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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 October 2008 (23.10.2008)

PCT

(10) International Publication Number
WO 2008/127616 A1

(51) International Patent Classification:
G11C 29/10 (2006.01)

(21) International Application Number:
PCT/US2008/004637

(22) International Filing Date: 10 April 2008 (10.04.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/735,182 13 April 2007 (13.04.2007) US

(71) Applicant *(for all designated States except US)*: **ATMEL CORPORATION** [US/US]; 2325 Orchard Parkway, San Jose, California 95131 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **CORCELLI, Ciro** [IT/IT]; Sandgata 31, I-7012 Trondheim (IT).

(74) Agents: **STEFFEY, Charles E.** et al.; Schwegman, Lundberg & Woessner, P.A., P.O. Box 2938, Minneapolis, Minnesota 55402 (US).

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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL,

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR TESTING PAGE DECODER

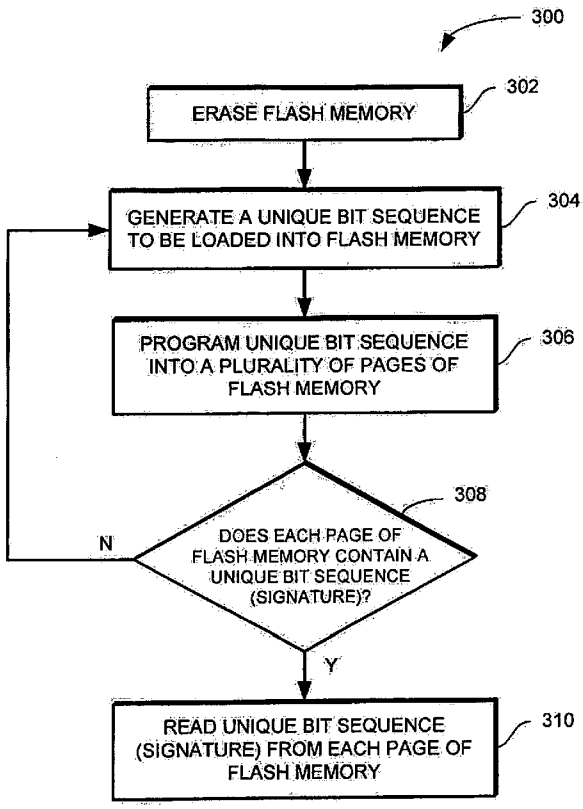

FIG. 3

(57) Abstract: A method (300) and apparatus for testing correct operation of a page decoder in a memory is provided. In one implementation, the method includes erasing the memory (302) to reset all memory cells associated with each of the N pages in the memory, and iteratively generating a unique bit sequence of M bits (304) and programming the unique bit sequence into a plurality of the N pages (306) at a given time until each of -the N pages contains a unique bit sequence relative to other pages in the memory (308). Responsive to each of the N pages having a unique bit sequence, the method further includes using the page decoder to read out each unique bit sequence associated with the N pages to verify correct operation of the page decoder (310).

WO 2008/127616 A1

NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

METHOD AND APPARATUS FOR TESTING PAGE DECODER

CLAIM OF PRIORTY

[0001] Benefit of priority is hereby claimed to U.S. Patent Application Serial Number 11/735,182, filed on April 13, 2007, which application is herein incorporated by reference.

FIELD

[0002] Embodiments disclosed in this document relate generally to digital circuits, and more particularly to address decoders.

BACKGROUND

[0003] Flash memory is a non-volatile computer memory that can be electrically erased and reprogrammed. FIG. 1 illustrates a conventional flash memory 100 organized into pages (e.g., pages 1-N). Each page (1-N) generally contains several words, and each word contains a pre-determined number of bits (e.g., 16 bits). Each word of a given page can be accessed with a pointer (or program counter) by means of an address decoder. As shown in FIG. 1, the address decoder is composed of a page decoder 102 and a word decoder 104. In particular, during a read from the flash memory 100, the page decoder 102 decodes the most significant bits of the program counter to determine which page to select, and the word decoder 104 decodes the least significant bits of the program counter to determine which word (within a selected page) is to be read. Each word has a group of bit lines (same size as the word size) that goes high when the corresponding word is selected. The word decoder 104 selects the correct bit lines that are used to multiplex the data onto an instructions bus (not shown). To write to the flash memory 100, page latches 106 are loaded with data. The least significant bits of the program counter are typically used to load the data to the correct portion of the page latches 106 in a similar fashion as in the word decoder (or the same word decoder can be used). Each page latch is connected to a corresponding bit line. During a read, the output from the page latches 106 is typically tri-stated to avoid drive contention.

