US008854884B2

(12) **United States Patent**
Kim

(10) **Patent No.: US 8,854,884 B2**
(45) **Date of Patent: Oct. 7, 2014**

(54) **NAND FLASH ARCHITECTURE WITH MULTI-LEVEL ROW DECODING**

(75) Inventor: **Jin-Ki Kim**, Kanata (CA)

(73) Assignee: **Conversant Intellectual Property Management Inc.**, Ottawa (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 192 days.

(21) Appl. No.: **13/467,491**

(22) Filed: **May 9, 2012**

(65) **Prior Publication Data**

US 2012/0218829 A1 Aug. 30, 2012

Related U.S. Application Data

(63) Continuation of application No. 12/495,089, filed on Jun. 30, 2009, now Pat. No. 8,189,390.

(60) Provisional application No. 61/157,594, filed on Mar. 5, 2009.

(51) **Int. Cl.**

G11C 16/04	(2006.01)
G11C 16/16	(2006.01)
G11C 8/10	(2006.01)
G11C 16/10	(2006.01)
G11C 8/12	(2006.01)
G11C 16/08	(2006.01)
G11C 8/14	(2006.01)
G11C 11/56	(2006.01)

(52) **U.S. Cl.**

CPC ***G11C 8/14*** (2013.01); *G11C 2211/5642* (2013.01); ***G11C 16/16*** (2013.01); ***G11C 8/10*** (2013.01); ***G11C 16/0483*** (2013.01); ***G11C 16/10*** (2013.01); ***G11C 8/12*** (2013.01); ***G11C 16/08*** (2013.01); ***G11C 11/5621*** (2013.01)

USPC **365/185.11**; 365/185.12; 365/185.13; 365/185.17; 365/230.03; 365/230.06

(58) **Field of Classification Search**

USPC 365/185.11, 185.12, 185.13, 185.17, 365/230.03, 230.06

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,473,563	A	12/1995	Suh et al.
5,621,690	A	4/1997	Jungroth et al.

(Continued)

FOREIGN PATENT DOCUMENTS

EP	0782145	7/1997
EP	1028433 A1	8/2000

(Continued)

OTHER PUBLICATIONS

Kirisawa, R. et al, a NAND Structured Cell With a New Programming Technology for Highly Reliable 5V-Only Flash EEPROM, 1990 Symposium on VLSI Technology, CH 2874-6, 90/0000-0129 1990 IEEE, Honolulu, US, 129-130, Jun. 4, 1990.

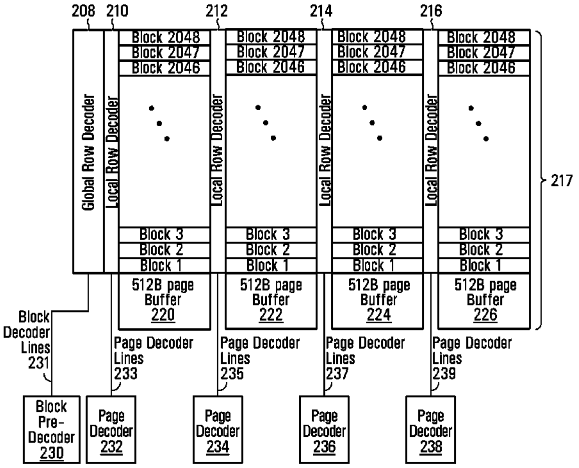
(Continued)

Primary Examiner — Vanthu Nguyen

(57) **ABSTRACT**

A NAND flash memory device is disclosed. The NAND flash memory device includes a NAND flash memory array defined as a plurality of sectors. Row decoding is performed in two levels. The first level is performed that is applicable to all of the sectors. This can be used to select a block, for example. The second level is performed for a particular sector, to select a page within a block in the particular sector, for example. Read and program operations take place to the resolution of a page within a sector, while erase operation takes place to the resolution of a block within a sector.

6 Claims, 14 Drawing Sheets



(56) References Cited

U.S. PATENT DOCUMENTS

5,896,340	A	4/1999	Wong et al.	
6,222,773	B1 *	4/2001	Tanzawa et al.	365/185.28
6,643,184	B2	11/2003	Pio	
6,662,263	B1 *	12/2003	Wong	711/103
6,735,116	B2 *	5/2004	Lee et al.	365/185.17
6,891,755	B2 *	5/2005	Silvagni et al.	365/185.13
6,950,336	B2 *	9/2005	Sowards et al.	365/185.12
7,035,162	B2	4/2006	Chung et al.	
7,079,417	B2	7/2006	Nam et al.	
7,080,192	B1 *	7/2006	Wong	711/103
7,333,369	B2 *	2/2008	Sakui et al.	365/185.28
7,406,649	B2	7/2008	Shimizume et al.	
7,423,910	B2	9/2008	Umezawa	
7,532,510	B2	5/2009	Lee et al.	
7,577,032	B2	8/2009	Umezawa	
7,577,059	B2 *	8/2009	Pyeon	365/238.5
7,778,107	B2 *	8/2010	Pyeon	365/233.5
7,821,833	B2	10/2010	Tanuma et al.	
7,940,578	B2	5/2011	Kang et al.	
2002/0186590	A1	12/2002	Lee	
2004/0047224	A1	3/2004	Ha	
2004/0246806	A1	12/2004	Ha	
2005/0226086	A1	10/2005	Sugawara	
2008/0074931	A1	3/2008	Kim et al.	
2008/0205164	A1	8/2008	Pyeon	

FOREIGN PATENT DOCUMENTS

EP	1041577		10/2000
EP	1047077	A1	10/2000
JP	2000235799		8/2000
JP	2002093182		3/2002
JP	2003517170		5/2003
JP	2004079161		3/2004
JP	2005302139		10/2005
JP	2008108382		5/2008

OTHER PUBLICATIONS

Samsung Electronics Co. Ltd., 2G x 8BIT NAND Flash Memory, K9XXG08UXM; K9GAG08U0M, 1-48, Sep. 21, 2006.

Gal, E. et al, “Algorithms and Data Structures for Flash Memories”, ACM Computing Surveys (CSUR), Vol. 37, No. 2; Jun. 2005, 138-163.

Kim, J. et al, a 120-MM2 64-Mb NAND Flash Memory Achieving 180 NS/Byte Effective Program Speed, IEEE Journal of Solid-State Circuits, Vol. 32, No. 5, May 1997, 670-680.

Imamiya, K et al, a 125-MM2 1-GB NAND Flash Memory With 10-Mbyte/S Program Speed, IEEE Journal of Solid-State Circuits, Vol. 37, No. 11, Nov. 2002, 1493-1501.

Shirota, R. et al, A 2.3 UM2 Memory Cell Structure for 16MB NAND EEPROMS, International Electron Devices Meeting 1990, Technical Digest; Dec. 1990, 103-106.

Suh, K. et al, A 3.3 V 32 MB NAND Flash Memory With Incremental Step Pulse Programming Scheme, IEEE Journal of Solid-State Circuits, Vol. 30, No. 11, Nov. 1995, 1149-1156.

Momodomi, M. et al, A 4MB NAND EEPROM With Tight Programmed VT Distribution, IEEE Journal of Solid-State Circuits, Vol. 26, Issue 4, Apr. 1991, 492-496.

Takeuchi, K. et al, A 56NM CMOS 99MM2 8GB Multi-Level NAND Flash Memory With 10MB/S Program Throughput,Solid-State Circuits, 2006 IEEE International Conference Digest of Technical Papers, Session 7, ISBN:1-4244-0079-1, 10 pages, Feb. 6, 2006.

Aritome, S. et al, A Reliable Bi-Polarity Write/Erase Technology in Flash EEPROMS, Int'l. Electron Devices Meeting, Technical Digest, 111-114, Dec. 9, 1990.

Ziaie, Kazem, “International Patent Application No. PCT/CA2010/000260,Search Report”, 1-4, Jun. 1, 2010.

* cited by examiner

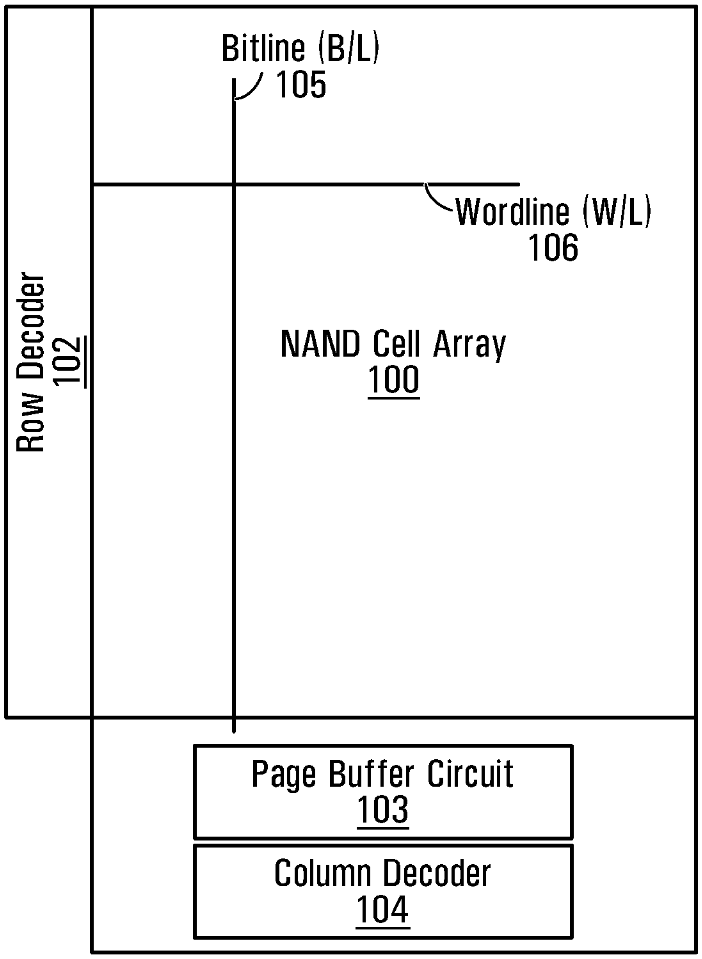

Bitline (B/L)
105
Wordline (W/L)
106
Row Decoder
102
NAND Cell Array
100
Page Buffer Circuit
103
Column Decoder
104

