New Encoding Method for Low Power Sequential Access ROMs

Seong-Ik Cho, Ki-Sang Jung, Sung-mi Kim, Namhee You, and Jong-Yeol Lee

Abstract—This paper propose a new ROM data encoding method that takes into account of a sequential access pattern to reduce the power consumption in ROMs used in applications such as FIR filters that access the ROM sequentially. In the proposed encoding method, the number of 1's, of which the increment leads to the increase of the power consumption, is reduced by applying an exclusive-or (XOR) operation to a bit pair composed of two consecutive bits in a bit line. The encoded data can be decoded by using XOR gates and D flip-flops, which are usually used in digital systems for synchronization and glitch suppression. By applying the proposed encoding method to coefficient ROMs of FIR filters designed by using various design methods, we can achieve average reduction of 43.7% over the unencoded original data in the power consumption, which is larger reduction than those achieved by previous methods.

Index Terms—ROM, sequential access, D flip-flop, exclusive OR, power consumption, FIR filter

I. INTRODUCTION

ROMs can be widely used in digital systems such as digital filters, FFTs, direct digital synthesizers (DDS), microprocessors, and digital signal processors. Usually, a ROM consumes the greatest percentage of the total power because it has many highly capacitive lines that

are frequently accessed in an SoC. Furthermore, the power consumption in a ROM tends to increase as the memory cell array size of a ROM continues to increase.

In the memory cell array of a ROM, the most of power is dissipated in word lines and bit lines during each access because the word line and the bit lines are highly capacitive lines due to a large number of memory cell transistors [1].

The authors of [1] survey various methods by which the power consumption of a ROM can be reduced including data encoding methods that focus on the reduction of the capacitance in bit lines and word lines. The bit line and the word line capacitances can be reduced by minimizing the number of 1's in the ROM data since 1's are implemented as transistors of which the junction capacitance and the gate capacitance are the sources of bit line and word line capacitances [2]. In [3] and [4], data encoding methods that invert column or row data to reduce the number of 1's in ROM data are proposed, respectively. In those methods, if the data contains more 1's than 0's, row or column data is inverted to reduce the number of 1's and hence, the load capacitance. In column inversion [3], a bit line is implemented with the inverted column data when the bit line has more 1's than 0's. An inverter is added to each inverted bit line. A word line is implemented with the inverted row data in row inversion [4] if the number of 1's is greater than that of 0's in a row. To indicate that a word line is inverted, a flag bit, that is which is '1' when the word line is inverted, is added to each word line, which means that row inversion needs an additional bit line. At the output, applying exclusive-or (XOR) operations between an inverted word line data and its flag bit, which is '1', produces the original data since

XORing with '1' is equivalent to an inversion. The authors of [5] propose word line collapsing that transforms ROM data by collapsing all rows having identical data pattern, which results in the reductions of the number of 1's and the memory cell area. Since word lines that have the same data are collapsed into a word line, the address decoder must be optimized to map different addresses to one memory word line. Although the number of 1's and the area of memory cell array are reduced by applying word line collapsing, the power consumption in each access to a word line remains almost the same because word line collapsing does not change the data pattern on which the power consumption depends by leaving one word line among the word lines that have the same data pattern.

This paper proposes a new encoding method that can reduce the number of transistors in a memory cell array of a ROM accessed sequentially by reducing the number of 1's in ROM data. In a sequentially accessed ROM with the size of N , the N locations are accessed sequentially from the first location with address 0 to the last location with address $(N-1)$. Examples of sequential access ROMs are used to store the coefficients of a FIR filter and the twiddle factors of a FFT. Unlike the previous encoding methods, the proposed encoding takes the sequential access pattern into account to reduce the number of 1's and the power consumption in a memory cell array, which leads to more reductions in the number of 1's and the power consumption than the previous works.

