

(21) Application No: 1310822.0  
 (22) Date of Filing: 04.10.2011  
 Date Lodged: 18.06.2013  
 (30) Priority Data:  
 (31) 2010293933 (32) 28.12.2010 (33) JP  
 (86) International Application Data:  
 PCT/JP2011/072891 Ja 04.10.2011  
 (87) International Publication Data:  
 WO2012/090564 Ja 05.07.2012

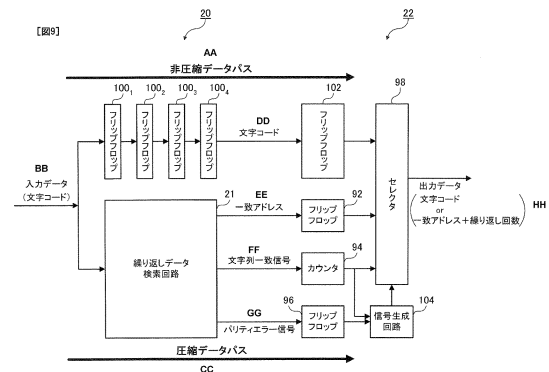
(51) INT CL:  
 H03M 7/40 (2006.01) G06F 17/30 (2006.01)  
 G11C 15/04 (2006.01)  
 (56) Documents Cited:  
 JP 630177242 A JP 080242176 A  
 JP 070114577 A JP 050252047 A  
 JP 2010268146 A  
 HISATADA MIYATAKE: High-speed parallel content-addressable memory (CAM) having parity check function during search, PROVISION, [Online] November 2010, pages 87 - 93 Retrieved from the Internet: URL: http://www06.ibm.com/ibm/jp/provision/no67/pdf/67 paper2. pdf

(71) Applicant(s):  
 International Business Machines Corporation  
 (Incorporated in USA - New York)  
 New Orchard Road, Armonk, N.Y. 10504,  
 United States of America  
 (72) Inventor(s):  
 Nobuyoshi Tanaka  
 Kiyoshi Nishino  
 Tadayuki Okada  
 Kiyoshi Takemura  
 (74) Agent and/or Address for Service:  
 IBM United Kingdom Limited  
 Intellectual Property Law, Hursley Park,  
 WINCHESTER, Hampshire, SO21 2JN,  
 United Kingdom

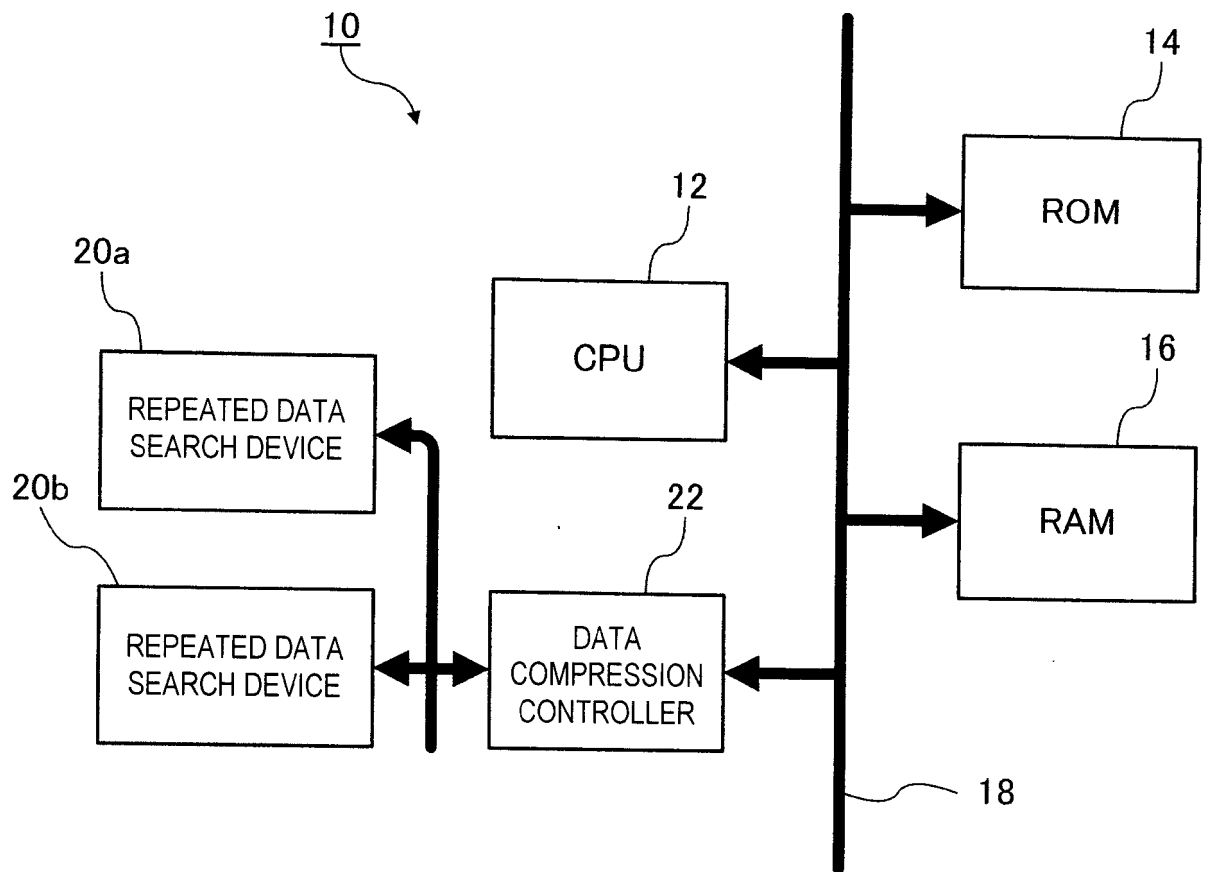
(58) Field of Search:  
 INT CL G06F, G11C, H03M  
 Other: Databases: IEEE Xplore, CiNii; Jitsuyo Shinan (JS) Koho 1922-96, JS Toroku Koho 1996-2011, Kokai JS Koho 1971-2011, Toroku JS Koho 1994-2011

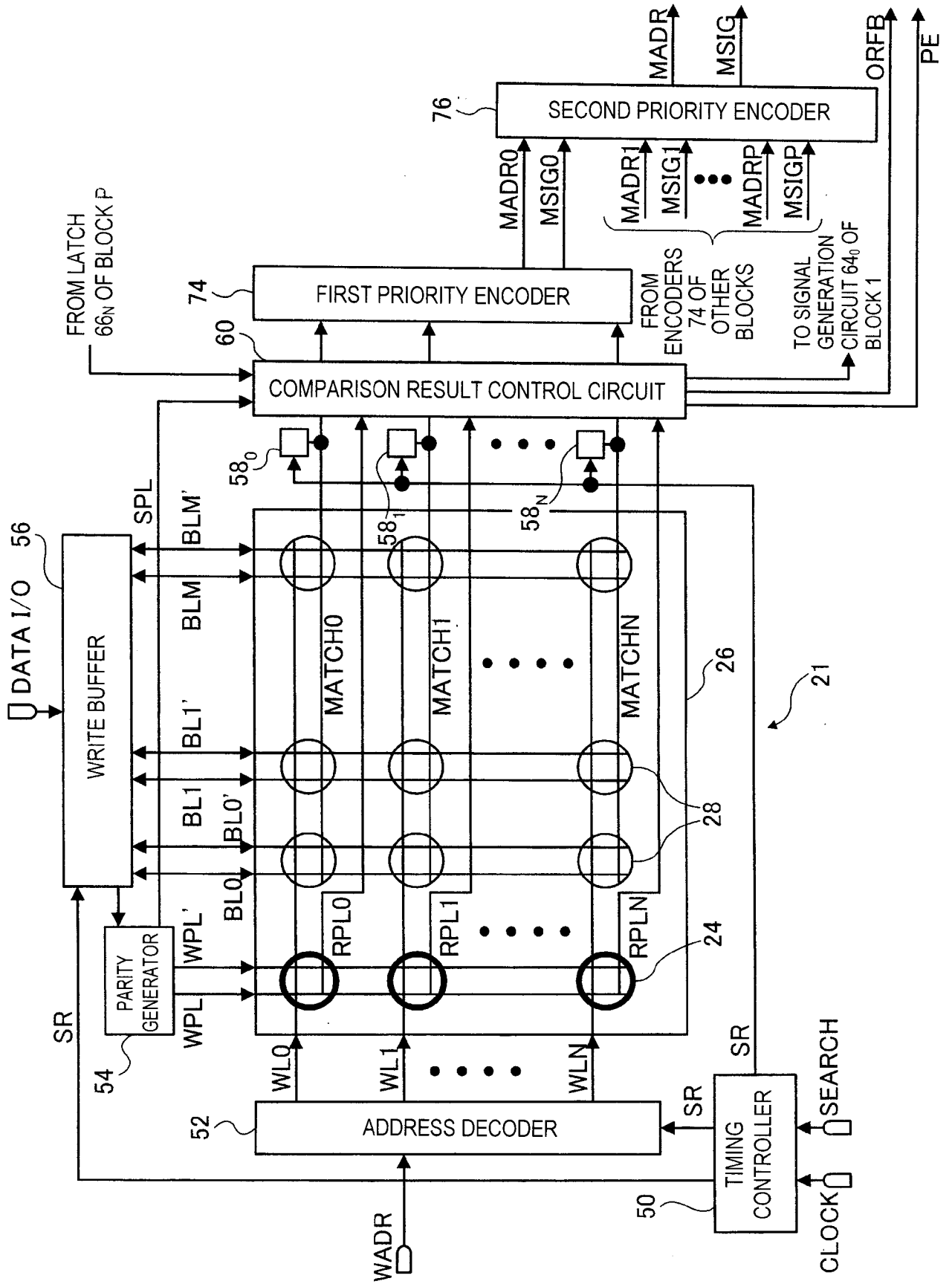
(54) Title of the Invention: **Apparatus and method for processing sequence of data element**  
 Abstract Title: **Apparatus and method for processing sequence of data element**

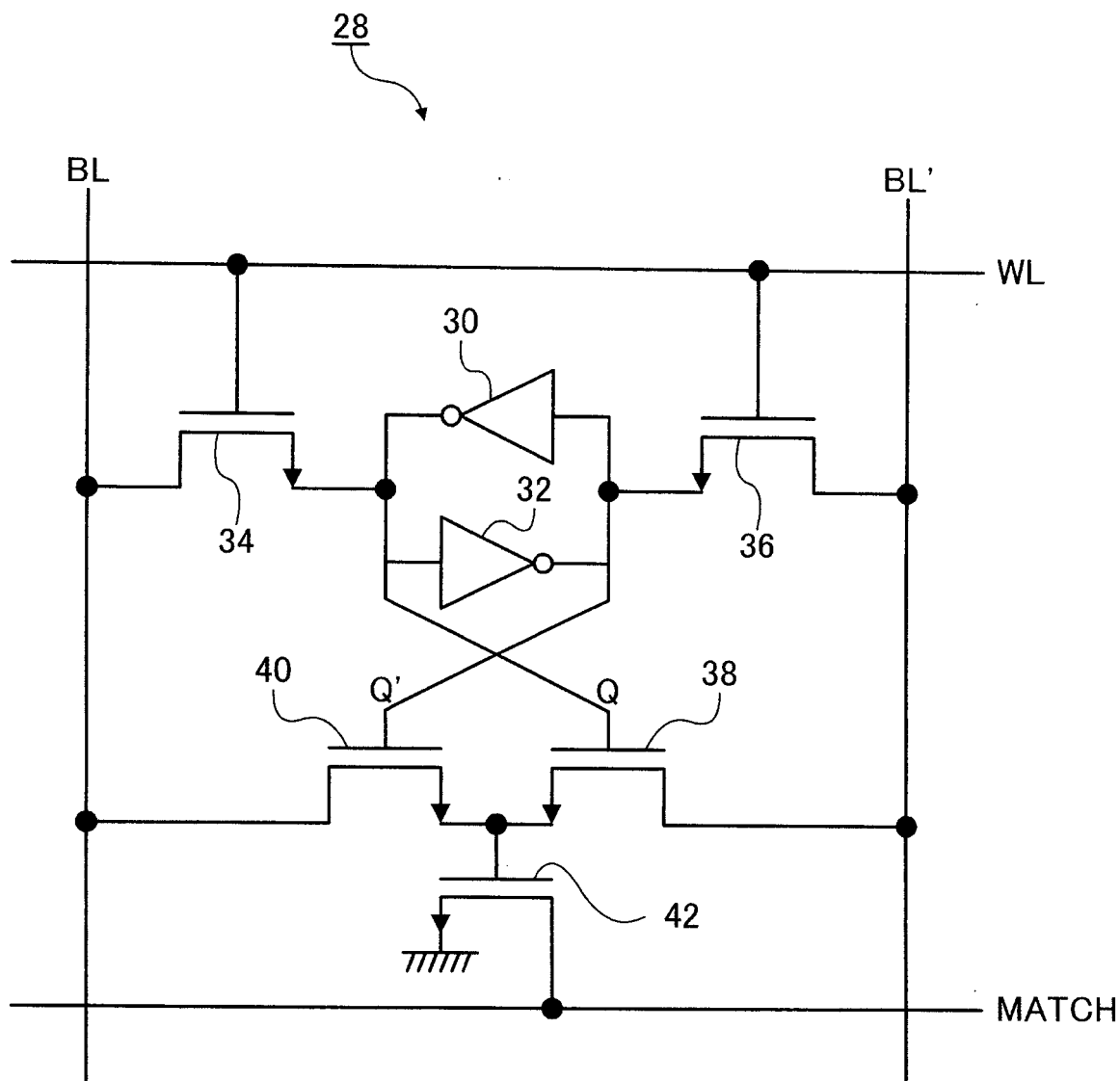
(57) A process performed using a CAM when an error is not generated in the data stored in the CAM, can be continued even if an error is generated in the data stored in the CAM. An input character code is provided to a compressed data path and a non-compressed data path. In the compressed data path, a repeat data retrieval circuit (21) outputs a coincident address of a character string in the CAM and a character string coincident signal which indicates the coincidence of the character string in the CAM. On the other hand, in the non-compressed data path, flip-flops (100<sub>1</sub> to 100<sub>4</sub>) send the character code without any changes. A selector (98) outputs the coincident address and the number of consecutive outputs of the character string coincident signals from the compressed data path if a parity error signal is not output from the repeat data retrieval circuit (21) and the number of consecutive outputs of the character string coincident signals is three or greater, and if not, outputs the character code from the non-compressed data path.