FIG. 1

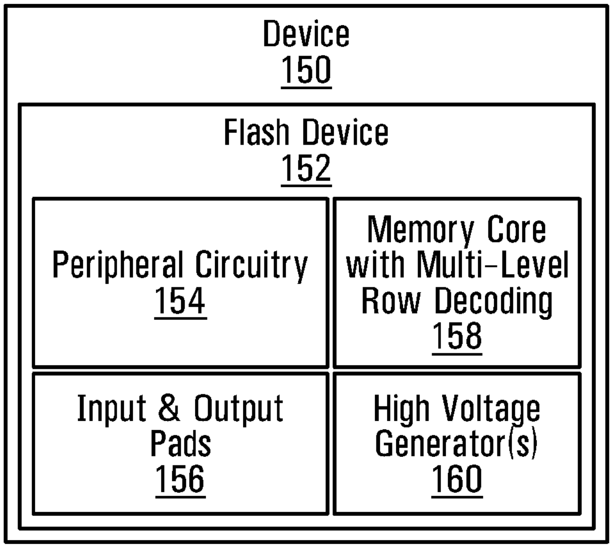

Device
150
Flash Device
152
Peripheral Circuitry
154
Memory Core with Multi-Level Row Decoding
158
Input & Output Pads
156
High Voltage Generator(s)
160

FIG. 2

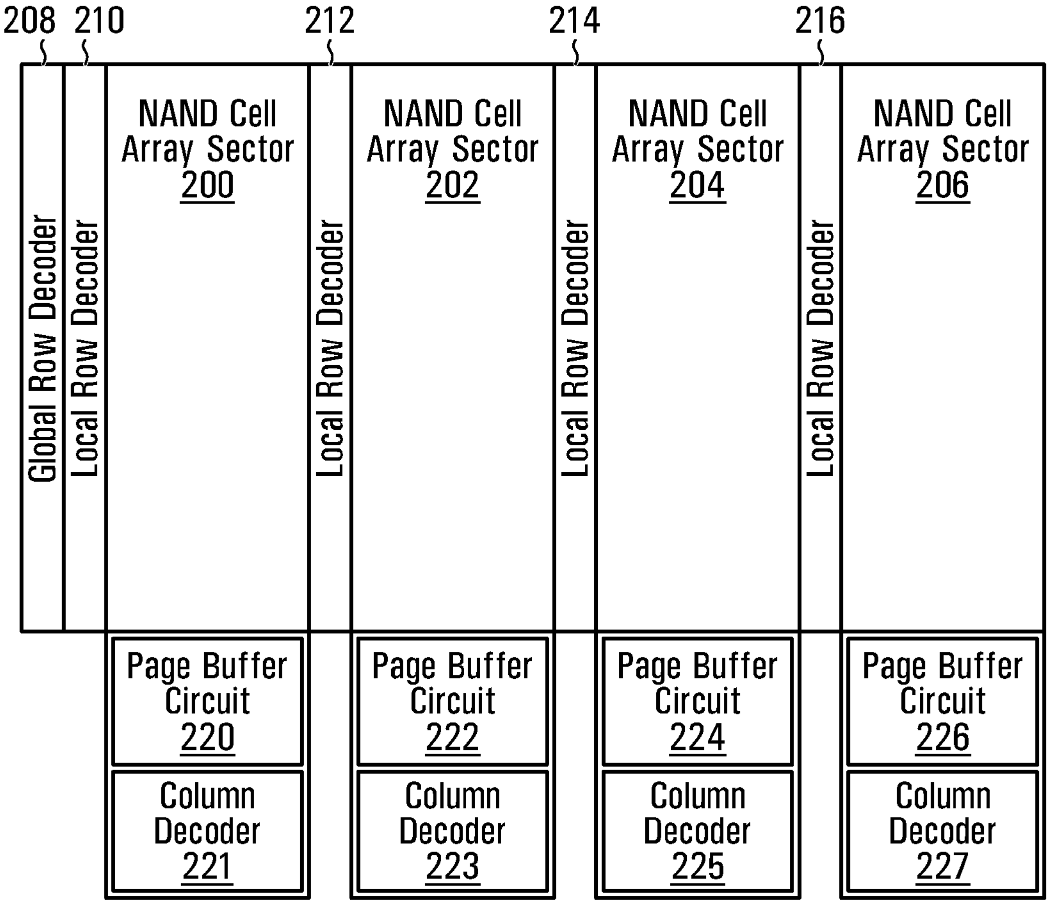

208
210
212
214
216
Global Row Decoder
Local Row Decoder
NAND Cell Array Sector
200
Local Row Decoder
NAND Cell Array Sector
202
Local Row Decoder
NAND Cell Array Sector
204
Local Row Decoder
NAND Cell Array Sector
206
Page Buffer Circuit
220
Column Decoder
221
Page Buffer Circuit
222
Column Decoder
223
Page Buffer Circuit
224
Column Decoder
225
Page Buffer Circuit
226
Column Decoder
227

FIG. 3

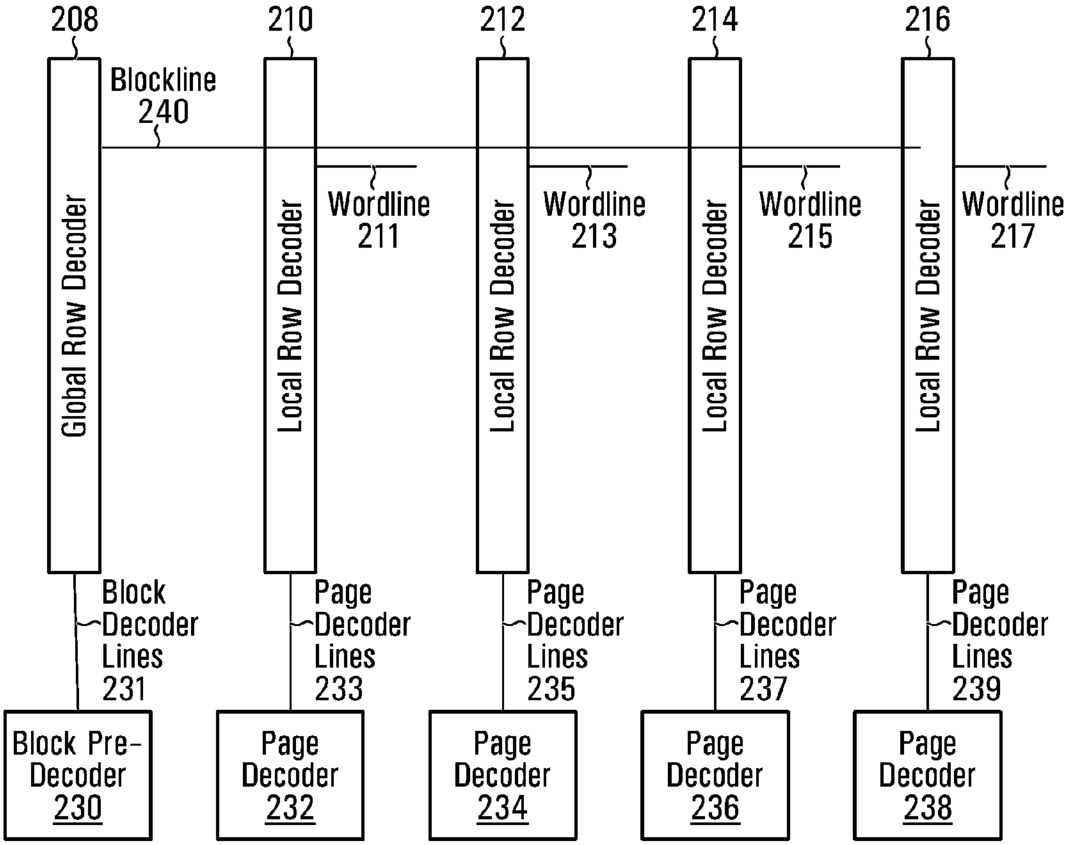

208
210
212
214
216
Blockline 240
Global Row Decoder
Local Row Decoder
Wordline 211
Local Row Decoder
Wordline 213
Local Row Decoder
Wordline 215
Local Row Decoder
Wordline 217
Block Decoder Lines 231
Page Decoder Lines 233
Page Decoder Lines 235
Page Decoder Lines 237
Page Decoder Lines 239
Block Pre-Decoder 230
Page Decoder 232
Page Decoder 234
Page Decoder 236
Page Decoder 238

FIG. 4

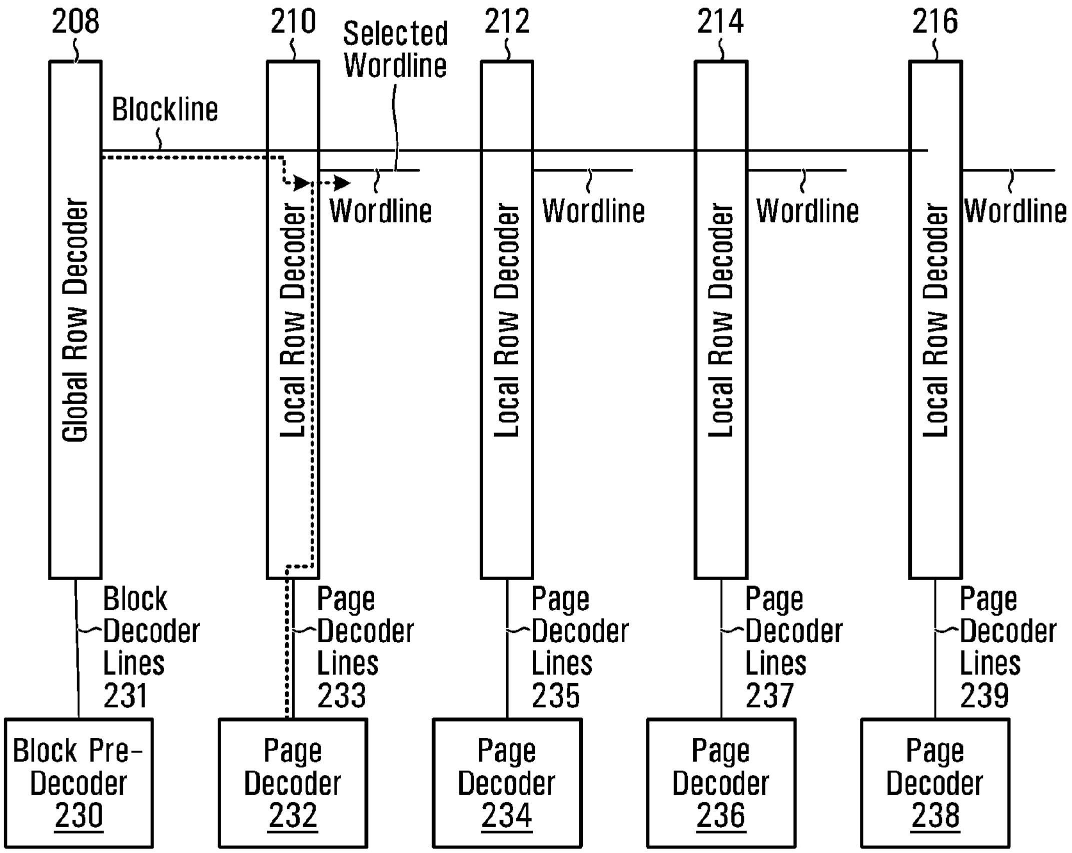

208
210
212
214
216
Selected Wordline
Blockline
Wordline
Wordline
Wordline
Wordline
Global Row Decoder
Local Row Decoder
Local Row Decoder
Local Row Decoder
Local Row Decoder
Block Decoder Lines 231
Page Decoder Lines 233
Page Decoder Lines 235
Page Decoder Lines 237
Page Decoder Lines 239
Block Pre-Decoder 230
Page Decoder 232
Page Decoder 234
Page Decoder 236
Page Decoder 238