The proposed method selectively encodes a bit line data by applying an XOR operation to a bit pair of two consecutive bits if the encoding reduces the number of transistors in the bit line. The effect of the proposed encoding is enlarged if the proposed encoding is applied after sign inversion which eliminates sign changes. By reducing the number of transistors in a memory cell array based on the proposed encoding, we can decrease junction capacitances of bit lines and gate capacitances of word lines, which achieves reduction in power consumption compared to a ROM implemented with unencoded data. The original data can be reconstructed from the encoded data by decoding process exploiting a D flip-flop (F/F) and an XOR gate at an output of the ROM. Although the proposed method is applied to NOR-type ROMs for a concise explanation in this paper, the

proposed method can also be applied to NAND-type ROMs achieving the reduction of power consumption due to the decrease of capacitances and on-resistances.

This paper is organized as follows. Section II describes the memory cell array architecture and the power consumption in a ROM. After section III and IV present the proposed ROM data encoding method and the experimental results, respectively, the paper is concluded in Section V.

II. ROM ARCHITECTURE AND POWER CONSUMPTION

A sequentially accessed ROM used in a synchronous digital system is shown in Fig. 1, where the output data of the ROM is synchronized by using D F/Fs. A sequential address, which is generated by using a counter, is input to an address decoder to select a word line. After a word line is selected, sense amplifiers (S/As) amplify the word line data, which is fed to D F/Fs that are usually used for the synchronization of signals and the glitch elimination.

1. Memory Cell Array Architecture

Memory cells designed in CMOS technologies usually utilize either a NAND array or a NOR array. In a NOR array memory cell a transistor lies between a bit line and a ground (GND) wire.

To reduce the power consumption in a memory cell array, reducing the number of transistors in the memory

Fig. 1. Sequentially accessed ROM with D F/Fs used for synchronization and glitch suppression.

Fig. 2. Memory cell array of a NOR-type ROM (a) Data, (b) schematic of the ROM.

cell array is important in that the capacitance of word lines and bit lines is proportional to the number of transistors. Since bit data '1' is implemented as a transistor between a bit line and a GND wire in general NOR-type ROM architecture as shown in Fig. 2, the number of NMOS transistors is equal to that of 1's. In Fig. 2(b), the voltage at node A is V_{DD} when word line $WL0$ is not selected. However, when $WL0$ is selected, which means $WL0$ is '1', the voltage at node A changes to GND by discharging the bit line.

2. POWER CONSUMPTION

Power consumption in a memory cell can be broken down into dynamic consumption ($P_{Dynamic}$), consumption due to short circuit current (P_{SC}) and static consumption (P_{Static}) [4].

$$P_{Total} = P_{Dynamic} + P_{SC} + P_{Static} \quad (1)$$

The main contributors of P_{Static} are leakage current due to reverse-biased diodes at the junctions and subthreshold current. P_{SC} can occur when both NMOS and PMOS devices of a logic gate are turned on during input and output transition. $P_{Dynamic}$, which constitutes the most significant component of power dissipation, is expressed as follows.

$$P_{Dynamic} = f \cdot C_{(BL+WL)} \cdot V_{Swing} \cdot V_{DD} \quad (2)$$

where $C_{(BL+WL)}$ is total parasitic junction capacitances and gate capacitances of bit lines and word lines, V_{Swing} is the voltage change seen at bit lines and word lines, and f is the frequency of the transition. In Eq. (2), we can find that the dynamic power consumption in a memory cell

array is proportional to the capacitance of bit lines and word lines.

III. PROPOSED ROM DATA ENCODING AND DECODING

In this section, the proposed ROM data encoding method by which the number of 1's can be reduced is described. The proposed encoding applies XOR operations over consecutive bits in a bit line, which produces zero when 1's appear consecutively. The encoded data is decoded by using D F/Fs with XOR gates.

1. Encoding and Decoding Based on XOR Operation

The proposed method encodes a ROM data by applying XOR operations to two consecutive bits in a bit line. The proposed encoding assumes that a '0' precedes the first bit. The bits in a bit line are encoded as follows. A bit pair, which is composed of two consecutive bits, is encoded to '1' when the pair has both '0' and '1'. Otherwise, a bit pair is encoded to '0'. The pairs (1, 0) and (0, 1) are encoded to '1' while the pairs (0, 0) and (1, 1) are encoded to '0'.