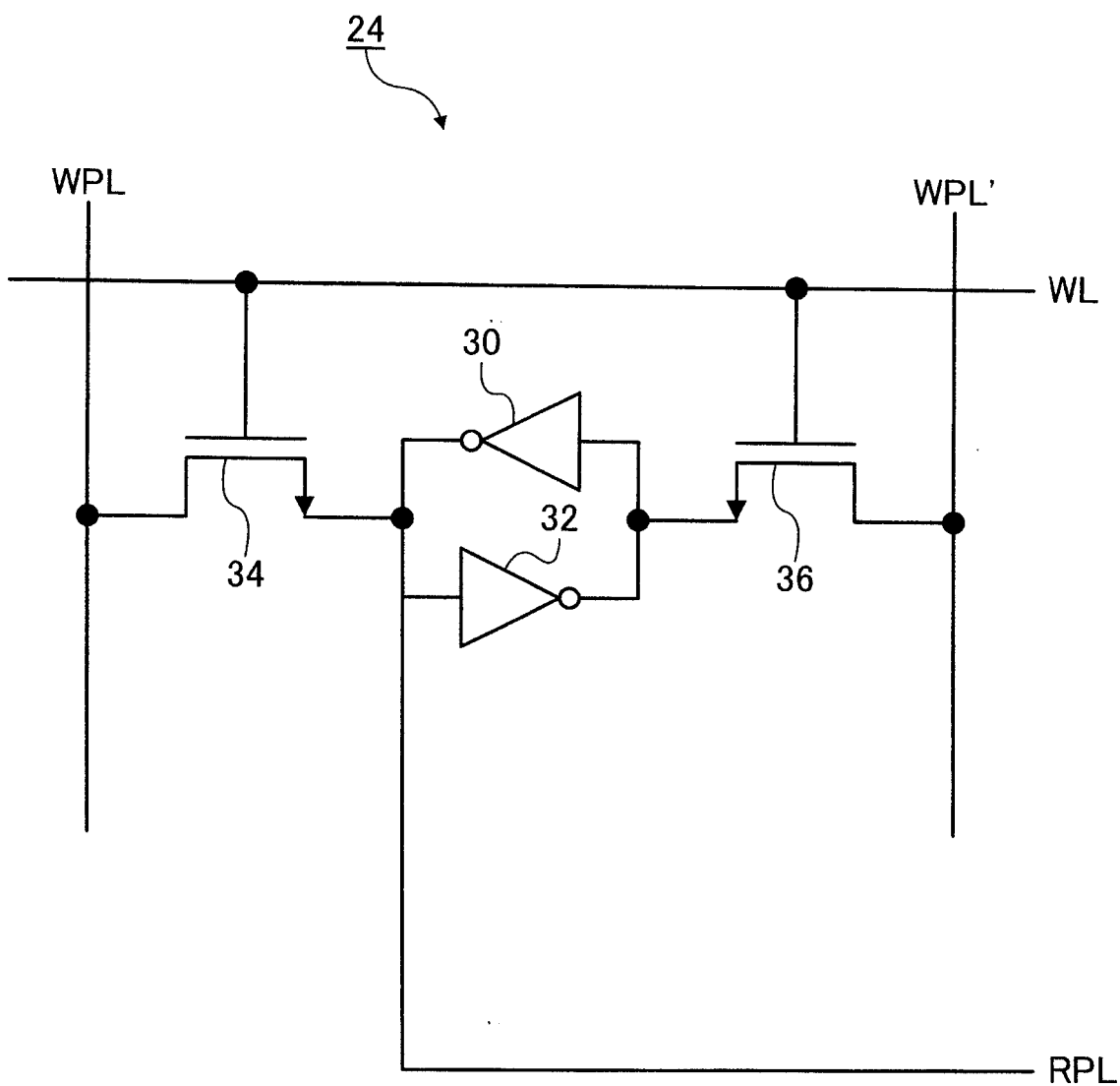


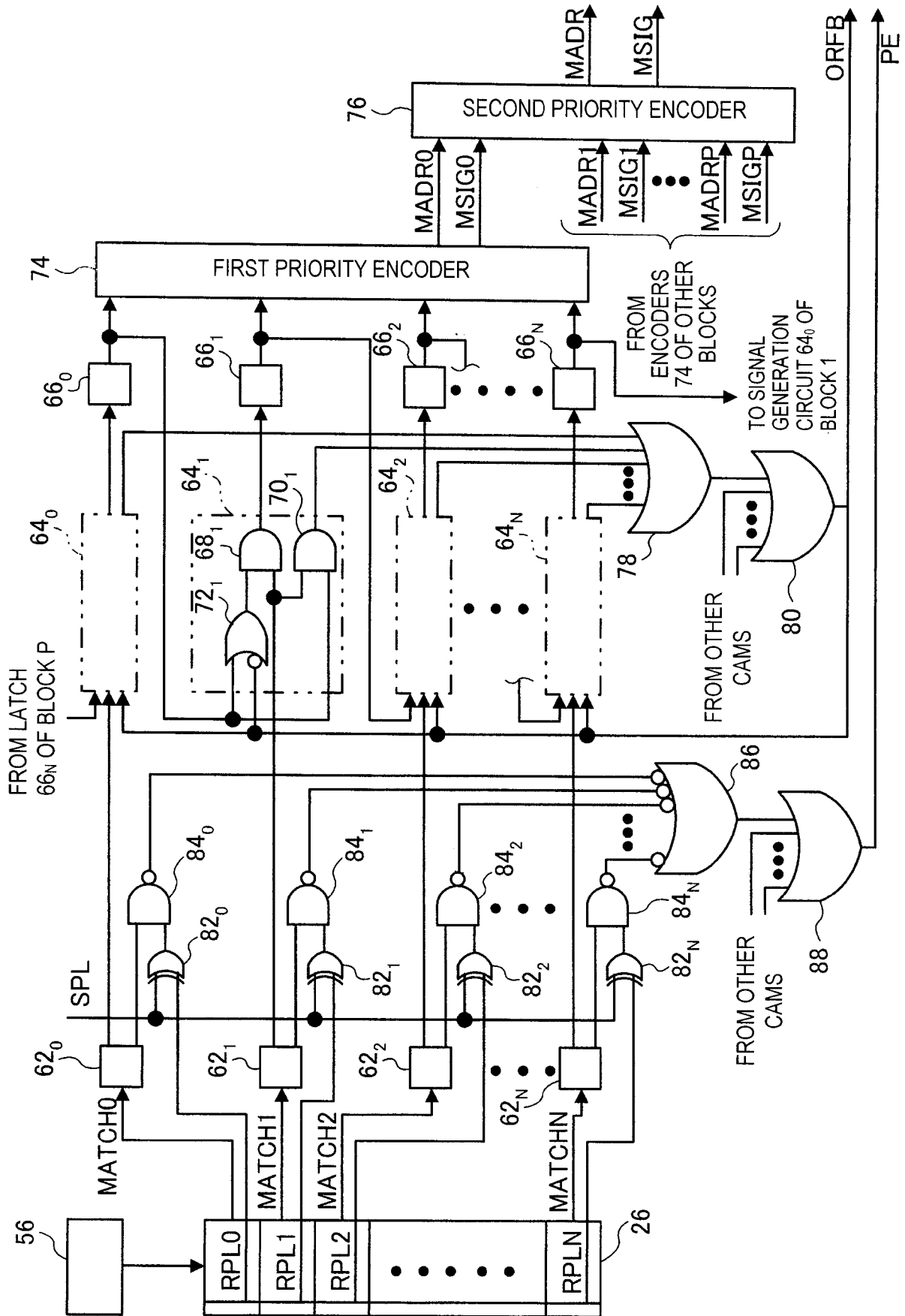
21 Repeat data retrieval circuit  
 92, 96, 100<sub>1</sub>, 100<sub>2</sub>, 100<sub>3</sub>, 100<sub>4</sub>, 102 Flip-flop  
 98 Selector  
 94 Counter  
 104 Signal generation circuit  
 AA Non-compressed data path  
 BB Input data (character code)  
 CC Compressed data path  
 DD Character code  
 EE Coincident address  
 FF Character string coincident signal  
 GG Parity error signal  
 HH Output data (character code or coincident address + the number of repeats)