FIG. 5

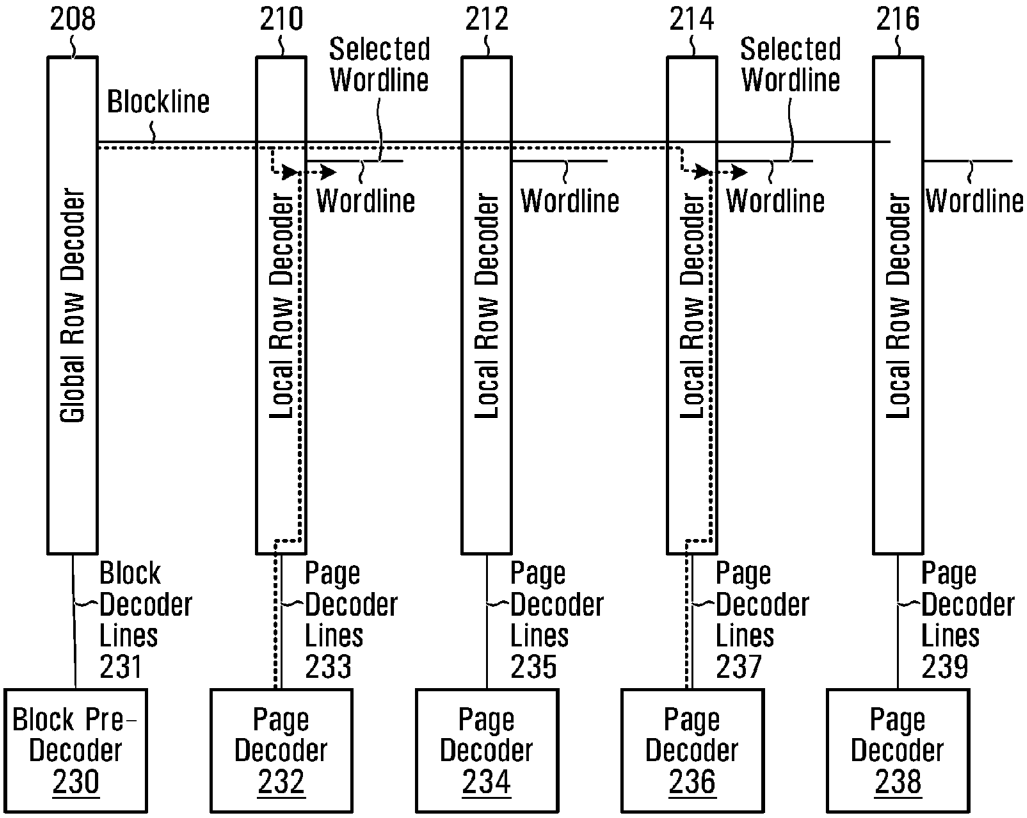

208
210
212
214
216
Selected Wordline
Selected Wordline
Blockline
Wordline
Wordline
Wordline
Wordline
Global Row Decoder
Local Row Decoder
Local Row Decoder
Local Row Decoder
Local Row Decoder
Block Decoder Lines 231
Page Decoder Lines 233
Page Decoder Lines 235
Page Decoder Lines 237
Page Decoder Lines 239
Block Pre-Decoder 230
Page Decoder 232
Page Decoder 234
Page Decoder 236
Page Decoder 238

FIG. 6

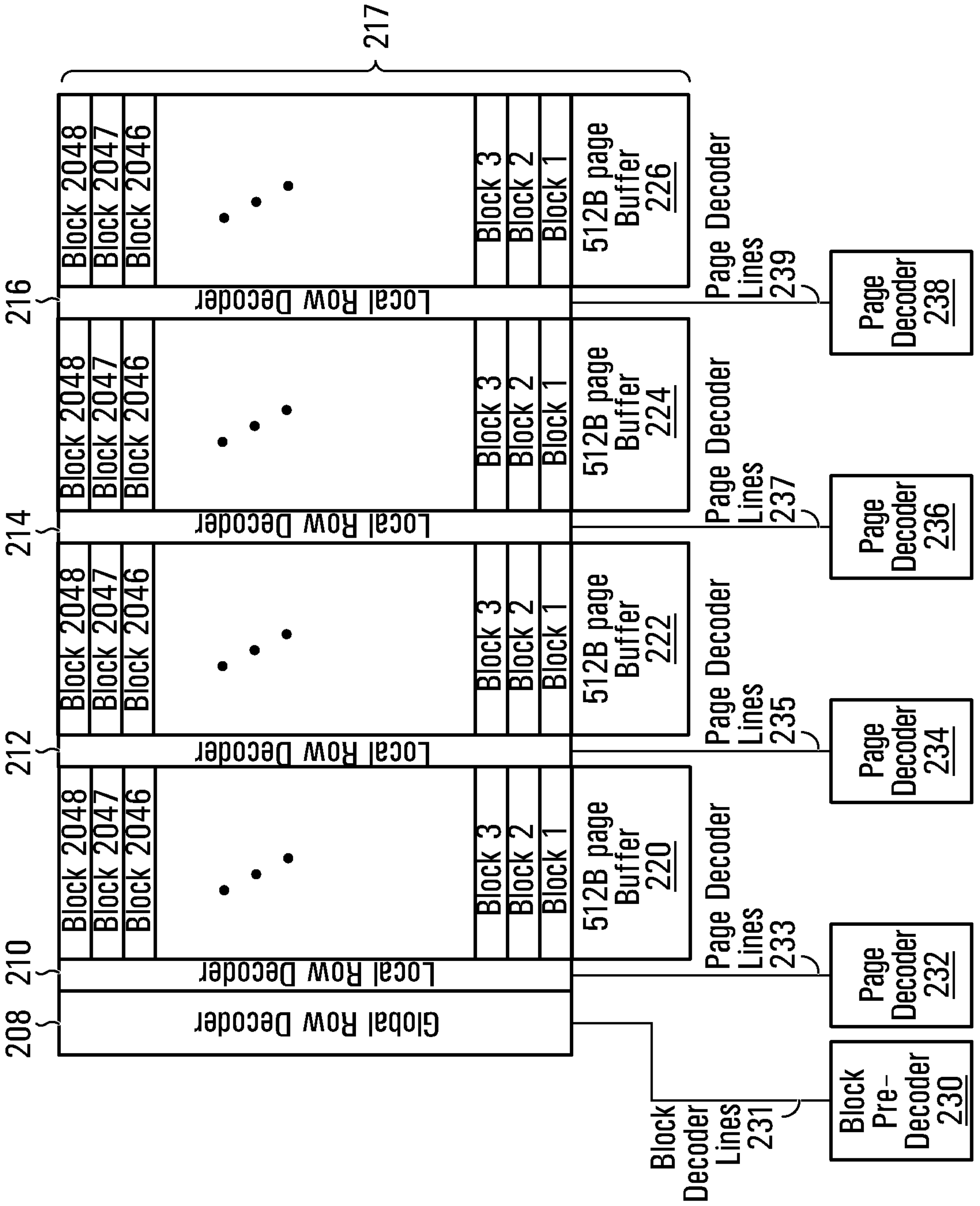

217
208
Global Row Decoder
210
Local Row Decoder
212
Local Row Decoder
214
Local Row Decoder
216
Local Row Decoder
Block 2048
Block 2047
Block 2046
Block 3
Block 2
Block 1
512B page Buffer 220
512B page Buffer 222
512B page Buffer 224
512B page Buffer 226
Block Decoder Lines 231
Page Decoder Lines 233
Page Decoder Lines 235
Page Decoder Lines 237
Page Decoder Lines 239
Block Pre-Decoder 230
Page Decoder 232
Page Decoder 234
Page Decoder 236
Page Decoder 238

FIG. 7

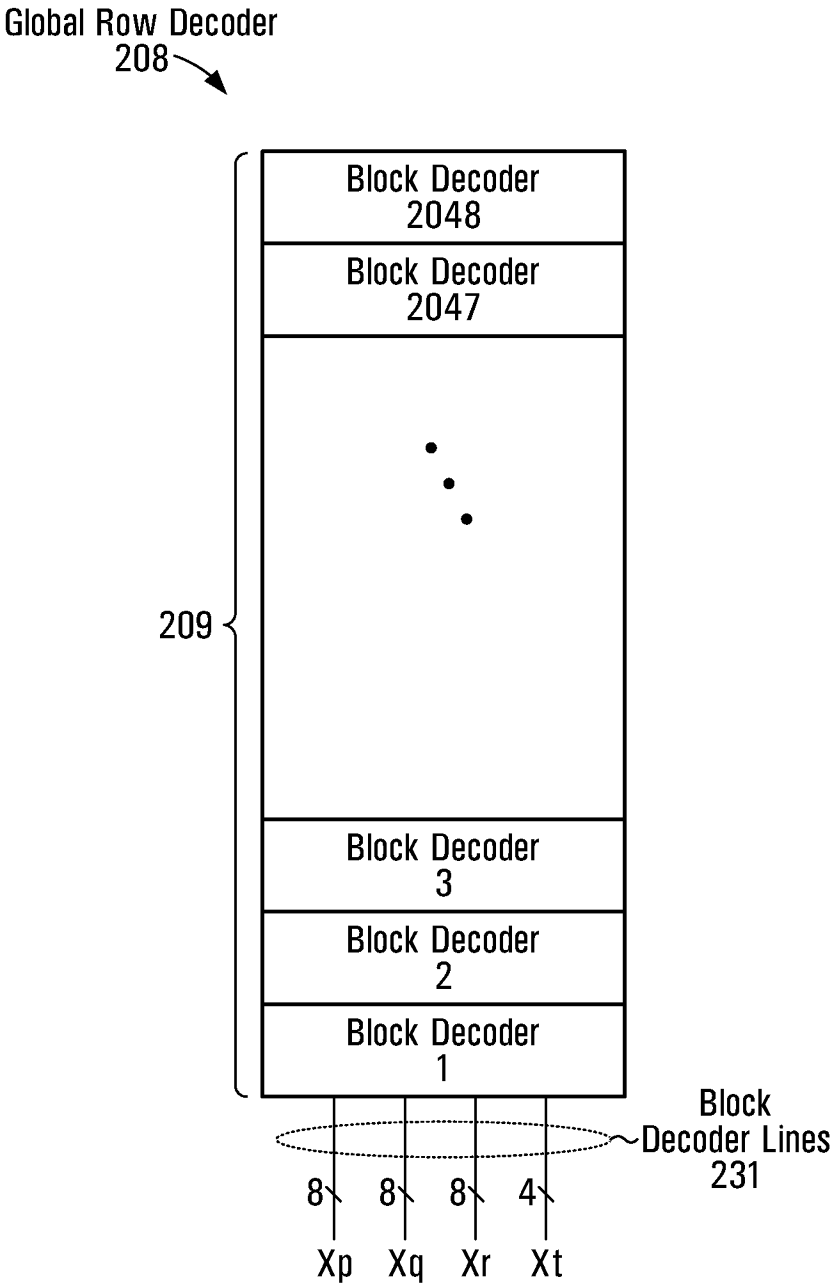

Global Row Decoder
208
Block Decoder
2048
Block Decoder
2047
209
Block Decoder
3
Block Decoder
2
Block Decoder
1
Block
Decoder Lines
231
8
8
8
4
Xp
Xq
Xr
Xt

