MEMORY REPAIR SYSTEM AND METHOD

**FIG. 1**

**Abstract:** A system and method for making efficient use of fuse ROM redundancy to increase yield and security. Some embodiments provide a memory repair system including a non-volatile memory component (104) and a controller (106) coupled to the non-volatile memory component. The non-volatile memory component includes a plurality of memory locations. The plurality of memory locations includes a replacement memory location to replace a faulty memory location and a replacement indicia memory location to store replacement memory location indicia. The controller coupled to the non-volatile memory component reads replacement memory location indicia from the replacement indicia memory location, determines an address for the replacement memory location using the indicia, reads the replacement memory location, and transfers a data value contained in the replacement memory location to a second memory component (112) to repair a defective memory location of the second memory component.
— as to the applicant’s entitlement to claim the priority of the earlier application (Rule 4.17(iii))

Published:

— with international search report
MEMORY REPAIR SYSTEM AND METHOD

This relates to memory cells in integrated circuits; and, in particular, to systems and methods for replacing defective memory cells.

BACKGROUND

As integrated circuit feature sizes shrink, memories embedded within an integrated circuit increase in density, and become more prone to failure. These failures reduce yield, resulting in higher cost per unit produced. One method of improving embedded memory yield involves including spare memory cells in the integrated circuit to replace memory cells found to be defective when the chip is tested. Employing redundancy in this manner can significantly improve the yield of embedded memories with minimal investment in die area.

"Fuse farms" are utilized as part of a system of embedded memory redundancy. A non-volatile memory in the fuse farm stores all the information necessary to repair embedded memories by replacing faulty memory cells with spare cells. During device initialization, a controller in the fuse farm reads the repair information from the fuse farm non-volatile memory ("fuse ROM") and loads the repair information into the on-chip embedded memories.

Being a form of embedded memory, the fuse ROM is itself subject to an increased rate of failure with reduced feature size. Because the fuse farm supplies repair information for the on chip memories, a fault in the fuse farm memory may result in a die that must be discarded. Moreover, users may be permitted to store limited information, such as serial numbers or part identifiers in fuse ROM after the part leaves the manufacturing facility. The fuse ROM may be implemented in a non-erasable memory technology, making pre-shipment testing of the fuse ROM impossible, and subjecting the manufacturer to an increased risk of loss of customer goodwill if the device is discarded due to an in-field fuse ROM programming failure.

SUMMARY

Various embodiments of systems and methods for making efficient use of fuse ROM redundancy to increase yield and security are disclosed. In some embodiments, a memory repair system includes a non-volatile memory component and a controller coupled to the non-volatile memory component. The non-volatile memory component includes a plurality of memory locations. The plurality of memory locations includes a replacement memory location to replace a faulty memory location and a replacement indicia memory location to store replacement memory location indicia. The controller reads replacement memory location
indicia from the replacement indicia memory location, determines an address for the replacement memory location using the indicia, reads the replacement memory location, and transfers a data value contained in the replacement memory location to a second memory component to repair a defective memory location of the second memory component.

In other embodiments, a method for memory repair includes reading a first memory word, determining whether a second memory word is selected to replace the first memory word, reading a second memory word location, determining a location of the second memory word, reading the second memory word, and transferring a data value from the second memory word to repair a defective memory location.

In other embodiments, a memory repair system includes a means for non-volatilely storing memory repair information and a means for non-volatilely storing replacement row indexing information. Additionally, the system includes means for accessing the replacement row indexing information, means for determining the location of the memory repair information from the replacement row indexing information, and means for accessing the memory repair information. Means for transferring the memory repair information to a defective memory component to repair the defective memory component is also included.

BRIEF DESCRIPTION OF THE DRAWINGS

In the following detailed description, reference will be made to the accompanying drawings, in which:

FIG. 1 shows an illustrative integrated circuit which includes a fuse farm coupled to an embedded memory system in accordance with various embodiments of the invention;

FIG. 2 shows an illustrative array of fuse ROM memory blocks in accordance with the preferred embodiments of the invention;

FIG. 3 shows an illustrative fuse ROM memory word in accordance with the preferred embodiments of the invention;

FIGS. 4A-B, 5A-E, and 6A-C show illustrative applications of repair addressing in a fuse ROM memory word in accordance with the preferred embodiments of the invention;

FIGS. 7 and 8 show flow diagrams for a fuse ROM repair method in accordance with the preferred embodiments of the invention;

FIGS. 9 and 10 show flow diagrams for a fuse ROM repair method in accordance with other embodiments of the invention; and
FIGS. 11 and 12 show flow diagrams for another fuse ROM repair method in accordance with other embodiments of the invention.

DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS

Disclosed herein are various systems and methods for making efficient use of redundancy to repair faults in fuse farm non-volatile memory, to thereby increase yield. The disclosed embodiments include a system for repairing a fuse ROM that stores fuse ROM repair information in a fuse ROM word, as well as methods for efficient use of fuse ROM redundant resources by storing fuse ROM repair information in a fuse ROM word.