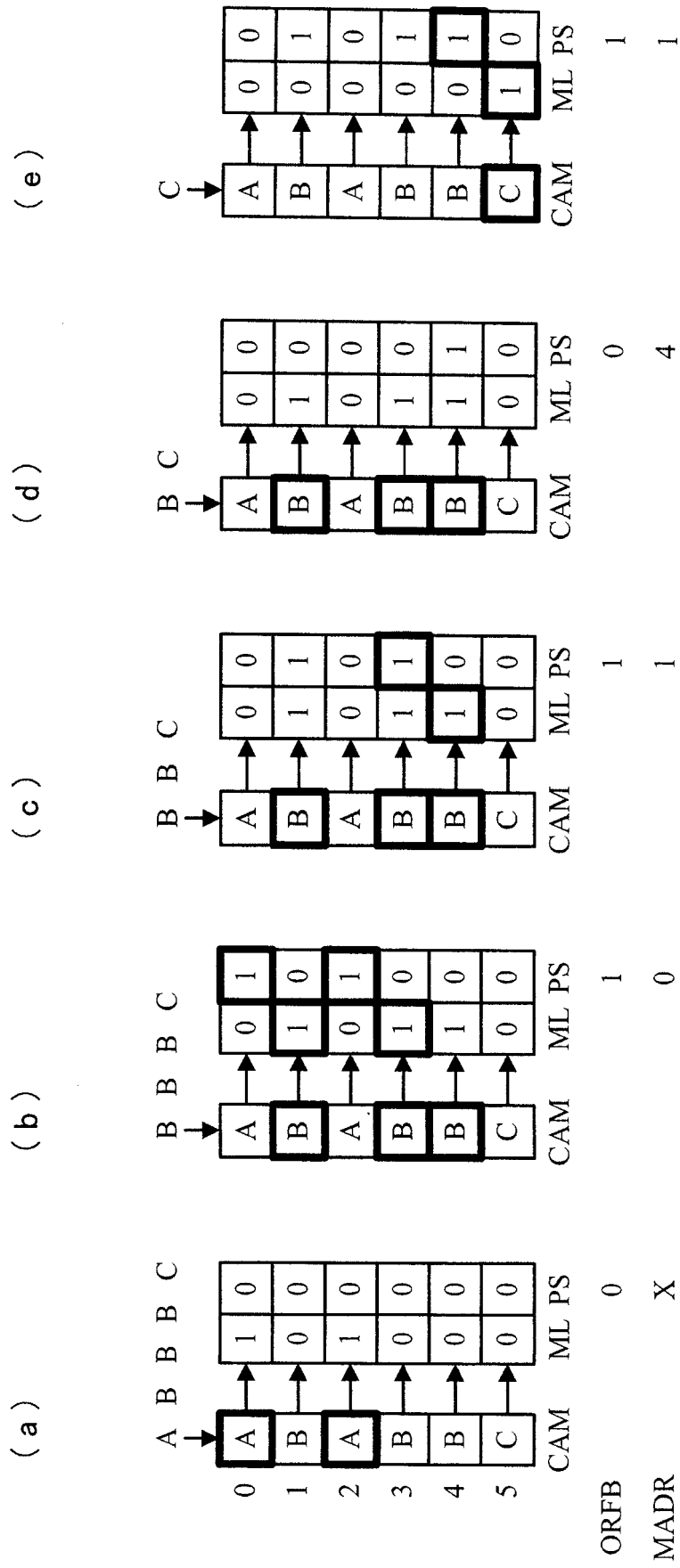


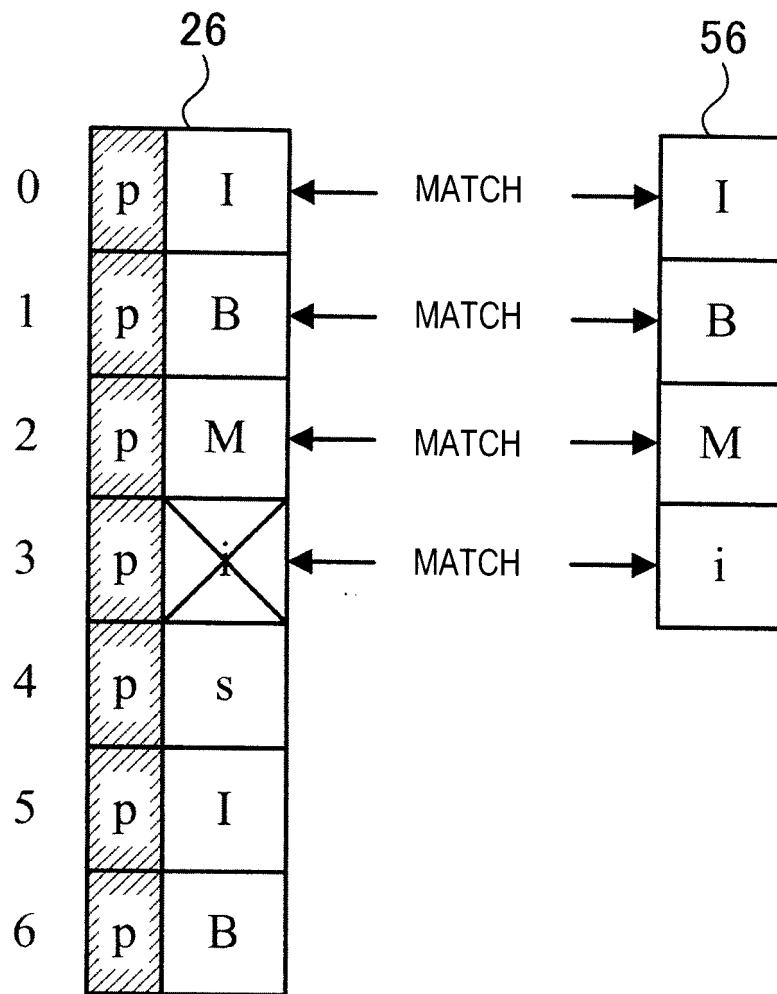




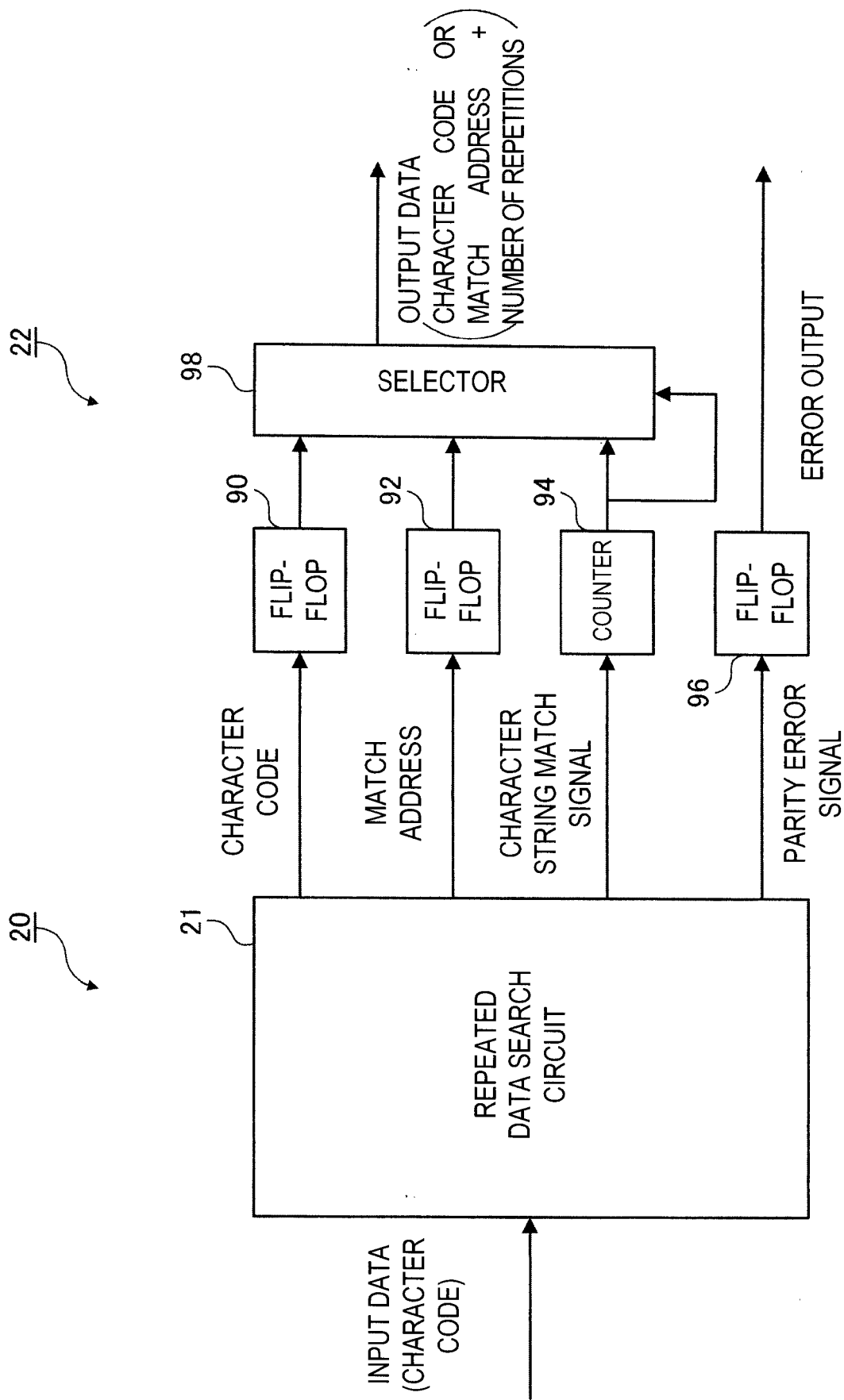


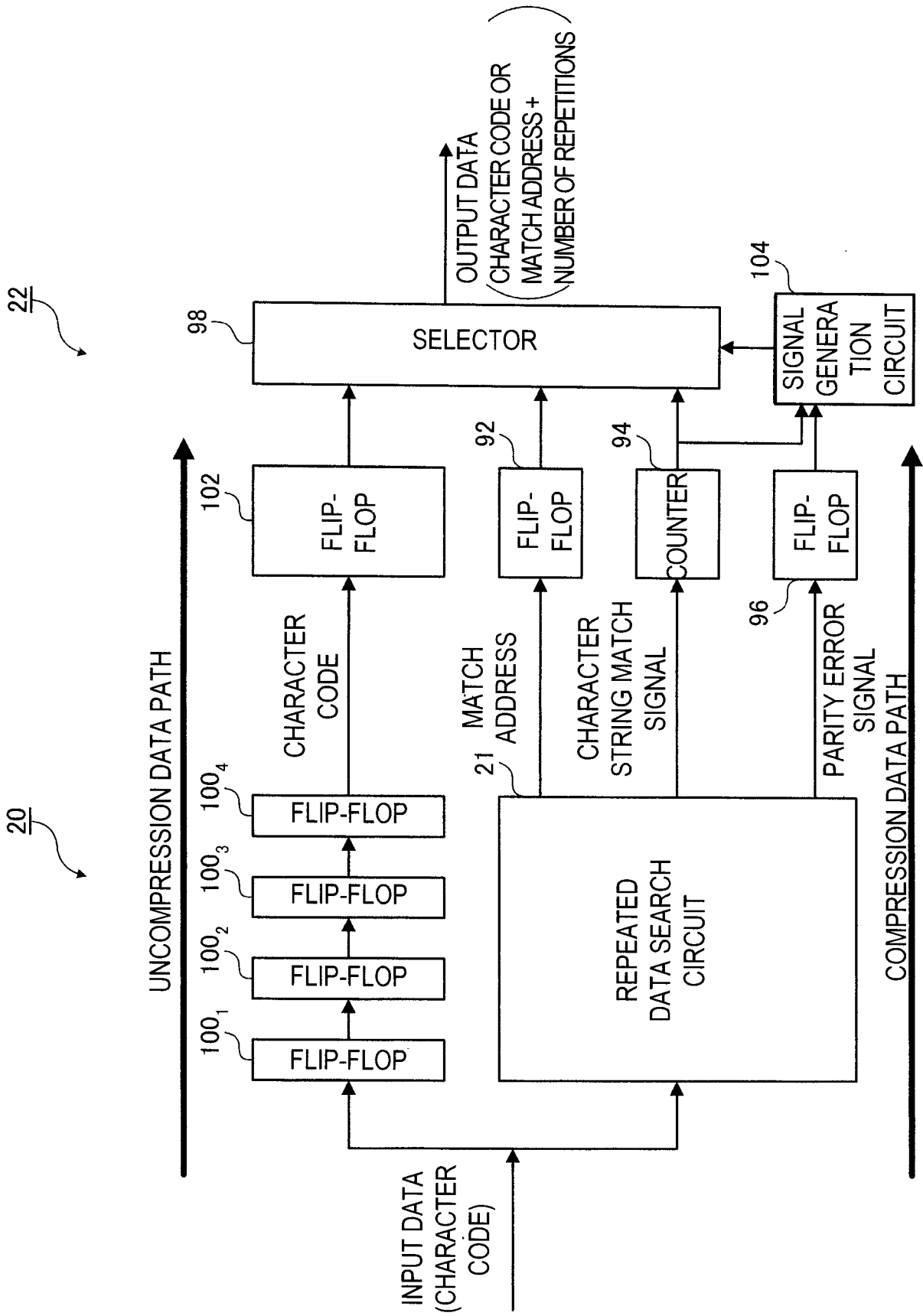






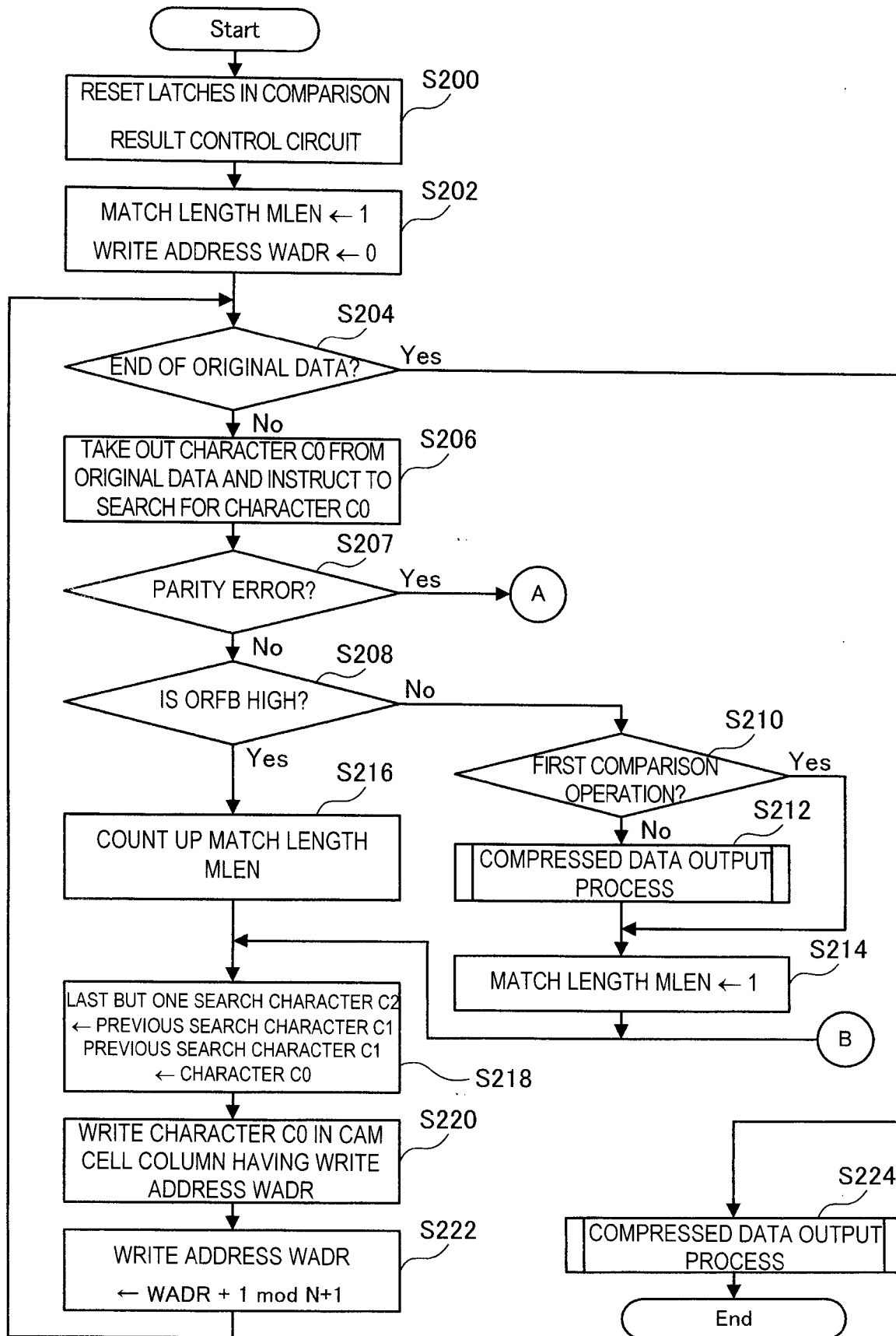


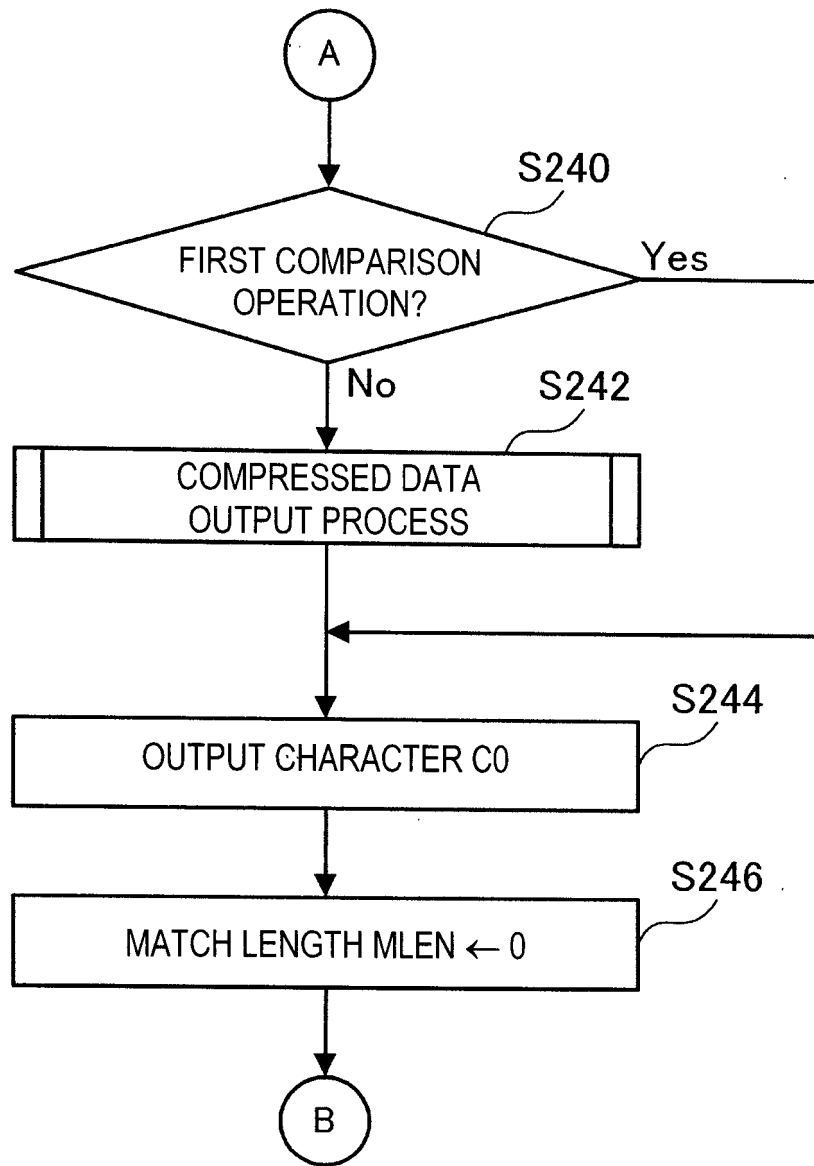


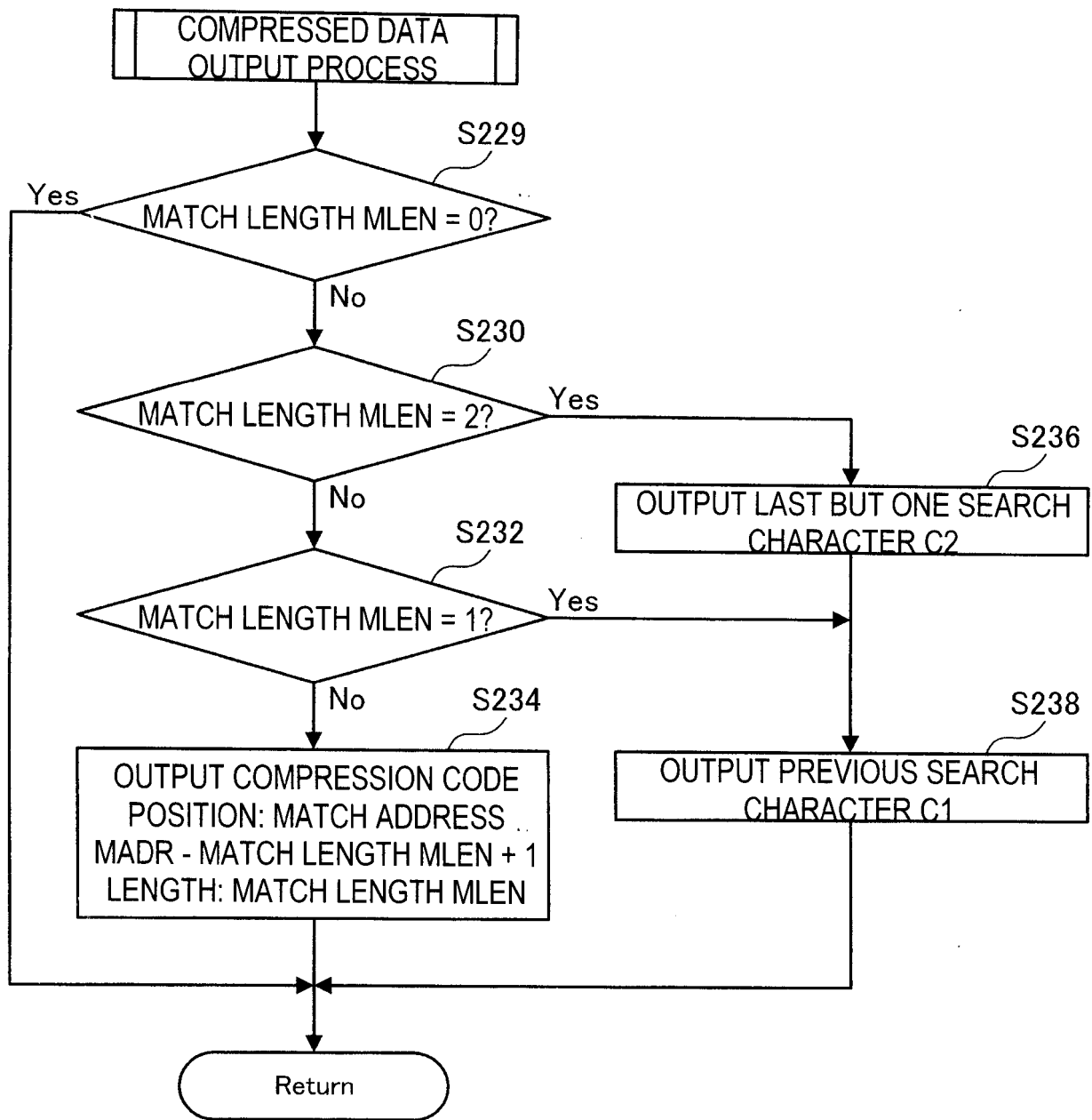


20

22







## Description

[Title of Invention]

DEVICE AND METHOD FOR PROCESSING DATA ELEMENT STRINGS

[Technical Field]

[0001]

The present invention relates to a device and method for processing data element strings and, in particular, to a device and method for processing a first partial data element string of a data element string on the basis of the position of a second partial data element string.

[Background Art]

[0002]

Deflate compression (RFC 1951) is a compression method on which ZLIB (RFC 1950) and GZIP (RFC 1952), which are currently widely used on computers, are based and uses LZ77 encoding. In the LZ77 encoding, a repeated portion of a character string in data is searched for and the character string is replaced with the position and length of the repeated portion to compress the data. For example, if a character string, "IBM is IBM", is encoded using the LZ77, the second occurrence of "IBM" is compressed since that portion is a repeated portion. Specifically, the portion is compressed by replacing it with a code such as "7, 3" which indicates that "a length of three characters that appears seven characters before is repeated". In this case, the longer the repeated portion, the higher the compression ratio is.

[0003]

In the specifications for the deflate compression, up to 32 K bytes of data preceding a character string is searched for a repeated occurrence of that character string, which requires an enormous amount of processing for comparing character strings for searching for repeated occurrences of character strings.

Accordingly, handling this processing by software requires a large amount of processing time. Hashing is usually used to reduce the search time by software. However, if many character strings have the same hash value, hashing can discard some of the character strings. That is, considering processing time and buffer capacity, it is difficult to find all of the character strings without omission.

[0004]

Therefore, the present applicant has proposed methods that can find all character strings by hardware without omission (see Patent Literatures 1 to 3, for example). In addition, the methods can find character strings considerably fast.

[0005]

In Patent Literature 1, when a search character string (ABCA) in a searched character string (BABCABB ...) stored in this order in content addressable memory cell columns is searched for, comparison operation with the first character (A) is made in all cell columns, comparison operation with the next character (B) is made only in cell columns having addresses (2) and (5) adjacent to the cell column in which a match has found previously, comparison operation with the next character (C) is similarly made only in the cell columns having addresses (3) and (6), and comparison operation with the last character (A) is similarly made only in the cell column having address (4), so that the search processing is completed in a short time.

[0006]

In Patent Literature 2, first switching means is provided between a portion of a match line opposite from the grounded end and a power supply with a switching element between them. The first switching means is turned on in a part of the cycle during which a content addressable memory turns on or off the switching element according to the result of comparison or during a part of a preparation cycle before comparison, thereby reducing the cycle during which a through-current is flowing to save power consumption.

[0007]

In Patent literature 3, the result of comparison of a search character in a write buffer with character data stored in each of the cell columns of a CAM is held in a first latch and a second latch in sequence, a signal generation circuit outputs the AND of an output from the first latch and an output from a third latch in the preceding stage to a priority encoder through the third latch and to an OR circuit when an input signal is low or outputs the AND of an output from the first latch and an output from the second latch in the preceding stage to the priority encoder through the third latch and to the OR circuit when an input signal is high, then first and second priority encoders output the OR of input signals, and a signal output from the OR circuit is input in each of signal generation circuits through a fourth latch and another OR circuit, thereby cutting the path through which the signal needs to pass through in one clock cycle in half to quicken processing.

[0008]

In these character string search methods, each of the bits that



make up a character is held in a memory cell of a content addressable memory (CAM). These memory cells are inherently more prone to errors called soft errors due to the influence of cosmic rays or other noise as the memory cells are more densely packed. Some conventional memories have error detection circuits such as parity check circuits in order to improve the reliability. The present applicant has proposed separately providing a parity bit for CAMs to perform parity error detection (see Patent Literature 4, for example).

[0009]

If an error occurs in a memory cell of a CAM, processing halts at that point in time. To avoid such a halt, it is actually desirable that an error correction circuit such as ECC be incorporated. However, a CAM needs an ECC circuit to be incorporated for each byte, which increases the circuit size too much. Therefore, the CAM in Patent Literature 4 incorporates only parity check circuits. Since the parity check circuits, if simply incorporated, increase the circuit size, the parity check circuits are creatively designed to minimize the circuit size.

[Citation List]

[Patent Literature]

[0010]

[Patent Literature 1]	JP7-114577A
[Patent Literature 2]	JP8-147986A
[Patent Literature 3]	JP8-242176A
[Patent Literature 4]	WO2010/050282

[Summary of Invention]

[Technical Problems]

[0011]

It is desirable that if a parity error occurs in the CAM described in Patent Literature 4, for example, during compression of a packet, processing of that packet can be continued. This also desirable in the case of an error in data stored in a CAM during processing that uses the CAM in general.

[0012]

An object of the present invention is to enable processing that is performed using a CAM unless an error occurs in data stored in the CAM to be continued even if an error occurs in the data stored in the CAM.

[Solution to Problems]

[0013]

