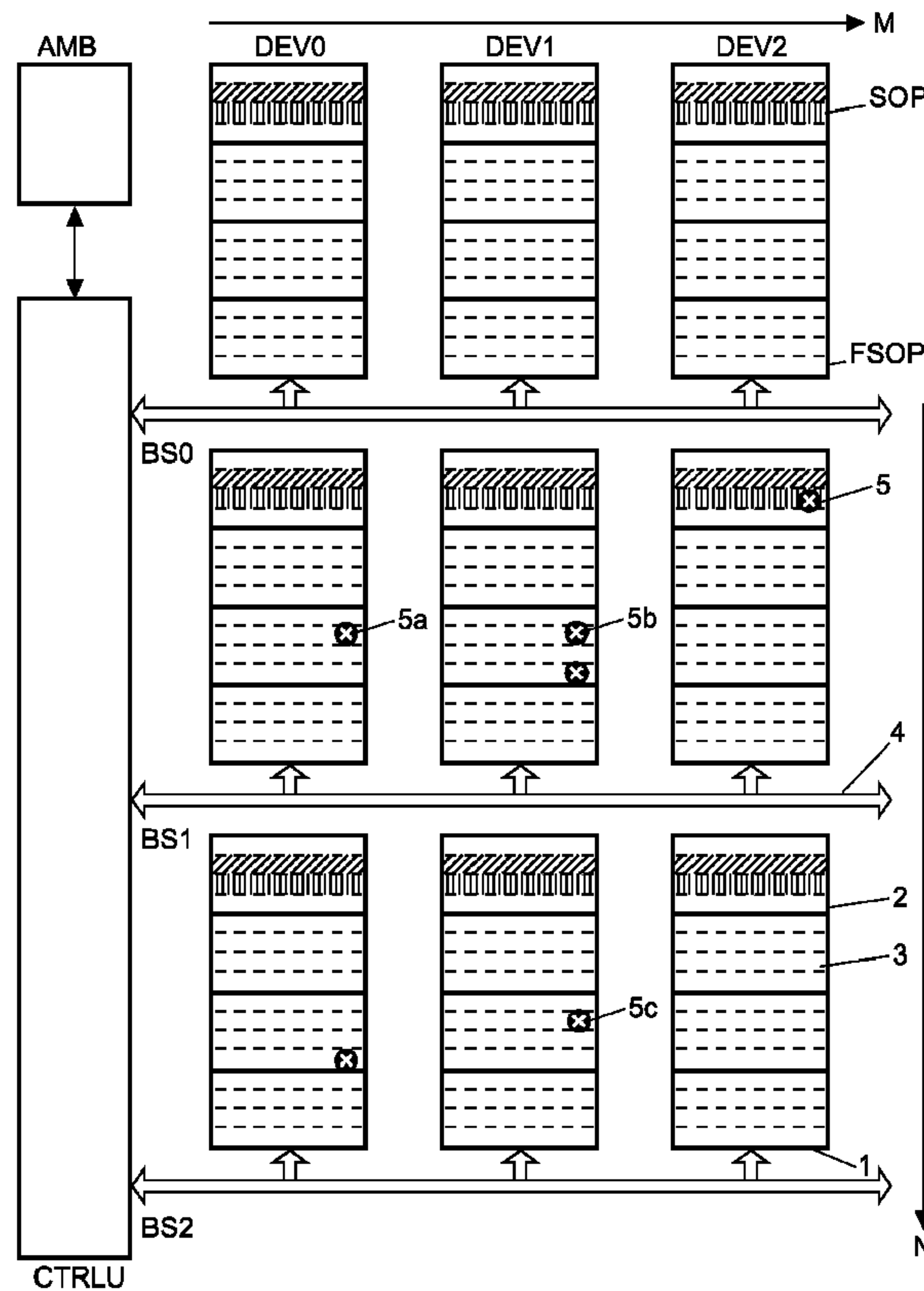




(86) Date de dépôt PCT/PCT Filing Date: 2006/12/04
 (87) Date publication PCT/PCT Publication Date: 2007/07/19
 (45) Date de délivrance/Issue Date: 2014/07/08
 (85) Entrée phase nationale/National Entry: 2008/07/03
 (86) N° demande PCT/PCT Application No.: EP 2006/069265
 (87) N° publication PCT/PCT Publication No.: 2007/080031
 (30) Priorité/Priority: 2006/01/16 (EP06090010.7)

(51) Cl.Int./Int.Cl. *G11C 16/10* (2006.01),
G06F 12/12 (2006.01), *G11C 29/00* (2006.01)
 (72) Inventeurs/Inventors:
BRUNE, THOMAS, DE;
WITTENBURG, JENS PETER, DE
 (73) Propriétaire/Owner:
THOMSON LICENSING, FR
 (74) Agent: FETHERSTONHAUGH & CO.

(54) Titre : PROCEDE ET APPAREIL DESTINES A ENREGISTRER DES DONNEES D'ENTREE A GRANDE VITESSE
DANS UNE MATRICE DE DISPOSITIFS DE MEMOIRE
 (54) Title: METHOD AND APPARATUS FOR RECORDING HIGH-SPEED INPUT DATA INTO A MATRIX OF MEMORY
DEVICES



(57) Abrégé/Abstract:

For recording or replaying in real-time digital high bandwidth video signals, e.g. HDTV, HD progressive or HD film capture signals, very fast memories are required. For storage of streaming HD video data NAND FLASH memory based systems could be used.



(57) **Abrégé(suite)/Abstract(continued):**

Flash memory devices are physically accessed in a page oriented mode. According to the invention, the input data are written in a multiplexed fashion into a matrix of multiple flash devices. A list processing is performed that is as simple and fast as possible, and defect pages of flash blocks of single flash devices are addressed within the matrix architecture. When writing in a sequential manner, the data content for the current flash device page of all flash devices of the matrix is copied to a corresponding storage area in an additional memory buffer. After the current series of pages has been written without error into the flash devices, the corresponding storage area in an additional memory buffer is enabled for overwriting with following page data. In case an error occurred in the current page in one or more flash devices, the content of these current pages is kept in the additional memory buffer.

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

(19) World Intellectual Property Organization
International Bureau(43) International Publication Date
19 July 2007 (19.07.2007)

PCT

(10) International Publication Number
WO 2007/080031 A1

(51) International Patent Classification:

G11C 16/10 (2006.01) *G06F 12/12* (2006.01)
G11C 29/00 (2006.01)

(21) International Application Number:

PCT/EP2006/069265

(22) International Filing Date:

4 December 2006 (04.12.2006)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

06090010.7 16 January 2006 (16.01.2006) EP

(71) Applicant (for all designated States except US): **THOMSON LICENSING** [FR/FR]; 46 Quai A. le Gallo, F-92100 Boulogne-Billancourt (FR).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **BRUNE, Thomas** [DE/DE]; Konkordiastr. 15, 30449 Hannover (DE). **WITTENBURG, Jens, Peter** [DE/DE]; Arnikaweg 33, 30916 Isernhagen (DE).