FIG. 8

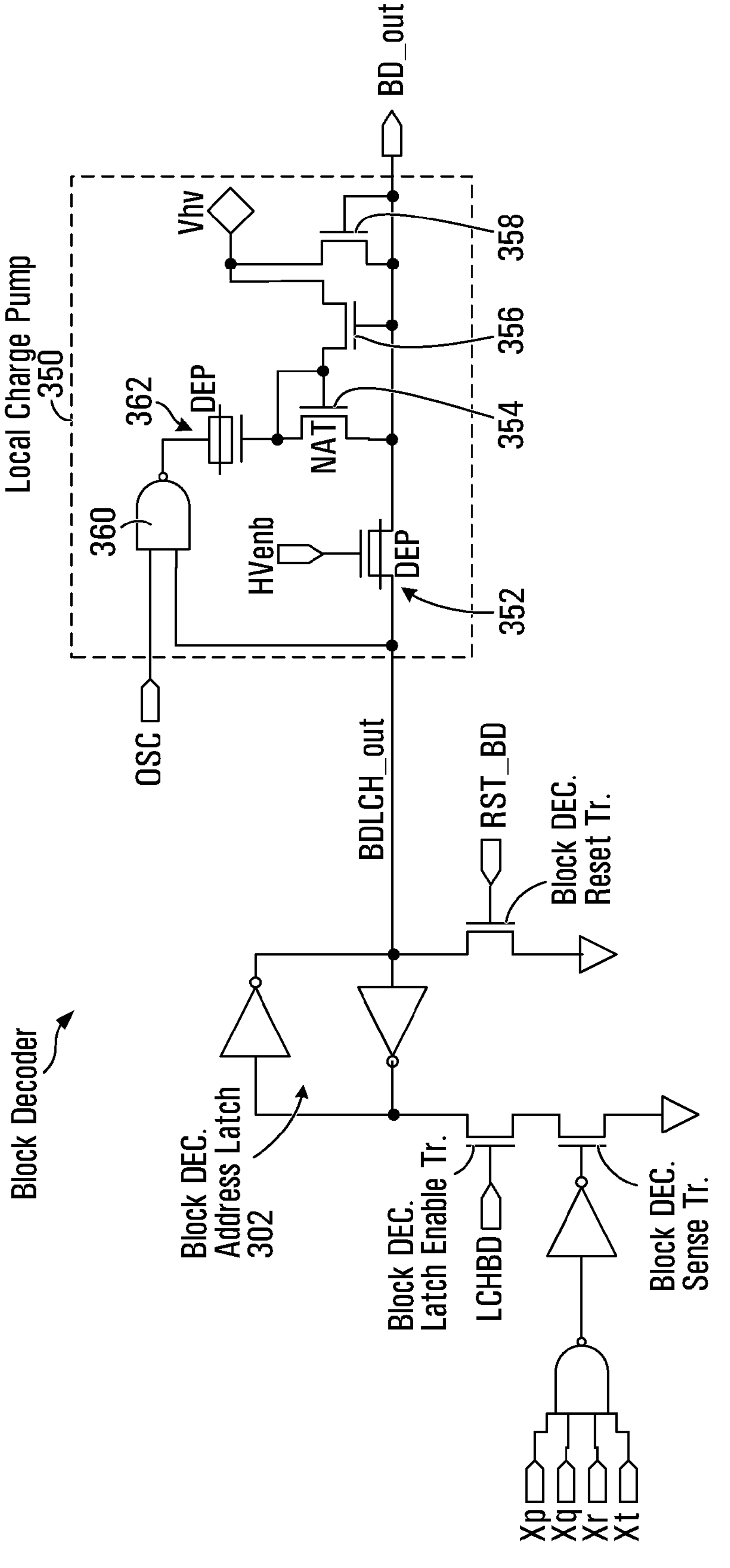

BD_out
Local Charge Pump
350
Vhv
358
356
354
362
DEP
NAT
360
HVenb
DEP
352
OSC
BDLCH_out
RST_BD
Block DEC.
Reset Tr.
Block Decoder
Block DEC.
Address Latch
302
Block DEC.
Latch Enable Tr.
LCHBD
Block DEC.
Sense Tr.
Xp
Xq
Xr
Xt

FIG. 9

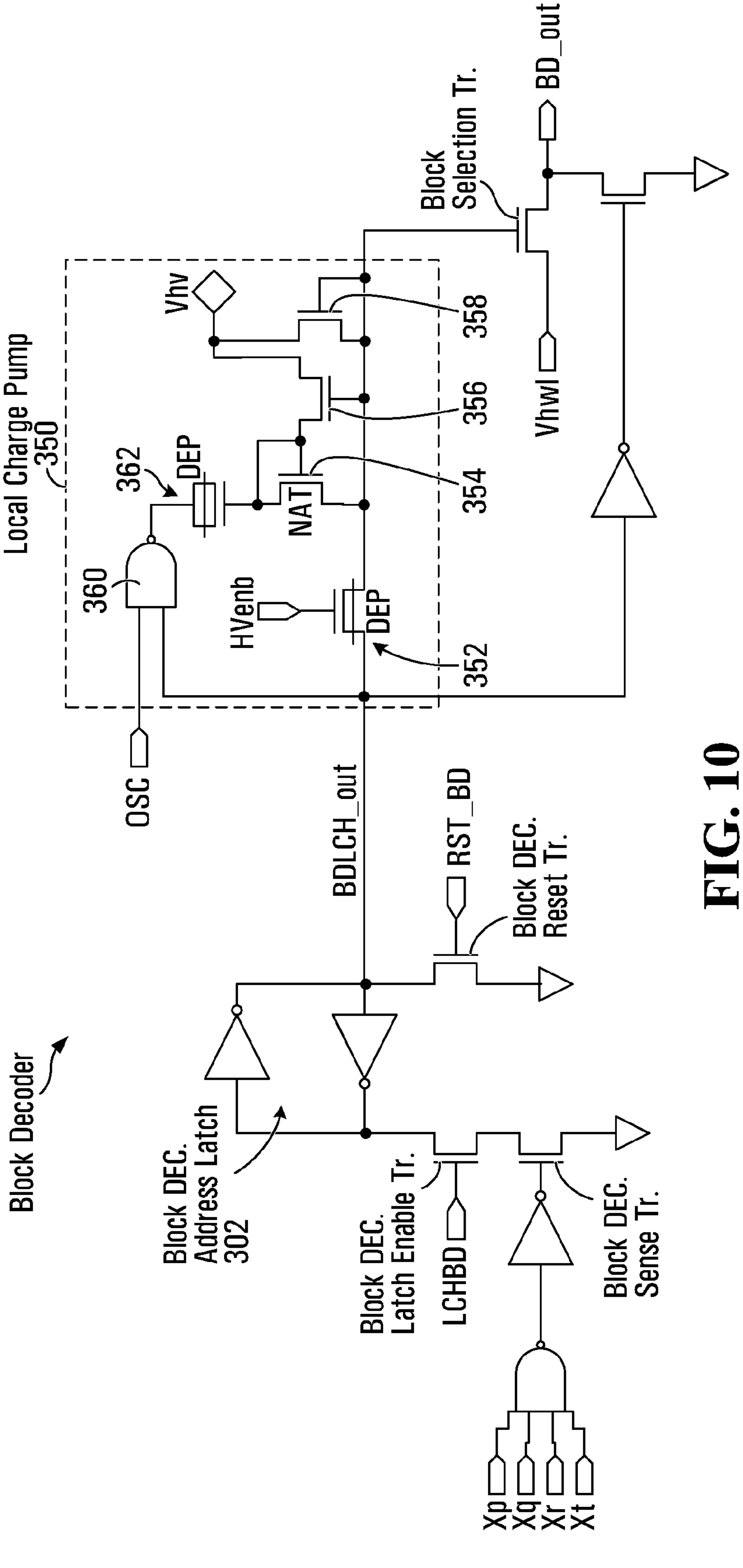

Block Decoder
Block DEC. Address Latch 302
Block DEC. Latch Enable Tr.
LCHBD
Block DEC. Sense Tr.
Xp
Xq
Xr
Xt
Block DEC. Reset Tr.
RST_BD
BDLCH_out
OSC
Local Charge Pump 350
360
362
DEP
NAT
HVenb
DEP
352
354
356
358
Vhv
Vhwl
Block Selection Tr.
BD_out

FIG. 10

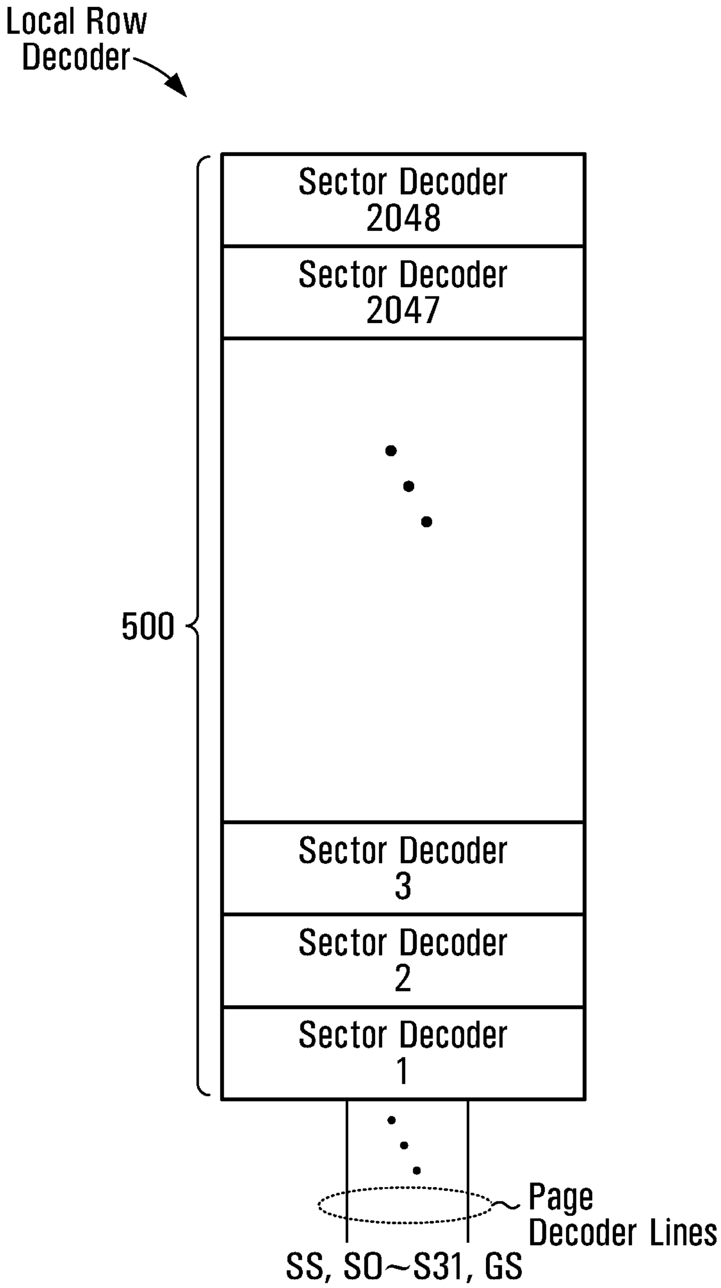

Local Row Decoder
Sector Decoder 2048
Sector Decoder 2047
500
Sector Decoder 3
Sector Decoder 2
Sector Decoder 1
Page Decoder Lines
SS, S0~S31, GS