Generally, a memory repair system incorporating fuse farm technology operates to maintain memory functionality by employing spare memory locations within a memory component in place of faulty memory locations within the component. In other words, faulty memory locations are replaced by, or remapped to, functional locations within the memory component. Because the fuse ROM is a critical element of the memory repair system, flexible and efficient means of utilizing redundant fuse ROM resources must be employed to overcome fuse ROM defects.

FIG. 1 shows an illustrative integrated circuit ("IC") 120 which includes a fuse farm 102 coupled to an embedded memory system 112, 114. Fuse farm 102 comprises a fuse ROM 104 for storage of embedded memory repair information and other user data, and a fuse farm controller 106 for accessing the fuse ROM 104. The fuse ROM 104 is a non-volatile memory that may be implemented in a variety of non-volatile memory technologies, for example, fusible link, anti-fuse, EPROM, EEPROM, FLASH, ferro-electric, magnetic, or equivalent non-volatile memory technologies. The fuse farm controller 106 orchestrates the transfer of memory repair information from the fuse ROM 104 to the embedded memories 112, 114 during device initialization. Controller 106 may be implemented as a state machine, a processor and associated program, or equivalent.

The embedded memories 112, 114 include spare memory cells that may replace memory cells found to be defective when the memories 112, 114 are tested during production. Repair information for enabling spare cells to replace defective cells is written into the fuse ROM 104 as part of the manufacturing process. During device initialization, controller 106 reads repair information from fuse ROM 104 and transfers the repair information to embedded memories 112, 114 via interface 116, which generally comprises a serial scan chain. Note that
in the context of the present disclosure, the term "repair" refers not to restoring a defective memory cell to proper operation, but to maintaining operability of the memory as whole in spite of a defective memory cell through application of redundancy and other techniques.

The controller 106 also manages access to the fuse ROM 104. An interface 108, for example a Joint Test Action Group ("JTAG") interface, allows external systems 122 to access the fuse farm 102. Memory repair information and other user information to be written into the fuse ROM 102 may be transferred via interface 108. On-chip systems 118, for example a processor core, may access the fuse farm 102 through other provided interfaces. Controller 106 operates to enhance device security by disabling read or write access to the programmed fuse ROM 104 by on-chip systems 118 or external systems 122.

The reliability of fuse farm 102 is one factor in achieving the yield improvements enabled by including redundant memory cells in embedded memories 112, 114. Existing systems have implemented a variety of reliability enhancement strategies. FIG. 2 shows an illustrative array of fuse ROM memory blocks 202. Each memory block 202 is further subdivided into rows or words 204, 206. One or more of memory rows 206 may be reserved to replace memory rows 204 found to be defective.

Referring now to FIG. 3, an example of a fuse ROM memory row 204 is illustrated. In the example of FIG. 3, the fuse ROM memory row 204 comprises a data field 306, a repair address ("RA") field 302, and set of access control flags 304. Some embodiments of fuse farm 102 apply the RA field 302 to implement single bit error repairs. When a value is written to data field 306, and subsequent read-back of data field 306 indicates a single defective bit, the column number of the defective bit may be written into the RA field 302. The column number stored in RA field 302 is used to correct the defective bit when the row 204 is subsequently read. When post-write verification of the data field 306 indicates multiple defective bits, or when an attempt to correct a single bit error by writing a column number to the RA field 302 fails, a replacement row 206 may be substituted for the defective row 204. Some fuse farm 102 embodiments set a repair flag 308 to indicate that the row 204 is defective and has been superseded by a replacement row 206. In some fuse farm implementations, selection of a replacement row 206 may be based on the address of the defective row. For example, when two rows of block 202 have been allocated for use as replacement rows 206, the least significant bit ("LSB") of the address of the defective row 204 may serve as an index to the
replacement row 206. Similarly, when four rows of block 202 have been allocated for use as replacement rows 206, the two LSBs of the address of the defective row 204 may serve as an index to the replacement row 206. Unfortunately, this indexing method limits row replacement to one of each of a plurality of rows having the same address LSBs. Consequently, notwithstanding the fact that adequate replacement rows are available, when two rows having the same address LSBs are defective, the fuse ROM cannot be repaired, and the device is discarded.

The embodiments of the present disclosure provide for efficient use of fuse ROM replacement rows, thereby reducing the number of non-repairable fuse ROM defects and improving device yield. FIG. 4A shows a first embodiment of an illustrative application of a reserved field, in this example the RA field 302 and control flag field 304, of a defective fuse ROM row 204 to identify a replacement row 206. In FIG. 4A, the repair flag 308 is set indicating substitution of a replacement row 206 for this defective row 204. The write protect ("WP") 312 and the read protect ("RP") 310 flags are reset indicating that the RA field 302 contains a replacement row index. By writing replacement row indexing information into a relatively large field of the defective fuse ROM row 204, such as the RA field 302, a substantial number of replacement rows 206 may be directly accommodated. To enhance data security, the data in defective data field 306 may be obfuscated by overwriting the defective data field 306 with, for example, a random value or all ones.