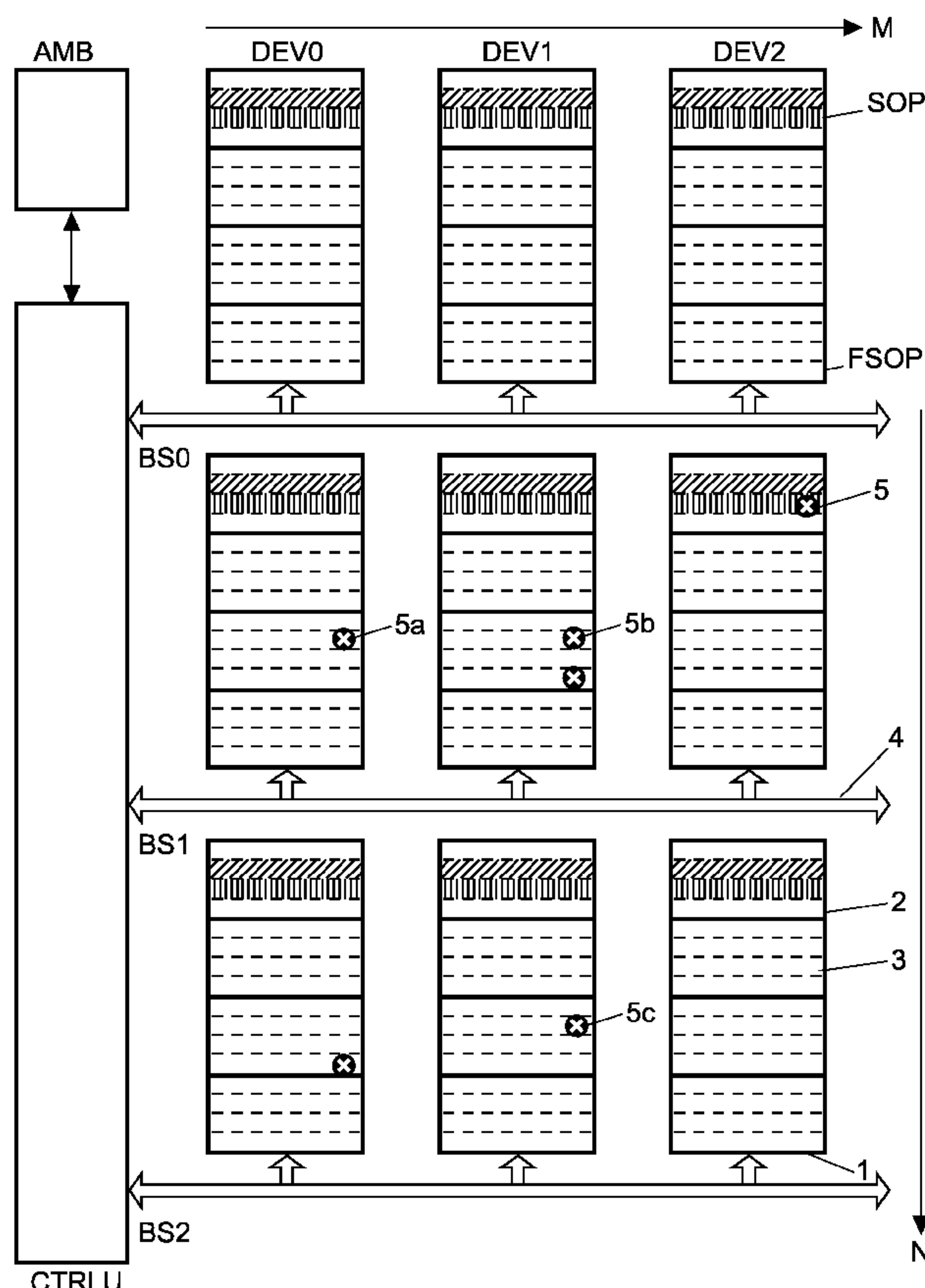
(74) Agent: **HARTNACK, Wolfgang**; Thomson Holding Germany GmbH & Co. OHG, European Patent Operations, Karl-Wiechert-Allee 74, 30625 Hannover (DE).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, 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, LV, LY, MA, MD, 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, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT,

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR RECORDING HIGH-SPEED INPUT DATA INTO A MATRIX OF MEMORY DEVICES



(57) Abstract: For recording or replaying in real-time digital high bandwidth video signals, e.g. HDTV, HD progressive or HD film capture signals, very fast memories are required. For storage of streaming HD video data NAND FLASH memory based systems could be used. Flash memory devices are physically accessed in a page oriented mode. According to the invention, the input data are written in a multiplexed fashion into a matrix of multiple flash devices. A list processing is performed that is as simple and fast as possible, and defect pages of flash blocks of single flash devices are addressed within the matrix architecture. When writing in a sequential manner, the data content for the current flash device page of all flash devices of the matrix is copied to a corresponding storage area in an additional memory buffer. After the current series of pages has been written without error into the flash devices, the corresponding storage area in an additional memory buffer is enabled for overwriting with following page data. In case an error occurred in the current page in one or more flash devices, the content of these current pages is kept in the additional memory buffer.

WO 2007/080031 A1

WO 2007/080031 A1



RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

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.

Published:

— *with international search report*

Method and Apparatus for recording high-speed input data into a matrix of memory devices

The invention relates to a method and to an apparatus for recording high-speed input data into a matrix of memory devices.

Background

10

For recording or replaying in real-time digital high bandwidth video signals, e.g. HDTV, HD progressive or HD film capture signals, very fast memories are required. For storage of streaming HD video data NAND FLASH memory based systems could be used. Flash memory devices are physically accessed in a page oriented mode, whereby one 'page' includes e.g. 1024 data words and related error correction code (ecc). Erase operations on a specific flash memory can be carried out on certain-size data blocks only. These data blocks are denoted by the term 'flash block' in the following. A flash block consists of e.g. 64 pages.

20

NAND flash memories have two basic disadvantages:

25

- the write access is rather slow;
- they have unmasked production defects and acquire even more defects during their lifetime. The required error handling is under user responsibility. This is also true for equivalent memory types.

30

Since a detection of defects in flash memory devices (e.g. NAND devices) takes place for instance during an erase operation, a defect in a page makes an entire flash block unusable. Such defect flash blocks must not be used by the file system. The handling of such defects is fully under user responsibility.

35

It is known to use redundancy codes like Reed-Solomon for such error processing, but that has other disadvantages like high dynamic time consumption versus real-time behaviour.