FIG. 11

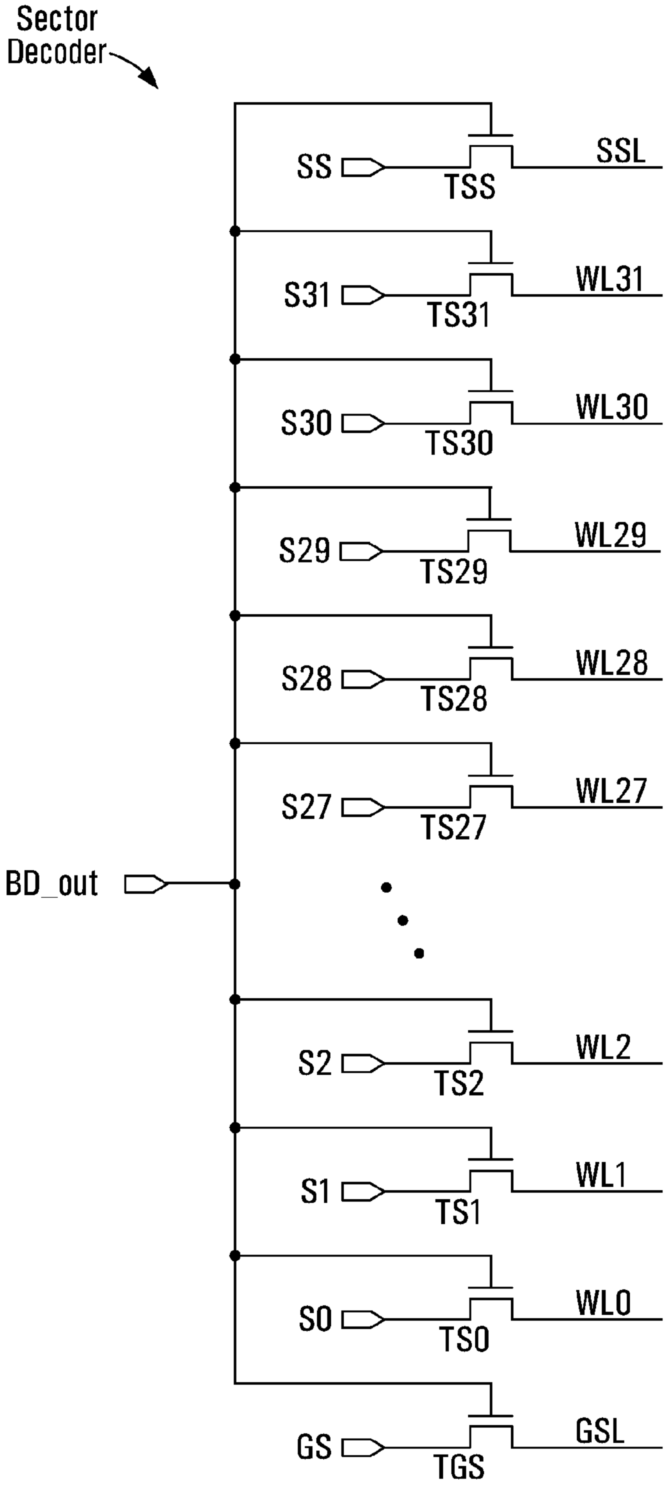

Sector
Decoder
SS
TSS
SSL
S31
TS31
WL31
S30
TS30
WL30
S29
TS29
WL29
S28
TS28
WL28
S27
TS27
WL27
BD_out
S2
TS2
WL2
S1
TS1
WL1
S0
TS0
WL0
GS
TGS
GSL

FIG. 12

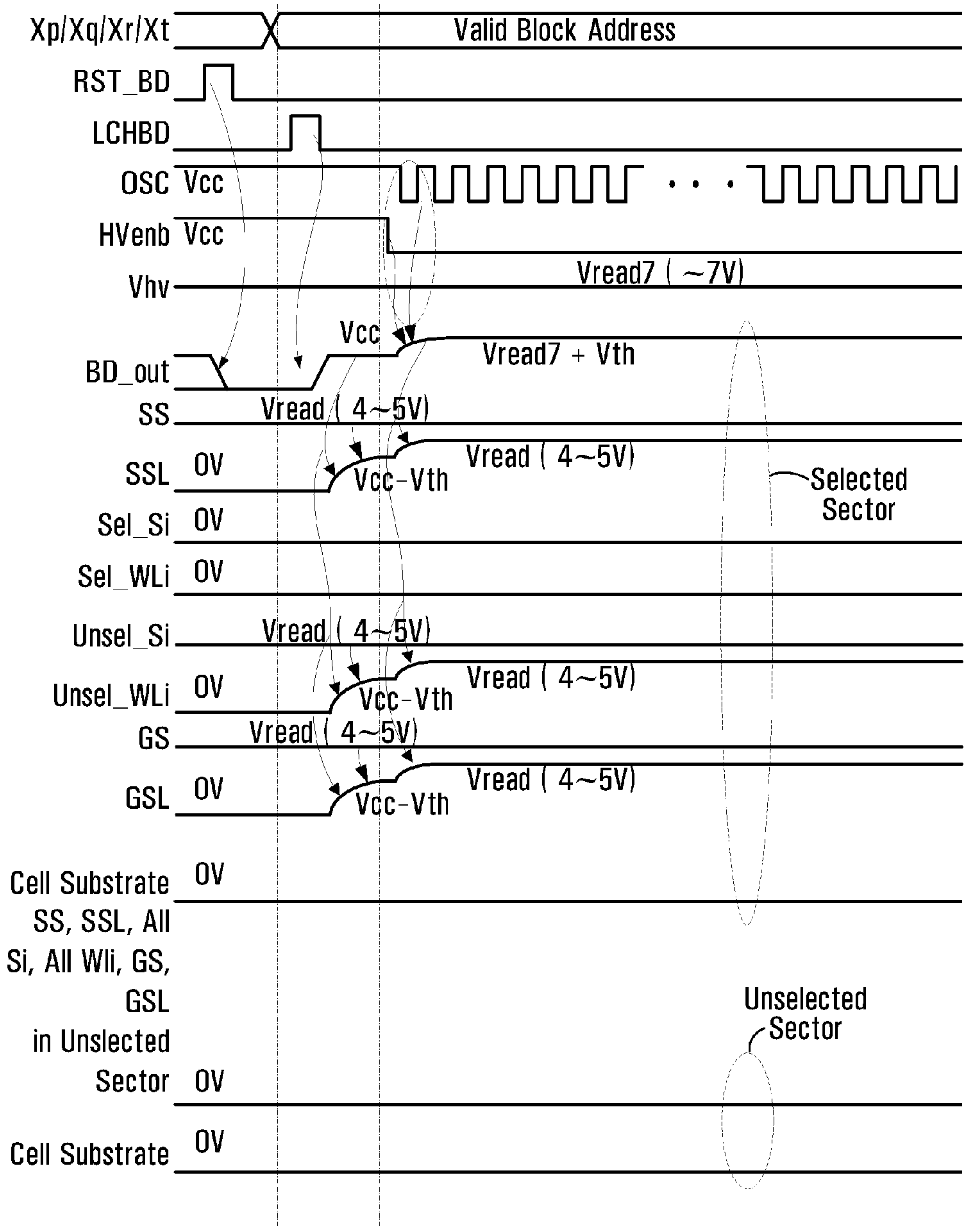

Xp/Xq/Xr/Xt
Valid Block Address
RST_BD
LCHBD
OSC
Vcc
HVenb
Vcc
Vhv
Vread7 (~7V)
BD_out
Vcc
Vread7 + Vth
SS
Vread (4~5V)
SSL
0V
Vcc-Vth
Vread (4~5V)
Selected Sector
Sel_Si
0V
Sel_WLi
0V
Unsel_Si
Vread (4~5V)
Unsel_WLi
0V
Vcc-Vth
Vread (4~5V)
GS
Vread (4~5V)
GSL
0V
Vcc-Vth
Vread (4~5V)
Cell Substrate
0V
SS, SSL, All Si, All Wli, GS, GSL in Unslected Sector
0V
Unselected Sector
Cell Substrate
0V