FIG. 4B illustrates a second aspect of the embodiment of FIG. 4A. In FIG. 4A, the WP 312 and RP 310 flags are "0," which, for purposes of this illustration, is the state of a fuse ROM memory cell prior to programming. The WP 312 and RP 310 flags, so reset, signify that the RA field 302 contains a replacement row index. If a defect in the RA field 302 is identified after writing the replacement row index, and the resultant RA field 302 value is unacceptable for use as a replacement row index, then a two bit non-zero index value may be written to WP 312 and RP 310. The two bit index value written to WP 312 and RP 310 may be used alone or in combination with the defective row address to form a replacement row index. Thus the use of the reserved fields of a defective row support both inter and intra row redundancy.

FIG. 7 shows a flow diagram for an illustrative fuse ROM repair method incorporating storage of replacement row indexing information in a reserved field of the defective row 204. In block 702, a value is written to a data field 306 of a fuse ROM
memory row 204. The value is read from the fuse ROM to determine whether the row 204 is
defective in block 704. If the write was successful, the operation is complete. If the write
failed due to a single bit error in block 706, the column number of the faulty bit is written to
the RA field 302 in block 708. The column number write is verified in block 710, and if
successful, the write operation is complete. If the column number write is unsuccessful, the
repair flag 308 is set in block 726 to indicate replacement of the defective row 204. Data
field 306 is overwritten in block 728 to obfuscate any value previously written. When an
unsuccesful column number write results in an RA field 302 value acceptable for use as a
replacement row index in block 720, the data value is written into the selected fuse ROM
replacement row 206 in block 724 and the operation is complete. When an unsuccessful
column number write results in an RA field 302 value unsuitable for use as a replacement
row index, a two bit non-zero index value is written to the WP 312 and RP 310 fields in
block 722, and the data value is written into the selected fuse ROM replacement row 206 in
block 724.

If a multi-bit error is detected in block 706, the repair flag 308 is set in block 712.
The set repair flag 308 indicates that a replacement row 206 is being substituted for the
defective row 204. Data field 306 is overwritten in block 714 to obfuscate any value
previously written, and a replacement row index is written to the RA field 302 in block 716.
The RA field 302 write is verified in block 718. If the RA field write was successful, the
data value is written to the selected fuse ROM replacement row 206 in block 724. If the RA
field write was unsuccessful, the acceptability of the RA field value for use as a replacement
row index is determined in block 720. For example, if an RA field value resulting from a
failed write can nevertheless be used to build a useable replacement row index, the RA field
value may be acceptable. If the RA field value is deemed acceptable in block 720, the data
value is written to the selected fuse ROM replacement row 206 in block 724. If the RA field
value is unacceptable, then a two bit non-zero index value is written to the WP 312 and RP
310 fields in block 722, and the data value is written to the selected replacement row 206 in
block 724.

FIG. 8 shows a flow diagram for an illustrative fuse ROM reading method
complementary to the replacement row index storage method of FIG. 7. In block 802, the
controller 106 reads a fuse ROM row 204. If, in block 804, the repair flag 308 is reset, then
the row 204 is deemed free of multi-bit errors, single-bit errors indicated by the RA field 302 are corrected, and the read operation is complete. If the repair flag 308 is set, the row 204 is defective and a replacement row 206 has been substituted. In block 806, if the row's WP 312 and RP 310 flags are reset, the replacement row index is read from the RA field 302 in block 808 and used to read the selected fuse ROM replacement row 206 in block 812. If the WP 312 and RP 310 flags are found to be not reset in block 806, then the two bit index value contained in the WP 312 and RP 310 flags may be used to build a replacement row index in block 810. For example, the two bit index value contained in the WP 312 and RP 310 flags may be combined with the address of the defective row to form a replacement row index. Numerous methods of combining of the index value stored in WP 312 and RP 310 with the defective row address can be used to generate the replacement row index, including using the index value in WP 312 and RP 312 as a complete replacement row index, or using the index value in WP 312 and RP 310 as the LSBs of a replacement row index. The replacement row index built in block 810 is used to read the selected fuse ROM replacement row 206 in block 812.

FIGS. 5A, 5B, and 5C show a first alternative embodiment of an illustrative application of a reserved field, in this example, the RA field 302 and control flag field 304, of a defective fuse ROM row 204 to identify a replacement row 206. In FIG. 5A, the repair flag 308 is set indicating that a replacement row 206 is to be substituted for this defective row 204 and the WP 312 and RP 310 flags are set as a security measure to prevent systems 118, 112 external to the fuse farm 102 from accessing this fuse ROM row 204. In this embodiment access protection redundancy may be provided by disabling both external read and write accesses to the row 204 when either of the WP 312 or RP 310 flags is set.

