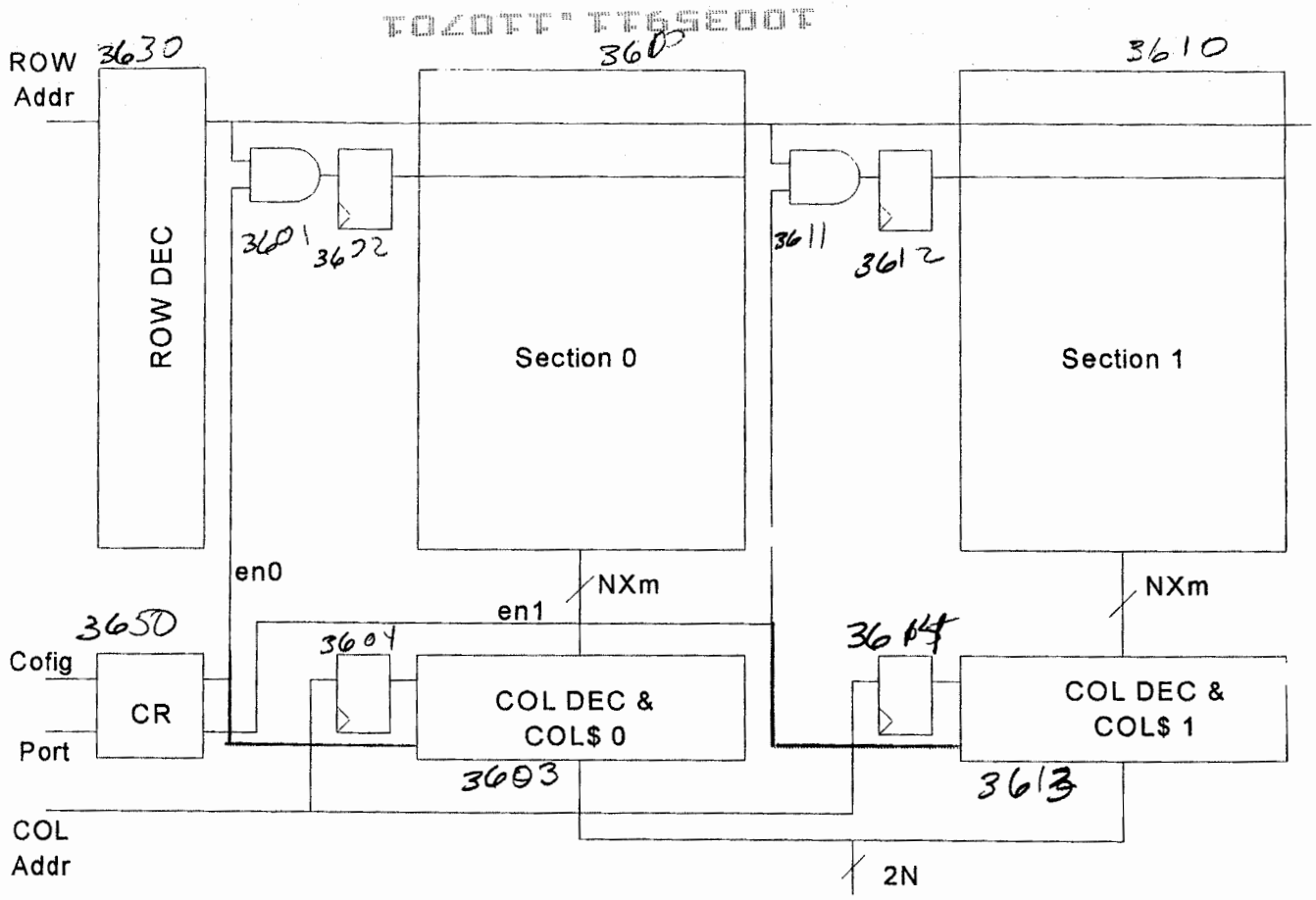
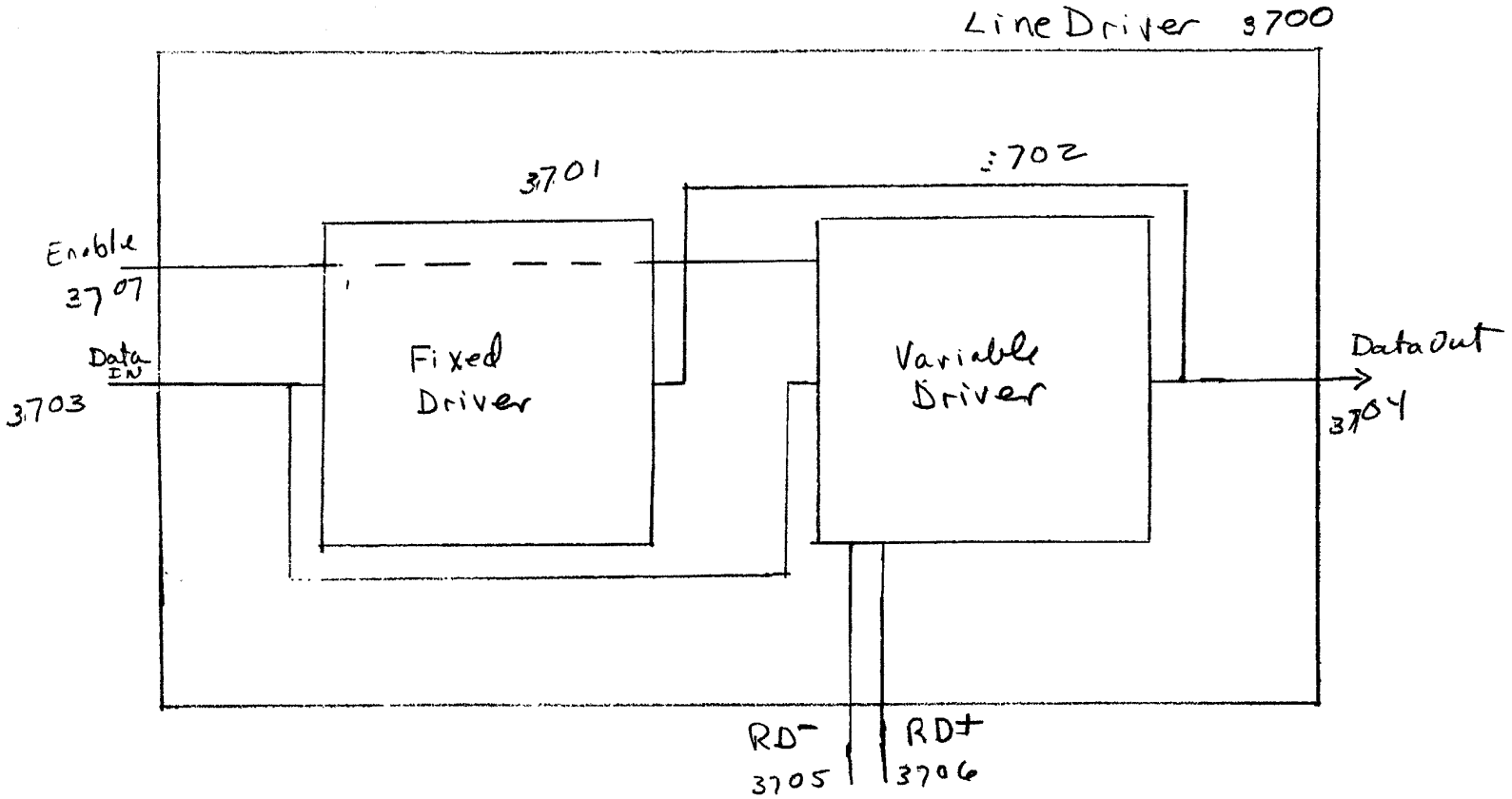