An example of the proposed encoding is shown in Fig. 3(a), where a bit line data '0111111101' is encoded to '0100000011'. As can be seen in Fig. 3(a), by encoding a bit line data that has small a number of bit changes, we can reduce the number of 1's in the data. In the example of Fig. 3(a) the number of 1's is reduced from nine to three. The reduction in the number of 1's leads to the decrease of the bit line capacitance. Therefore, by using the proposed encoding we can reduce the power consumption that is proportional to the amount of capacitive load.

The encoded data is decoded by using D F/Fs and XOR gates when the data is read as shown in Fig. 3(b) and (c). Since the data is read sequentially, the bit stored in $(i+1)$ -th memory cell is XORed with the decoding result of i -th cell stored in a D F/F. The operation of decoding block in Fig. 3(b) can be readily explained by using the example in Fig. 3(c). At the beginning of sequential read, the D F/F is cleared according to the value assumed when encoding the data, which is '0' in Fig. 3(a). Because the first encoded bit is '0', the first

Fig. 3. Proposed ROM data encoding and decoding with a D F/F and an XOR gate (a) An XOR encoding example of a bit line, (b) a decoding block, (c) a $2^N \times M$ -bit ROM with the proposed XOR encoding, (d) a decoding example.

output of the D F/F remains unchanged. The next encoded bit, which is ‘1’, makes the input of D F/F toggle to ‘1’. Therefore, the second output of the D F/F is ‘1’. The next seven outputs remain as ‘1’ because the next seven encoded bits are all 0’s. After 11 clock cycles, all the data is decoded.

For some data that has the large number of bit changes, the proposed encoding may increase the number of 1’s as shown in Fig. 4(a). To prevent the increase of the number of 1’s, the proposed encoding is selectively applied to the bit lines where it can reduce the number of 1’s. Fig. 4(b) shows an example having bit line data of which the encoding results have more 1’s than the unencoded bit lines. The proposed encoding increases the number of 1’s when applied to the first and the third bit lines in Fig. 4(b). In this example, the proposed XOR encoding is applied only to the second and the fourth bit lines as shown in Fig. 4(c).

2. Enhanced XOR encoding with sign inversion

The proposed XOR encoding is less effective for the implementation of a ROM which stores two’s

Fig. 4. (a) Example where the number of 1’s increases after encoding, (b) a ROM data example, (c) a sequentially accessed ROM implemented by selectively applying the proposed XOR encoding to the data in (b).

Table 1. Number of 1’s in encodings of FIR filter coefficients

Filter bandwidth	Total number of 1’s	Number of 1’s due to sign change	Percentage of 1’s due to sign change (%)
$0.10f_s^*$	1579	258	16.3
$0.12f_s$	1708	348	20.4
$0.21f_s$	1859	574	30.9
$0.23f_s$	1978	664	33.6
$0.31f_s$	2119	884	41.7

* f_s is a sampling frequency.

complement values such as the FIR filter coefficients that have many sign changes. For example, when two values “0000000001000001” and “1111111110011110”, which are two’s complement representations of 0.002 and -0.003, respectively, are stored in consecutive locations in a ROM, the second value is encoded as “111111111011111”, where the number of 1’s increases after encoding. The effect of sign changes on the number of 1’s in encoding results is shown in Table 1, where the numbers of 1’s in the encoding of FIR filter coefficients are shown. As can be seen in Table 1, the number of 1’s due to the sign change is up to 41.7% of the total number of 1’s generated in encoding.

To eliminate the sign changes which cause the generation of 1’s, the signs of data are unified by applying sign inversion before applying XOR encoding. In sign inversion, the numbers of positive and negative row data are counted and the bits of the minority data are inverted. To indicate the inversion of each datum, a flag bit is added as shown in Fig. 5(b) which is the sign inversion result of Fig. 5(a) where sign changes two

Fig. 5. Example of sign inversion (a) Original data with two sign changes, (b) sign inversion result of the original data, (c) data after XOR encoding is applied to (b), (d) XOR encoding result without sign inversion.

times. When XOR encoding is applied to the sign inverted data in Fig. 5(b), the number of 1's is more reduced compared with the encoding result Fig. 5(d) that is generated by applying XOR encoding without sign inversion.