29316-48

2

For example, the following NAND flash memories are on the market:

Samsung K9K2G16U0M-YCB000 (2Gbit, 16bit oriented), K9W4G08U0M-YCB000

(4Gbit, 8bit oriented), Toshiba TH58NVG2S3BFT00 (4Gbit, 8bit oriented),

MICRON MT29G08AAxxx (2Gbit, 8bit oriented), MT29G16AAxxx

5 (2Gbit, 16bit oriented), Samsung K9K4G08 (4Gbit, 8bit oriented).

Invention

The invention can be used for real-time recording of high definition streaming video data on NAND flash memory based devices. Using flash devices while recording in real-time at high data throughput will result in big files or takes, respectively. It is
10 important to have a resource and performance efficient processing for dynamical defect remapping from faulty flash blocks towards free flash blocks. Today's most NAND flash applications are Reed-Solomon error protected and not real-time capable.

A problem to be solved by some embodiments of the invention is to provide a
15 resource and performance efficient algorithm for remapping defects in FLASH devices that occur while in operation, to support low power consumption, short operation time and real-time capability.

According to some embodiments of the invention, high-speed input data are written in a multiplexed fashion into a matrix of multiple flash devices. A list processing is
20 performed that is as simple and fast as possible, and defect pages of flash blocks of single flash devices are addressed within the matrix architecture. When writing in a sequential manner, the data content for the current flash device page of all flash devices of the matrix is also copied to a corresponding storage area

in an additional memory buffer. After the current series of pages has been written without error into the flash devices, the corresponding storage area in an additional memory buffer is enabled for overwriting with following page data.

5 In case an error occurred in the current page in one or more flash devices, the content of these current pages is kept in the additional memory buffer.

In principle, the inventive method is suited for recording
10 high-speed input data into a matrix of memory devices of a first type, said matrix including at least two memory devices per row and at least two memory devices per column, wherein the memory devices of a row are connected to a common bus and to each row of said matrix a separate common bus
15 is assigned, and wherein said memory devices each are internally arranged in multiple pages to which pages input data can be written in a sequential manner, and wherein when writing into said first type memory devices defects may occur at different locations in said memory devices, said
20 method including the steps:

A) writing a first section of input data in a multiplexed manner, using said common buses, in the same current page of the memory devices, starting with the first column of memory devices, column by column, and writing this first section of
25 input data also in a corresponding section of an additional memory device of a second type different from said first type;

B) checking whether or not at least one defect in at least one memory device has occurred while writing said first section of input data into said current page of the memory devices;
30

C) writing the following section of input data in a multiplexed manner, using said common buses, in the following page of the memory devices, starting with the first column
35 of memory devices, column by column,
and if no defect has been found in the foregoing step, also

writing this following section of input data in the same section of said additional memory device,
and if a defect has been found in the foregoing step, also writing this following section of input data in a following
5 corresponding section of said additional memory device;
D) checking whether or not at least one defect in at least one memory device has occurred while writing said following section of input data into said following page;
E) continuing with steps C) and D) until all sections of input
10 put data have been written into said matrix of memory devices and into said additional memory device;
F) copying input data stored in corresponding sets of pages of said matrix of memory devices, which sets include per set at least one defect in at least one memory device, to corresponding sets of pages in said matrix of memory devices
15 which are not yet occupied with said input data, thereby taking input data of defect single pages not from said matrix of memory devices but from corresponding single page data stored in said additional memory device.

20

In principle the inventive apparatus is suited for recording high-speed input data into a matrix of memory devices of a first type, said matrix including at least two memory devices per row and at least two memory devices per column,
25 wherein the memory devices of a row are connected to a common bus and to each row of said matrix a separate common bus is assigned, and wherein said memory devices each are internally arranged in multiple pages to which pages input data can be written in a sequential manner, and wherein when
30 writing into said first type memory devices defects may occur at different locations in said memory devices, said apparatus including an additional memory device of a second type different from said first type and means being adapted for carrying out the following functions:
35 A) writing a first section of input data in a multiplexed manner, using said common buses, in the same current page of

29316-48

5

the memory devices, starting with the first column of memory devices, column by column, and writing this first section of input data also in a corresponding section of said additional memory device;

5 B) checking whether or not at least one defect in at least one memory device has occurred while writing said first section of input data into said current page of the memory devices;

C) writing the following section of input data in a multiplexed manner, using said common buses, in the following page of the memory devices, starting with the first column of memory devices, column by column,

10 and if no defect has been found in the foregoing function, also writing this following section of input data in the same section of said additional memory device,

and if a defect has been found in the foregoing function, also writing this following section of input data in a following corresponding section of said additional memory device;

15 D) checking whether or not at least one defect in at least one memory device has occurred while writing said following section of input data into said following page;

E) continuing with functions C) and D) until all sections of input data have been written into said matrix of memory devices and into said additional memory device;

20 F) copying input data stored in corresponding sets of pages of said matrix of memory devices, which sets include per set at least one defect in at least one memory device, to corresponding sets of pages in said matrix of memory devices which are not yet occupied with said input data, thereby taking input data of defect single pages not from said matrix of memory devices but from corresponding single page data stored in said additional memory device.

29316-48

5a

Advantageous additional embodiments of the invention are disclosed herein.

In some embodiments, the first type memory devices are FLASH memories.

In some embodiments, the defects are stored in a defects list that contains information items about the corresponding page, the corresponding memory device
5 and the corresponding bus for each defect.

In some embodiments, the defects are stored in a defects list that contains information items about the corresponding block, the corresponding page, the corresponding memory device and the corresponding bus for each defect.

In some embodiments, the defect flash pages of a flash block are determined as
10 follows:

- a) initialising a register array for 'page' values with invalid values;
- b) making the first valid 'block' value BLNO out of the defects list a 'reference block' value;
- c) copying the corresponding 'page' value from the defects list into the appropriate
15 field in said page register array;
- d) analysing the next defects list entry;
- e) if the current 'block' value BLNO equals the 'reference block' value, continuing with step c) else continuing with step d);
- f) continuing steps c) to e) until the end of said defects list is reached.

20 In some embodiments, a bitmap register for storing the bitmaps for the page entries processed in step c) is generated and every N bits of the bitmaps for these pages are processed in a sequential processing order and said bitmap register is used to shortcut non-defect flash devices, the bitmap register being generated as follows:

29316-48

5b

- g) initialising a pointer for the bitmap and for a 'device counter';
- h) initialising a pointer for said page register array and initialising said bitmap register with zeroes;
- i) addressing a single cell in said page register array;
- 5 j) if its entry is valid, continuing with step k) else continuing with step l);
- k) loading N bits from the corresponding bitmap stored in said defects list and logically ORing that with the corresponding bitmap register values;
- l) incrementing the page register array pointer and continuing with step i) until the end of said page register array is reached;
- 10 m) if the bitmap register value equals zero, incrementing the bitmap pointer and said 'device counter' and continuing with step h) until the end of the bitmap is reached.