ROW
Addr
3630
ROW DEC
3605
Section 0
3610
Section 1
3601
3602
3611
3612
en0
en1
NXm
NXm
3650
Cofig
Port
CR
3604
COL DEC &
COL$ 0
3606
COL DEC &
COL$ 1
3603
3613
COL
Addr
2N

Fig 36



Variable Driver
- $RD^+ \wedge \overline{DataIN}$ = pull down
- $RD^- \wedge DataIN$ = pull up

Fig 37A

IPR2024-001428
HP – EX1002, Page 470 of 487

FOZOTT TT6SE00T

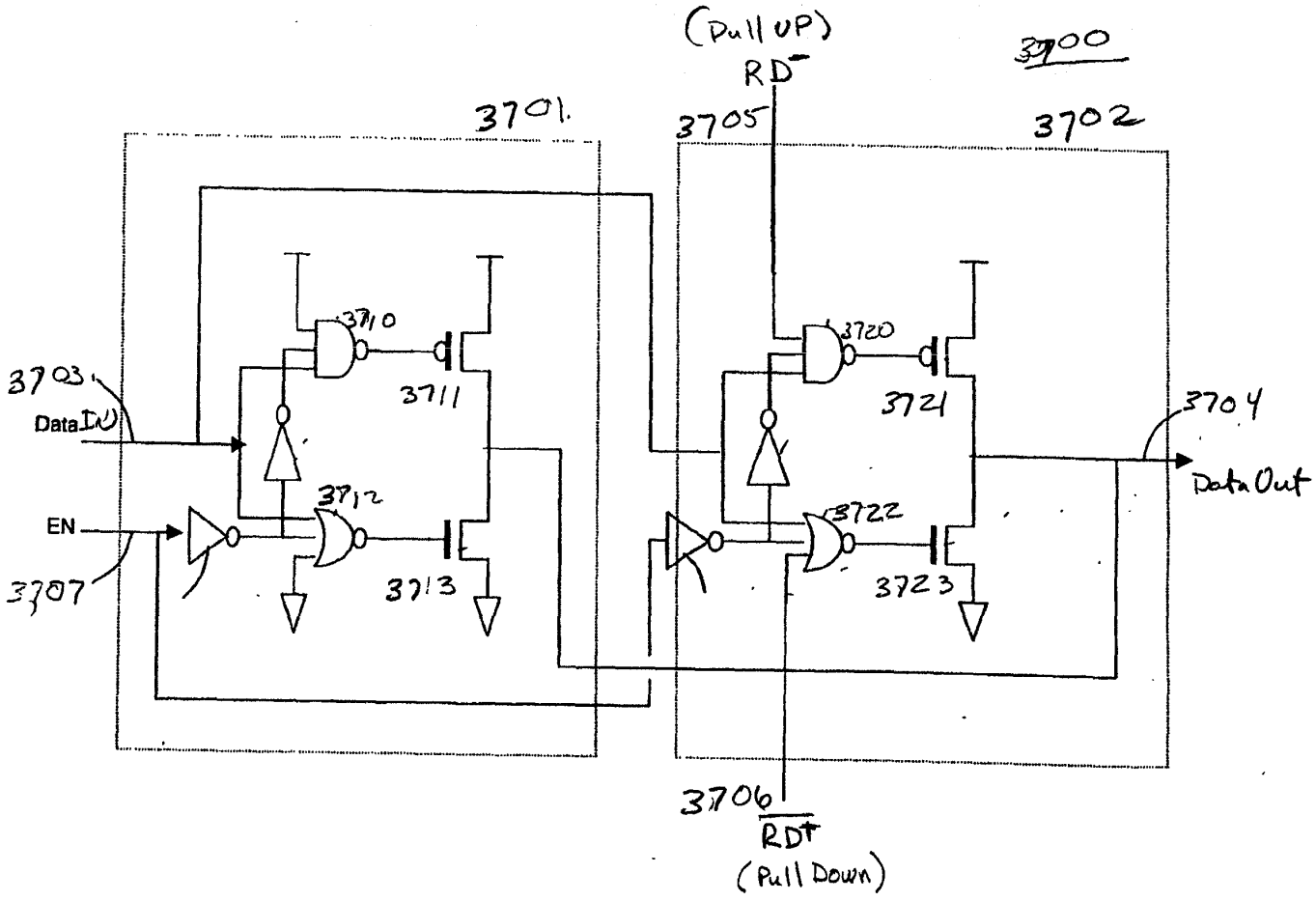


Fig 37B

IPR2024-001428
HP – EX1002, Page 471 of 487

TO:OFF TT6SE00T

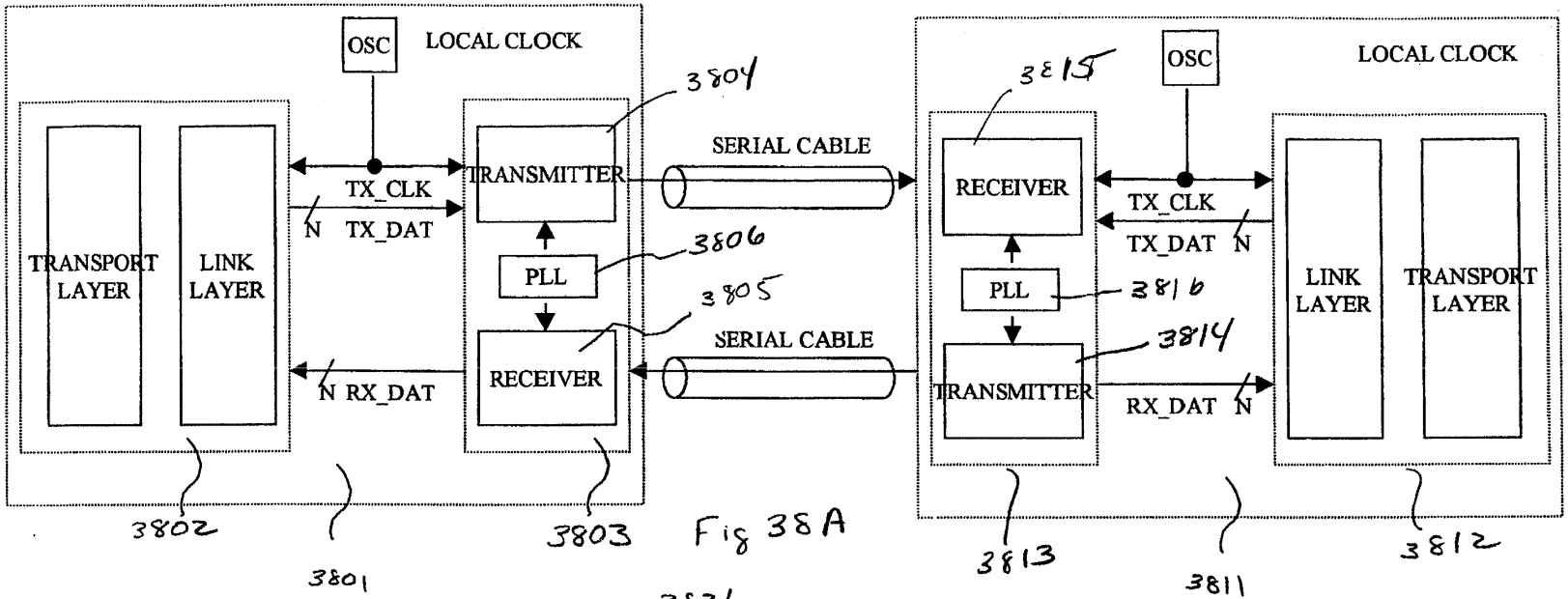


Fig 38A

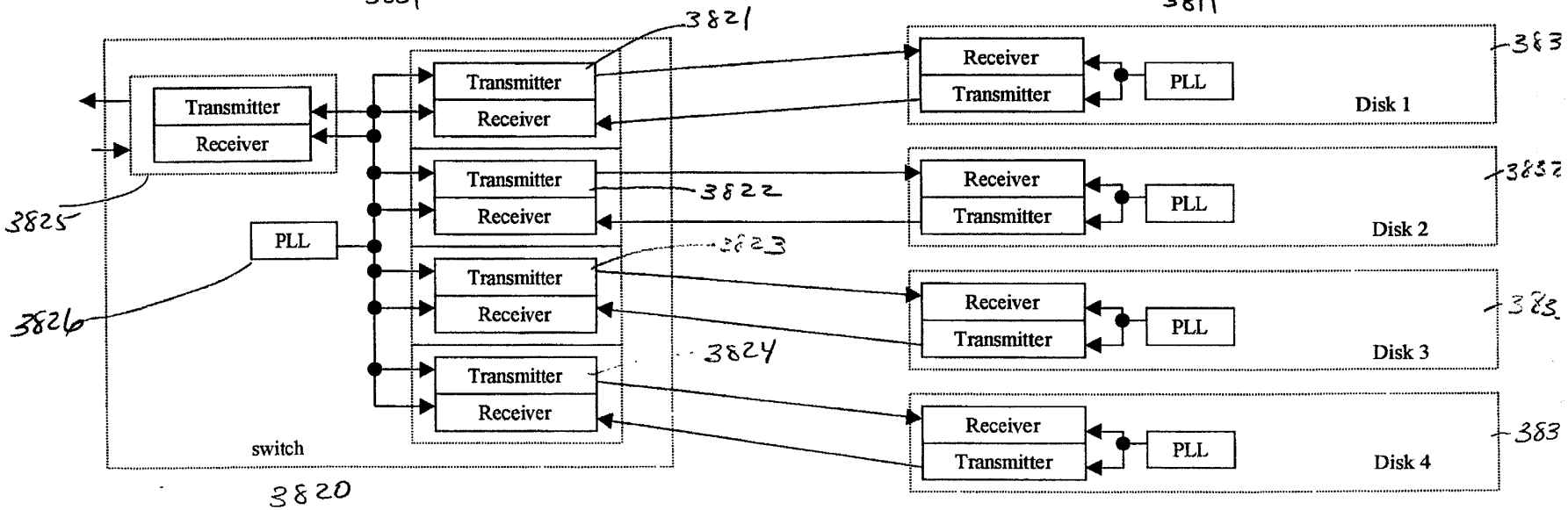


Fig 38B

TOZOTT TTGSEDOT

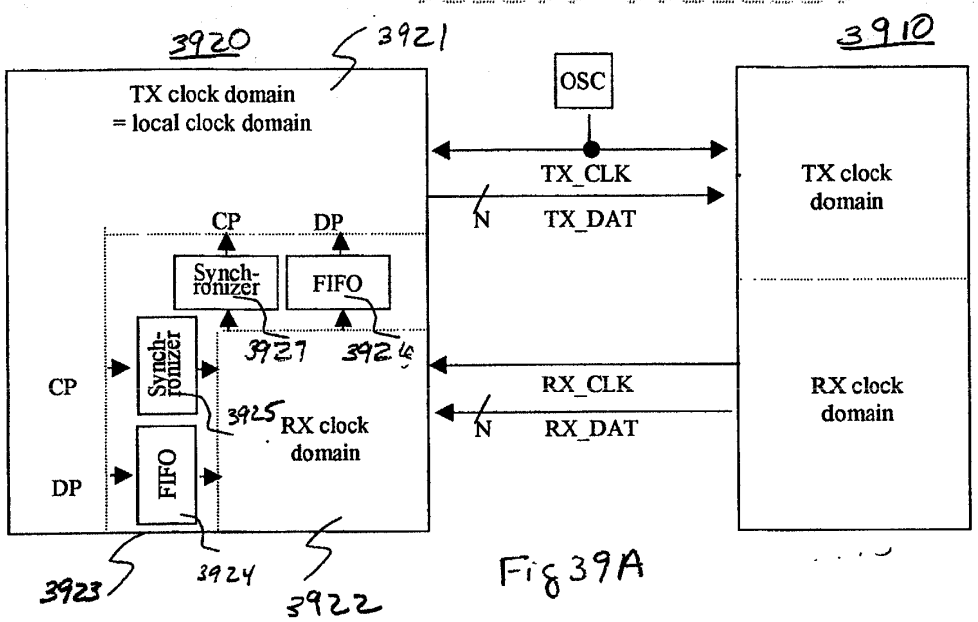


Fig 39A