To achieve the object, the present invention provides a device that processes a first partial data element string of a data element string on the basis of the position of a second partial data element string of the data element string and includes a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored, a first path providing an input data element to the content addressable memory as the search data element and outputting the address output from the content addressable memory, a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element, and a processing part

processing the first partial data element string by using the position of the second partial data element string determined by the address output from the first path if no error occurs in a match data element that matches the search data element when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and processing the first partial data element string by using the particular data element output from the second path if an error occurs in a match data element that matches the search data element.

[0014]

The second path may have a pipeline configuration that outputs the data element in synchronization with the output of the address from the first path.

[0015]

The present invention also provides a device that replaces a first partial data element string of a data element string with information indicating the position and length of a second partial data element string of the data element string to compress the data element string and includes a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored, a generation circuit generating, on the basis of the address output from the content addressable memory, string presence information indicating that a string of match data elements that matches a string of search data elements previously

provided is stored and column address information indicating an address at which the string of the match data elements is stored if the string of the match data elements is stored in the content addressable memory when the search data element is provided to the content addressable memory, a first path providing an input data element to the content addressable memory as the search data element and outputting the string presence information and the column address information generated by the generation circuit, a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element, and a processing part performing a process for replacing the first partial data element string with information indicating the position and length of the second partial data element string determined by the number of successive outputs of the string presence information from the first path and the column address information output from the first path if one condition that no error occurs in a match data element that matches the search data element is satisfied when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and performing a process using the particular data element output from the second path to prevent a part of the first partial data element string from being compressed if an error occurs in a match data element that matches the search data element.

[0016]

The processing part may perform the process for replacing with information indicating the position and length of the second partial data element string that is determined by the number of successive outputs and the column address information output from the first path if a further condition that the number of successive outputs of the string presence information from the

first path exceeds a predetermined number is satisfied in addition to the one condition when the first path provides the particular data element to the content addressable memory as the search data element.

The processing part may perform a process using the particular data element output from the second path to prevent a part of the first partial data element string from being compressed if the number of successive outputs of the string presence information from the first path does not exceeds a predetermined number when the first path provides the particular data element to the content addressable memory as the search data element.

The second path may have a pipeline configuration that outputs the data element in synchronization with the output of the string presence information and the column address information from the first path.

[0017]

The present invention also provides a device that replaces a first partial data element string of a data element string with information indicating the position and length of a second partial data element string of the data element string to compress the data element string and includes a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string and a parity element corresponding to the data element in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored, a generation circuit generating, on the basis of the address output from the content addressable memory, string presence

information indicating that a string of match data elements that matches a string of search data elements previously provided is stored and column address information indicating an address at which the string of the match data elements is stored if the string of the match data elements is stored in the content addressable memory when the search data element is provided to the content addressable memory, a determination circuit determining whether or not a value of the parity element stored at an address at which a match data element that matches the search data element is stored is generated from the search data element if the match data element is stored in the content addressable memory when the search data element is provided to the content addressable memory, a first path providing an input data element to the content addressable memory as the search data element and outputting the string presence information and the column address information generated by the generation circuit, a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element, and a processing part performing a process for replacing the first partial data element string with information indicating the position and length of the second partial data element string determined by the number of successive outputs and the column address information output from the first path if both of a condition that the determination circuit determines that a value of the parity element is generated and a condition that the number of successive outputs of the string presence information from the first path exceeds a predetermined number are satisfied when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and performing a process using the particular data element output from the second path to prevent the part of the particular data element of the first partial data element string from being compressed if at least one of a condition that the

determination circuit determines that a value of the parity element is not generated and a condition that the number of successive outputs of the string presence information from the first path does not exceeds the predetermined number is satisfied.