[0004] Testing correct operation of a flash memory typically requires testing the functionality of both the word decoder and the page decoder. Testing the functionality of a word decoder is usually not an issue because it is generally sufficient to program - i.e., write values into - only one page and then read back the values programmed into the page. Testing the functionality of a page decoder, however, typically requires programming *all* pages within a flash memory. That is, a unique signature is written into each page of the flash memory, which unique signature is subsequently read from each page of the flash memory. This guarantees that the page decoder points to correct locations within the flash memory. Yet, programming each page of a flash memory is a very time-consuming process, and test time is an important factor in production flow. Another method for testing the functionality of a page decoder is to insert observation points (e.g., a scan chain) at the output of the page decoder. If the page decoder works properly, then only one line (used to access a page) is active at a time. Such a method, however, does not guarantee that the connection between the output lines and the pages is free of any faults.

[0005] The details of one or more implementation embodiments are set forth in the accompanying drawings and the description below. Other features and advantages will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 is a block diagram of a conventional flash memory;

[0007] FIG. 2 is a block diagram of a flash memory in accordance with some embodiments;

[0008] FIG. 3 is a method for testing the functionality of a page decoder of a flash memory in accordance with some embodiments;

[0009] FIG. 4 is a schematic diagram of a page decoder in accordance with some embodiments; and

[0010] FIG. 5 is a block diagram of a memory including a page decoder in accordance with some embodiments.

DETAILED DESCRIPTION

[0011] Embodiments of the present invention relate generally to digital circuits, and more particularly to address decoders. The following description of the various embodiments is presented to enable one of ordinary skill in the art to make and use the invention and is provided in the context of a patent application and its requirements. The present invention is not intended to be limited to the embodiments shown but is to be accorded the widest scope consistent with the principles and features described and claimed herein. Like reference symbols in the various drawings indicate like elements.

[0012] FIG. 2 illustrates a flash memory 200 in accordance with one implementation. In one implementation, the flash memory 200 is organized into pages (e.g., pages 1-N), and includes an address decoder (comprised of a page decoder 202 and a word decoder 204) for accessing each word of a given page within the flash memory 200. In one implementation, during a read from the flash memory 200, the page decoder 202 decodes the most significant bits of a program counter (not shown) to determine which page to select, and the word decoder 204 decodes the least significant bits of the program counter to determine which word (within a selected page) is to be read. In one implementation, to write to the flash memory 200, page latches 206 are loaded with data (or bit sequences) that are to be written to corresponding bit locations (or memory cells) within the flash memory 200 - i.e., each page latch is connected to a corresponding bit line. The least significant bits of the program counter can be used to load the data to the correct portion of the page latches 206. In one implementation, each memory cell of the flash memory 200 is written to by setting a logic level of the memory cell to zero. Accordingly, in this implementation, erasing a memory cell requires setting the logic level of the memory cell to one. In one implementation, during a programming operation, an entire page is written to at once and the resulting logical value of the memory cells associated with the page depends upon what was previously loaded in the page latches 206. That is, a logic one in a page latch will leave the content (or logic level) of a corresponding memory cell in the flash memory untouched, whereas a logic zero in the page latch will write the corresponding memory cell in the flash memory (the logic level of the memory cell is set to zero).

[0013] As discussed above, testing correct operation of a flash memory typically requires testing the functionality of both the word decoder and the page decoder. Testing the functionality of a word decoder is usually not an issue because it is generally sufficient to program only one page and then read back into the page, the values programmed. Testing the functionality of a page decoder, however, in some embodiments typically requires programming all pages (within a flash memory) with a unique signature that is subsequently read from each page of the flash memory. Unlike a conventional page decoder that requires N operations to program N pages of a flash memory, the page decoder 202 is operable to program a plurality of pages at a given time to reduce the amount of time required to program all the pages of a flash memory with a unique signature. Reducing the amount of time required to program pages of a flash memory reduces the overall time and cost required to test functionality of a flash memory. In one implementation, the number of operations required to program each of N pages of a flash memory (with a unique signature) is reduced to $log_2 N$, as described in greater detail below.

[0014] FIG. 3 illustrates a method 300 for testing the functionality of a page decoder (e.g., page decoder 202) of a flash memory in accordance with one implementation. A flash memory (e.g., flash memory 200) is erased (step 302). In one implementation, erasing a flash memory comprises setting a logic level of each memory cell (of the flash memory) to one. After the flash memory is erased, a unique bit sequence (or signature) to be loaded into the flash memory is generated (e.g., by logic) (step 304). In one implementation, the number of bits in the unique bit sequence is equal to the number of bits in a given page of the flash memory. For example, if a page contains 32 bits, then the unique bit sequence contains 32 bits. A bit sequence having such a length permits an entire page to be programmed at once. The unique bit sequence is programmed into a plurality of pages of the flash memory at one time (step 306). In one implementation, $N/2$ pages are programmed with the unique bit sequence, where N is the total number of pages within the flash memory.

[0015] A determination is made (e.g., by logic) whether each page of the flash memory contains a unique bit sequence (or unique signature) relative to other pages in the flash memory (step 308). If each page of the flash memory has not yet been programmed to have a unique signature, then the method 300

returns to step 304 to generate another unique bit sequence to be loaded into the flash memory. In one implementation, during each iteration, *N/2* pages of the flash memory are programmed. Thus, in this implementation, only log_2N iterations (or operations) are required to program each page of the flash memory with a unique signature. If, at step 306, a determination is made that each page of the flash memory has been programmed with a unique signature, then each unique signature is read from a corresponding page of the flash memory (step 310).