FIG. 13

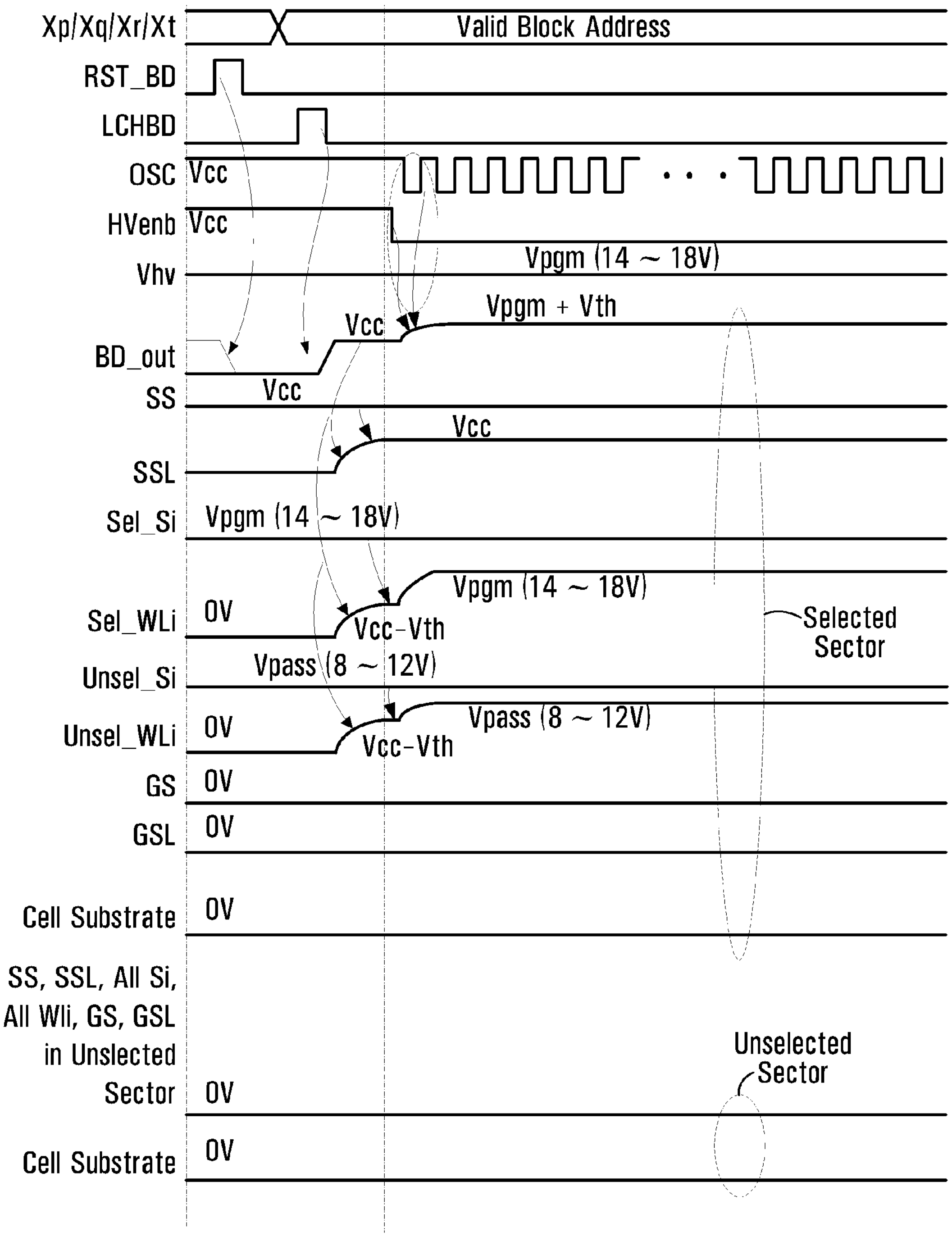

Xp/Xq/Xr/Xt
Valid Block Address
RST_BD
LCHBD
OSC
Vcc
HVenb
Vcc
Vhv
Vpgm (14 ~ 18V)
Vpgm + Vth
Vcc
BD_out
SS
Vcc
Vcc
SSL
Sel_Si
Vpgm (14 ~ 18V)
Vpgm (14 ~ 18V)
Sel_WLi
0V
Vcc-Vth
Selected Sector
Unsel_Si
Vpass (8 ~ 12V)
Unsel_WLi
0V
Vpass (8 ~ 12V)
Vcc-Vth
GS
0V
GSL
0V
Cell Substrate
0V
SS, SSL, All Si, All Wli, GS, GSL in Unslected Sector
0V
Unselected Sector
Cell Substrate
0V

FIG. 14

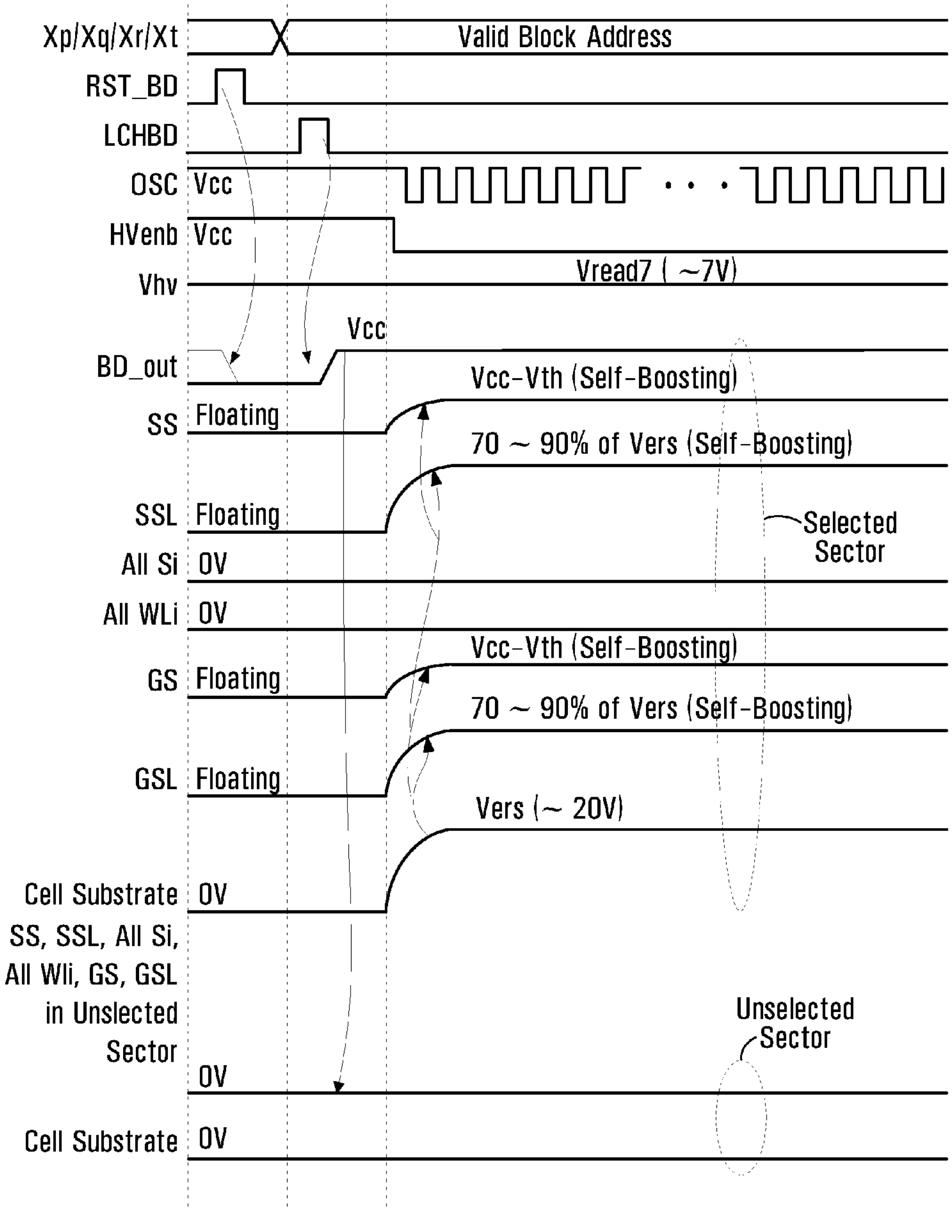

Xp/Xq/Xr/Xt
Valid Block Address
RST_BD
LCHBD
OSC
Vcc
. . .
HVenb
Vcc
Vhv
Vread7 (~7V)
Vcc
BD_out
Vcc-Vth (Self-Boosting)
SS
Floating
70 ~ 90% of Vers (Self-Boosting)
SSL
Floating
Selected Sector
All Si
0V
All WLi
0V
Vcc-Vth (Self-Boosting)
GS
Floating
70 ~ 90% of Vers (Self-Boosting)
GSL
Floating
Vers (~ 20V)
Cell Substrate
0V
SS, SSL, All Si, All Wli, GS, GSL in Unslected Sector
0V
Unselected Sector
Cell Substrate
0V

FIG. 15

NAND FLASH ARCHITECTURE WITH MULTI-LEVEL ROW DECODING

CROSS REFERENCE TO RELATED APPLICATION

This application claims the benefit of U.S. Provisional Application No. 61/157,594, filed Mar. 5, 2009, and U.S. patent application Ser. No. 12/495,089 filed Jun. 30, 2009, the content which is incorporated herein by reference in their entirety.

FIELD

A NAND flash memory device is disclosed.

BACKGROUND

In conventional NAND flash memory, erasing is performed on a per-block basis. In contrast, read and program operation takes place on a per-page basis.

BRIEF DESCRIPTION OF THE DRAWINGS

Example embodiments will now be described with reference to the attached drawings in which:

FIG. **1** is a block diagram of a typical memory core architecture in NAND flash memory;

FIG. **2** is a block diagram of an example NAND flash device within which one of the NAND core architectures described herein might be implemented;

FIG. **3** is a block diagram of memory core architecture for NAND flash memory provided by an example embodiment;

FIG. **4** is a block diagram of a memory core architecture in a NAND flash memory provided by an example embodiment;

FIGS. **5** and **6** show single page read and multiple page read operation for the example embodiment of FIG. **3**, respectively;

FIG. **7** is a block diagram of a memory core architecture in NAND flash memory in accordance with an example embodiment;

FIG. **8** is a block diagram of a global row decoder in accordance with an example embodiment;

FIG. **9** is a circuit diagram of an example implementation of a single block decoder of FIG. **8**;

FIG. **10** is a block diagram of another example implementation of a single block decoder of FIG. **8**;

FIG. **11** is a block diagram of a local row decoder in accordance with an example embodiment;

FIG. **12** is a circuit diagram of an example implementation of a single sector decoder of FIG. **10**;

FIG. **13** is a timing diagram for read in accordance with an example embodiment;

FIG. **14** is a timing diagram for program in accordance with an example embodiment; and

FIG. **15** is a timing diagram for erase in accordance with an example embodiment.

DETAILED DESCRIPTION

FIG. **1** illustrates a memory core architecture in a NAND flash memory. The NAND flash memory core comprises a NAND memory cell array **100**, a row decoder **102** and a page buffer circuit **103** and column decoder **104**. The row decoder **102** is connected to the NAND memory cell array **100** by a set of wordlines, only one wordline **106** being shown in FIG. **1** for simplicity. The page buffer circuit **103** is connected to the NAND memory cell array **100** through a set of bitlines, only one bitline **108** being shown in FIG. **1** for simplicity.

The cell array structure of the NAND flash memory comprises a set of n erasable blocks. Each block is subdivided into m programmable pages (rows).

Erasing for the memory core architecture of FIG. **1** is performed on a per-block basis. In contrast, read and program operation takes place on a per-page basis.

A NAND flash memory having the core architecture of FIG. **1** flash suffers from at least three limitations. First, bits can only be programmed only after erasing a target memory array. Second, each cell can only sustain a limited number of erasures, after which it can no longer reliably store data. In other words, there is a limitation in the number of erase and program cycle to cells (i.e. endurance, typically 10,000~100,000 cycles). Third, the minimum erasable array size is much bigger than the minimum programmable array size. Due to these limitations, sophisticated data structures and algorithms are implemented to effectively use flash memories.