[0018]

The present invention also provides a method for processing a first partial data element string of a data element string on the basis of the position of a second partial data element string of the data element string. The method includes a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored, a first path providing an input data element to the content addressable memory as the search data element and outputting the address output from the content addressable memory, and a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element, and includes the steps of: determining whether or not an error has occurred in a match data element matching the search data element when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element; and if it is determined that no error has occurred in the match data element, using the address output from the first path to process the first partial data element string and, if it is determined that an error has occurred in the match data element, using the particular data element output from the second path to process the first

partial data element string.

[Advantageous Effect of Invention]

[0019]

According to the present invention, processing that is performed using a CAM unless an error occurs in data stored in the CAM can be continued even if an error occurs in the data stored in the CAM.

[Brief Description of Drawings]

[0020]

Figure 1 is a diagram illustrating an exemplary configuration of a microcomputer to which an embodiment of the present invention is applied;

Figure 2 is a block diagram illustrating an exemplary configuration of a repeated data search circuit in an embodiment of the present invention;

Figure 3 is a circuit diagram illustrating an exemplary configuration of a content addressable memory cell;

Figure 4 is a circuit diagram illustrating an exemplary configuration of a parity memory cell;

Figure 5 is a block diagram illustrating an exemplary configuration of a comparison result control circuit in an embodiment of the present invention;

Figures 6(a) to 6(e) are conceptual diagrams illustrating an operation of a comparison result control circuit in an embodiment of the present invention;



Figure 7 is a diagram illustrating an example of a parity error that occurs during a character string search;

Figure 8 is a diagram illustrating an exemplary configuration of a conventional repeated data search device and data compression controller;

Figure 9 is a diagram illustrating an exemplary configuration of a repeated data search device and a data compression controller in an embodiment of the present invention;

Figure 10-1 is a flowchart illustrating an exemplary operation of a data compression controller in an embodiment of the present invention;

Figure 10-2 is a flowchart illustrating an exemplary operation of the data compression controller in an embodiment of the present invention; and

Figure 11 is a flowchart illustrating details of a compressed data output process in the flowchart of Figure 10.

[Description of Embodiment]

[0021]

Embodiments of the present invention will be described below in detail with reference to the accompanying drawings.

Figure 1 illustrates (a portion of) a configuration of a microcomputer 10 to which an embodiment is applied. The microcomputer 10 includes a CPU 12, a ROM 14, and a RAM 16, which are interconnected through a bus 18 including an address bus, a data bus, and a control bus.

[0022]

The microcomputer 10 also includes a plurality of repeated data search devices 20a, 20b, ... according to the present embodiment, which are connected to the bus 18 through a data compression controller 22 that controls operations of the repeated data search devices. Note that the repeated data search devices 20a, 20b, ... have the same configuration and  $P + 1$  repeated data search devices are provided in the present embodiment. The  $P + 1$  repeated data search devices will be hereinafter collectively referred to as the "repeated data search devices 20" and a configuration of the repeated data search devices 20 will be described. The individual repeated data search devices will be distinguished from one another by referring to them as "block 0" to "block P".

[0023]

Figure 2 is a diagram illustrating a repeated data search circuit 21 which constitutes a part of the repeated data search devices 20 in the present embodiment.

The repeated data search circuit 21 includes a CAM cell array 26 having a matrix of content addressable memory cells (hereinafter the content addressable memory is abbreviated as CAM) 28 indicated by thin circles in the figure. The CAM cell array 26 has a matrix of  $N + 1$  word lines  $WL_0$  to  $WL_N$ , a matrix of  $N + 1$  match lines  $MATCH_0$  to  $MATCH_N$ , and a matrix of  $M + 1$  pairs of bit lines  $BL_0, BL_0'$  to  $BL_M, BL_M'$ . Each of the CAM cells 28 is connected to any one of the word lines  $WL_0$  to  $WL_N$ , any one of the match lines  $MATCH_0$  to  $MATCH_N$ , and any one of the pairs of bit lines  $BL_0, BL_0'$  to  $BL_M, BL_M'$ .

[0024]

The multiple CAM cells 28 have the same configuration. As

illustrated in Figure 3, each CAM cell 28 includes two NOT circuits 30 and 32. Each of the NOT circuits 30 and 32 has an input terminal connected to an output terminal of the other of the NOT circuits 30 and 32. The loop of the NOT circuits 30 and 32 constitutes a memory circuit (SRAM) which stores 1 bit of data. The output terminal of the NOT circuit 30 is connected to the source of an N-MOSFET 34, the output terminal of the NOT circuit 32 is connected to the source of an N-MOSFET 36, and the gate of each of the N-MOSFETs 34 and 36 is connected to a word line WL. The drains of the N-MOSFETs 34 and 36 are connected to bit lines BL and BL', respectively.

[0025]

Here, when 1 bit of data is to be written in the CAM cell 28, the word line WL is driven high, the bit line BL is driven to a level corresponding to the data D to be written (high when D is "1" or low when D is "0"), and the bit line BL' is driven to the level that is the inverse of that of the bit line BL (low when the bit line BL is high or high when the bit line BL is low). As a result, the N-MOSFETs 34 and 36 turn on and the data provided through the bit line BL or BL' is held in the loop of the NOT circuits 30 and 32.

[0026]

The output terminal of the NOT circuit 30 is connected to the gate of an N-MOSFET 38 and the output terminal of the NOT circuit 32 is connected to the gate of an N-MOSFET 40. Note that data (Q') that is the inverse of the data D is output from the NOT circuit 32 and data Q (= D) that is the inverse of data Q' is output from the NOT circuit 30. The drains of the N-MOSFETs 38 and 40 are connected to the bit lines BL' and BL, respectively, and the sources are connected to the gate of the N-MOSFET 42. The drain of the N-MOSFET 42 is connected to a match line MATCH and the source is grounded.

[0027]

Note that,  $M + 1$  CAM cells 28, for example, are connected to the same word line and the same match line in the present embodiment so that  $M + 1$  bits of data can be stored in the plurality of CAM cells 28 connected to the same word and match lines (hereinafter these CAM cells will be referred to as the CAM cell column). An address is assigned to each CAM cell column in the CAM cell array 26.

[0028]

$N + 1$  parity memory cells 24 indicated by thick circles in the figure are arranged in a column in the CAM cell array 26.  $N + 1$  read parity lines RPL0 to RPLN and one pair of write parity lines WPL and WPL' are provided in the CAM cell array 26 and each parity memory cell 24 is connected to any one of the word lines WL0 to WLN and any one of the read parity lines RPL0 to RPLN and to the pair of write parity lines WPL and WPL'.

[0029]

The  $N + 1$  parity memory cells 24 have the same configuration. As illustrated in Figure 4, each parity memory cell 24 has a configuration similar to that of the CAM cell 28 in Figure 3, except that the parity memory cell 24 does not have the N-MOSFETs 38, 40 and 42 included in the CAM cell 28 and the output terminal of the NOT circuit 30 is connected to a read parity line RPL so that data held in the loop of the NOT circuits 30 and 32 is directly read.

Note that an address is assigned to each parity memory cell 24 in the CAM cell array 26.

[0030]

Returning to Figure 2, the repeated data search circuit 21 includes a timing controller 50. The timing controller 50 is connected to the data compression controller 22. When a search instruction SEARCH is input from the data compression controller 22 into the timing controller 50, the repeated data search circuit 21 enters a "search" mode in which the repeated data search circuit 21 repeatedly compares input search data with data stored in each CAM cell column of the CAM cell array 26 and then writes the search data in any of the CAM cell columns of the CAM cell array 26.

[0031]

A system clock CLOCK from the data compression controller 22 is input into the timing controller 50 and the timing controller 50 generates and outputs a clock SR that is in synchronization with the system clock CLOCK. The clock SR is output to a write buffer 56 connected to the timing controller 50.

[0032]

The write buffer 56 is connected to the data compression controller 22 and search data is sequentially input in the write buffer 56 from the controller 22. The pairs of bit lines BL0, BL0' to BLM, BLM' are connected to the write buffer 56 so that the write buffer 56 holds input search data and changes the levels of the bit line pairs BL0, BL0' to BLM, BLM' (drives the bit line pairs) according to values of the held search data in synchronization with the clock SR input from the timing controller 50. While the write buffer 56 is driving each bit line pair according to given search data, data stored in each CAM cell column is compared with the given search data and the given search data is written in a given CAM cell column.

[0033]

A parity generator 54 is connected to the write buffer 56 and search data is sequentially input in the parity generator 54 from the write buffer 56. The pair of write parity lines WPL and WPL' are connected to the parity generator 54. The parity generator 54 calculates a parity bit on the basis of input search data and changes the level of the write parity line pair WPL and WPL' (drives the bit line pair) according to the calculated parity bit in synchronization with the clock SR input from the timing controller 50. While the parity generator 54 is driving the bit line pair according to the parity bit, the parity bit is written in a predetermined parity memory cell 24. A search parity line SPL is also connected to the parity generator 54 and the parity generator 54 changes the level of the search parity line SPL according to the calculated parity bit.

[0034]

An address decoder 52 is connected to the timing controller 50 and the clock SR is output to the address decoder 52 as well. The address decoder 52 is connected to the data compression controller 22 and the address of a CAM cell column (a write address WADR) in which search data is to be written from the data compression controller 22 is specified. The word lines WL0 to WLN are connected to the address decoder 52. In synchronization with the clock SR, more specifically, at a timing in a period during which the write buffer 56 is driving each bit line pair according to given search data after comparison of the given search data with the data stored in each CAM cell column, a word line corresponding to a specified write address WADR is asserted (enabled) to write the given search data into a CAM cell column corresponding to the specified address and a parity bit calculated from the given search data is written in a parity memory cell 24 corresponding to the specified address.

[0035]

$N + 1$  match line controllers  $58_0$  to  $58_N$  connected to the match lines MATCH0 to MATCHN are also connected to the timing controller 50 and the clock SR is also output to each of the match line controllers  $58_0$  to  $58_N$ . The match line controllers  $58_0$  to  $58_N$  charge (precharge) the match lines MATCH0 to MATCHN to a high level on the basis of the input clock SR prior to comparison of search data with data stored in each CAM cell column. The precharge of the match lines by the match line controllers  $58_0$  to  $58_N$  is performed in a period between the end of the driving of each bit line pair according to the given search data by the write buffer 56 and the start of driving of each bit line pair according to next search data.

[0036]

The match lines MATCH0 to MATCHN are connected to a comparison result control circuit 60. As illustrated in Figure 5, the comparison result control circuit 60 includes latches  $62_0$  to  $62_N$ , signal generation circuits  $64_0$  to  $64_N$  and latches  $66_0$  to  $66_N$ . Note that the match line controllers  $58_0$  to  $58_N$  are omitted from Figure 5. While a detailed configuration of only the signal generation circuit  $64_1$  connected to the match line MATCH1 is depicted in Figure 5 among the signal generation circuits  $61_0$  to  $64_N$ , the other signal generation circuits have the same configuration as the signal generation circuit  $64_1$  and only a circuit portion associated with the match line MATCH1 in the comparison result control circuit 60 will be described below.

[0037]

The match line MATCH1 is connected to the input terminal of the latch  $62_1$  and an output terminal of the latch  $62_1$  is connected

to one of two input terminals of each of AND circuits 68<sub>1</sub> and 70<sub>1</sub> of the signal generation circuit 64<sub>1</sub>. The output terminal of an OR circuit 72<sub>1</sub> is connected to the other of the two input terminals of the AND circuit 68<sub>1</sub> and the other of the two input terminals of the OR circuit 72<sub>1</sub> and the other of the two input terminals of the AND circuit 70<sub>1</sub> are connected to the output terminal of a latch 66<sub>0</sub> in the preceding stage. The output terminal of the AND circuit 68<sub>1</sub> is connected to the input terminal of a latch 66<sub>1</sub> and the output terminal of the latch 66<sub>1</sub> is connected to the input terminals of an OR circuit 72<sub>2</sub> and an AND circuit 70<sub>2</sub>, not depicted, of a signal generation circuit 64<sub>2</sub> in the subsequent stage. The output terminal of the latch 66<sub>1</sub> is also connected to the input terminal of a first priority encoder 74. Furthermore, the output terminal of the AND circuit 70<sub>1</sub> is connected to one input terminal of an OR circuit 78 and the output terminals of AND circuits 70<sub>0</sub> and 70<sub>2</sub> to 70<sub>N</sub>, not depicted, are connected to the other input terminals of the OR circuit 78. The output terminal of the OR circuit 78 is connected to one input terminal of an OR circuit 80 and the output terminal of the OR circuit 78 of another repeated data search circuit 21 is connected to the other input terminal of the OR circuit 80. The output terminal of the OR circuit 80 is connected to the input terminals of OR circuits 72<sub>0</sub> to 72<sub>N</sub> of the signal generation circuits 64<sub>0</sub> to 64<sub>N</sub> through an inverter, not depicted (only the OR circuit 72<sub>1</sub> is depicted), and a feedback signal ORFB output from the OR circuit 80 is inverted and input in the OR circuits 72<sub>0</sub> to 72<sub>N</sub>. The feedback signal ORFB output from the OR circuit 80 is also output to the data compression controller 22 (see also Figure 2).