[0016] An example will now be provided. Table 1 below shows contents of a flash memory after the flash memory has been erased - i.e., each memory cell has a logic level of one. In the example, the flash memory has only 8 pages, and each page contains 5 bits.

Page 1	1	1	1	1	1
Page 2	2	2	2	2	2
Page 3	3	3	3	3	3
Page 4	4	4	4	4	4
Page 5	5	5	5	5	5
Page 6	6	6	6	6	6
Page 7	7	7	7	7	7
Page 8	8	8	8	8	8

Table 1

[0017] Table 2 below shows the contents of the flash memory after a first plurality of pages (e.g., N/2 pages) have been programmed with a unique signature. The unique signature programmed into the flash memory is [01111], and this signature is first loaded into the page latches prior to being written into the flash memory. The "H" means that the page lines corresponding to pages 1-4 are set high. Therefore, because only the first bit in the page latches is set to zero while all the other bits are set at one, only the first bit in each of the accessed pages (i.e., pages 1-4) are written while the other bits are left untouched.

	Page latches	0	1	1	1	1
H	Page 1	0	1	1	1	1
H	Page 2	0	1	1	1	1
H	Page 3	0	1	1	1	1
H	Page 4	0	1	1	1	1
	Page 5	1	1	1	1	1

	Page 6	1	1	1	1	1
	Page 7	1	1	1	1	1
	Page 8	1	1	1	1	1

Table 2

[0018] In a second programming step, *N/2* pages of the flash memory are access again at once. However, as shown in Table 3 below, the first two pages of the first half of the flash memory and the first two pages of the second half of the flash memory are accessed. And in this programming step, only the second bit in the page latches is set to zero. Table 3 shows the contents of the flash memory after the second programming step.

	Page latches	1	0	1	1	1
H	Page 1	0	0	1	1	1
H	Page 2	0	0	1	1	1
	Page 3	0	1	1	1	1
	Page 4	0	1	1	1	1
H	Page 5	1	0	1	1	1
H	Page 6	1	0	1	1	1
	Page 7	1	1	1	1	1
	Page 8	1	1	1	1	1

Table 3

[0019] In a third (and final) programming step, *N/2* pages are again accessed at once, however, this time in an alternating fashion. In this programming step, only the third bit in the page latches is set to a logic level of zero. Table 4 shows the contents of the flash memory after the third programming step.

	Page latches	1	1	0	1	1
H	Page 1	0	0	0	1	1
	Page 2	0	0	1	1	1
H	Page 3	0	1	0	1	1
	Page 4	0	1	1	1	1
H	Page 5	1	0	0	1	1
	Page 6	1	0	1	1	1
H	Page 7	1	1	0	1	1
	Page 8	1	1	1	1	1

[0020] As shown in Table 4 above, the programming test sequence leaves a unique signature in each page with only log_2N (or three) programming operations. These signatures can then be used to test the functionality of a page decoder through a read operation. As shown in the example above, page 1 is programmed with a given unique bit sequence during each programming step. In contrast, page 8 remains unchanged throughout all the programming steps.

[0021] In general, to apply such a programming technique to a flash memory having N pages, one needs the highest log_2N bits of the program counter. Each of these bits of the program counter can be combined with a test mode signal (also of size log_2N) to access $N/2$ pages of the flash memory at a time. FIG. 4 illustrates one implementation of a page decoder 400 operable to program N/2 pages of a flash memory at a time. The page decoder 400 includes inverters (INVO-INV2), OR gates (ORO-OR2), and AND gates (ANDO-AND11). To access $N/2$ pages at a given time as set forth in the example above, the following programming constraints can be implemented:

Programming step 1:

PC[2:0] = 3'000

TM[2:0] = 3'b101

Programming step 2:

PC[2:0] = 3'b000

TM[2:0] = 3'b110

Programming step 3:

PC[2:0] = 3'b000

TM[2:0] = 3'b110

CONCLUSION

[0022] This specification describes method embodiments for testing correct operation of a page decoder in a memory comprised of N pages, in which each of the N pages has M memory cells. The method comprises, inter alia, erasing the memory to reset all memory cells associated with each of the N pages

in the memory; iteratively generating a unique bit sequence of *M* bits and programming the unique bit sequence into a plurality of the N pages at a given time until each of the *N* pages contains a unique bit sequence relative to other pages in the memory; and responsive to each of the *N* pages having a unique bit sequence, using the page decoder to read out each unique bit sequence associated with the *N* pages to verify correct operation of the page decoder.