In FIGS. 5A, 5B, and 5C, the RA field 302 is subdivided into three sets of two bit sub-fields to provide index field redundancy. As exemplified in FIG. 5A, when sub-field RA[1 :0] contains a non-zero value, that is, when at least one bit of the sub-field is a "1," the non-zero value denotes an index that may alone or in combination with the address of the defective row be used to identify the replacement row 206. When an index value containing sub-field is identified, higher order RA sub-fields are ignored. Consequently, in FIG. 5A, sub-fields RA[3:2] and RA[5:4] need not be read. In FIG. 5B, the RA[1 :0] sub-field is unprogrammed, and therefore zeroed. The zeroed RA[1 :0] sub-field indicates that RA[1 :0] does not contain an
index value, but rather that an index value may be read from one of the higher order RA sub-fields. Sub-field RA[3:2] contains a non-zero index value in FIG. 5B, so sub-field RA[5:4] need not be read. Finally, FIG. 5C illustrates a situation in which both sub-field RA[1:0] and sub-field RA[3:2] are zeroed, indicating that they are unprogrammed and that a two bit index value may be read from sub-field RA[5:4].

FIG. 9 shows a flow diagram for an illustrative first alternative fuse ROM 104 repair method incorporating storage of a replacement row index in a reserved field of a defective row 204. In block 902, a value is written to the data field 306 of a fuse ROM memory row 204. The value is read from the fuse ROM 104 to determine whether the row 204 is defective in block 904. If the write was successful, the operation is complete. If the write failed due to a single bit error in block 906, the column number of the faulty bit is written to the RA field 302 in block 908. The column number write is verified in block 910, and if successful, the write operation is complete. If, in block 910, the column number write failed, the repair flag 308 is set in block 912 indicating that a replacement row 206 is being substituted for the defective row 204. The RA sub-field values resulting from the failed write of block 908 may be used as an index by the method of FIG. 10. In block 926, the RP 310 and WP 312 flags are set to inhibit access to the defective row by systems 118, 122 external to the fuse farm 102.

If a multi-bit error was detected in block 906, the repair flag 308 is set in block 914. The repair flag 308 indicates that a replacement row 206 is being substituted for the defective row 204. In block 916, an index value which may be used to identify a selected replacement row 206 is written to sub-field RA[5:4]. If, in block 918, the index value write was successful or the resultant sub-field value is acceptable as an index value, the WP 312 and RP 310 flags are set in block 926 to inhibit access to the defective row 204 by systems 118, 122 external to the fuse farm, and the data value is written to the selected fuse ROM replacement row 206 in block 928. If the RA[5:4] sub-field value is unacceptable in block 918, an index value is written to sub-field RA[3:2]. If in block 922, the index value write was successful or the resultant RA[3:2] sub-field value is acceptable as an index value, the WP 312 and RP 310 flags are set in block 926 to inhibit access to the defective row 204 by systems 118, 122 external to the fuse farm 102. The data value is written to the selected fuse ROM replacement row 206 in block 928. If the RA[3:2] sub-field value is unacceptable in block
922, an index value is written into sub-field RA[1:0] in block 924, the WP 312 and RP 310 flags are set in block 926 to inhibit access to the defective row 204 by systems 118, 122 external to the fuse farm 102. The data value is written to the selected fuse ROM replacement row 206 in block 928.

FIG. 10 shows a flow diagram for an illustrative fuse ROM 104 reading method complementary to the index value storage method of FIG. 9. The method employs successive testing of RA sub-fields to identify a valid index value for location of replacement row 206. In block 1002, the controller 106 reads a fuse ROM row 204. If the repair flag 308 is not set in block 1004, then the row 204 is deemed free of multi-bit errors, single-bit errors indicated by the RA field 302 are corrected, and the read operation is complete. If the repair flag 308 is set in block 1004, the row 204 is defective, and a replacement row 206 has been substituted. In block 1006, if sub-field RA[1:0] is non-zero, then sub-field RA[1:0] contains an index value for identifying a replacement row 206. In block 1008, the contents of sub-field RA[1:0] alone or in combination with the defective row address may be used to build a replacement row index and to read the selected fuse ROM replacement row 206 in block 1016. A replacement row index may be built in block 1008 by, for example, combining the index value extracted from the RA sub-field with the upper bits of the defective row address, such that:

\[ \text{ReplacementRowIndex} = (\text{DefectAddress} \& \text{SubFieldMask}) \equiv \text{IndexValue}, \]

where \( \text{DefectAddress} \) is the address of the defective row, \( \text{SubFieldMask} \) zeros the address field to be replaced by \( \text{IndexValue} \), \( \text{IndexValue} \) is the index value extracted from the RA sub-field, and "&" and "\( \equiv \)" denote bit-wise "AND" and "OR" respectively. As in any of the disclosed embodiments, the replacement row index may constitute any value leading to the location of the replacement row 206 including the address of the replacement row 206.