Fig. 6 shows the four possible cases of sign inversion of two consecutive rows in a ROM data. For example, both rows are inverted in **Case 4** of Fig. 6(b), which means the j -th bits in those rows are $\sim a$ and $\sim b$ after sign inversion. XOR encoding results of the cases in Fig. 6(b) are shown in Fig. 6(c), where, for example, the XOR encoding result of the j -th bit in **Case 2** is $a \oplus \sim b$. Flag bits are also encoded as shown in **Case 3**, where two flag bits '1' and '0' are encoded to '1'. Fig. 6(d) and (e) show examples of the possible four cases in sign inversion and their XOR encoding results, respectively.

Decoding logic for XOR encoding with sign inversion is shown in Fig. 7(a), where XOR gates are exploited with D F/Fs. For an encoded bit line, the bit from memory cell, the encoded flag bit, and the decoding result of the previous row stored in a D F/F are XORed to complete decoding. As shown in Fig. 7(b) and (c), the decoding logic works for all four cases generating the original bit 'b'. Fig. 7(c) shows the operation of the decoding logic in **Case 3**, where the flag bit is encoded to '1' and the encoded result of the j -th bit in the $(i+1)$ -th row is $\sim a \oplus b$. Since the decoding result of the j -th bit in

Fig. 6. XOR encoding of two consecutive rows with four possible cases of sign inversion (a) Original data, (b) data after sign inversion, (c) XOR encoding results of the $(i+1)$ -th rows in (b), (d) examples of four possible cases, and (e) XOR encoding results of the examples in (d). “ $\oplus$ ” represents an XOR operation and “x” is a don't-care condition.

Fig. 7. Decoding of XOR encoding with sign inversion (a) Decoding logic, (b) decoding of the $(i+1)$ -th rows in Fig. 6(c), (c) an illustration of decoding logic operation in **Case 3**.

the i -th row is 'a', the decoding result of the j -th bit in $(i+1)$ -th row is found by calculating $a \oplus (\sim a \oplus b) \oplus 1$, where $(\sim a \oplus b)$ and '1' are the encoding values of the j -th bit and the flag bit in the current row, respectively. To retrieve the original data from an unencoded bit line, an XOR operation is applied between the bit from memory cell and the decoded flag bit by using the decoding logic in Fig. 3(b).

IV. EXPERIMENTAL RESULTS

To compare the power consumption reductions achieved by the column inversion [3], the row inversion [4], the word line collapsing [5] and the proposed encoding, we model ROMs on a schematic level using 0.18 μm CMOS process with the data obtained by applying the encodings and the transformation to the coefficients of FIR filters. We simulate the ROM models and estimate power consumptions by using Cadence Spectre that is a transistor-level circuit simulator. Table 2 summarizes the descriptions of the filters used in the experiments.

Table 3 compares the number of 1's in FIR filter coefficient ROMs which are implemented by using original data, column inverted data [3], row inverted data

[4], word-line collapsed data [5] and the data encoded by the proposed method, respectively. By applying the proposed encoding, the number of 1's is reduced by 50.7% on average, which is greater than 5.9% and 38.8% reductions achieved by column inversion and row inversion, respectively. The number of 1's in the word-line collapsed data is smaller than that of 1's in the data generated by applying the proposed method.

Total power consumption, which is estimated in simulations by multiplying the supply voltage and the current drawn from the power supply, is compared in Table 4, where the proposed encoding reduces the largest amount of the power consumption, which is 43.7% on average. The average reductions of the column inversion, the row inversion and the word line collapsing are 5.7%, 33.0% and 8.8%, respectively. The power consumption reduction of the word line collapsing is small because the power consumption depends on data pattern and the data pattern read in each access to a word-line collapsed ROM is the same as the data pattern read in the original ROM and hence the power consumption in each access is almost the same in both ROMs.