When the flash controller requests data write or data modification into even only a small portion of the page, typically the block containing a page to be modified will be reprogrammed to one of free (empty) blocks declared by an erase-unit reclamation process. In this case, valid pages containing original data in the original block are copied to the selected free block. After that, the new block having modified data in a page with original data in the rest of pages is remapped to the valid block address by a virtual mapping system in the flash controller. The original block is now obsolete and will be declared as a free block by the erase-unit reclamation process after it has been erased.

The limited number of erase-program cycles (endurance) limits the lifetime of a flash device. It would be advantageous to have a lifetime that is as long as possible, and this depends on the pattern of access to the flash device. Repeated and frequent rewrites to a single cell or to a small number of cells will bring the onset of failures soon and so end the useful lifetime of the device quickly.

Moreover, in a flash memory system having multiple flash devices, if there is significantly uneven use among devices in the flash memory system, one device will reach an end of lifetime at a time when other devices have significant life left in them. When the one device reaches an end of life time, the entire memory system may have to be replaced, and this greatly reduces the life time of the flash memory system.

If rewrites can be evenly distributed to all cells of the device, the onset of failures will be delayed as much as possible, maximizing the lifetime of the device. To extend the device lifetime by even use across all the cells of the device, many wear-levelling techniques and algorithms have been proposed and implemented in flash memory systems.

The cell arrays of NAND flash have been so miniaturized over the course of time that they have reached the point where any further reduction in process technology is expected to drastically reduce the maximum number of erase-program cycles.

According to one broad aspect, provided is a NAND flash memory core with multi-level row decoding.

According to another broad aspect, provided is a NAND flash memory device comprising: peripheral circuitry, input/output pads, and a high voltage generator; a NAND flash memory core comprising: a NAND memory cell array comprising a plurality of rows by a plurality of columns, the cells arranged into a plurality of sectors, each sector comprising the cells of a plurality of said columns; the cells arranged into a plurality of blocks, each block comprising cells of a plurality of said rows; the NAND memory cell array configured for erasure to a resolution of one block within one sector, and configured for read and program to a resolution of one row within one sector.

According to another broad aspect of, provided is a method in a NAND flash memory core comprising: performing multi-level row decoding. Due to the size mismatch between read/program and erase, the block copy operations described above introduce unnecessary program operations because unaffected data in pages of the block are reprogrammed (copied) to the new block along with the modified data. There could be a dramatic extension to the device lifetime if the minimum erasable array size is smaller than an entire block.

FIG. **2** is a block diagram of a device **150** containing a NAND flash memory device **152**. The NAND flash memory device **152** has a memory core with multi-level row decoding, generally indicated at **158**. In addition, the NAND flash memory device **152** has a peripheral circuitry **154**, input and output pads **156**, and high voltage generator(s) **160**. The peripheral circuitry **154** may, for example, comprise one or more of input and output buffers for address and data, input buffers for control and command signals, state machine including command decoder, address counter, row and column per-decoder, and status registers. Device **150** may be any device having a use for NAND flash memory device **152**. Specific examples include a mobile device, a memory stick, a camera, a solid state disk drive, and an MP3 player. Flash device **152** may form a permanent part of the device **150**, or may be removable. Detailed example implementations of the memory core with multi-level row decoding are provided below. More generally, any memory core with multi-level row deciding is contemplated. The cell array is formed of sectors, each sector comprising a plurality of columns of cells. The cells also form blocks, each block comprising a plurality of rows, also referred to as pages. In some embodiments, multi-level row decoding involves performing a first level of row decoding for all of the sectors, and for each sector, performing a second level of row decoding only for that sector. In some embodiments, erasing within the memory core is performed to a resolution of one block within one sector, and read and program operations take place to a resolution of one row within one sector.

Referring now to FIG. **3**, shown is a core architecture provided by an example embodiment. The core architecture includes a NAND memory cell array that is implemented as at least two NAND memory cell array sectors, hereinafter simply “sectors”, there being four sectors **200**,**202**,**204**,**206** shown in the illustrated example. The NAND memory cell array is formed of a plurality of blocks which in turn are formed of pages, also referred to as rows. The cells of each sector of the NAND memory cell array are also arranged in columns (not shown). Row decoding functionality is provided by a global row decoder **208** that performs row decoding to the level of blocks, in combination with a set of local row decoders **210**,**212**,**214**,**216** that perform decoding to the level of a page within a block selected by the global row decoder. More generally, the global row decoder **208** performs a first level of row decoding to select a subset of the plurality of rows. In example embodiments described herein in detail, the selectable subsets are contiguous blocks, but this need not be the case in all implementations. The local row decoders **210**, **212**, **214**, **216** perform a second level of row decoding to select a row within the subset of the plurality of rows selected by the global row decoder **208**. The local row decoders **210**,**212**,**214**,**216** include one local row decoder associated with each respective sector **200**,**202**,**204**,**206** and perform page selection local to the associated sector. Page buffer functionality is implemented with four page buffer circuits **220**,**222**,**224**,**226**, one per sector **200**,**202**,**204**,**206**. Column decoder functionality is implemented with four column decoders **221**,**223**,**225**,**227**, one per sector **200**,**202**,**204**,**206**.

Read operation is performed to the resolution of a page within a block within a sector. Program operation is also performed to the resolution of a page within a block within a sector. However, a page within a block within a sector is erased before it is programmed. Erase operation is performed to the resolution of a block within a sector.

For a read operation, the global row decoder **208** is used to select a block of the plurality of blocks of the NAND memory cell array. Sector selection is performed by performing column selection with the page buffer circuit and column decoder associated with the desired memory sector. This can be achieved, for example, by a memory controller enabling the associated page buffer circuit and column decoder and/or sending column decoder signals to the associated page buffer circuit and column decoder. Page selection is performed by the local row decoder associated with the selected sector. In this manner, a selected page within a selected block within a selected sector can be read. During a read operation, the data of the selected page within the selected block and within the selected sector is sensed and latched into sense amplifier (not shown) and page buffer circuit of the selected sector. After that the data stored in the page buffer circuit is sequentially read out through the associated column decoder and, for example, stored in a global buffer (not shown).

For an erase operation, the global row decoder **208** is used to select a block of the plurality of blocks of the NAND memory cell array. Sector selection is performed by performing column selection with the page buffer circuit and column decoder associated with the desired memory sector. Then an appropriate erase signal is applied. In this manner, a selected block within a selected sector can be erased.

For a program operation, the global row decoder **208** is used to select a block of the plurality of blocks of the NAND memory cell array. Sector selection is performed by performing column selection with the page buffer circuit and column decoder associated with the desired memory sector. Page selection is performed by the local row decoder associated with the selected sector. Then, the contents of the page buffer circuit associated with the selected sector are programmed to the selected page within the selected block within the selected sector. During a program operation, the input data (for example from a global buffer circuit, not shown) is sequentially loaded into the page buffer circuit of the selected sector via the associated column decoder. The input data latched in the page buffer circuit is then programmed into the selected page of the selected sector.

FIG. **4** shows another example of a core architecture provided by an example embodiment. This example embodiment is similar to FIG. **2** and like components have been labelled using like reference numbers. The example embodiment of FIG. **4** has a block pre-decoder **230** connected through block decoder lines **231** to the global row decoder **208**. The global row decoder **208** is connected to the memory array through a plurality of blocklines, one per block although only one blockline **240** is shown in the illustrated example. The blocklines are commonly connected to all local row decoders **210**, **212**,**214**,**216**. Each local row decoder **210**,**212**,**214**,**216** is also driven by a respective set of page decoder lines **233**,**235**,**237**, **239** from a respective page decoder **232**,**234**,**236**,**238**. Each local row decoder **210**,**212**,**214**,**216** is connected to the corresponding sector though a plurality of wordlines, only one shown per sector indicated at **211**,**213**,**215**,**219**.

In operation, to select a particular block, the block predecoder **230** converts an input, for example from a memory controller, into an appropriate signal on block decoder lines **231**. The global row decoder **208** selects one of block lines. To select a particular page within a particular sector, the page decoder of the associated sector (one of page decoders **232**, **234**,**236**,**238**) is enabled and used to select the particular page within the selected block.

An example of single sector selection is depicted in FIG. **5** which shows selection of a page within a block for local row detector **210**. In some example embodiments, the circuit is configured to allow multiple page decoders to be enabled simultaneously. In such example embodiments, within the selected block, selection of a respective page within multiple sectors can be performed by enabling multiple page decoders. An example of multiple sector selection is shown in FIG. **6** which shows selection of a row within a block by each of row decoders **210** and **214**. The blockline selects one of blocks within all of the sectors while page decoder lines select one of pages (i.e. wordlines) within the selected block in each sector.

In this example embodiment, a read operation will result in one or multiple page buffer circuits containing read-out data. The contents of these page buffer circuits are then individually read out. A program operation will result in the contents of one or multiple page buffer circuits being programmed simultaneously. Typically, this will have been preceded by a series of write to page buffer operations by which the multiple page buffer circuits are written to sequentially.

FIG. **7** shows more detailed core architecture provided by an example embodiment wherein again this example embodiment is similar to FIG. **3** and like components have been labelled using like reference numbers. In FIG. **7**, as in other block diagrams, certain components (such as, for example, column decoders) are not shown so as not to obscure features of example embodiments. In the example, a NAND core (this can be an entire device core architecture, a plane or a bank) comprises four sectors and the page size of each sector is 512 bytes. More generally, the page size of each sector is at least one byte. In this example, there are 2048 blocks collectively indicated at **217**. Each block is split into four sectors. The global row decoder **208** is connected to all of the local row decoders **210**, **212**, **214**, **216** in common by 2048 blocklines (not shown), one per block. Each block has 32 pages.

An example implementation of the global row decoder **208** of FIG. **7** is depicted in FIG. **8**. The global row decoder **208** has a respective block decoder for each block, namely 2048 block decoders collectively indicated at **209** corresponding to the number of blocks. Each of the block decoders is connected to the block decoder lines **231**. In this example, the block decoder lines **231** comprises lines xp,xq,xr,xt for carrying block decoder address signals Xp,Xq,Xr and Xt. Xp,Xq,Xr and Xt are the pre-decoded lines. Xp corresponds to Address A_0~A_2. Xq corresponds to Address A_3~A_5. Xr corresponds to Address A_6~A_8. Xt corresponds to Address A_9~A_{10} Each block decoder drives a respective blockline (not shown). The block decoder associated with the block indicated by the address signals on block decoder lines **231** drives the respective blockline to be in a select state, and all other blocklines are in a de-select state.

An example circuit implementation of a single block decoder is depicted in FIG. **9**. It is noted that there are many variations on circuit implementation for the block decoder, and that such variations should be readily apparent to one skilled in the art.