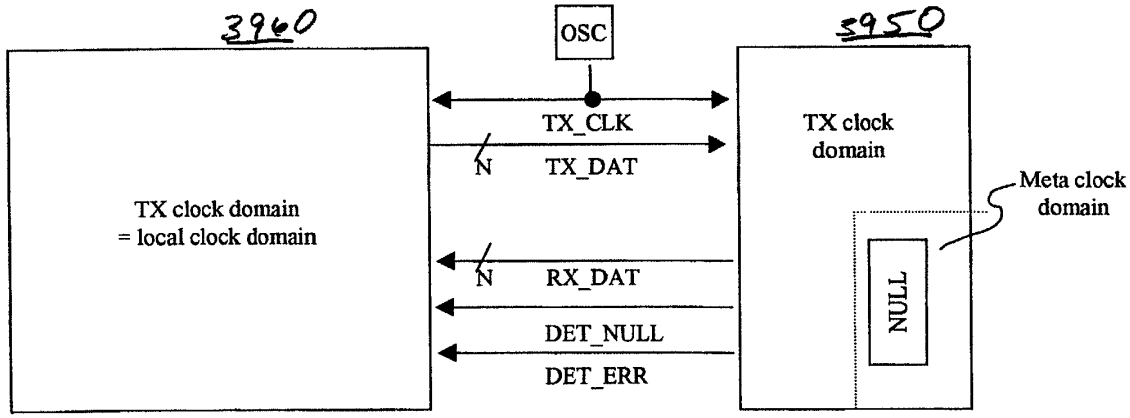


Fig 39B

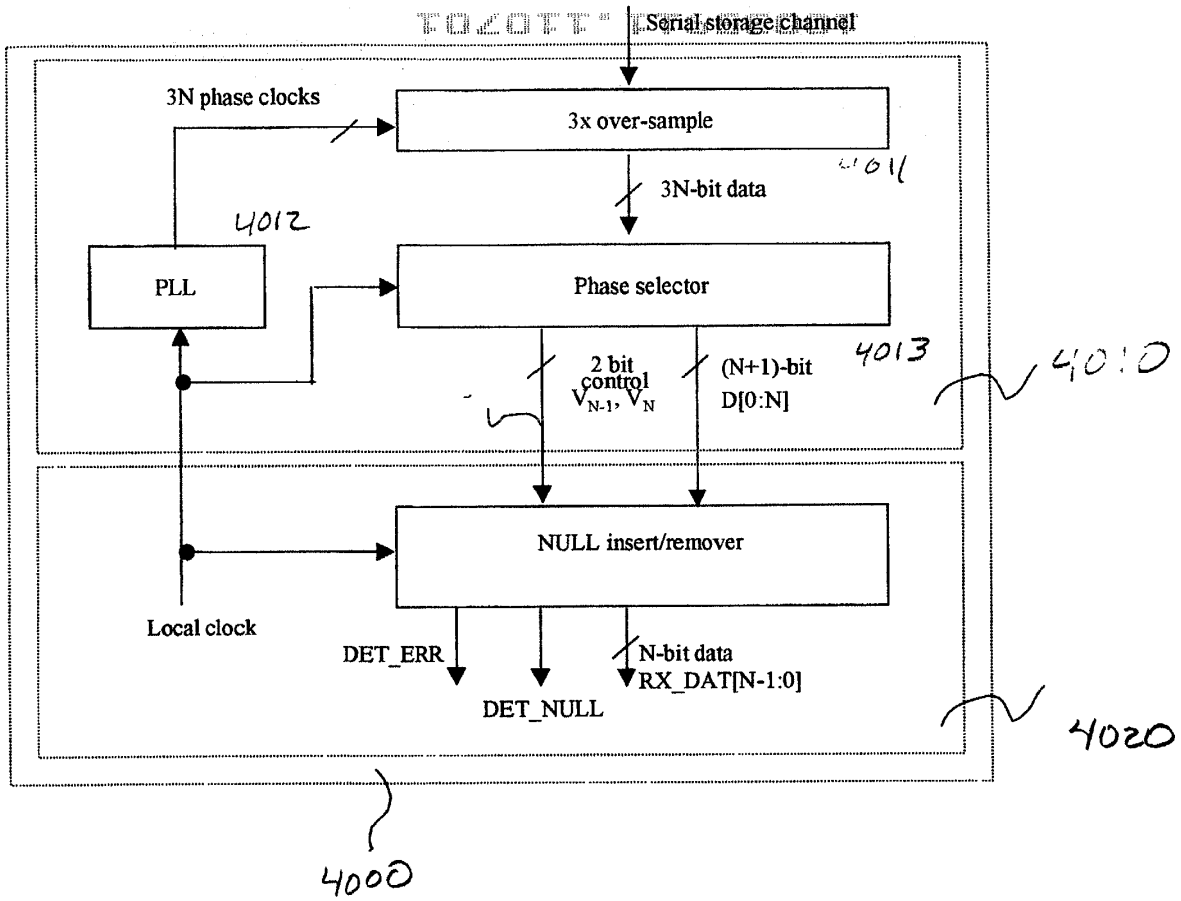

Serial storage channel
3N phase clocks
3x over-sample
4011
3N-bit data
4012
PLL
Phase selector
4013
2 bit control
V_{N-1}, V_N
(N+1)-bit
D[0:N]
4010
NULL insert/remover
Local clock
DET_ERR
N-bit data
RX_DAT[N-1:0]
DET_NULL
4020
4000