In some embodiments, if in step m) the bitmap register value not equals zero, the following steps are carried out in order to determine the bus for a defect block in a defect flash device and to copy corresponding flash pages:

- 15 n) initialising a 'bus count' and the pointer for said page register array;
- o) reading said page register array for a valid value;
- p) analysing the bit in said bitmap register where 'bus count' is pointing to: if that bit equals zero, incrementing 'bus count' and continuing with step o), if that bit equals one, continuing with step q);
- 20 q) for all page values between zero and the value read in step o) minus one, copying flash pages from the faulty flash block to a new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';

29316-48

5c

- r) for the page value read in step o), copying the corresponding flash page data from said additional memory buffer to the new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';
- 5 s) incrementing the pointer for said page register array and continuing with step o) until the end of said page register array is reached;
- t) updating a 'block remap table' for the appropriate device and marking the corresponding block entry in said defects list as invalid;
- u) continuing with step a) as described herein.

Drawings

Exemplary embodiments of the invention are described with reference to the accompanying drawings, which show in:

- 5 Fig. 1 matrix of flash devices;
- Fig. 2 extra buffer containing non-faulty data copies of faulty flash pages;
- Fig. 3 entry of a defects list;
- Fig. 4 example defects list;
- 10 Fig. 5 page register array;
- Fig. 6 bitmap register;
- Fig. 7 page replacement sequence.

15 Exemplary embodiments

It is advantageous for real-time recording to have the flash devices organised within a matrix that is connected by several busses BS0, BS1, BS2, ... to a controller unit CTRLU as depicted in Fig. 1. Several flash devices are sharing the same data bus, for example devices DEV0 to DEV2 share bus BS0. More generally, the number of flash devices sharing the same data bus is M. There are several such flash device/common bus rows in the matrix, their number is N. The matrix includes at least two memory devices per row and at least two memory devices per column. The controller unit CTRLU may receive the high-speed input data to be stored or recorded. Regarding an 'erase' operation, the smallest addressable unit in a NAND flash memory like device 1 is a flash block 2 (depicted with a bold surrounding, i.e. in the figure device 1 has four flash blocks).

Regarding a 'write' operation, the smallest addressable unit in a NAND flash memory is a flash page 3 (in the figure each flash block contains four flash pages).

35 In one implementation, the total number of flash blocks in the matrix is e.g. 4096. Each block may contain 64 pages.

The input data to be stored are multiplexed and are written via the N parallel buses into the same current page in each of the flash devices in the first column of the matrix, i.e. into device DEV0 and the devices arranged below it. After
5 this current page is full, the following input data are written into the same current page in the second column of the matrix, i.e. into device DEV1 and the devices arranged below it. This write processing is continued M times until the current page of all flash devices of the matrix is
10 filled with input data. This status is depicted by the diagonal hatching of the second page in the flash devices. Thereafter the write processing is continued with the next (i.e. the third) page in the flash devices, which third page is marked in Fig. 1 by a vertical hatching. These pages form
15 a set of pages SOP.

For achieving a high-speed real-time recording and defect recovery, the data content for the current flash page of all flash devices of the matrix is in each case copied in a sequential manner to an additional memory buffer AMB (e.g. an
20 SRAM) which can be connected to controller unit CTRLU or to the buses, and these data will remain in buffer AMB if one or more defects have occurred in the current page in one or more of the flash devices of the matrix.

25 The extra buffer AMB is shown in Fig. 2 in more detail. The matrix depicted in Fig. 1 includes nine flash devices. The first column of devices in the matrix is related to the first column in memory AMB, the second column in the matrix is related to the second column in memory AMB, and so on. A
30 line in memory AMB is related to a page in a row of devices in the matrix.

When writing into the third page (marked by vertical hatching) in the flash devices of the matrix an error or defect
5 has occurred. Therefore the original input data content of these pages is kept stored in the corresponding area (marked
35 by vertical hatching) in memory AMB, i.e. it is kept stored

in AMB without defect 5. Therefore, when the next series of pages (i.e. the fourth page) is written in the flash devices, the original input data are written into the following section in memory AMB (i.e. lines 3 to 6 in Fig. 2). In case no defect has occurred in any of the fourth pages, this section (lines 3 to 6) in memory AMB is overwritten by the next series of pages (i.e. the fifth page). Because three defects 5a, 5b and 5c occur in the tenth page (i.e. in block three of the flash devices), the original input data content of these pages is kept stored in the corresponding area (lines 3 to 6) in memory AMB without defects at corresponding locations 5A, 5B and 5C.

In effect, under the control of controller unit CTRLU, buffer AMB stores non-faulty data copies of faulty flash pages but its capacity is significantly smaller than that of the flash matrix. For example, RAM memory AMB can have a capacity such that the data content of up to 15 different pages of each device of the matrix can be stored therein.

The page errors (e.g. 5, 5a, 5b and 5c) which occurred during real-time recording are marked in a defects list, an entry of which defects list is shown in Fig. 3. That entry may contain the block number BLNO (0 ... B), the page number PGNO (0 ... P), and the device column number DEV0, ..., DEVm in the FLASH matrix. The bits BT0, BT1, ..., BTn in the column number indicate in which row number (= bus number BS0, BS1, ..., BSn) of the matrix a defect or multiple defects have occurred in a flash page. It is not necessary to store the location within a page of a defect.

30

An example defects list for the nine flash devices depicted in Fig. 1 is shown in Fig. 4. This list can be stored in controller unit CTRLU. This example defects list defines the defect pages in the Fig. 1 matrix.

35

A first defect 5 is located in block number BLNO=0 in page number PGNO=2, and according to the bitmap part BTMP in the

third device column DEV2 with respect to the second bus BS1. A second defect 5a is located in block number BLNO=2 in page number PGNO=1 in the first device column DEV0 with respect to the second bus BS1.

5 A third defect 5b is located in block number BLNO=2 in page number PGNO=1 in the second device column DEV1 with respect to the second bus BS1.

A fourth defect 5c is located in block number BLNO=2 in page number PGNO=1 in the second device column DEV1 with respect
10 to the third bus BS2.

Further defects are located in block number BLNO=2 in page number PGNO=3 in the first device column DEV0 with respect to the third bus BS2 and in the second device column DEV1 with respect to the second bus BS1.

15 Upon or while writing the input data into the flash devices, the flash devices report any defects via the buses BS0 to BS2 to controller unit CTRLU.

The respective flash page copies have to remain buffered.

20 When recovering the defects, it is advantageous that the defects list is analysed in a resource and time efficient manner, to support low power consumption, short operation time and real-time behaviour. Target is that the content of a flash block that contains one or more faulty flash pages is
25 copied to corresponding sets FSOP of free flash pages in a free flash block within the same flash device. At last the faulty flash block has to be logically re-mapped by the new flash block in e.g. a file system.