The circuit has a block decoder address latch **302** having a latch output BDLCH_out that is reset to 0V when the RST_BD is high (actually short pulse) and latched when the LCHBD is high (which may be a short pulse) with valid predecoded address signals of Xp, Xq, Xr and Xt (block decoder lines) received at NAND logic gate **303**. Detailed timing information is shown in FIGS. **12**, **13** and **14** described subsequently.

The block decoder has a local charge pump **300** that is a high voltage switching circuit to provide voltages during read, program and erase operations. Local charge pump **300** includes a depletion mode n-channel pass transistor **352**, a native n-channel diode-connected boost transistor **354**, a high breakdown voltage n-channel decoupling transistor **356**, a high breakdown voltage n-channel clamp transistor **358**, a NAND logic gate **360**, and a capacitor **362**. NAND logic gate **360** has one input terminal for receiving the latch output BDLCH_out and another input terminal for receiving control signal OSC, for driving one terminal of capacitor **362**. Pass transistor **352** is controlled by the complement of signal HVen, referred to as HVenb. The common terminals of decoupling transistor **356** and clamp transistor **358** are coupled to high voltage Vhv.

The final output signal BD_out of the each block decoder is commonly connected to all of the local row decoders, for example as depicted in FIG. **9**.

The operation of local charge pump **350** will now be described. During a read operation, HVenb is at the high logic level and OSC is maintained at the low logic level. Therefore, circuit elements **362**, **354**, **356** and **358** are inactive, and the output terminal BD_out reflects the logic level appearing on BDLCH_out. During a program operation, HVenb is at the low logic level, and OSC is allowed to oscillate between the high and low logic levels at a predetermined frequency. If the latch output BDLCH_out is at the high logic level, then capacitor **362** will repeatedly accumulate charge on its other terminal and discharge the accumulated charge through boost transistor **354**. Decoupling transistor **356** isolates Vhv from the boosted voltage on the gate of boost transistor **354**. Clamp transistor **358** maintains the voltage level of output terminal BD_out at about Vhn+Vth, where Vth is the threshold voltage of clamp transistor **358**. The local charge pump **300** shown in FIG. **9** is one example circuit which can be used to drive signals to a voltage levels higher than the a supply voltage VCC, but persons skilled in the art will understand other charge pump circuits can be used with similar or equal effectiveness. Table 1 below shows example bias conditions for the local charge pump **300** during read and program operations.

TABLE 1

	Read		Program	
	Selected	Unselected	Selected	Unselected
BDLCH_out	Vcc	Vss	Vcc	Vss
Hvenb	Vss	Vss	Vss	Vss
OSC	Oscillation	Oscillation	Oscillation	Oscillation
Vhn	Vread7 (~7 V)	Vread7 (~7 V)	~Vpgm (14 V~18 V)	~Vpgm (14 V~18 V)
BD_out	Vread7 (~7 V) + Vth	Vss	Vpgm (14 V~18 V) + Vth	Vss

The output signal BD_out of the block decoder is raised to Vhv when the block decoder latch output BDLCH_out is Vcc, HVenb is 0V and the OSC is oscillating.

Referring to FIG. **10**, another example of a block decoder uses a block selection transistor. Vhwl is a high voltage source which has various levels based on operations. In this example embodiment, the drivability of BD_out is determined by the size of the block selection transistor, not the local charge pump. Therefore this circuit provides stronger drivability in the case of higher number of local row decoders in the NAND memory core.

FIG. **11** depicts an example of a local row decoder. The local row decoder has 2048 sector decoders collectively indicated at **500**, one per block. These are referred to as sector decoders because a page within a sector is selected, as opposed to a page within the overall memory array. The inputs to the local row decoder are page decoder lines which in the illustrated example include string select (SS), wordline select signals S**0**-S**31** (one per wordline), and ground select (GS). The wordline select signals S**0**-S**31** commonly connect to the sector decoders.

Referring now to FIG. **12**, an example circuit for a single sector decoder will be described. String select line SSL, wordlines WL**0** to WL**31** and ground select line GSL are driven by common signals of SS, S**0** to S**31** and GS through pass transistors TSS, TS**0** to TS**31** and TGS which are commonly controlled by the output signal BD_out of the associated block decoder. The page decoder lines, namely string select signal SS, ground select signal GS and common string decode signals S**0** to S**31**, are provided by the page decoder.

In operation, for the block that is selected, the BD_out input of all the corresponding sector decoders is activated. This will include one sector decoder for that block in each sector. For all the remaining blocks that were not selected, the BD_out of all the corresponding sector decoders is deactivated. For a sector for which an operation is to be performed, within that sector, all of the sector decoders are commonly controlled by common page decoder lines. There may be one or more sectors for which an operation is to be performed. For a sector for which no operation is to be performed, all of the common page decoder lines are inactive such that all of the commonly connected sector decoders are inactive. For a sector decoder that is selected both by a BD_out in a select state, and by page decoder lines that are active, the sector decoder causes a respective selected wordline (one of WL**0** through WL**31**) to be in a selected state while the remaining wordlines are in a de-selected state.

Table 2 shows an example set of bias conditions to the block decoder, local row decoder and NAND cell array during read, program and erase. It is to be understood that all values may vary based on cell characteristics and process technology.

TABLE 2

	Read	Program	Erase
Selected Global Row Decoder: BD_out	Vread7 (~7 V) + Vth	Vpgm + Vth	Vcc
Unselected Global Row Decoder: BD_out	Vss (0 V)	Vss (0 V)	Vss (0 V)
	Local Decoder in Selected Sector		
SS	Vread (4~5 V)	2 V~Vcc	Floating and Vcc-Vth (Self-booting)
Selected Si	Vss (0 V)	Vpgm (14 V~18 V)	Vss (0 V)
Unselected Si	Vread (4~5 V)	Vpass (8 V~12 V)	—
SSL	Vread (4~5 V)	2 V~Vcc	Floating & Self-boosting (70~90% of Vers)
Selected WLi	Vss (0 V)	Vpgm (14 V~18 V)	Vss (0 V)
Unselected WLi	Vread (4~5 V)	Vpass (8 V~12 V)	—
GS	Vread (4~5 V)	Vss (0 V)	Floating and Vcc-Vth (Self-booting)
GSL	Vread (4~5 V)	Vss (0 V)	Floating & Self-boosting (70~90% of Vers)
Bitlines	Pre-charged & Sensed	Vss (0 V) for program & Vcc for Program Inhibit	Clamp to Vers-0.6 V
Cell Substrate	Vss (0 V)	Vss (0 V)	Vers (~20 V)
	Local Decoder in Unselected Sector		
SS	Vss (0 V)	Vss (0 V)	Vss (0 V)
All Si	Vss (0 V)	Vss (0 V)	Vss (0 V)
GS	Vss (0 V)	Vss (0 V)	Vss (0 V)
SSL	Vss (0 V)	Vss (0 V)	Vss (0 V)
All WLi	Vss (0 V)	Vss (0 V)	Vss (0 V)
GSL	Vss (0 V)	Vss (0 V)	Vss (0 V)
Bitlines	Vss (0 V)	Vss (0 V)	Vss (0 V)
Cell Substrate	Vss (0 V)	Vss (0 V)	Vss (0 V)

With this example embodiment, either single sector operation or multiple sector operation can be performed. For read operations, a single sector page read and up to a four sector page read in parallel can be performed. More generally, the maximum number of sectors that can be read in parallel is determined by the number of sectors in the NAND memory core. For program operations, a single sector page program and up to a four sector page program in parallel can be performed. More generally, the maximum number of sectors that can be programmed in parallel is determined by the number of sectors in the NAND memory core. For erase, a single sector block erase and up to a four sector block erase in parallel can be performed. More generally, the maximum number of sectors that can be erased in parallel is determined by the number of sectors in the NAND memory core.

FIG. **13** shows an example of read operation timing in accordance with some example embodiments. The voltage bias conditions during read for this example are defined in Table 2 above for this example. All signals in each unselected sectors remain at 0V. This operation timing is based on the use of the block decoder shown in FIG. **9**.

FIG. **14** shows an example of program operation timing in accordance with some example embodiments. The voltage bias conditions during program for this example are defined in Table 2 above for this example. All signals each unselected sectors remain at 0V. This operation timing is based on the use of the block decoder shown in FIG. **9**.

FIG. **15** shows an example of erase operation timing in accordance with some example embodiments. The voltage bias conditions during erase are defined in Table 2 above for this example. All signals in unselected sectors remain at 0V. This operation timing is based on the use of the block decoder shown in FIG. **9**.

In FIGS. **13**, **14**, **15**, Sel_Si is short form for any "selected" Si input signal (where Si={S_0 . . . S_{31}}). Unsel_Si is short form for any "unselected" Si input signal (where Si={S_0 . . . S_{31}}). Sel_WLi is short form for any "selected" word line signal (where WLi={WL_0 . . . WL_{31}}). Unsel_WLi is short form for any "unselected" word line signal (where WLi={WL_0 . . . WL_{31}}).

It will be understood that when an element is herein referred to as being "connected" or "coupled" to another element, it can be directly connected or coupled to the other element or intervening elements may be present. In contrast, when an element is herein referred to as being "directly connected" or "directly coupled" to another element, there are no intervening elements present. Other words used to describe the relationship between elements should be interpreted in a like fashion (i.e., "between" versus "directly between", "adjacent" versus "directly adjacent", etc.).

Certain adaptations and modifications of the described embodiments can be made. Therefore, the above discussed embodiments are considered to be illustrative and not restrictive.

I claim:

1. A method for performing mutli-level row decoding in a NAND flash memory core, the NAND flash memory core including a plurality of memory cell array sectors and a plurality of blocks, each of the plurality of blocks being divided into portions each belonging to one of the plurality of memory cell array sectors, the method comprising:

performing, using a global row decoder, a first level of row decoding to select a block spanning the plurality of memory cell array sectors; and

performing, using a first local row decoder associated with a first memory cell array sector of the plurality of memory cell array sectors, a second level of row decoding to select a page within the selected block and the first memory cell array sector.

2. The method of claim **1**, wherein the second level of row decoding further comprises selecting, using a second local row decoder associated with a second memory cell array sector of the plurality of memory cell array sectors, an additional page within the selected block and the second memory cell array sector, the selection of the additional page being performed in parallel with the selection of the page.

3. The method of claim **1** wherein:

each memory cell array sector of the plurality of memory cell array sectors includes wordlines and a cell substrate; and

all wordlines and cell substrates of memory cell array sectors for which a page has not been selected are biased to a common voltage.

4. The method of claim **3** wherein the common voltage is substantially 0V.

5. The method of claim **1**, wherein the plurality of memory cell array sectors comprise at least four memory cell array sectors.

6. The method of claim **1**, wherein:

the first memory cell array sector includes a plurality of columns and a plurality of rows; and

the second level of row decoding comprises transferring contents of a page buffer circuit of the NAND flash memory core to a selected row of the first memory cell array sector.

* * * * *