Fig 40

T0ZOTT" TT6SE00T

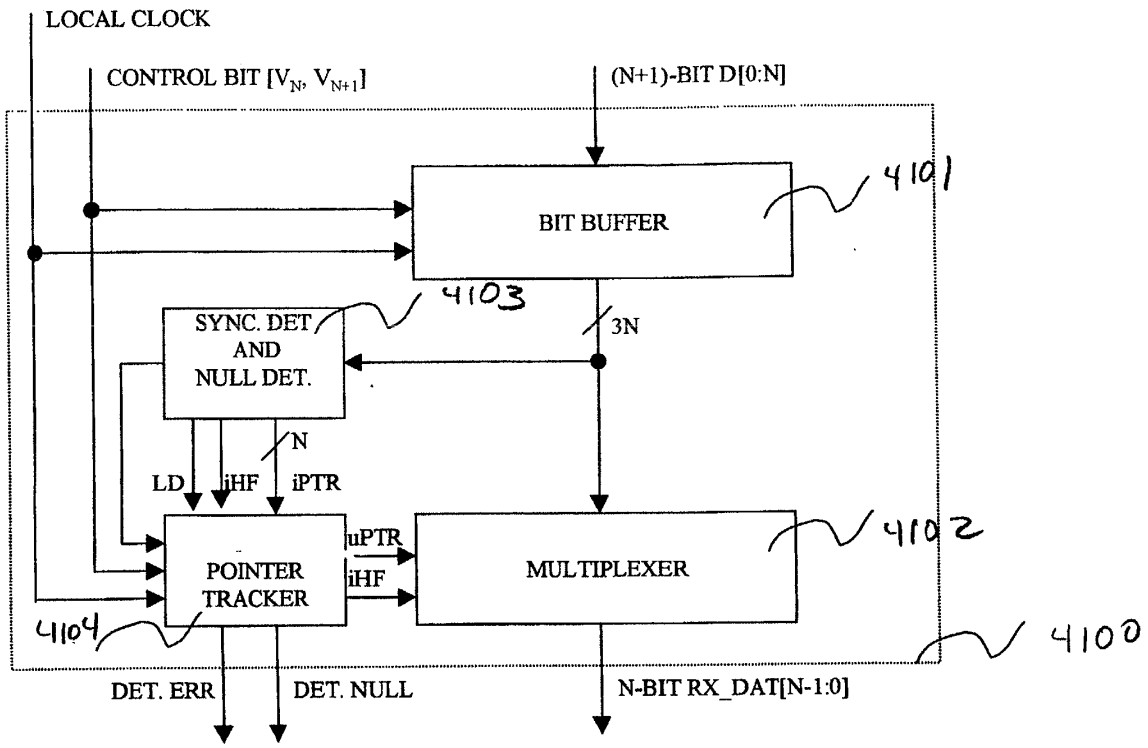


Fig 41

T0Z0TT' TT6SE00T

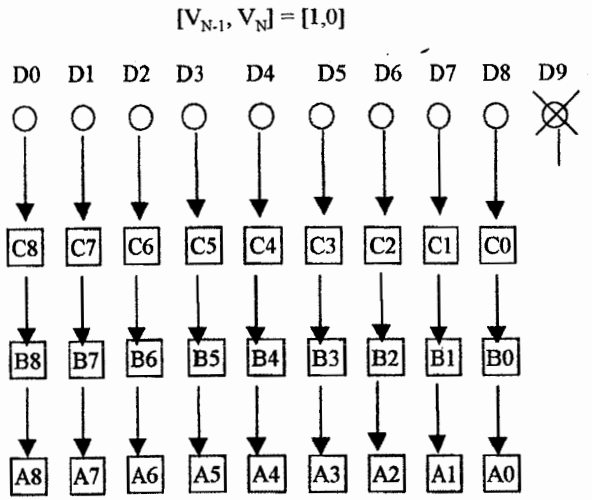


Fig 42A

TOZOTT' TT6SE00T

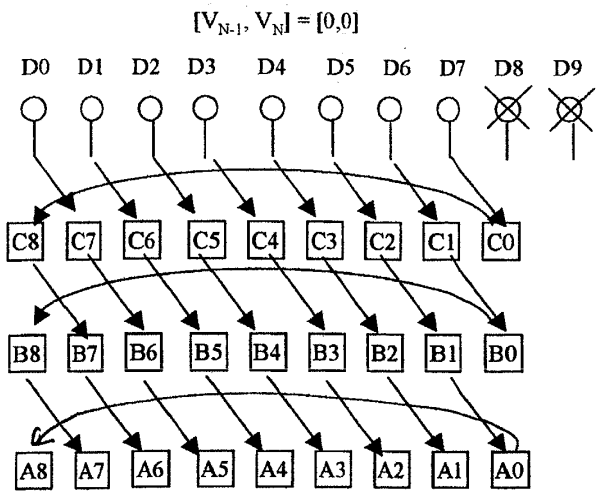


Fig 42B

T02OTT" TT6SE00T

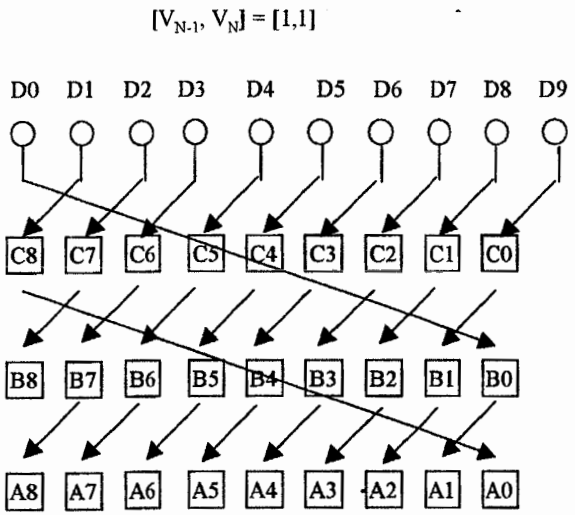


Fig 42C

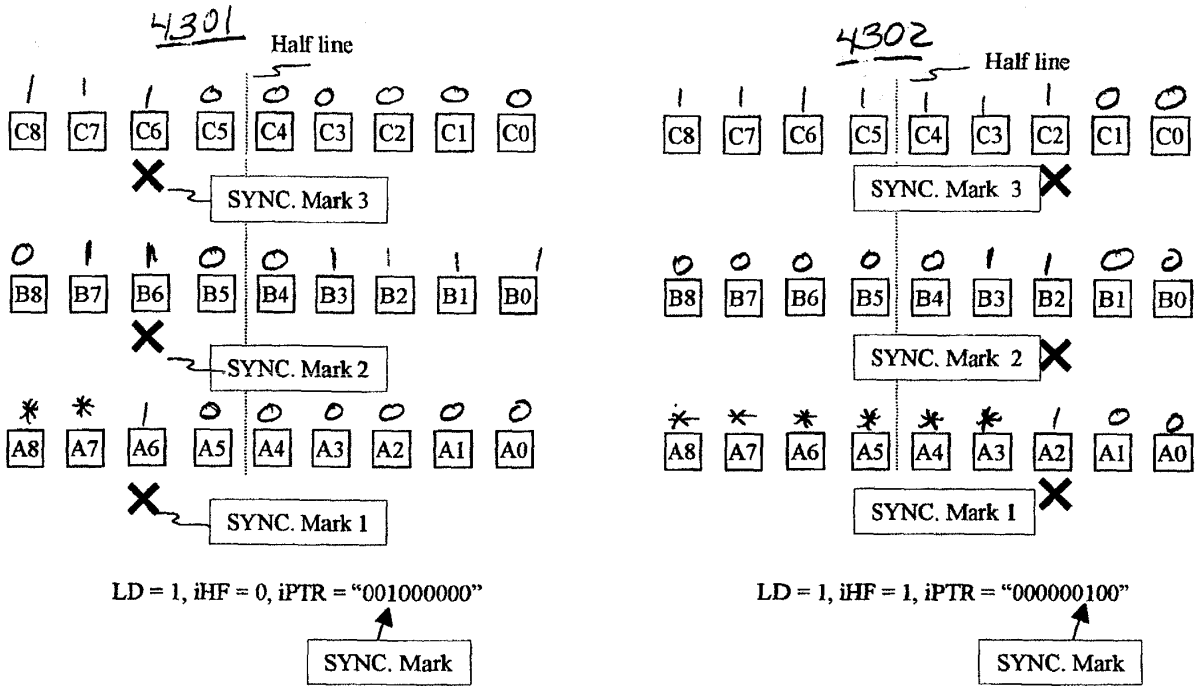


Fig. 43

IPR2024-001428
HP – EX1002, Page 479 of 487