Hereby the task of analysing the defects list that contains
30 information about the flash block, the flash page, device and bus of every defect is described in detail below. During analysing processing, the appropriate defect flash page within a block is to be copied from the buffer to the new flash block. All other flash pages within a block are to be
35 copied from the faulty flash block to the new flash block.

Determining defect flash pages of a flash block:

- a) Initialise a register array for 'page' values with invalid values (Fig. 5 shows a corresponding page register array containing defect pages for flash block BLNO=2, which page register contains the corresponding page number values from column PGNO for the rows concerning BLNO=2);
- b) Make the first valid 'block' value BLNO out of the defects list a 'reference block' value;
- 10 c) Copy the corresponding 'page' value from the defects list into the appropriate field in the page register array;
- d) Analyse the next defects list entry;
- e) If the current 'block' value BLNO equals the 'reference block' value go to step c) else go to step d);
- 15 f) Loop steps c) to e) until the end of the defects list is reached.

Fig. 6 shows a bitmap register for storing the bitmaps for the page entries processed in step c). The bitmaps for these pages are processed in a sequential processing order PRO (every N bits). The bitmap register is used to shortcut non-defect flash devices:

- g) Initialise a pointer for the bitmap and for a 'device counter';
- 25 h) Initialise a pointer for the page register array. Initialise the bitmap register with zeroes;
- i) Address a single cell in the page register array;
- j) If its entry is valid go to step k) else go to step l);
- k) Load N bits from the corresponding bitmap (stored in the defects list) and logically OR that with the corresponding bitmap register values (N=number of busses);
- 30 l) Increment the page register array pointer and go to step i) until the end of the page register array is reached;
- m) If the bitmap register value equals zero then increment the bitmap pointer and the 'device counter' and go to
- 35 step h) until the end of the bitmap is reached. If the

bitmap register value not equals zero then start the following steps n) to u).

Fig. 7 explains a page replacement sequence according to a
5 bitmap register and bitmap analysis. The goal is to determine the bus for a defect block in a defect flash device and to copy corresponding flash pages:

- n) Initialise a 'bus count' and the pointer for the page register array;
- 10 o) Read the page register array (Fig. 5) for a valid value;
- p) Analyse the bit in the bitmap register where 'bus count' is pointing to: if that bit equals zero, increment 'bus count' and go to step o), if that bit equals one, go to step q);
- 15 q) For all page values between zero and the value read in step o) minus one, copy flash pages from the faulty flash block to a new flash block. The reference flash device number corresponds to value 'device counter', the reference bus number corresponds to value 'bus count';
- 20 r) For the page value read in step o), copy the corresponding flash page data from the additional memory buffer AMB to the new flash block. The reference flash device number corresponds to value 'device counter', the reference bus number corresponds to value 'bus count';
- 25 s) Increment the pointer for the page register array and go to step o) until the end of the page register array is reached;
- t) Update a 'block remap table' for the appropriate device and mark the corresponding block entry in the defects
30 list as invalid;
- u) Go to step a).

The above registers, pointers, list and table can be controlled by controller unit CTRLU.

35 The invention can be used in any block oriented storage that can be affected with defects during lifetime.

29316-48

12

CLAIMS:

1. A method for recording high-speed input data into a matrix of memory devices of a first type, said matrix including at least two memory devices per row and at least two memory devices per column, wherein the memory devices of a row are
5 connected to a common bus and to each row of said matrix a separate common bus is assigned, and wherein said memory devices each are internally arranged in multiple pages to which pages input data can be written in a sequential manner, and wherein when writing into said first type memory devices defects may occur at different locations in said memory devices, said method comprising the steps of:
- 10 A) writing a first section of input data in a multiplexed manner, using said common buses, in the same current page of the memory devices, starting with the first column of memory devices, column by column, and writing this first section of input data also in a corresponding section of an additional memory device of a second type different from said first type;
- 15 B) checking whether or not at least one defect in at least one memory device has occurred while writing said first section of input data into said current page of the memory devices;
- C) writing the following section of input data in a multiplexed manner, using said common buses, in the following page of the memory devices, starting with the first
20 column of memory devices, column by column, and if no defect has been found in the foregoing step, also writing this following section of input data in the same section of said additional memory device, and if a defect has been found in the foregoing step, also writing this following section of input data in a following corresponding section of said additional memory device;
- 25 D) checking whether or not at least one defect in at least one memory device has occurred while writing said following section of input data into said following page;

29316-48

13

E) continuing with steps C) and D) until all sections of input data have been written into said matrix of memory devices and into said additional memory device;

F) copying input data stored in corresponding sets of pages of said matrix of memory devices, which sets include per set at least one defect in at least one memory device,
5 to corresponding sets of pages in said matrix of memory devices which are not yet occupied with said input data, thereby taking input data of defect single pages not from said matrix of memory devices but from corresponding single page data stored in said additional memory device.

2. The method according to claim 1, wherein said first type memory
10 devices are FLASH memories.

3. The method according to claim 1 or 2, wherein said defects are stored in a defects list that contains information items about the corresponding page, the corresponding memory device and the corresponding bus for each defect.

4. The method according to claim 2, wherein said defects are stored in a
15 defects list that contains information items about the corresponding block, the corresponding page, the corresponding memory device and the corresponding bus for each defect.

5. The method according to claim 4, wherein defect flash pages of a flash block are determined as follows:

20 a) initialising a register array for 'page' values with invalid values;

b) making the first valid 'block' value BLNO out of the defects list a 'reference block' value;

c) copying the corresponding 'page' value from the defects list into the appropriate field in said page register array;

29316-48

14

d) analysing the next defects list entry;

e) if the current 'block' value BLNO equals the 'reference block' value, continuing with step c) else continuing with step d);

f) continuing steps c) to e) until the end of said defects list is reached.

5 6. The method according to claim 5, wherein a bitmap register for storing the bitmaps for the page entries processed in step c) is generated and every N bits of the bitmaps for these pages are processed in a sequential processing order and said bitmap register is used to shortcut non-defect flash devices, said bitmap register being generated as follows:

10 g) initialising a pointer for the bitmap and for a 'device counter';

h) initialising a pointer for said page register array and initialising said bitmap register with zeroes;

i) addressing a single cell in said page register array;

j) if its entry is valid, continuing with step k) else continuing with step l);

15 k) loading N bits from the corresponding bitmap stored in said defects list and logically ORing that with the corresponding bitmap register values;

l) incrementing the page register array pointer and continuing with step i) until the end of said page register array is reached;

20 m) if the bitmap register value equals zero, incrementing the bitmap pointer and said 'device counter' and continuing with step h) until the end of the bitmap is reached.