[0023] Particular implementation embodiments may include one or more of the following features. Erasing the memory can include setting all memory cells associated with each of the *N* pages to a logic level of one. Programming the unique bit sequence into a plurality of the *N* pages can include setting one or more of the *M* memory cells of a given page to a logic level of zero. Programming the unique bit sequence into a plurality of the *N* pages at a given time can include programming the unique bit sequence into *N/2* pages of the memory at a given time. In one aspect, only log_2N iterations are required until each of the *N* pages contains a unique bit sequence relative to other pages in the memory. At least one of the plurality of *N* pages can be programmed with a given unique bit sequence during each iteration. The memory can be a flash memory.

[0024] This specification also describes embodiments of a memory including *N* pages, in which each of the *N* pages has *M* memory cells. The memory further includes a page latch to iteratively load a unique bit sequence of *M* bits into a plurality of the *N* pages at a given time until each of the *N* pages contains a unique bit sequence relative to other pages in the memory, and a page decoder to read out each unique bit sequence associated with the *N* pages when each of the *N* pages has a unique bit sequence relative to other pages in the memory.

[0025] In some embodiments, a method and system for fully testing the functionality of a page decoder is provided that does not require use of observation points - e.g., a scan chain. In some embodiments, the techniques described herein reduce the overall test time of a flash memory by reducing the number of programming operations required to place a unique signature into each page of the flash memory.

[0026] Various implementation embodiments of a page decoder for a flash memory have been described. Nevertheless, various modifications may be

made to the implementations. For example, though example embodiments of page decoder described above are generally described with respect to flash memories, a page decoder in accordance with the present invention can be implemented within other types of memories, as shown in FIG. 5.

[0027] FIG. 5 illustrates a memory 500 including a page decoder 502 and a word decoder 504. The page decoder 502 is operable to program a plurality of pages (or memory blocks) (not shown) within the memory at a given time to reduce the amount of time required to program all the pages (or memory blocks) with a unique signature. The memory can be, for example, a RAM (random access memory), a ROM (read-only memory), a hybrid memory, in addition to a multi-level cell device (that can store more than 1 bit per memory cell). In addition, steps of the method discussed above can be performed in a different order and still achieve desirable results. Accordingly, many modifications may be made without departing from the scope of the following claims.

CLAIMS

What is claimed is:

1. A method for testing correct operation of a page decoder in a memory comprised of N pages, each of the N pages having M memory cells, the method comprising:

erasing the memory to reset all memory cells associated with each of the N pages in the memory;

iteratively generating a unique bit sequence of M bits and programming the unique bit sequence into a plurality of the N pages at a given time until each of the N pages contains a unique bit sequence relative to other pages in the memory; and

responsive to each of the N pages having a unique bit sequence, using the page decoder to read out each unique bit sequence associated with the N pages to verify correct operation of the page decoder.

2. The method of claim 1, wherein:

erasing the memory comprises setting all memory cells associated with each of the N pages to a logic level of one; and

programming the unique bit sequence into a plurality of the N pages comprises setting one or more of the M memory cells of a given page to a logic level of zero.

3. The method of claim 1, wherein programming the unique bit sequence into a plurality of the N pages at a given time comprises programming the unique bit sequence into $N/2$ pages of the memory at a given time.

4. The method of claim 3, wherein only $log_2 N$ iterations are required until each of the N pages contains a unique bit sequence relative to other pages in the memory.

5. The method of claim 4, wherein at least one of the plurality of N pages is programmed with a given unique bit sequence during each iteration.

6. The method of claim 5, wherein the memory is a flash memory.

7. A memory comprised of N pages, each of the N pages having M memory cells, the memory comprising:

means for erasing the memory to reset all memory cells associated with each of the N pages in the memory;

means for iteratively generating a unique bit sequence of M bits and programming the unique bit sequence into a plurality of the N pages at a given time until each of the N pages contains a unique bit sequence relative to other pages in the memory; and

responsive to each of the N pages having a unique bit sequence, means for reading out each unique bit sequence associated with the N pages to verify correct operation of the memory.

8. The memory of claim 7, wherein:

the means for erasing the memory comprises means for setting all memory cells associated with each of the N pages to a logic level of one; and

the means for iteratively generating a unique bit sequence of M bits and programming the unique bit sequence into a plurality of the N pages comprises means for setting one or more of the M memory cells of a given page to a logic level of zero.

9. The memory of claim 7, wherein the means for iteratively generating a unique bit sequence of M bits and programming the unique bit sequence into a plurality of the N pages comprises means for programming the unique bit sequence into $N/2$ pages of the memory at a given time.

10. The memory of claim 9, wherein the means for iteratively generating a unique bit sequence of M bits and programming the unique bit sequence into a plurality of the N pages requires only $log_2 N$ iterations until each of the N pages contains a unique bit sequence relative to other pages in the memory.

11. The memory of claim 10, wherein at least one of the plurality of *N* pages is programmed with a given unique bit sequence during each iteration.