Fig. 8 shows the reduction in the number of 1's in

Table 2. Low-pass 256-tap FIR filters used in experiments

Filter	Bandwidth	Description
F1	$0.02f_s$	Filters designed by using Hamming window
F2	$0.21f_s$	
F3	$0.03f_s$	Filter designed by using Kaiser window with $\beta = 3.3953$
F4	$0.05f_s$	Least-squares linear-phase filter
F5	$0.05f_s$	Equiripple optimal linear-phase filter designed by using Park-McClellan(PM) algorithm
F6	$0.05f_s$	Compensation FIR filter in a digital down converter of a GSM system by using PM algorithm in [6]

Table 3. Number of 1's in coefficient ROMs of FIR filters

Filter	Org.*	[3]	[4]	[5]	Prop.	Reduction (%)			
						[3]	[4]	[5]	Prop.
F1	2394	2250	1536	1183	1139	6.0	35.8	50.6	52.4
F2	2490	2428	1652	1241	1477	2.5	33.7	50.2	40.7
F3	2603	2459	1018	1851	704	5.5	60.9	28.9	73.0
F4	2476	2304	1526	1244	1223	6.9	38.4	49.8	50.6
F5	2461	2359	1798	1240	1507	4.1	26.9	49.6	38.8
F6	2603	2337	1635	1364	1336	10.2	37.2	47.6	48.7

*Org. and Prop. mean original data and data encoded by using the proposed method, respectively.

Table 4. Total power consumption in coefficient ROMs of FIR filters under condition that $f=100$ MHz, $V_{DD}=1.8$ V and $V_{Swing}=1.8$ V

Filter	Power consumption (μW)					Reduction (%)			
	Org.	[3]	[4]	[5]	Prop.	[3]	[4]	[5]	Prop.
F1	1076.7	1020.4	765.6	985.1	605.0	5.2	28.9	8.5	43.8
F2	1155.8	1134.9	833.9	1051.7	799.1	1.8	27.9	9.0	30.9
F3	1147.3	1075.7	520.7	976.8	425.1	6.2	54.6	14.9	62.9
F4	1111.2	1050.0	746.2	1043.3	643.5	5.5	32.8	6.1	42.1
F5	1136.1	1095.8	890.9	1056.7	794.1	3.5	21.6	7.0	30.1
F6	1175.2	1075.7	808.2	1092.5	708.8	8.5	31.2	7.0	39.7
Average						5.7	33.0	8.8	43.7

Fig. 8. Reductions in the number of 1's in ROM data and the power consumption of memory cell array.

ROM data and the power consumption reduction of the memory cell array. By investigating Fig. 8, we can find that the amount of the power consumption reduction in memory cell array is very close to that of the reduction in the number of 1's, which agrees with Eqs. (1) and (2) that state the power consumption is proportional to the bit line and the word line capacitances that increase with the increment in the number of 1's.

By encoding ROM data, most of the power consumption reduction is achieved in memory cell area as can be seen in Table 5, where the detailed power consumptions in the coefficient ROM of a digital down converter exploited in a GSM system are shown. Data encoding methods need blocks, which consume additional power, to decode the encoded data. XOR gates are used in row inversion and the proposed encoding whereas inverters are used in column inversion. In Table 5, although we can achieve 495.9 μ W of power reduction in memory cell array by applying the proposed encoding, total power reduction decreases to 466.4 μ W since F/Fs and decoding blocks consume the additional power of 29.5 μ W. By comparing Tables 3 and 4, we can recognize that the reduction of total power consumption is smaller than that of the number of 1's due to the additional blocks when data encodings are applied. The word-line collapsed ROM consumes larger amount of power than the ROM exploiting the proposed method since the word line collapsing achieves smaller power consumption reduction in memory cell array.

As can be seen in Table 5, even though the decoding block of the proposed encoding consumes larger amount of power than those of the previous methods, the total

power consumption of the proposed encoding is smaller than those of the previous methods since the proposed encoding effectively reduces the power consumption by encoding ROM data with the consideration of the sequential access pattern in addition to sign inversion that reduces the number of 1's generated in sign extension.

V. CONCLUSIONS

In this paper, we propose a new access pattern-aware ROM data encoding that can be applied to the design of a ROM that is accessed sequentially. The proposed method encodes each bit line by grouping two consecutive bits and decodes the encoded data by using D F/Fs and XOR gates. By applying the proposed method, we can reduce the number of 1's and the power consumption by 50.7% and 43.7% on the average, respectively.