7. The method according to claim 6 wherein, if in step m) said bitmap register value not equals zero, the following steps are carried out in order to determine the bus for a defect block in a defect flash device and to copy corresponding flash pages:

29316-48

15

- n) initialising a 'bus count' and the pointer for said page register array;
- o) reading said page register array for a valid value;
- p) analysing the bit in said bitmap register where 'bus count' is pointing to: if that bit equals zero, incrementing 'bus count' and continuing with step o), if that bit equals one, continuing with step q);
- q) for all page values between zero and the value read in step o) minus one, copying flash pages from the faulty flash block to a new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';
- r) for the page value read in step o), copying the corresponding flash page data from said additional memory buffer to the new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';
- s) incrementing the pointer for said page register array and continuing with step o) until the end of said page register array is reached;
- t) updating a 'block remap table' for the appropriate device and marking the corresponding block entry in said defects list as invalid;
- u) continuing with step a) in claim 5.

8. An apparatus for recording high-speed input data into a matrix of memory devices of a first type, said matrix including at least two memory devices per row and at least two memory devices per column, wherein the memory devices of a row are connected to a common bus and to each row of said matrix a separate common bus is assigned, and wherein said memory devices each are internally arranged in multiple pages to which pages input data can be written in a sequential

29316-48

16

manner, and wherein when writing into said first type memory devices defects may occur at different locations in said memory devices, said apparatus including an additional memory device of a second type different from said first type and means being adapted for carrying out the following functions:

- 5 A) writing a first section of input data in a multiplexed manner, using said common buses, in the same current page of the memory devices, starting with the first column of memory devices, column by column, and writing this first section of input data also in a corresponding section of said additional memory device;
- B) checking whether or not at least one defect in at least one memory device has
10 occurred while writing said first section of input data into said current page of the memory devices;
- C) writing the following section of input data in a multiplexed manner, using said common buses, in the following page of the memory devices, starting with the first column of memory devices, column by column, and if no defect has been found in the
15 foregoing function, also writing this following section of input data in the same section of said additional memory device, and if a defect has been found in the foregoing function, also writing this following section of input data in a following corresponding section of said additional memory device;
- D) checking whether or not at least one defect in at least one memory device has
20 occurred while writing said following section of input data into said following page;
- E) continuing with functions C) and D) until all sections of input data have been written into said matrix of memory devices and into said additional memory device;
- F) copying input data stored in corresponding sets of pages of said matrix of memory devices, which sets include per set at least one defect in at least one memory device,
25 to corresponding sets of pages in said matrix of memory devices which are not yet

29316-48

17

occupied with said input data, thereby taking input data of defect single pages not from said matrix of memory devices but from corresponding single page data stored in said additional memory device.

9. Apparatus according to claim 8, wherein said first type memory devices
5 are FLASH memories.

10. Apparatus according to claim 8 or 9, wherein said defects are stored in a defects list that contains information items about the corresponding page, the corresponding memory device and the corresponding bus for each defect.

11. Apparatus according to claim 9, wherein said defects are stored in a
10 defects list that contains information items about the corresponding block, the corresponding page, the corresponding memory device and the corresponding bus for each defect.

12. Apparatus according to claim 11, wherein defect flash pages of a flash block are determined as follows:

- 15 a) initialising a register array for 'page' values with invalid values;
- b) making the first valid 'block' value BLNO out of the defects list a 'reference block' value;
- c) copying the corresponding 'page' value from the defects list into the appropriate field in said page register array;
- 20 d) analysing the next defects list entry;
- e) if the current 'block' value BLNO equals the 'reference block' value, continuing with step c) else continuing with step d);
- f) continuing steps c) to e) until the end of said defects list is reached.

29316-48

18

13. Apparatus according to claim 12, wherein a bitmap register for storing the bitmaps for the page entries processed in step c) is generated and every N bits of the bitmaps for these pages are processed in a sequential processing order and said bitmap register is used to shortcut non-defect flash devices, said bitmap register

5 being generated as follows:

g) initialising a pointer for the bitmap and for a 'device counter';

h) initialising a pointer for said page register array and initialising said bitmap register with zeroes;

i) addressing a single cell in said page register array;

10 j) if its entry is valid, continuing with step k) else continuing with step l);

k) loading N bits from the corresponding bitmap stored in said defects list and logically ORing that with the corresponding bitmap register values;

l) incrementing the page register array pointer and continuing with step i) until the end of said page register array is reached;

15 m) if the bitmap register value equals zero, incrementing the bitmap pointer and said 'device counter' and continuing with step h) until the end of the bitmap is reached.

14. Apparatus according to claim 13 wherein, if in step m) said bitmap register value not equals zero, the following steps are carried out in order to determine the bus for a defect block in a defect flash device and to copy

20 corresponding flash pages:

n) initialising a 'bus count' and the pointer for said page register array;

o) reading said page register array for a valid value;

29316-48

19

- p) analysing the bit in said bitmap register where 'bus count' is pointing to: if that bit equals zero, incrementing 'bus count' and continuing with step o), if that bit equals one, continuing with step q);
- q) for all page values between zero and the value read in step o) minus one, copying
5 flash pages from the faulty flash block to a new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';
- r) for the page value read in step o), copying the corresponding flash page data from
10 said additional memory buffer to the new flash block, whereby the reference flash device number corresponds to value 'device counter' and the reference bus number corresponds to value 'bus count';
- s) incrementing the pointer for said page register array and continuing with step o) until the end of said page register array is reached;
- t) updating a 'block remap table' for the appropriate device and marking the
15 corresponding block entry in said defects list as invalid;
- u) continuing with step a) in claim 12.