If sub-field RA[1:0] contains all zeros in block 1006, sub-field RA[1:0] does not contain an index value, and sub-field RA[3:2] may be tested for an index value in block 1010. If, in block 1010, sub-field RA[3:2] is found to contain a non-zero value, then sub-field RA[3:2] contains an index value which in block 1012 may be used alone or in combination with the defective row address to build a replacement row index and to read the selected fuse ROM replacement row 206 in block 1016. If sub-field RA[3:2] contains all zeros in block 1010, field RA[5:4] contains an index value, and in block 1014, the contents
of sub-field RA[5:4] may be used alone or in combination with the defective row address to build a replacement row index and to read the selected fuse ROM replacement row in block 1016.

FIGS. 5D and 5E show an alternative embodiment of an illustrative application of a reserved field to identify a replacement row 206 wherein the RA field 302 is subdivided into two three bit sub-fields. The present embodiment is similar in many respects to the embodiment of 5A-5C, but employs a three bit rather than a two bit index field. In FIG. 5D, the repair flag 308 is set indicating that a replacement row 206 is to be substituted for the defective row 204. The WP 312 and RP 310 flags are set as a security measure to prevent systems 118, 122 external to the fuse farm 102 when either of the WP 312 or RP 310 flags is set. In this embodiment, access protection redundancy is provided by disabling access to the defective row 204 by systems 118, 122 external to fuse farm 102 when either of the WP 312 or RP 310 flags is set.

The RA field 302 is subdivided into a set of two three bit sub-fields to provide redundant storage for replacement row identifying index values. As illustrated in FIG. 5D, when sub-field RA[2:0] contains a non-zero value, that value denotes an index that may alone or in combination with the address of the defective row 204 be used to identify the replacement row 206. In this embodiment, when an index containing sub-field is identified, higher order RA sub-fields are ignored. Consequently, in FIG. 5D sub-field RA[5:3] need not be read. In FIG. 5E, the RA[2:0] sub-field is unprogrammed, and therefore contains zeros. The zeroed RA[2:0] sub-field indicates that RA[2:0] does not contain an index value, but rather that an index value may be read from the RA[5:3] sub-field.

FIGS. 6A, 6B, and 6C show yet another alternative embodiment of an illustrative application of a reserved field, in this example, the RA field 302 and control flag field 304, of a defective fuse ROM row 204 to identify a replacement row 206. In FIG. 6A, the repair flag 308 is set indicating that a replacement row 206 is to be substituted for this defective row 204. The WP 312 and RP 310 flags are set as a security measure to prevent access to the defective fuse ROM row 204 by systems 118, 122 external to the fuse farm 102. In this embodiment access protection redundancy is provided by disabling access to the defective row 204 by systems 118, 122 external to fuse farm 102 when either of the WP 312 or RP 310 flags is set.
The RA field 302 is subdivided into a set of three two bit sub-fields to provide index field redundancy. As illustrated in FIG. 6A, when sub-field RA[5:4] contains both a one and a zero, that value denotes an index that may alone or in combination with the address of the defective row be used to identify a replacement row 206. In this embodiment, when an index containing sub-field is identified, lower order RA sub-fields are ignored. Consequently, in FIG. 6A sub-fields RA[3:2] and RA[1:0] need not be read. In FIG. 6B, the RA[5:4] sub-field contains either all ones or all zeros, indicating that the RA[5:4] sub-field does not contain an index value, but rather that an index value may be read from one of the lower order RA sub-fields. Sub-field RA[3:2] contains both a zero and a one, and consequently contains an index value that may alone or in combination with the address of the defective row be used to identify the replacement row 206. In FIG. 6C, both RA[5:4] and RA[3:2] contain either an all ones or all zeros value, indicating that a replacement row identifying index value may be read from a lower order RA sub-field, e.g., sub-field RA[1:0].

FIG. 11 shows a flow diagram for an illustrative second alternative fuse ROM 104 repair method incorporating storage of a replacement row index in a reserved field of the defective row 204. In block 1102, a value is written to the data field 306 of a fuse ROM memory row 204. The value is read from the fuse ROM 104 to determine whether the row 204 is defective in block 1104. If the write was successful, the operation is complete. If the write failed due to a single bit error in block 1106, the column number of the faulty bit is written to the RA field 302 in block 1108. The column number write is verified in block 1110, and if successful, the write operation is complete. If, in block 1110, the column number write failed, the repair flag 308 is set in block 1112 indicating that a replacement row 206 is being substituted for the defective row 204. The RA sub-field values resulting from the failed write of block 1108 may be used as a replacement row index by the method of FIG. 12. In block 1126, the RP 310 and WP 312 flags are set to prevent access to the defective row 204 by systems 118, 122 external to fuse farm 102, and the data value is written to the selected fuse ROM replacement row 206 in block 1128.