TOZOFF TTGSEOOT

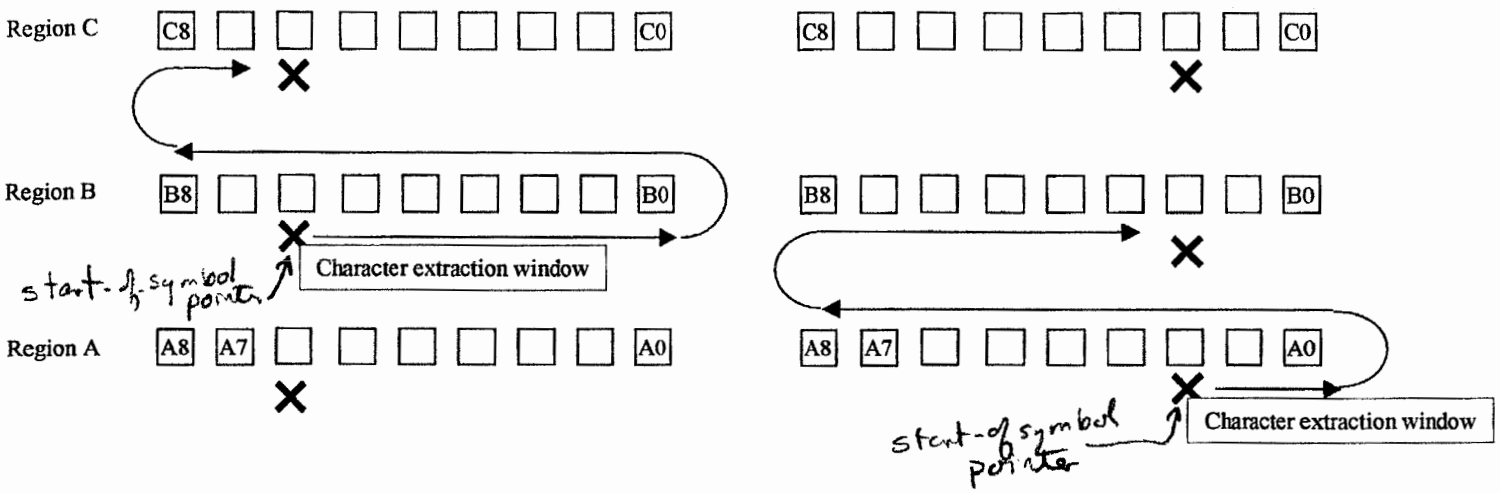


Fig 44

FOZOFF FT6SE00F

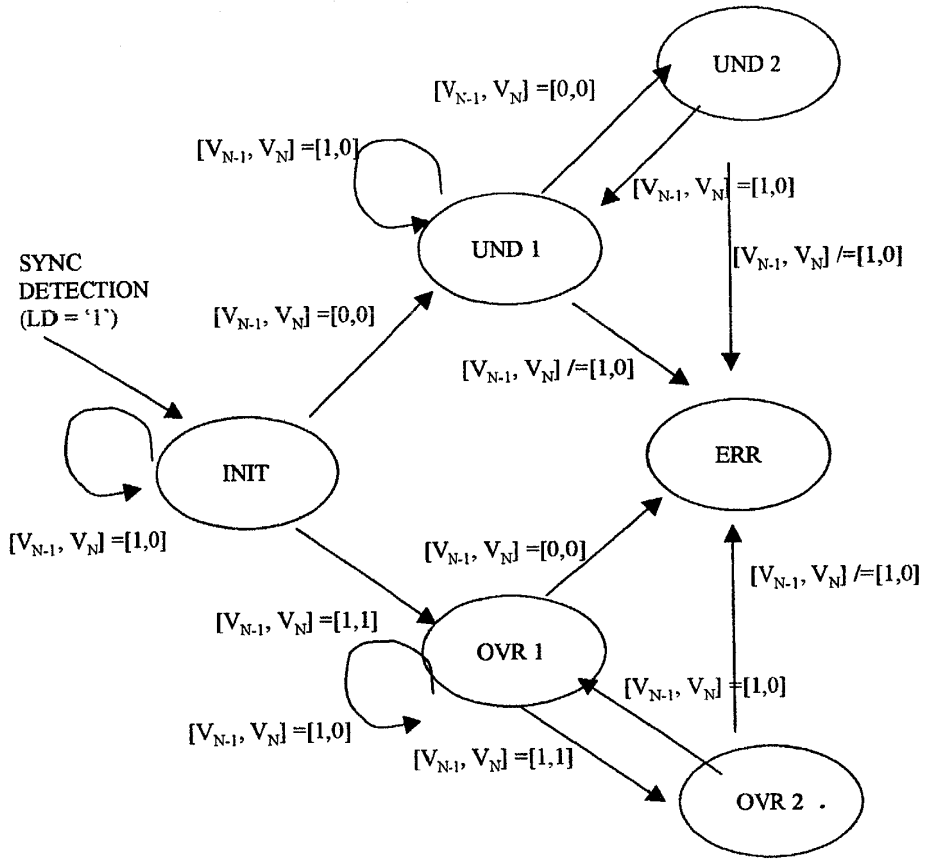


Fig 45

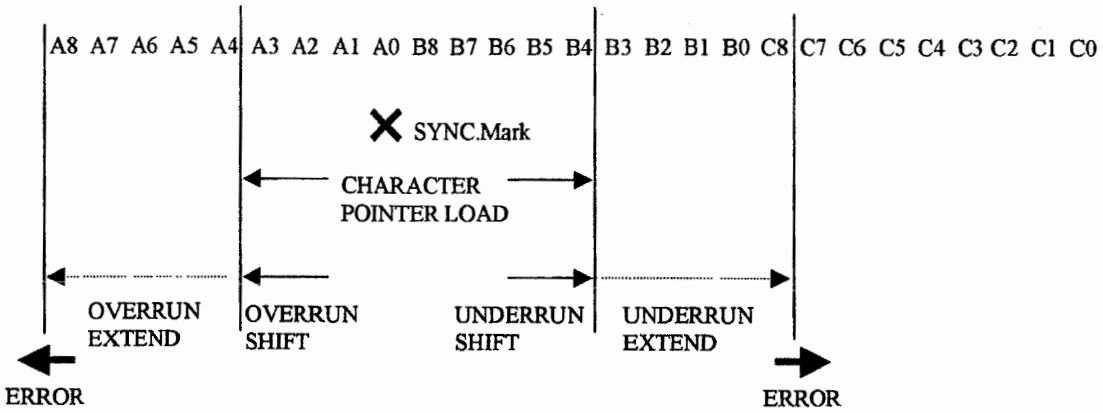

A8 A7 A6 A5 A4 A3 A2 A1 A0 B8 B7 B6 B5 B4 B3 B2 B1 B0 C8 C7 C6 C5 C4 C3 C2 C1 C0
SYNC.Mark
CHARACTER
POINTER LOAD
OVERRUN
EXTEND
OVERRUN
SHIFT
UNDERRUN
SHIFT
UNDERRUN
EXTEND
ERROR
ERROR