[0038]

Note that Figure 5 illustrates the comparison result control circuit 60 of the repeated data search circuit 21 of block 0 and one of the two input terminals of the OR circuit 72<sub>0</sub> of the signal generation circuit 64<sub>0</sub> is connected to the output



terminal of the latch  $66_N$  of the comparison result control circuit 60 of block P. Furthermore, the output terminal of the latch  $66_N$  of the comparison result control circuit 60 of block 0 is connected to one of the two input terminals of the OR circuit  $72_0$  of the comparison result control circuit 60 of block 1 (see also Figure 2).

[0039]

An output terminal of the first priority encoder 74 is connected to an input terminal of a second priority encoder 76. The first priority encoder 74 encodes an address corresponding to an input signal that has gone high after a "search" operation among signals input from the latches  $66_0$  to  $66_N$  of the comparison result control circuit 60, outputs the encoded address as a match address MADR0 to the second priority encoder 76, and also outputs the OR of the signals input from the latches  $66_0$  to  $66_N$  to the second priority encoder 76 as a match signal MSIG0. Note that if more than one signal is high among the signals input from the latches  $66_0$  to  $66_N$ , the first priority encoder 74 outputs the address of a content addressable memory column that has a higher priority according to a predetermined criterion, for example the lowest address or the address that is closest to the current write address WADR, as a match address MADR0. Note that the reason why the latter address is output is that the amount of distance information required for compression is small and consequently the compression ratio increases.

[0040]

Match addresses MADR and match signals MSIG that are output from the first priority encoder 74 of the other repeated data search circuits 21 are also input into input terminals of the second priority encoder 76. The output terminals of the second priority encoder 76 are connected to the data compression

controller 22.

[0041]

The second priority encoder 76 has a configuration similar to that of the first priority encoder 74. The second priority encoder 76 outputs an address that has a higher priority according to a predetermined criterion among addresses MADR<sub>0</sub> to MADR<sub>P</sub> input from a plurality of first priority encoders 74 that have MSIG<sub>0</sub> to MSIG<sub>P</sub> at the high level, for example the lowest address, plus the address of that block (the address assigned to each of the plurality of repeated data search devices), or the address that is closest to the current write address WADR to the data compression controller 22 as a match address MADTR and outputs a signal representing the OR of match signals MSIG<sub>0</sub> to MSIG<sub>P</sub> input from the plurality of first priority encodes 74 to the data compression controller 22 as a match signal MSIG.

[0042]

The comparison result control circuit 60 also includes EXOR circuits 82<sub>0</sub> to 82<sub>N</sub> and NAND circuits 84<sub>0</sub> to 84<sub>N</sub>. Only a circuit portion associated with the read parity line RPL<sub>1</sub> among these circuits will be described below.

[0043]

The read parity line RPL<sub>1</sub> is connected to one of the two input terminals of the EXOR circuit 82<sub>1</sub> and a search parity line SPL is connected to the other of the input terminals of the EXOR circuit 82<sub>1</sub>. The output terminal of the EXOR circuit 82<sub>1</sub> is connected to one of the two input terminals of a NAND circuit 84<sub>1</sub> and the other of the two input terminals of the NAND circuit 84<sub>1</sub> is connected to the output terminal of the latch 62<sub>1</sub>. The output terminal of the NAND circuit 84<sub>1</sub> is connected to one input terminal of an OR circuit 86 through an inverter

and the output terminals of NAND circuits  $84_0$  and  $84_2$  to  $84_N$  are connected to the other input terminals of the OR circuit 86 through an inverter. The output terminal of the OR circuit 86 is connected to one input terminal of an OR circuit 88 and the output terminals of OR circuits 86 of other repeated data search circuits 21 are connected to the other input terminals of the OR circuit 88. A parity error signal PE output from the OR circuit 88 is output to the data compression controller 22 (see also Figure 2).

[0044]

A comparison operation in the repeated data search circuit 21 will be described first as an operation of the present embodiment. When data to be compressed (original data) is compressed, the data compression controller 22 takes out a unit data having a predetermined bit length from the original data in sequence as search data as will be described later and outputs the search data to the repeated data search circuit 21 in sequence along with a search instruction SEARCH and a write address WADR.

[0045]

When the search instruction SEARCH is input in the repeated data search circuit 21, the timing controller 50 in the repeated data search circuit 21 outputs the clock SR to the write buffer 56, the address decoder 52 and the match line controllers  $58_0$  to  $58_N$  and the match line controllers 58 precharge the match lines MATCH0 to MATCHN in preparation for comparison operation in the CAM cell array 26, and the write buffer 56 holds input search data and drives the bit line pairs BL0, BL0' to BLM, BLM' according to the search data.

[0046]

In each CAM cell 28 of the CAM cell array 26, the N-MOSFET 38 is on when data Q output from the NOT circuit 30 is "1" (high) or the N-MOSFET 40 is on when data Q' output from the NOT circuit 32 is "1". Accordingly, if the data Q (Q') held in the loop of the NOT circuit 30 and 32 matches data D (D') provided through the bit line pairs BL, BL', the N-MOSFET 42 is not turned on; if they do not match, a current flows from the drain of one of the N-MOSFETs 38 and 40 that is turned on to the source to turn the N-MOSFET 42 on. This lowers the level of the precharged match line MATCH to low (discharge).

[0047]

The data comparison described above is performed in the CAM cells 28 simultaneously. Since a plurality of CAM cells 28 (CAM cell column) are connected to one match line MATCH, the comparison operation of comparing input character data with character data stored in the CAM cell column is completed almost in the first half of one cycle of the clock SR and each match line is held high only when the N-MOSFETs 42 do not turn on in all of the CAM cells 28 connected to the match line, that is, only when the character data stored in the CAM cell column matches the character data input in the write buffer 56. When they do not match, the match line is driven low.

[0048]

In the latter half of the clock SR cycle, the word line of the CAM cell column corresponding to the input write address WADR is asserted (enabled) by the address decoder 52 and the search data is written in that CAM cell column. When the write of the search data in the CAM cell column is completed and the word line is driven low and the write buffer 56 stops driving each bit line pair, the match line controller 58 causes the match lines to be precharged again.

[0049]

The operation described above (comparison operation) is performed in one cycle of the clock SR and is repeated in synchronization with inputs of search data, a search instruction SEARCH and a write address WADR from the data compression controller 22.

[0050]

In this operation, the repeated data search circuit 21 also checks the parity of the character data stored in each CAM cell column. The parity check operation will be described.

The EXOR circuit 82 compares the level of the search parity line SPL output from the parity generator 54 and the level of the read parity line RPL output from the parity memory cell 24 and outputs a low-level signal if they match or a high-level signal if they do not match.

[0051]

When a signal output from the latch 62 is driven high to indicate a data match and the signal output from the EXOR circuit 82 is driven low to indicate a parity bit match, or when the signal output from the latch 62 is driven low to indicate a data mismatch, the NAND circuit 84 outputs a high-level signal.

On the other hand, when the signal output from the latch 62 is driven high to indicate a data match and the signal output from the EXOR circuit 82 is driven high to indicate a parity bit mismatch, the NAND circuit 84 outputs a low-level signal.

[0052]

If at least one of the signals output from the NAND circuits 84<sub>0</sub> to 84<sub>N</sub> is low, the OR circuit 86 outputs a high-level signal and the OR circuit 88 outputs a parity error signal PE to the data compression controller 22.

[0053]

Note that the character data in the foregoing description is one example of a data element and the parity bit is one example of a parity element. The signal when ORFB is "1" is one example of string presence information indicating that a string of match data elements that matches a string of search data elements is stored, MADR is one example of column address information indicating an address in which the match data element string that matches the search data element string is stored, and the portion including the signal generation circuits 64, the latches 66, the first priority encoder 74, the second priority encoder 76, the OR circuits 78 and 80 is one example of a generation circuit. Furthermore, the portion including the latches 62, the EXOR circuit 82, the NAND circuit 84, the OR circuits 86 and 88 is one example of a determination circuit.

[0054]

An operation of the comparison result control circuit 60 will be described below with reference to Figure 6. Figure 6 illustrates an example in which original data is text data, a character data string, "ABABBC", is already stored in the CAM cell columns with addresses "0" to "5" in sequence, and character data, "ABBBC ...", is input in sequence as search data. Latches 62<sub>0</sub> to 62<sub>5</sub> are referred to as ML<sub>0</sub> to ML<sub>5</sub>, and latches 66<sub>0</sub> to 66<sub>5</sub> are referred to as PS<sub>0</sub> to PS<sub>5</sub>. Furthermore, it is assumed in the following description that comparison operations in the other repeated data search devices have resulted in "mismatches" in succession.

[0055]

As illustrated in Figure 6(a), when character data "A" is input first as search data, comparisons in the CAM cell columns with addresses "0" and "2" (indicated by thick-line boxes in Figure 6) result in a "mismatch", the match lines other than match lines MATCH0 and MATCH2 among the match lines MATCH0 to MATCH5 are discharged to low, and only match lines MATCH0 and MATCH2 are held high. The levels of the match lines MATCH0 to MATCH5 are held in the latches ML0 to ML5, respectively, of the comparison result control circuit 60 and are then output to the signal generation circuits 64<sub>0</sub> to 64<sub>5</sub> in the next cycle.

[0056]

As illustrated in Figure 6(a), since the level held in each of the latches PS0 to PS5 is low at this point in time ("0" in the Figure 6; the same applies to latches PS6 to PSN, not shown, and the latches PS0 to PSN in the other blocks), signals output from the AND circuits 70<sub>0</sub> to 70<sub>5</sub> are low and a feedback signal ORFB0 output from the OR circuit 78 and a feedback signal ORFB output from the OR circuit 80 are also low. In this way, in the present embodiment, the fact that only one character is matched, that is, the presence of a CAM cell column in which a "match" is found in one comparison operation is not enough to drive the feedback signal ORFB high. Furthermore, since the feedback signal ORFB is low, outputs from the OR circuits 72<sub>0</sub> to 72<sub>5</sub> go high and the levels of signals output from the latches ML0 to ML5 are held unchanged in the latches PS0 to PS5 (see Figure 6(b)).

[0057]

As illustrated in Figure 6(b), when character data "B" is input as the next search data, comparisons in the CAM cell columns

with addresses "1", "3" and "4" result in a "match" and the levels of the match lines (only the match lines MATCH 1, 3 and 5 are high) are held in the latches ML0 to ML5. In the next cycle, outputs from the latches ML1, ML3 and ML5 go high and outputs from the latches PS0 and PS2 (that is, the latches PS in the stage preceding the latches ML1 and ML3) go high, and therefore signals output from the AND circuits 70<sub>1</sub> and 70<sub>3</sub> go high and a feedback signal ORFB0 output from the OR circuit 78 and a feedback signal ORFB output from the OR circuit 80 go high.

[0058]

In this way, the feedback signal ORFB goes high only when two or more successive characters have matches, that is, only when a data string made up of a plurality of pieces search data input in succession is stored in the CAM cell array 26. Note that if the first priority encoder 74 is configured to output the lowest address as a match address MADR0, the match address MADR0 is address "0" which corresponds to PS0. The value of a lower bit of a match address MADR output from the second priority encoder 76 depends on the value of match addresses input from the other repeated data search devices, and is equal to the value of the match address MADR0 if the data does not have a match in the other repeated data search devices. The values of an upper bit of the match address MADR is equal to the address of the encoder block (the address assigned to each of the plurality of repeated data search devices) in which the match has been found. In this case, the block is the block with address 0 and therefore the values are all 0s.

[0059]

As illustrated in Figure 6(c), when character data "B" is input as the next search data, the same match lines that have become high previously go high, the levels of the match lines are



first held in the latches ML0 to ML5, and then output to the signal generation circuits 64<sub>0</sub> to 64<sub>5</sub>, respectively. Since only the latch ML4 among the latches ML1, ML3 and ML4 that output high-level signals has the preceding latch PS that also outputs a high-level signal at this time, only a signal output from the AND circuit 70<sub>4</sub> among signals output from the AND circuits 70<sub>1</sub>, 70<sub>3</sub> and 70<sub>4</sub> goes high. Accordingly, the feedback signal ORFB0 output from the OR circuit 78 is held high and the match address MADR0 is address "4" which corresponds to PS4.

[0060]

Furthermore, since the feedback signal ORFB is held high, a signal equivalent to the AND of an output ML(n, m) from the latch ML with address n and an output PS(n-1, m) of the latch PS in the preceding stage is output to the latch PS with address n, where m is the current cycle. As a result, only the latch PS4 holds the level high.

[0061]

Then, as illustrated in Figure 6(d), character data "B" is input as the next search data. In this case, the levels of the match lines held in the latches ML0 to ML5 first and then input in the signal generation circuits 64<sub>0</sub> to 64<sub>5</sub>, respectively, are the same as the levels in the previous cycle. However, for all the latches ML1, ML3 and ML4 that output high-level signals, signals output from their preceding latches PS are low and therefore signals output from the AND circuits 70<sub>0</sub> to 70<sub>5</sub> are low and feedback signals ORFB0 and ORFB are also low. When the feedback signal ORFB goes low, signals output from the OR circuits 72<sub>0</sub> to 72<sub>5</sub> of the signal generation circuits 64<sub>0</sub> to 64<sub>5</sub> go high and therefore the levels of signals output from the latches ML1 to ML5 are held in the latches PS0 to PS5 as is.