If a multi-bit error is detected in block 1106, the repair flag 308 is set in block 1114. The repair flag 308 indicates that a replacement row 206 is being substituted for the defective row 204. In block 1116, a replacement row identifying index value is written to sub-field RA[IrO]. If, in block 1118, the index value write was successful, the WP 312 and RP 310
flags are set to inhibit systems 118, 122 external to the fuse farm 102 from accessing the
defective row 204 in block 1126, and the data value is written to the selected replacement
row 206 in block 1128. If, in block 1118, the index value write failed, a replacement row
index is written to sub-field RA[3:2] in block 1120. If in block 1122 the index value write
was successful, the WP 312 and RP 310 flags are set in block 1126 to inhibit systems 118, 122 external to the fuse farm 102 from accessing the defective row 204. The data value is
written to the selected replacement row in block 1128. If the index value write to sub-field
RA[3:2] failed in block 1122, an index value is written into sub-field RA[5:4] in block 1124,
the WP 312 and RP 310 flags are set in block 1126 to prevent fuse farm external systems
118, 112 from accessing the defective row 204, and the data value is written to the selected replacement row in block 1128.

FIG. 12 shows a flow diagram for an illustrative fuse ROM reading method
complementary to the replacement row index storage method of FIG. 11. The method
employs successive testing of RA sub-fields to identify a replacement row locating index
value. In block 1202, the controller 106 reads a fuse ROM row 204. If the repair flag 308 is
not set in block 1204, then the row 204 is deemed free of multi-bit errors, single-bit errors
indicated by the RA field 302 are corrected, and the read operation is complete. If the repair
flag 308 is set, the row 204 is defective and a replacement row 206 has been substituted. In
block 1206, if sub-field RA[5:4] contains both a zero and a one, then sub-field RA[5:4]
contains a replacement row identifying index value, and the contents of sub-field RA[5:4]
may be used in block 1208 to build a replacement row index, and to read the selected fuse
ROM replacement row 206 in block 1216. A replacement row index may be formed in block
1208 by, for example, combining the index value extracted from the RA sub-field with the
upper bits of the defective row address, such that:

ReplacementRowIndex = (DefectAddress & SubFieldMask) \( \& \) IndexValue,

where DefectAddress is the address of the defective row, SubFieldMask zeros the address field
to be replaced by IndexValue, and IndexValue is the index value extracted from the RA sub-
field.

If sub-field RA[5:4] contains all zeros or all ones in block 1206, sub-field RA[5:4]
does not contain a replacement row index, and sub-field RA[3:2] is tested for a valid
replacement row index value in block 1210. If in block 1210 sub-field RA[3:2] is found to
contain neither all ones nor all zeros, then sub-field RA[3:2] contains an index value, and the contents of sub-field RA[5:4] may be used in block 1208 to build a replacement row index, and to read the selected fuse ROM replacement row 206 in block 1216. If field sub-RA[3:2] contains all ones or all zeros in block 1210, then sub-field RA[1:0] contains an index value. In block 1214, the contents of sub-field RA[1:0] are used to build a replacement row index, and to read the selected fuse ROM replacement row 206 in block 1216.

As disclosed, some embodiments apply the RA field 302 to correct single bit memory faults within the data field 306. Control flag field 304 failures are also correctable using the RA field 302. Referring now to FIG. 3 and Table 1 below, where Table 1 contains a set of illustrative RA values for correcting data field 306 and control field 304 bit errors. When a multi-bit data field 306 error or an RA field 302 error necessitates row replacement, repair flag 308 is set. As illustrated in Table 1, various RA field 302 values may be reserved to correct a repair flag 302 memory failure. When a repair flag 308 write fails, an appropriate value may be written to the RA field 302 to accomplish a repair flag 308 correction. When a single data field bit error is detected, and the subsequent RA field column write fails, and the requisite repair flag 308 write also fails, the repair flag 308 failure is correctable by setting the appropriate RA field bits. The resultant RA field 302 value may be used as an index value to build a replacement row index. Thus, this disclosure also supports repair flag 308 redundancy.