12. The memory of claim 11, wherein the memory is a flash memory.

13. A memory comprising:

N pages, each of the *N* pages having *M* memory cells;

a page latch to iteratively load a unique bit sequence of *M* bits into a plurality of the *N* pages at a given time until each of the *N* pages contains a unique bit sequence relative to other pages in the memory; and

a page decoder to read out each unique bit sequence associated with the *N* pages when each of the *N* pages has a unique bit sequence relative to other pages in the memory.

14. The memory of claim 13, wherein the page latch is operable to iteratively load a given unique bit sequence into *N/2* pages of the memory at a given time.

15. The memory of claim 14, wherein only $log_2 N$ iterations are required until each of the *N* pages contains a unique bit sequence relative to other pages in the memory.

16. The memory of claim 15, wherein the memory is a flash memory.

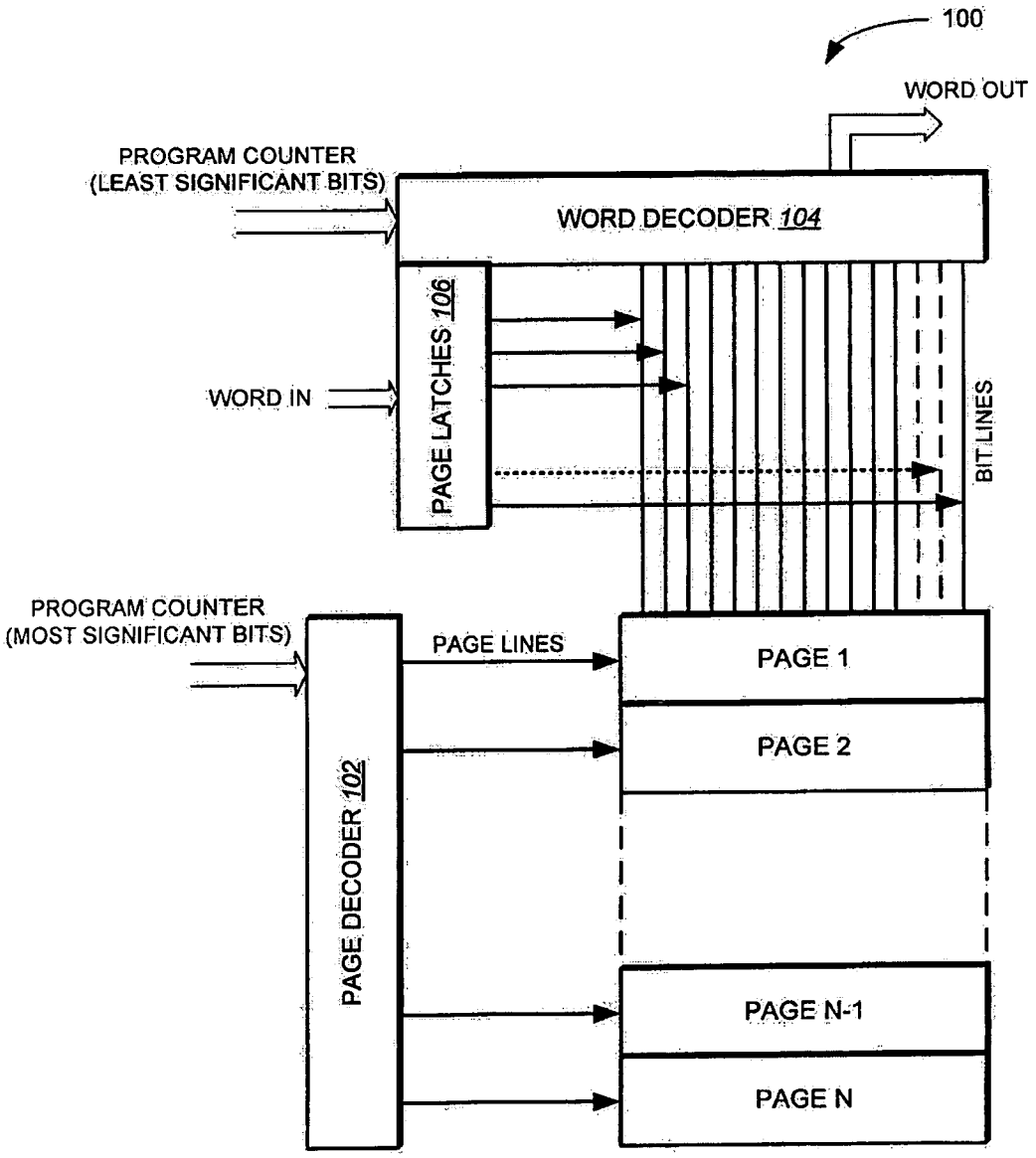

FIG. 1 (PRIOR ART)