Fig 46

T0ZOTT' TT6SE00T

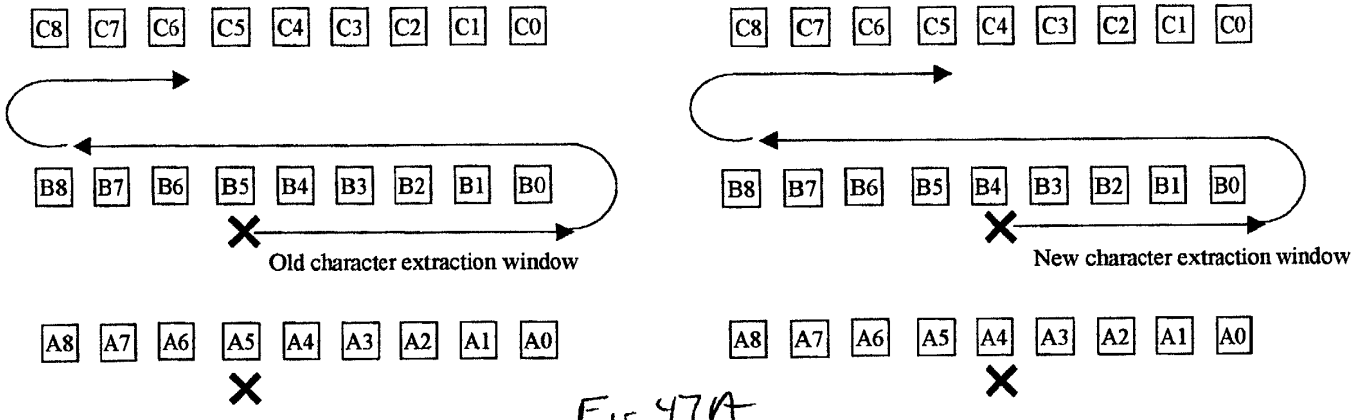


Fig 47A

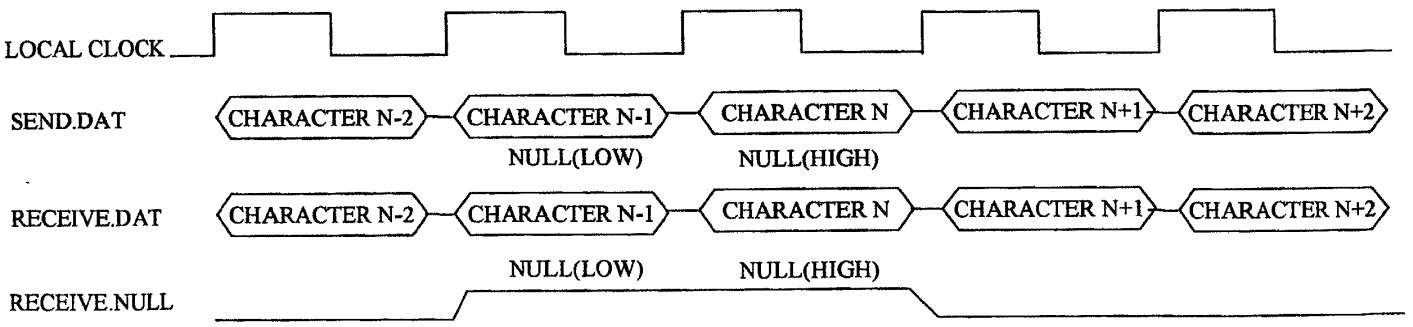


Fig 47B

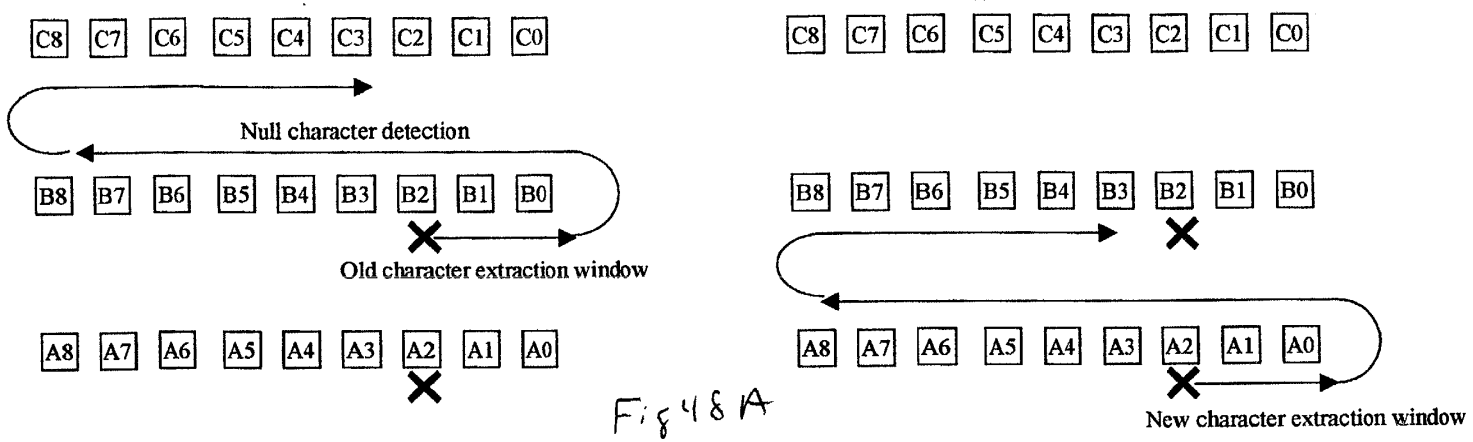


Fig 48A

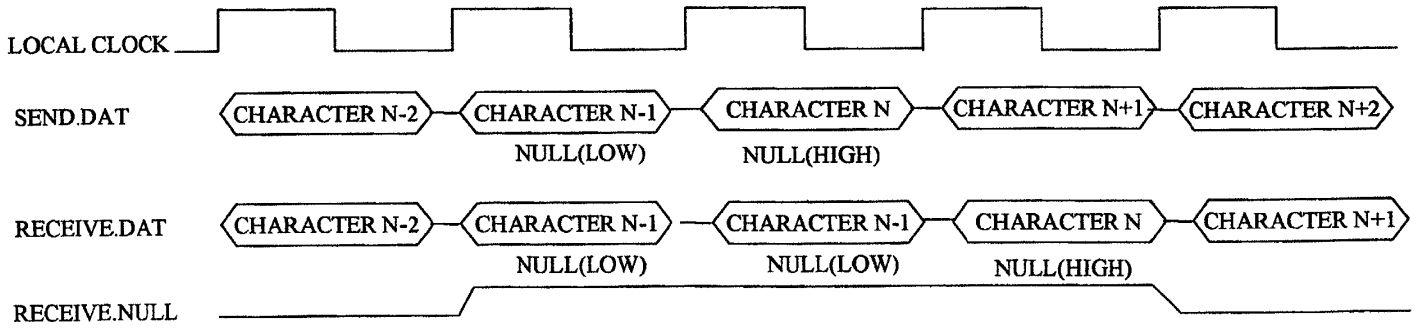


Fig 48B

TOZOTT" TT6SE00T

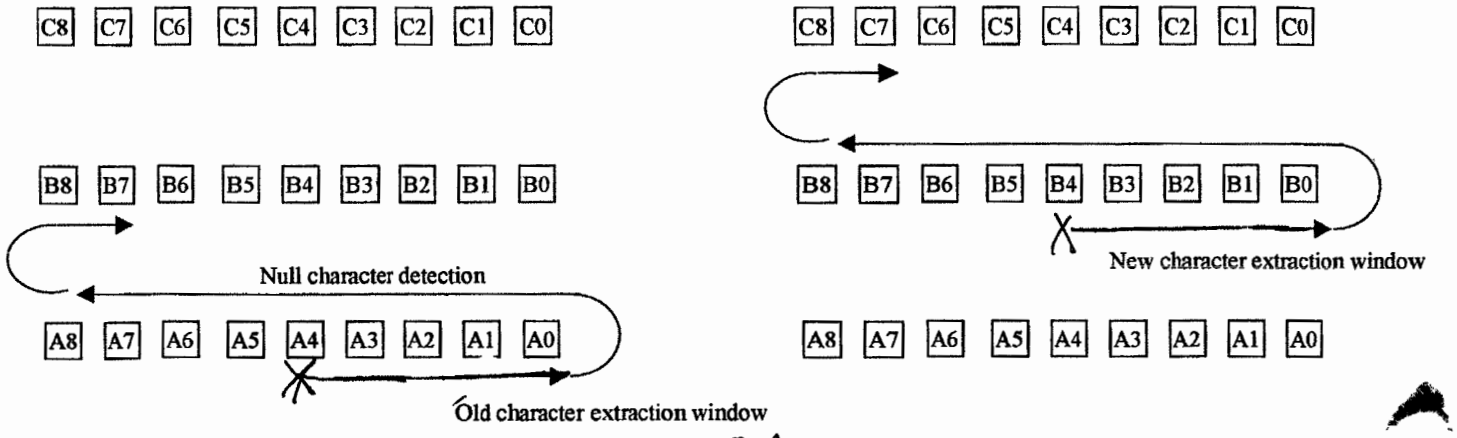


Fig. 49A

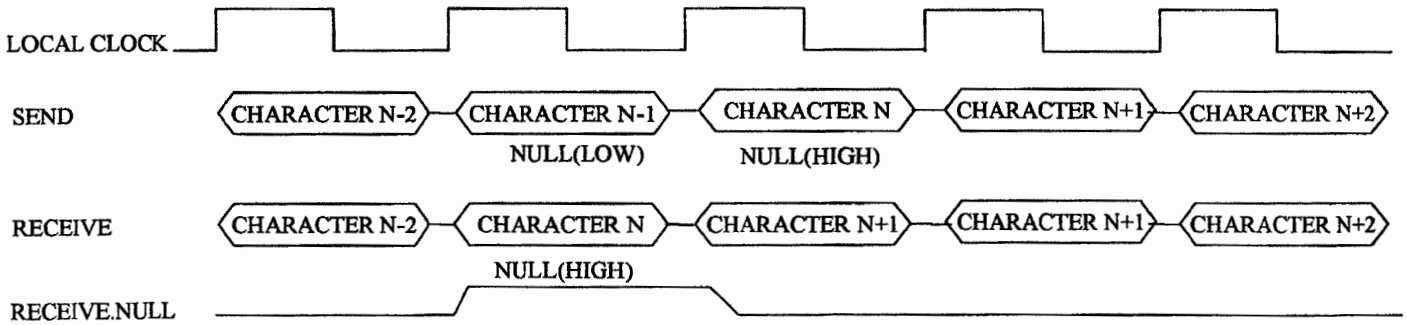


Fig 49B

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2001

Application or Docket Number

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	30	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	30 minus 20=	* 10
INDEPENDENT CLAIMS	4 minus 3 =	* 1
MULTIPLE DEPENDENT CLAIM PRESENT		☐

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐		OR	OTHER THAN SMALL ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	370.00	OR	BASIC FEE	740.00
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL		OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. 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 number found in the appropriate box in column 1.

CLAIMS ONLY

SERIAL NO.	FILING DATE
APPLICANT(S)	

CLAIMS

	AS FILED		AFTER 1st AMENDMENT		AFTER 2nd AMENDMENT			*		*		*	
	IND.	DEP.	IND.	DEP.	IND.	DEP.		IND.	DEP.	IND.	DEP.	IND.	DEP.
1	1						51						
2							52						
3							53						
4							54						
5							55						
6							56						
7							57						
8							58						
9	1						59						
10							60						
11							61						
12							62						
13							63						
14							64						
15							65						
16							66						
17	1						67						
18							68						
19							69						
20							70						
21							71						
22							72						
23							73						
24	1						74						
25							75						
26							76						
27							77						
28							78						
29							79						
30							80						
31							81						
32							82						
33							83						
34							84						
35							85						
36							86						
37							87						
38							88						
39							89						
40							90						
41							91						
42							92						
43							93						
44							94						
45							95						
46							96						
47							97						
48							98						
49							99						
50							100						
TOTAL IND.	4	⇓		⇓		⇓	TOTAL IND.		⇓		⇓		⇓
TOTAL DEP.	26	⇐		⇐		⇐	TOTAL DEP.		⇐		⇐		⇐
TOTAL CLAIMS	30						TOTAL CLAIMS						

*MAY BE USED FOR ADDITIONAL CLAIMS OR ADMENDMENTS

FORM PTO-2022 (1-98)

U.S.DEPARTMENT OF COMMERCE
Patent and Trademark Office

*U.S. GPO: 1998-443-593/89152