1/3

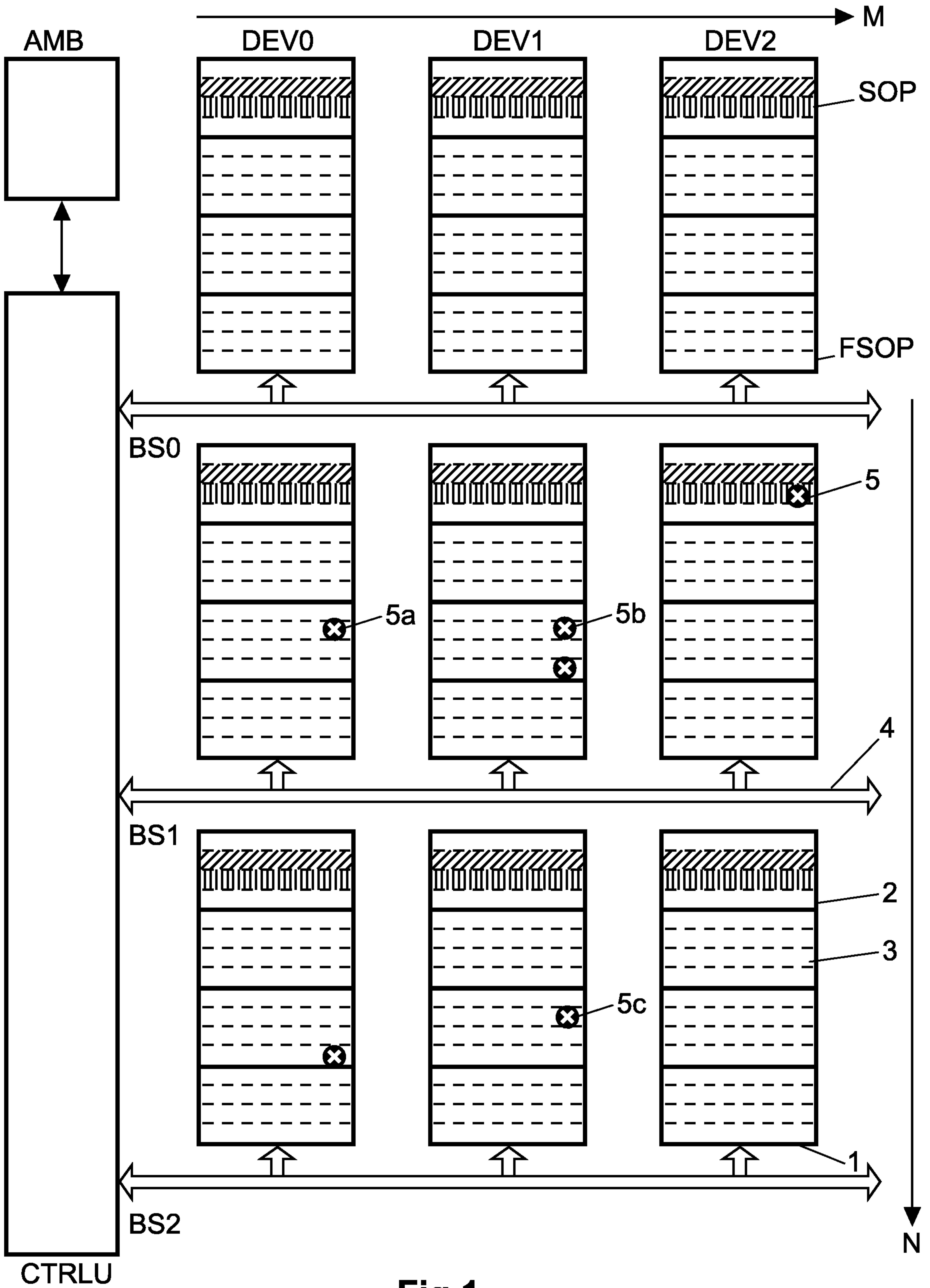


Fig.1

2/3

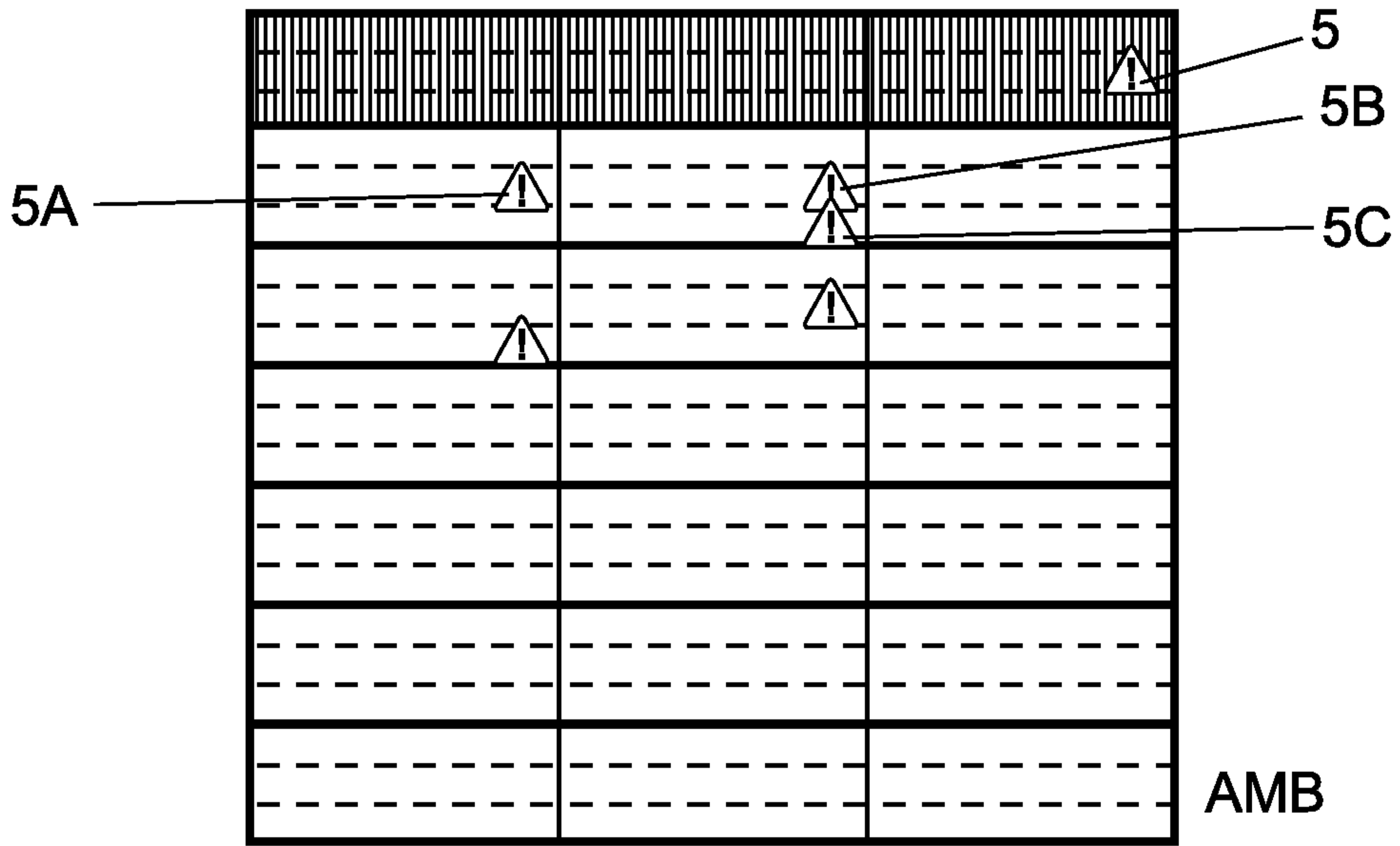


Fig.2

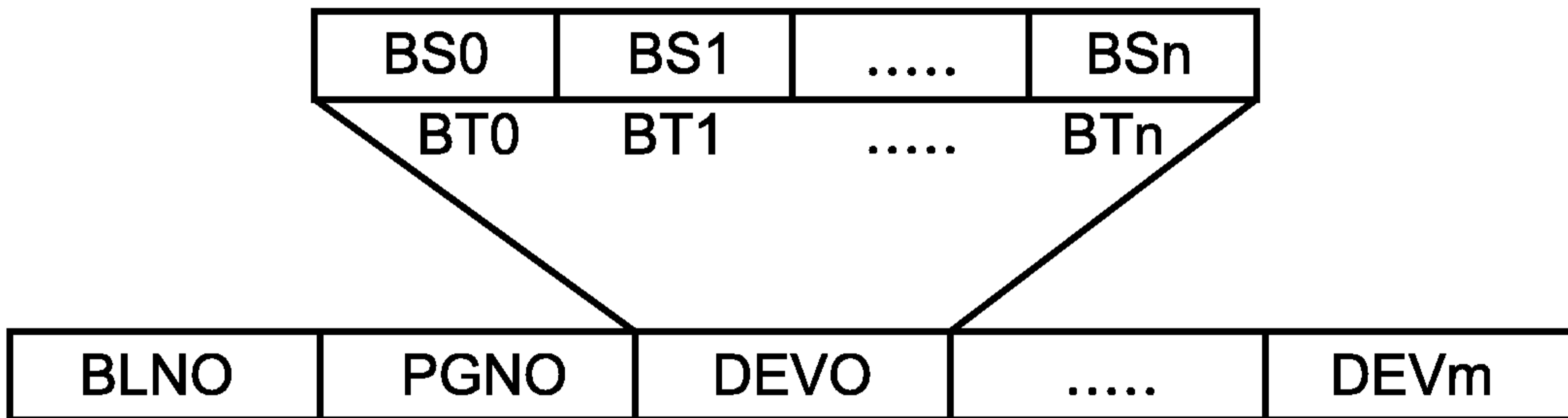


Fig.3

"0"	"2"	"000.000.010"