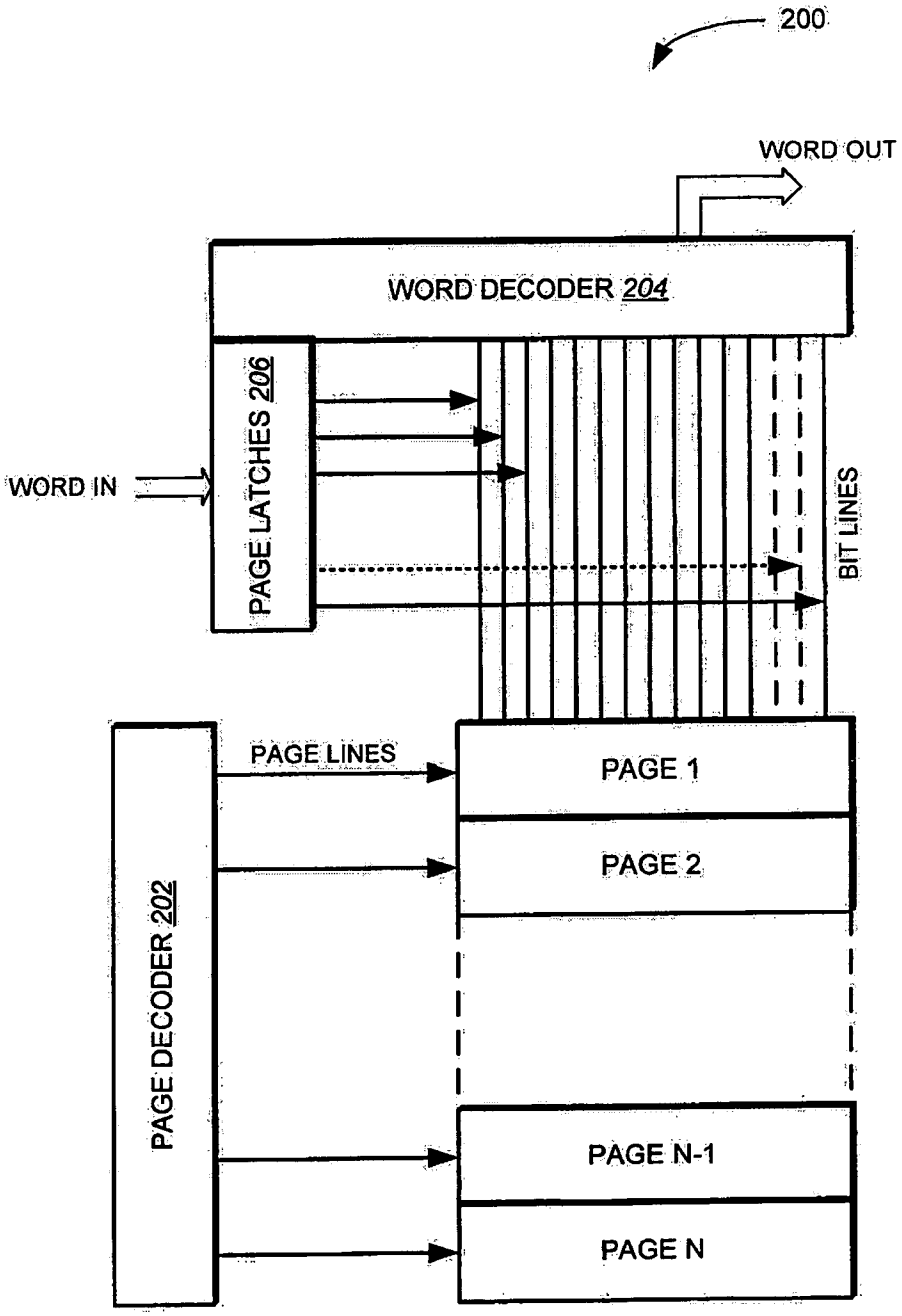
200
WORD OUT
WORD DECODER 204
PAGE LATCHES 206
WORD IN
BIT LINES
PAGE LINES
PAGE 1
PAGE 2
PAGE DECODER 202
PAGE N-1
PAGE N