[0062]

As illustrated in Figure 6(e), when character data "C" is input as the next search data, only the match line MATCH5 goes high, the levels of the match lines are first held in the latches ML<sub>0</sub> to ML<sub>5</sub> and then output to the signal generation circuits 64<sub>0</sub> to 64<sub>5</sub>, respectively. Since only the latch ML5 outputs a high-level signal and its preceding latch PS4 also outputs a high-level signal, only a signal output from the AND circuit 70<sub>5</sub> among signals output from the AND circuits 70<sub>0</sub> to 70<sub>5</sub> is high. Accordingly, the feedback signal ORFB0 output from the OR circuit 78 is held high and the match address MADR0 is address "1" which corresponds to PS1.

[0063]

In this way, the repeated data search circuit 21 can search for a match with one character in one clock cycle and enables a character string search that is faster by far than any other methods. Specifically, each time one character is input, the input character can be compared with all of the characters in the CAM cell array 26 simultaneously, which enables a fast search.

[0064]

Here, consider a situation where a parity error has occurred in the repeated data search circuit 21.

Figure 7 illustrates a parity error occurring in a repeated data search operation by a repeated data search circuit 21.

Assume that a character string, "IBMisIB", is input in sequence in a CAM cell array 26. Also assume that parity bits (represented by "P") each of which is associated with each of the characters are stored in the CAM cell array 26. Note that the numbers that appear to the left of the CAM cell array 26 in

the figure represent the addresses of the cell columns of the CAM and correspond to the values N of the word lines WLN (match lines MATCHN and read parity lines RPLN).

[0065]

When the next character string, "IBMi", is input to the write buffer 56 in the repeated data search circuit 21 in this state, one character at a time, the repeated data search circuit 21 sequentially compares the characters with the character strings stored in the CAM cell array 26. Specifically, when the first character, "I", is input in the write buffer 56, the match line MATCH0 is held high because the character data in address 0 matches the character. When the second character, "B", is input in the write buffer 56, the match line MATCH1 is held high because the character data in address 1 matches the character. When the third character, "M", is input in the write buffer 56, the match line MATCH2 is held high because the character data in address 2 matches the character.

[0066]

Assume that when the fourth letter, "i", is input in the write buffer 56, a parity error (indicated by the "x" mark in the figure) is detected while the match line MATCH3 is held high. The cause may be that the arrangement of bits of an original, different character has become equal to the arrangement of bits of the character "i" because of a bit-flip caused by noise, for example. Accordingly, the match with the character "i" is unreliable. That is, data in the CAM cell array 26 is unreliable. Therefore, the process can be no longer continued and the process by the repeated data search circuit 21 needs to be aborted at this point in time.

In this case, typically, all the process is performed again from the beginning. Specifically, the parity error is reported

to the data source to request the data source to resend the data from the beginning.

If the parity error is a temporary error caused by noise, the resend of the data enables recovery from the error. However, if the error is not a temporary error but a persistent parity error due to a bit corruption or the like of the CAM cell array 26, recovery is impossible because it is a failure of the CAM cell array 26.

[0067]

A configuration that performs the process again from the beginning in the event of a parity error will be described below.

Figure 8 is a diagram illustrating (a part of) an example of a configuration of a repeated data search device 20 and a data compression controller 22 configured to perform this.

As illustrated, the repeated data search device 20 includes a repeated data search circuit 21 described with reference to Figures 2 to 6. The data compression controller 22 includes a flip-flop 90 that holds character data (a character code) output from the repeated data search circuit 21, a flip-flop 92 that holds a match address (MADR in Figure 2) output from the repeated data search circuit 21, a counter 94 that counts successive outputs of a character string match signal (ORFB in Figure 2) from the repeated data search circuit 21, and a flip-flop 96 that holds a parity error signal (PE in Figure 2) output from the repeated data search circuit 21. The data compression controller 22 further includes a selector 98 that selects and outputs one of character data held by the flip-flop 90 and a set of a match address held by the flip-flop 92 and a count of the counter 94 on the basis of the count of the counter 94.

[0068]

Operations of the repeated data search device 20 and the data compression controller 22 that have the exemplary configuration will be described below.

First, character data is input in the repeated data search circuit 21, one byte at a time. Note that although not clearly depicted in the figure, the character data is input from the data compression controller 22. The repeated data search circuit 21 sequentially writes the input character data in the CAM, compares the input character data with all pieces of character data written so far and, if there is a match piece of character data, outputs a "1" as a character string match signal. However, the repeated data search circuit 21 does not output a "1" as the character string match signal when only one character has a match; it outputs a "1" when two or more successive characters have matches. The repeated data search circuit 21 continues to output a "1" while there is a match with a character of the character string. That is, if a "1" is output during one clock cycle and then a "0" is output in the next clock cycle as the character string match signal, it indicates matches of two characters. If "1"s are output during two clock cycles and a "0" is output in the next clock cycle as the character string match signal, it indicates matches of three characters. In this way, if "1"s are output in K clock cycles and a "0" is output in the next clock cycle, it indicates matches of K + 1 characters. Accordingly, the data compression controller 22 can count outputs of the character string match signal with the counter 94 to detect how many characters have matches.

[0069]

The repeated data search circuit 21 outputs an address of a

character data match in the CAM as a match address and the flip-flop 92 holds the match address.

Furthermore, the repeated data search circuit 21 outputs input character data regardless of whether or not there is a match with the character data, and the flip-flop 90 holds the character data. The character data is required as uncompressed data if there is not a match with the character data.

[0070]

When there is not a match with the character data, the selector 98 outputs the character data held in the flip-flop 90; when there is a match with the character data, the selector 98 outputs the match address held in the flip-flop 92 and the count (the number of successive matches with the character data) by the counter 94.

[0071]

If a parity error occurs in the repeated data search circuit 21 in this exemplary configuration, the repeated data search circuit 21 outputs a "1" to the data compression controller 22 as a parity error signal. This allows the data compression controller 22 to know that the character string match signal and the character code are unreliable. In that case, the compression process cannot be continued and therefore data to be compressed is input anew.

[0072]

To avoid this, the present embodiment enables the reliability of compressed data to be ensured even if a parity error occurs in a repeated data search circuit 21 and the process by the repeated data search circuit 21 halts.

Figure 9 is a diagram illustrating (a part of) an exemplary configuration of a repeated data search device 20 and a data compression controller 22 in the present embodiment.

As illustrated, the repeated data search device 20 includes flip-flops  $100_1$ ,  $100_2$ ,  $100_3$  and  $100_4$  in addition to the components in the configuration in Figure 8. The data compression controller 22 includes, instead of the flip-flop 90 in Figure 8, a flip-flop 102 that holds character data output from the flip-flop  $100_4$  and a signal generation circuit 104 generating a signal to be output to a selector 98 on the basis of a count of the counter 94 and a parity error signal from the flip-flop 96. In the present embodiment, the data compression controller 22 is provided as an example of a processing part.

[0073]

The compression data path in the figure is an example of a first path for performing compression based on the result of search by the repeated data search circuit 21. The uncompression data path, on the other hand, is an example of a path through which input character data is passed without change. That is, the two paths are provided in the present embodiment to provide redundancy for the repeated data search device 20.

[0074]

Operations of the repeated data search device 20 and the data compression controller 22 in the present embodiment will be described below.

The operations are similar the operations in Figure 8 until the repeated data search circuit 21 outputs a match address and a character string match signal, the flip-flop 92 holds the match address, and the counter 94 counts successive matches.

[0075]

However, in the present embodiment, the flip-flop 102 holds character data that passed through the flip-flops 100<sub>1</sub>, 100<sub>2</sub>, 100<sub>3</sub> and 100<sub>4</sub> provided outside the repeated data search circuit 21, rather than character data that passed through the repeated data search circuit 21 as in Figure 8.

[0076]

If a parity error occurs in the repeated data search circuit 21 and the repeated data search circuit 21 outputs a parity error signal, the signal generation circuit 104 outputs a signal instructing the selector 98 to select data passing through the uncompression data path to the selector 98. In response to this, the selector 98 selects the character data held in the flip-flop 102 and outputs the character data to a processing part of the data compression controller 22.

[0077]

On the other hand, if no parity error occurs in the repeated data search circuit 21 and the repeated data search circuit 21 does not output the parity error signal, the signal generation circuit 104 outputs a signal illustrating the selector 98 to select data passing through the uncompression data path to the selector 98 if the number of successive match characters is less than or equal to 2. In response to this, the selector 98 selects the character data held in the flip-flop 102 and outputs the character data to the processing part of the data compression controller 22. If the number of successive match characters is greater than or equal to 3, the signal generation circuit 104 outputs a signal instructing the selector 98 to select data passing through the compression data path to the selector 98. In response to this, the selector 98 selects the



set of character data held in the flip-flop 92 and the count of the counter 94 and outputs the data and count to the processing part of the data compression controller 22. Note that the reason why data passing through the uncompression data path is selected if the number of successive match characters is less than or equal to 2 and data passing through the compression data path is selected if the number of successive match character is greater than or equal to 3 is that compression using the position and length of a match character string including two or less successive match characters does not have no compression effect.

[0078]

Since no parity error has occurred for the first "IBM" in the example in Figure 7, the repeated data search circuit 21 is properly operating and the selector 98 outputs data, (0, 3), indicating that the three characters "IBM" have matches. Then, a parity error occurs and therefore the selector 98 outputs character data "i" as the fourth character without change. That is, the data (0, 3) and "i" are output in this order.

On the other hand, if no parity error has occurred in the repeated data search circuit 21, the selector 98 outputs (0, 4) because it means that the four characters "IBMi" have matches.

[0079]

Since the configuration described above is used in the present embodiment, data is not lost even if a parity error occurs in the compression data path, because the original data remains in the uncompression data path. Thus, the present embodiment has the effect that if a parity error occurs, the process does not halt and correct data is output, although the compression ratio decreases because the compressed data becomes invalid.

[0080]

Note that the repeated data search circuit 21 in the compression data path in Figure 9 is configured to take an input of one character (1 byte) as character data every one clock cycle and process the input character data in a pipeline having four stages. That is, character data is written in the CAM cell array 26 in the first clock cycle, a match of the character data is held in the latch 62 in the second clock cycle, a match of the character string preceding that character data is held in the latch 66 in the third clock cycle, and a match address is held in the flip-flop 92 in the fourth clock cycle. Accordingly, the match address and the character string match signal are output four clock cycles after the input of the character data. Therefore, the uncompression data path needs to have a pipeline configuration with four stages, namely the flip-flops 100<sub>1</sub>, 100<sub>2</sub>, 100<sub>3</sub> and 100<sub>4</sub>, which is equal to the number of the stages of the compression data path. However, the number of stages is illustrative only; if the number of the stages of the compression data path is K, the uncompression data path needs to have K stages.

[0081]

An operation of the data compression controller 22 in the present embodiment will be described below.

Figures 10-1 and 10-2 are flowcharts illustrating a process in the data compression controller 22. The process is initiated when data to be compressed (original data) is transferred to the data compression controller 22 through the bus 18 and an instruction to compress the original data is issued from the CPU 12.

[0082]

The data compression controller 22 first resets the latches 62<sub>0</sub> to 62<sub>N</sub> and latches 66<sub>0</sub> to 66<sub>N</sub> of the comparison result control circuit 60 of the repeated data search device 20 (step 200). The data compression controller 22 then initializes the match length MLEN to "1" and the write address WADR to "0" (step 202). The data compression controller 22 then determines whether or not the output of the original data to the repeated data search device 20 has been completed (step 204). If the determination is negative, the data compression controller 22 takes out character C0 data corresponding to the first one character from the original data as search data and outputs the data to the repeated data search device 20 along with a search instruction SEARCH and a write address WADR (step 206). As a result, the character C0 data is provided to the uncompression data path and the compression data path in the repeated data search device 20 and the search operation described above is performed.

[0083]

The data compression controller 22 then determines whether or not a parity error signal PE has gone high (step 207). A process performed if the parity error signal PE has gone high and the determination is affirmative will be described later. If the parity error signal PE is low and the determination is negative, the data compression controller 22 determines whether or not a character string match signal ORFB output from the repeated data search device 20 has gone high (step 208). In this case, since the latches 62 and the latches 66 have been previously reset at step 200, the character string match signal ORFB is held low. Therefore the determination is negative and the process proceeds to step 210, where the data compression controller 22 determines whether or not the current comparison operation is the first comparison operation for given original data (step 210). If the determination is affirmative, the process proceeds to step 214 without performing a process for outputting compressed data at step 212. Since the character

string match signal ORFB does not go high unless there are matches with two or more successive characters as described earlier, the character previously searched for is output as compressed data if the comparisons result in successive "mismatches" as will be described later. Accordingly, there is not data to be output as compressed data at this point in time and therefore step 212 is not performed.

[0084]

Then, the data compression controller 22 assigns "1" to the match length MLEN (step 214) and proceeds to step 218. The data compression controller 22 then sets the data of the previous search character C1 in place of the data of the last but one search character C2 (step 218) and sets the data of the current search character C0 in place of the previous search character C1 (step 218) and writes the data of the character C0 in the CAM cell column corresponding to the current write address WADR (0 for the first comparison operation) (step 220). The write operation is performed actually by the address decoder 52 asserting (enabling) the word line of the CAM cell column corresponding to the write address WADR. The data compression controller 22 adds 1 to the write address WADR (accordingly, the character data is written in ascending order of CAM cell column addresses), divides the result by the size N + 1 of the CAM cell array 26 and sets the remainder as the next write address WADR (step 222).

[0085]

Once the data has been written in all CAM cell columns in this way, the data is written in the CAM cell column with the address "0". Thus the CAM cell array 26 is used as the so-called ring buffer and an overflow of the CAM cell array 26 does not occur.