<table>
<thead>
<tr>
<th>RA Value</th>
<th>Bit Corrected</th>
</tr>
</thead>
<tbody>
<tr>
<td>000001</td>
<td>Data 0</td>
</tr>
<tr>
<td>000010</td>
<td>Data 1</td>
</tr>
<tr>
<td>000011</td>
<td>Data 2</td>
</tr>
<tr>
<td>011111</td>
<td>Data 30</td>
</tr>
<tr>
<td>100000</td>
<td>Data 31</td>
</tr>
<tr>
<td>100010</td>
<td>RP</td>
</tr>
<tr>
<td>100011</td>
<td>WP</td>
</tr>
<tr>
<td>11xxxx</td>
<td>R</td>
</tr>
<tr>
<td>lxlxxx</td>
<td></td>
</tr>
<tr>
<td>lxxlxx</td>
<td></td>
</tr>
</tbody>
</table>

Table 1

Those skilled in the art to which the invention relates will appreciate that many other embodiments and variations are possible within the scope of the claimed invention.
CLAIMS

What is claimed is:

1. A system comprising:
   a non-volatile memory component comprising a plurality of memory locations,
   wherein the plurality of memory locations comprise a replacement memory location to replace
   a faulty memory location, and a replacement indicia memory location to store replacement
   memory location indicia; and
   a controller coupled to the non-volatile memory component that reads the
   replacement memory location indicia from the replacement indicia memory location,
   determines an address for the replacement memory location using the indicia, reads the
   replacement memory location, and transfers a data value from the replacement memory
   location to a second memory component to repair a defective memory location of the second
   memory component.

2. The system of claim 1, wherein the replacement memory location indicia
   further comprises a flag indicating that a memory location is defective, and an index value
   determining the address of the replacement memory location.

3. The system of claim 1, wherein each of the plurality of memory locations
   further comprises the replacement indicia memory location.

4. The system of claim 1, wherein the replacement indicia memory location
   comprises a plurality of fields, each field containing the replacement memory location indicia.

5. The system of claim 1, wherein the plurality of memory locations further
   comprise a write protect memory location to prevent an external system from writing at least
   one of the plurality of memory locations, and a read protect memory location to prevent the
   external system from reading at least one of the plurality of memory locations.

6. A method for memory repair comprising:
   reading a first memory word;
   determining whether a second memory word is selected to replace the first
   memory word;
   reading a second memory word location indicia;
   determining a location of the second memory word from the second memory
   word location indicia;
reading the second memory word;
transferring a repair value from the second memory word to a memory device having a defective memory location to repair the memory device.

7. The method of claim 6, further comprising:
writing a data value to the first memory word;
reading the data value from the first memory word to determine whether the first memory word is defective;
selecting the second memory word to replace the first memory word; and
writing the second memory word location indicia.

8. The method of claim 6, further comprising writing indicia of a single bit error correcting a single bit error in the first memory word.

9. The method of claim 6, further comprising writing indicia of a single bit error wherein indicia of a single bit error comprises the second memory word location indicia.

10. The method of claim 6, wherein the first memory word comprises a plurality of fields, each field dimensioned to store the second memory word location indicia.
<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/0</td>
<td>1/0</td>
<td>1/0</td>
<td>1/0</td>
<td>1/0</td>
<td>1/0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**FIG. 4A**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 4B**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX</td>
<td>XX</td>
<td>01/10/11</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 5A**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX</td>
<td>01/10/11</td>
<td>00</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 5B**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-11</td>
<td>00</td>
<td>00</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 5C**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>XXX</td>
<td>001-111</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 5D**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>000-111</td>
<td>000</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 5E**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/10</td>
<td>XX</td>
<td>XX</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 6A**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>00/11</td>
<td>01/10</td>
<td>XX</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 6B**

<table>
<thead>
<tr>
<th>RA5</th>
<th>RA4</th>
<th>RA3</th>
<th>RA2</th>
<th>RA1</th>
<th>RA0</th>
<th>WP</th>
<th>RP</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>00/11</td>
<td>00/11</td>
<td>01/10</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 6C**
FIG. 7

START

WRITE DATA TO FUSE ROM

FUSE ROM WRITE SUCCESSFUL?

NO

SINGLE BIT ERROR?

YES

SET REPAIR FLAG

OVERWRITE DATA

WRITE REPLACEMENT ROW INDEX TO RA

RA WRITE SUCCESSFUL?

NO

WRITE REPLACEMENT ROW INDEX TO WP:RP

WRITE DATA TO FUSE ROM

STOP

WRITE ERROR COLUMN TO RA

RA WRITE SUCCESSFUL?

NO

SET REPAIR FLAG

OVERWRITE DATA

YES
**FIG. 8**

1. **START**
2. READ FUSE ROM ROW
3. **REPAIR FLAG SET?**
   - **NO**
   - **YES**
     - WP:RP RESET?
       - **NO**
       - **YES**
         - **EXTRACT REPLACEMENT ROW INDEX FROM RA**
         - **USE WP:RP TO BUILD REPLACEMENT ROW INDEX**
3. **STOP**

**FIG. 10**

1. **START**
2. READ FUSE ROM ROW
3. **REPAIR FLAG SET?**
   - **NO**
   - **YES**
     - RA[1:0] == 0?
       - **NO**
       - **YES**
         - **USE RA[1:0] TO BUILD REPLACEMENT ROW INDEX**
     - RA[3:2] == 0?
       - **NO**
       - **YES**
         - **USE RA[3:2] TO BUILD REPLACEMENT ROW INDEX**
     - **USE RA[5:4] TO BUILD REPLACEMENT ROW INDEX**
4. READ FUSE ROM REPLACEMENT ROW
5. **STOP**
START

WRITE DATA TO FUSE ROM

FUSE ROM WRITE SUCCESSFUL?