"2"	"1"	"010.011.000"
"2"	"3"	"001.010.000"
"INV"	n.a.	n.a.
"INV"	n.a.	n.a.

BTMP

Fig.4

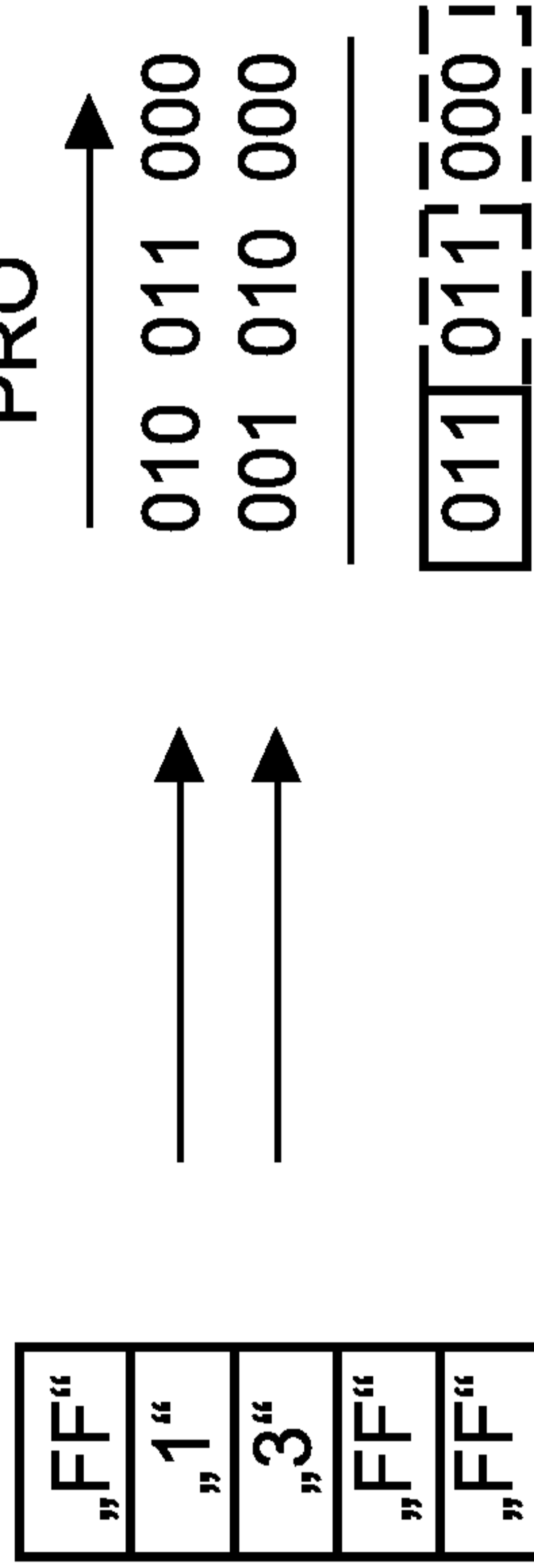


Fig.6

Fig.5

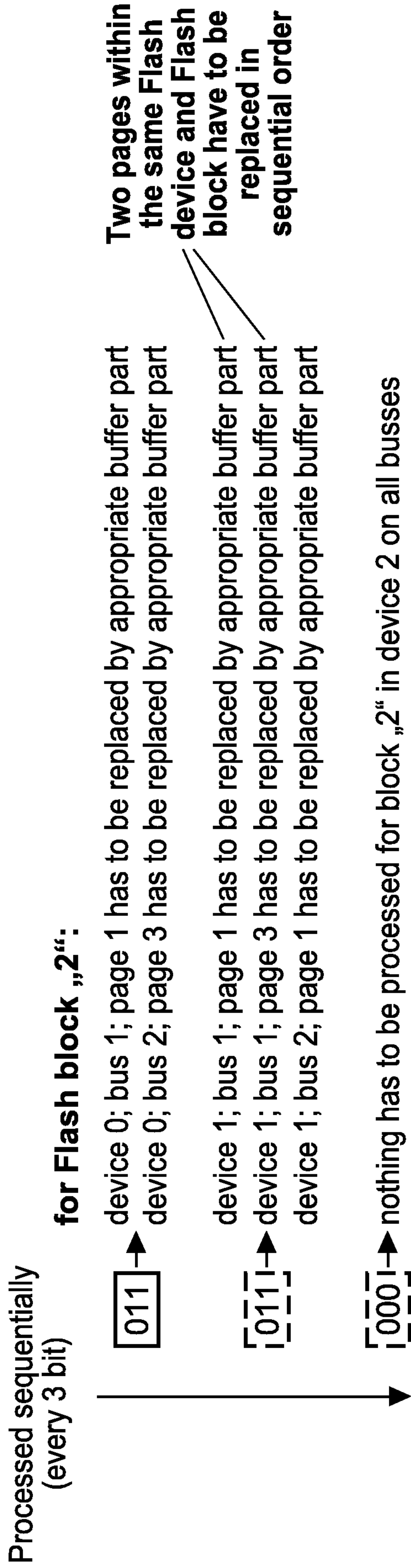


Fig.7