FIG. 2

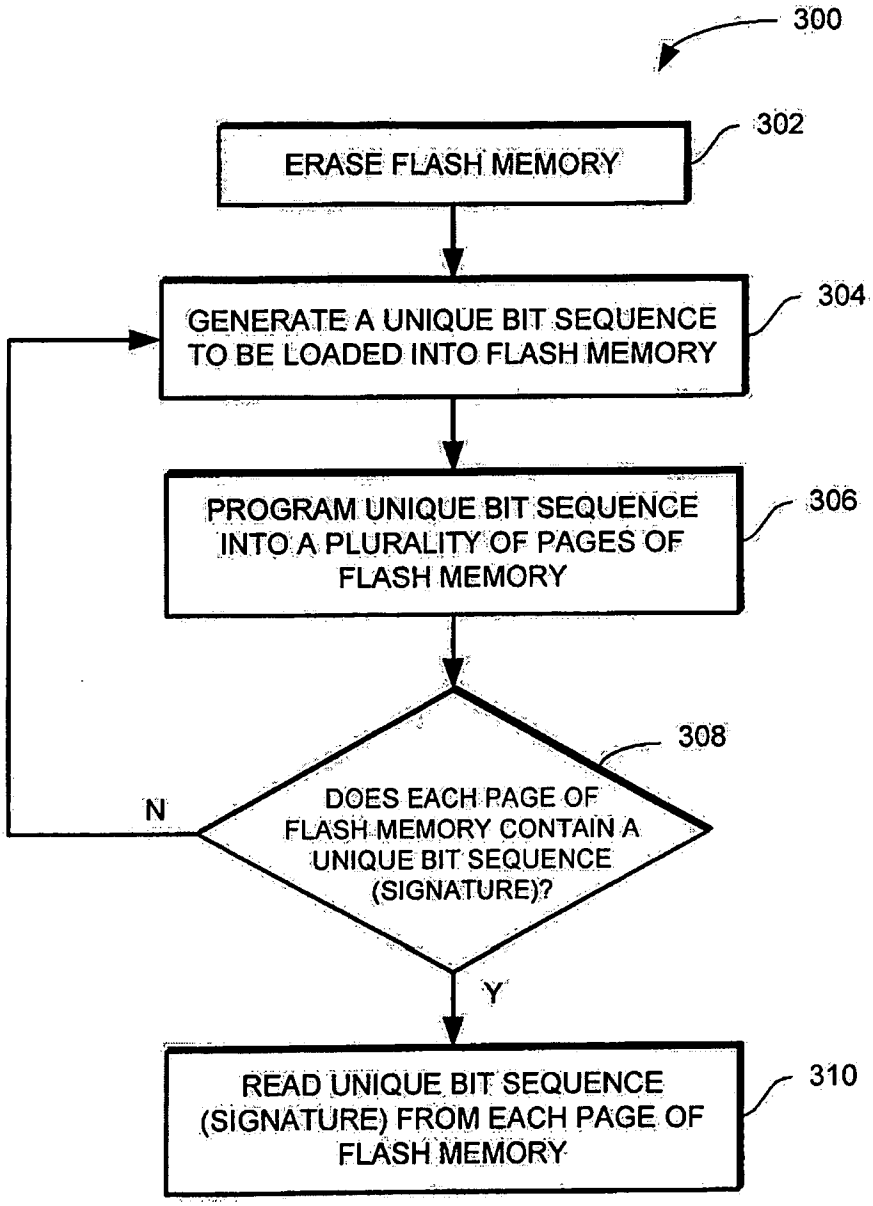
300
ERASE FLASH MEMORY
302
GENERATE A UNIQUE BIT SEQUENCE TO BE LOADED INTO FLASH MEMORY
304
PROGRAM UNIQUE BIT SEQUENCE INTO A PLURALITY OF PAGES OF FLASH MEMORY
306
DOES EACH PAGE OF FLASH MEMORY CONTAIN A UNIQUE BIT SEQUENCE (SIGNATURE)?
308
N
Y
READ UNIQUE BIT SEQUENCE (SIGNATURE) FROM EACH PAGE OF FLASH MEMORY
310