YES

NO

SINGLE BIT ERROR?

YES

WRITE ERROR COLUMN TO RA

NO

SET REPAIR FLAG

RA[5:4] = REPLACEMENT ROW IDX

WRITE SUCCESSFUL?

YES

NO

SET REPAIR FLAG

RA[3:2] = REPLACEMENT ROW IDX

WRITE SUCCESSFUL?

YES

NO

RA[1:0] = REPLACEMENT ROW IDX

SET WP AND RP FLAGS

WRITE DATA TO FUSE ROM

STOP

FIG. 9
FIG. 11

START

WRITE DATA TO FUSE ROM

FROM DATA WRITE SUCCESSFUL?

YES

SINGLE BIT ERROR?

YES
WRITE ERROR COLUMN TO RA

NO

SET REPAIR FLAG

RA[1:0] = REPLACEMENT ROW IDX

WRITE SUCCESSFUL?

YES

RA WRITE SUCCESSFUL?

YES
SET REPAIR FLAG

NO

RA[3:2] = REPLACEMENT ROW IDX

WRITE SUCCESSFUL?

YES

RA[5:4] = REPLACEMENT ROW IDX

SET WP AND RP FLAGS

WRITE DATA TO FUSE ROM

STOP
FIG. 12

1202 READ FUSE ROM ROW

1204 REPAIR FLAG SET?


1208 USE RA[5:4] TO BUILD REPLACEMENT ROW INDEX

1210 USE RA[3:2] TO BUILD REPLACEMENT ROW INDEX

1214 USE RA[1:0] TO BUILD REPLACEMENT ROW INDEX

1216 READ FUSE ROM REPLACEMENT ROW

STOP
A. CLASSIFICATION OF SUBJECT MATTER

GIIC 29/00(2006.01)1, GIIC 16/00(2006.01)1

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 8 G1C 16/00, G01C 29/00

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

Korean Utility models and applications for Utility Models since 1975

Japanese Utility Models and applications for Utility Models since 1975

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

eKIPASS(KIPO internal) "redundancy, non-volatile, repair, defective"

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No</th>
</tr>
</thead>
<tbody>
<tr>
<td>P.X</td>
<td>JP 2007200527 A (SAMSUNG ELECTRONICS CO, LTD) 09 August 2007</td>
<td>1.6</td>
</tr>
<tr>
<td></td>
<td>See paragraphs &lt;33&gt; - &lt;40&gt;</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>JP 2007035245 A (SAMSUNG ELECTRONICS CO, LTD) 08 February 2007</td>
<td>1-10</td>
</tr>
<tr>
<td></td>
<td>See paragraphs &lt;29&gt;, &lt;34&gt; - &lt;36&gt;</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>JP 2004055100 A (ELPIDA MEMORY INC) 19 February 2004</td>
<td>1-10</td>
</tr>
<tr>
<td></td>
<td>See paragraph [0049]</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>JP 2005050442 A (SANYO ELECTRIC CO, LTD) 24 February 2005</td>
<td>1-10</td>
</tr>
<tr>
<td></td>
<td>See paragraph &lt;35&gt; and claim 1</td>
<td></td>
</tr>
</tbody>
</table>

☐ 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 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 25 SEPTEMBER 2008 (25 09 2008)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
Government Complex-Daejeon, 139 Seonsa-ro, Seo-gu, Daejeon 302-701, Republic of Korea

Facsimile No 82-42-472-7140

Date of mailing of the international search report 25 SEPTEMBER 2008 (25.09.2008)

Authorized officer

JANG Ho Keun

Telephone No 82-42-481-8187
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP 2007200527 A</td>
<td>09.08.2007</td>
<td>KR 100735542 B1</td>
<td>28.06.2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KR 1020060 1173 14 A</td>
<td>16.11.2006</td>
</tr>
<tr>
<td>JP 2007035245 A</td>
<td>08.02.2007</td>
<td>DE 1006033649 A1</td>
<td>29.03.2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KR 1020070 13110 A</td>
<td>30.01.2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 20070033449 A1</td>
<td>08.02.2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 7280415 BB</td>
<td>09.10.2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 20040062086 A1</td>
<td>01.04.2004</td>
</tr>
<tr>
<td>JP 2005050442 A</td>
<td>24.02.2005</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>