[0086]

After the process at step 222 has been performed, the process returns to step 204. If the determination at step 204 is negative, step 206 and the subsequent steps are performed again. At step 206, data of the character C0 following the previous search character C1 is taken out from the original data as search data and is output to the repeated data search device 20 along with a search instruction SEARCH and a write address WADR. If the determination at step 208 is negative, the process proceeds to step 210. If the determination at step 210 is negative, a compressed data output process at step 212 is performed.

[0087]

In the compressed data output process, the data compression controller 22 determines whether the match length MLEN is 0 or not as illustrated in Figure 11 (step 229). If the previous parity error signal PE is high, all of the past data has been output by a process, which will be described later, and the match length MLEN has been set to 0. Accordingly, data is not output as compressed data and the process proceeds to step 214 of the flowchart of Figure 10-1.

If the determination at step 229 is negative, the data compression controller 22 determines whether the match length MLEN is 2 or not (step 230). If the result of the previous comparison is a "match" but the result of the last comparison but one is not a "match", then the determination is affirmative and the data of the data of the last but one search character C2 is output as compressed data (step 236), then the data of the previous search character C1 is output (step 238), and then the process proceeds to step 214 of the flowchart of Figure 10-1.

If the determination at step 230 is negative, the data compression controller 22 determines whether the match length MLEN is 1 or not (step 232). If the result of the previous comparison is not a "match", the determination is affirmative and the data of the previous search character C1 is output as compressed data (step 238), then the process proceeds to step 214 of the flowchart of Figure 10-1.

[0088]

On the other hand, if the determination at step 208 of the flowchart of Figure 10-1 is affirmative, the match length MLEN is counted up (step 216), then the process proceeds to step 218. Accordingly, compressed data is not output while the character string match signal ORFB stays high.

[0089]

If the characters string match signal ORFB, which was high in the previous comparison operation, goes low and the determination at step 210 is negative, it means that the end of a repeated character string having a length greater than or equal to two characters has been found and therefore the compressed data output process is performed at step 212. Here, if the results of both of the previous comparison and the last but one comparison are "matches", the match length MLEN has been counted up to 3 or greater at step 216 described above and therefore the determinations at steps 229, 230 and 232 are negative and the process proceeds to step 234.

[0090]

The data compression controller 22 obtains a compression code for compressing the repeated character string. The present embodiment uses a compression code including a first code representing a pointer pointing to the position of the same

character string as a repeated character string and a second code representing the length of the repeated character string. The data compression controller 22 sets the difference between the match address MADR and the match length MLEN plus 1 ( $MADR - MLEN + 1$ ) as the first code, sets the match length MLEN as the second code, and outputs the values (step 234). Consequently, the compressed data output from the data compression controller 22 is shorter than the original data.

[0091]

Note that to output the compressed data, the data compression controller 22 also inserts a code representing a separator between the character data and the compression code in order to allow the compression code and the character data to be distinguished from each other when the compressed character string is decompressed. In this way, the step 234 is performed every time a repeated character string is found and the found repeated character string is converted to a conversion code and output, thereby eliminating the redundancy of the original data to compress the original data into compressed data.

[0092]

If the determination at step 207 of the flowchart of Figure 10-1 is affirmative, the process proceeds to step 240 of the flowchart of Figure 10-2. If the determination at step 240 is negative, the compressed data output process at step 242 is performed. The compressed data output process has been described with reference to the flowchart of Figure 11 and therefore the description of the compressed data output process will be omitted.

If the parity error signal PE goes high, the character string match signal ORFG is not reliable. Accordingly, the character C0 obtained through the uncompression data path of the repeated

data search device 20 is output at this point in time (step 244). Then "0" is assigned to the match length MLEN (step 246), the process proceeds to step 218 of the flowchart of Figure 10-1, and then the process described above is continued.

[0093]

The process described above is repeated and, when the character C data takeout position in the original data reaches the end of the original data, the determination at step 204 will be affirmative and the data compression controller 22 performs the compressed data output process again (step 224). Here, if the value of the match length MLEN is greater than or equal to 3, a compression code is output at step 234; if the value of the match length MLEN is 2, the data of the last but one search character C2 and the data of the previous search character C1 are output at steps 236 and 238, respectively, as compressed data; if the value of the match length MLEN is 1, the data of the previous search character C1 is output at step 238 as compressed data. Then the process ends.

[0094]

Note that while a pointer pointing to the position of the same character string as a repeated character string is used as the first code in the compression code in this exemplary operation, the distance between the position of the same character string as a repeated character string and the position of the repeated character string may be used according to the example described above. In that case, the first code may be WADR minus the position of the same character string as the repeated character string.

[0095]

An embodiment of the present invention has been described above.



In the present embodiment, the uncompression data path through which input character data is passed without change and the compression data path that outputs the position and length of a character string in the CAM that includes successive matches with input character data are provided in the repeated data search device 20 as described above. If a parity error occurs in the CAM, the character data from the uncompression data path is used. This enables the process using the character data to be continued without interruption even though the parity error occurs in the character data in the CAM.

[0096]

The cases where a parity error occurs are summarized below:

- Case 1: Temporary error

This error occurs incidentally and singly due to the influence of cosmic rays or noise. In this case, when the data is read again, no error occurs.

- Case 2: Cell failure

Since memory cells of a memory such as a CAM are usually tested at the factory prior to shipment, defective memory cells are rarely shipped. Cell failures occur in operation.

The present embodiment is effective in both of cases 1 and 2.

Case 1 error is temporary and therefore decreases the compression ratio only temporarily. In case 2, the compression ratio decreases whenever the failed cell is hit. However, since data itself is not corrupted in either case, the errors do not cause any problem in the present embodiment.

[0097]

While it is assumed that a particular portion of a character string is replaced with information indicating the position and length of another portion to compress the character string in the present embodiment, a particular portion of a character string may be processed on the basis of the position of another portion. In that case, the compression data path may be a path from the point of the CAM cell array 26 at which character data is input to the point at which the address of character data that matches the character data is output and the uncompression data path may be a path having a pipeline configuration that outputs character data in synchronization with the output of the address from the compression data path.

[0098]

While a parity bit is stored in the CAM cell array 26 in the present embodiment, the parity bit may be stored in a memory different from the CAM cell array 26. While a parity error has been given above as an error that occurs in data stored in the CAM cell array 26, the present invention is also applicable to a situation where an error other than a parity error occurs.

[0099]

While the present invention has been described with an embodiment thereof, the technical scope of the present invention is not limited to the embodiment described above. It will be apparent to those skilled in the art that various modifications can be made and variations can be used without departing from the spirit and scope of the present invention.

[Reference Signs List]

[0100]

20 ... Repeated data search device, 21 ... Repeated data search circuit, 22 ... Data compression controller, 24 ... Parity memory cell, 26 ... CAM cell array, 28 ... CAM cell, 52 ... Address decoder, 54 ... Parity generator, 60 ... Comparison result control circuit, 64 ... Signal generation circuit, 74 ... First priority encoder, 76 ... Second priority encoder, 90, 92, 96, 100, 102 ... Flip-flop, 94 ... Counter, 98 ... Selector

## Claims

1. A device processing a first partial data element string of a data element string on the basis of the position of a second partial data element string of the data element string, the device comprising:

a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored;

a first path providing an input data element to the content addressable memory as the search data element and outputting the address output from the content addressable memory;

a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element; and

a processing part processing the first partial data element string by using the position of the second partial data element string if no error occurs in a match data element that matches the search data element when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and processing the first partial data element string by using the particular data element output from the second path if an error occurs in a match data element that matches the

search data element, the position of the second partial data element string being determined by the address output from the first path.

2. The device according to claim 1, wherein the second path has a pipeline configuration that outputs the data element in synchronization with the output of the address from the first path.
3. A device replacing a first partial data element string of a data element string with information indicating the position and length of a second partial data element string of the data element string to compress the data element string, the device comprising:

a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored;

a generation circuit generating, on the basis of the address output from the content addressable memory, string presence information indicating that a string of match data elements that matches a string of search data elements previously provided is stored and column address information indicating an address at which the string of the match data elements is stored if the string of the match data elements is stored in the content addressable memory when the search data element is provided to the content addressable memory;

a first path providing an input data element to the

content addressable memory as the search data element and outputting the string presence information and the column address information generated by the generation circuit;

a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element; and

a processing part performing a process for replacing the first partial data element string with information indicating the position and length of the second partial data element string if one condition that no error occurs in a match data element that matches the search data element is satisfied when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and performing a process using the particular data element output from the second path to prevent a part of the first partial data element string from being compressed if an error occurs in a match data element that matches the search data element, the position and length of the second partial data element string being determined by the number of successive outputs of the string presence information from the first path and the column address information output from the first path.

4. The device according to claim 3, wherein the processing part performs the process for replacing with information indicating the position and length of the second partial data element string that is determined by the number of successive outputs and the column address information output from the first path if a further condition that the number of successive outputs of the string presence information from the first path exceeds a predetermined number is satisfied in addition to the one condition when

the first path provides the particular data element to the content addressable memory as the search data element.

5. The device according to claim 3 or 4, wherein the processing part performs a process using the particular data element output from the second path to prevent a part of the first partial data element string from being compressed if the number of successive outputs of the string presence information from the first path does not exceeds a predetermined number when the first path provides the particular data element to the content addressable memory as the search data element.
6. The device according to any of claims 3 to 5, wherein the second path has a pipeline configuration that outputs the data element in synchronization with the output of the string presence information and the column address information from the first path.
7. A device replacing a first partial data element string of a data element string with information indicating the position and length of a second partial data element string of the data element string to compress the data element string, the device comprising:

a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string and a parity element corresponding to the data element in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored;

a generation circuit generating, on the basis of the

address output from the content addressable memory, string presence information indicating that a string of match data elements that matches a string of search data elements previously provided is stored and column address information indicating an address at which the string of the match data elements is stored if the string of the match data elements is stored in the content addressable memory when the search data element is provided to the content addressable memory;

a determination circuit determining whether or not a value of the parity element stored at an address at which a match data element that matches the search data element is stored is generated from the search data element if the match data element is stored in the content addressable memory when the search data element is provided to the content addressable memory;

a first path providing an input data element to the content addressable memory as the search data element and outputting the string presence information and the column address information generated by the generation circuit;

a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element; and

a processing part performing a process for replacing the first partial data element string with information indicating the position and length of the second partial data element string if both of a condition that the determination circuit determines that a value of the parity element is generated and a condition that the number of successive outputs of the string presence information from the first path exceeds a predetermined number are satisfied when the first path provides a



particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element, and performing a process using the particular data element output from the second path to prevent the part of the particular data element of the first partial data element string from being compressed if at least one of a condition that the determination circuit determines that a value of the parity element is not generated and a condition that the number of successive outputs of the string presence information from the first path does not exceeds the predetermined number is satisfied, the position and length of the second partial data element string being determined by the number of successive outputs and the column address information output from the first path.

8. A method for processing a first partial data element string of a data element string on the basis of the position of a second partial data element string of the data element string, the method comprising:

a content addressable memory storing each of a plurality of data elements constituting at least a part of the data element string in an address corresponding to the position of the data element in the data element string and, if a match data element that matches a search data element is stored as one of the plurality of data elements when the search data element is provided, outputting an address at which the match data element is stored;

a first path providing an input data element to the content addressable memory as the search data element and outputting the address output from the content addressable memory; and

a second path outputting an input data element without providing the input data element to the content addressable memory as the search data element; and

the method comprising the steps of:

determining whether or not an error has occurred in a match data element matching the search data element when the first path provides a particular data element included in the first partial data element string to the content addressable memory as the search data element in response to an input of the particular data element; and

if it is determined that no error has occurred in the match data element, using the address output from the first path to process the first partial data element string and, if it is determined that an error has occurred in the match data element, using the particular data element output from the second path to process the first partial data element string.

**INTERNATIONAL SEARCH REPORT**

International application No.

PCT/JP2011/072891

**A. CLASSIFICATION OF SUBJECT MATTER**

H03M7/40(2006.01) i, G06F17/30(2006.01) i, G11C15/04(2006.01) i

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)

H03M7/40, G06F17/30, G11C15/04

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

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2011
Kokai Jitsuyo Shinan Koho	1971-2011	Toroku Jitsuyo Shinan Koho	1994-2011

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

IEEE Xplore, CiNii

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 7-114577 A (International Business Machines Corp.), 02 May 1995 (02.05.1995), paragraphs [0100] to [0137]; fig. 10, 11 & US 5448733 A & EP 634839 A1	1-8
Y	Hisatada MIYATAKE, "High-speed parallel content-addressable memory (CAM) having parity check function during search", PROVISION, 2010.11, no.67, pages 87 to 93 <URL: <a href="http://www-06.ibm.com/ibm/jp/provision/no67/pdf/67_paper2.pdf">http://www-06.ibm.com/ibm/jp/provision/no67/pdf/67_paper2.pdf</a> >	1-8

Further documents are listed in the continuation of Box C.

See patent family annex.

\* Special categories of cited documents:

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

"E" earlier application or patent 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  
15 November, 2011 (15.11.11)

Date of mailing of the international search report  
29 November, 2011 (29.11.11)

Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/072891

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 8-242176 A (International Business Machines Corp.), 17 September 1996 (17.09.1996), entire text; all drawings & US 5877714 A	1-8
A	JP 2010-268146 A (International Business Machines Corp.), 25 November 2010 (25.11.2010), entire text; all drawings & US 2010/0293344 A1	1-8
A	JP 5-252047 A (Fujitsu Ltd.), 28 September 1993 (28.09.1993), entire text; all drawings (Family: none)	1-8
A	JP 63-177242 A (Hitachi, Ltd.), 21 July 1988 (21.07.1988), entire text; all drawings (Family: none)	1-8