FIG. 3

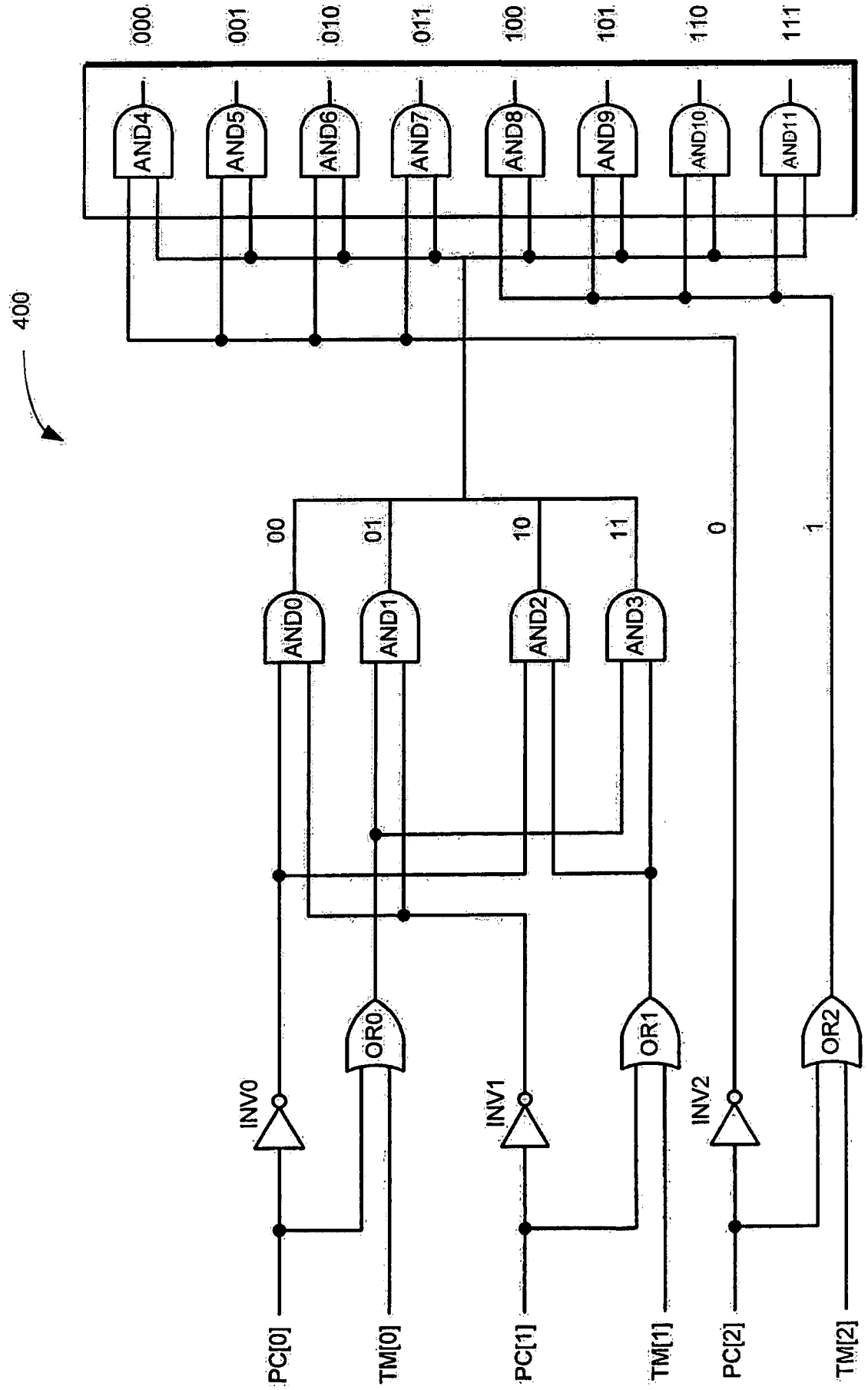
400
000
001
010
011
100
101
110
111
AND4
AND5
AND6
AND7
AND8
AND9
AND10
AND11
00
01
10
11
0
1
AND0
AND1
AND2
AND3
OR0
OR1
OR2
INV0
INV1
INV2
PC[0]
TM[0]
PC[1]
TM[1]
PC[2]
TM[2]

FIG. 4

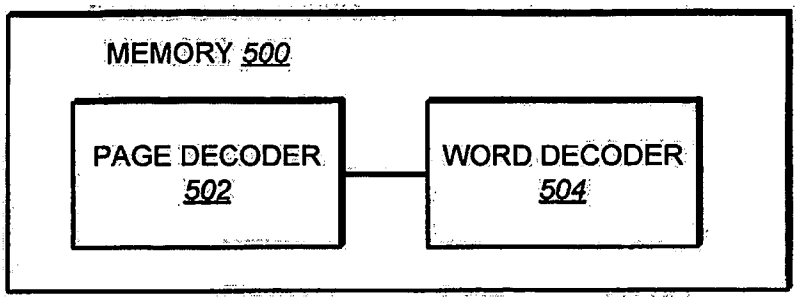

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2008/004637

A. CLASSIFICATION OF SUBJECT MATTER
INV. G11C29/10

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)
G11C

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, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 6 212 112 B1 (NAURA DAVID [FR] ET AL) 3 April 2001 (2001-04-03) column 3, line 11 - column 6, line 22; figures 4-6	1,2,7,8
Y	column 2, line 20 - column 2, line 58	3,5,6,9, 11-14,16
Y	DATABASE INSPEC [Online] THE INSTITUTION OF ELECTRICAL ENGINEERS, STEVENAGE, GB; March 1990 (1990-03), VAN DER GOOR A J ET AL: "An overview of deterministic functional RAM chip testing" XP002488797 Database accession no. 3889555 the whole document -/--	3,5,9, 11,13,14

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

[X] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 July 2008	30/07/2008

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	Balaguer López, J

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	& Computing Surveys USA, vol. 22, no. 1, March 1990 (1990-03), pages 5-33, ISSN: 0360-0300	
Y	US 2004/015756 A1 (CHIU SAU-KWO [TW] ET AL) 22 January 2004 (2004-01-22) paragraph [0060] - paragraph [0074]; figures 1-3	6,12,16
A	US 5 954 831 A (CHANG EDWARD C M [US]) 21 September 1999 (1999-09-21) column 1, line 11 - column 2, line 63; figures 3-6 column 3, line 60 - column 6, line 18; figures 8-12	1-16
A	US 4 672 610 A (SALICK JOHN E [US]) 9 June 1987 (1987-06-09) column 4, line 47 - column 5, line 13; figures 2A,2B	1-15

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2008/004637

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6212112 B1	03-04-2001	FR 2787233 A1	16-06-2000
US 2004015756 A1	22-01-2004	TW 567500 B	21-12-2003
US 5954831 A	21-09-1999	CN 1218961 A JP 11306794 A	09-06-1999 05-11-1999
US 4672610 A	09-06-1987	NONE	