ACKNOWLEDGMENTS

This work was financially supported by the second stage of Brain Korea 21. The CAD Tools were provided of IDEC (IC Design Education Center).

REFERENCES

- [1] E. de Angel and E. E. Swartzlander, Jr., "Survey of low-power techniques for ROM, " in *Proc. Int. Symp. Low Power Electronics and Design*, 1997. pp. 7-11
- [2] Neil H. E. Weste and David Harris, "CMOS VLSI Design, A Circuits and Systems Perspective", Addison Wesley.
- [3] You-Sung Chang and Chong-Min Kyung, "Conforming Block Inversion for Low Power Memory", *IEEE Tran. Very Large Scale Integer.*, VOL. 10. NO. 1 FEBRUARY 2002. pp. 15-19
- [4] Paul L. Chou, "Low Power ROM Generation", *M.S. thesis, Massachusetts Institute of Technology, Boston*, JUNE 1996
- [5] Venkat R. V. and Prasad A. V. S. S., "Data Dependent Optimization for ROM Structures," in *Proc. ISCAS*, 2009. pp. 2479-2752
- [6] Ricardo A. Losaga, "Practical FIR Filter Design in MATLAB", The MathWorks, Inc. 2004.

Table 5. Detailed power consumption in the coefficient ROM of F6 under condition that $f=100$ MHz, $V_{DD}=1.8$ V and $V_{Swing}=1.8$ V

Data encoding	Power consumption (μ W)			Total power (μ W)
	Address decoder	Memory cell array	F/Fs with decoding block	
Org.	25	1070.8	79.4	1175.2
[3]	25(0)	970.6 (-100.2, -9.4%)	80.1 (+0.7)	1075.7 (-8.5%)
[4]	25(0)	689.4 (-381.4, -35.6%)	93.8 (+14.4)	808.2 (-31.2%)
[5]	13 (-12, -48%)	1016.5 (-54.3, -5.1%)	63.0 (-16.4)	1092.5 (-7.0%)
Prop.	25(0)	574.9 (-495.9, -46.3%)	108.9 (+29.5)	708.8 (-39.7%)

*"(- or +)" means the amount of reduction in μ W.

Seong-Ik Cho received the B.S., M.S., and Ph.D. degrees in electrical engineering from Chonbuk National University, Korea, in 1987, 1989, and 1994, respectively. From 1996 to 2004, he joined System IC R&D Center and Memory R&D Center in

Hynix Semiconductor Inc., Korea, where he worked for the High-speed Graphic DRAM, Data converter, and Analog Circuits. Since March 2004, I have been with the Department of Electronic Engineering, Chonbuk National University, where I am associate professor. My current research includes Low-voltage Low-power High-speed Analog Circuits, High-speed I/O Interface, DLL/PLL, Memory, Data Converter and Power Management IC.

Ki-Sang Jung received the B.S. and M.S. degree in electronic engineering from Chonbuk National University, Korea, in 2007 and 2009, respectively, where he is working toward the Ph.D. degree. His research

interests include High-speed I/O circuit, Low-power High performance Mixed signal circuit and Low power circuit design.

Sung-Mi Kim received the B.S. and M.S. degree in electronic engineering from Chonbuk National University, Korea, in 1998 and 2000, respectively, where she is working toward the Ph.D. degree. Hers research

interests include High-speed I/O circuit, Low-power High performance Mixed signal circuit and Low power circuit design.

Namhee Yoo received the B.S., M.S. degrees and Ph. D. from Chonbuk National University in 1991, 1993 and 1997, respectively. During 1997-2000, he worked in Technology Institute of Syngenta Co., Ltd. of

Korea as a crop breeding and biotechnology researcher. He now with professor for Industrial Cooperation Foundation, Chonbuk National University and center director for Social Enterprise Support Center, Jeonbuk Province.

Jong-Yeol Lee received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 1993, 1996, and 2002, respectively. After graduation from

KAIST, he was with Hynix Semiconductor, Inc. from 2002 to 2003, where he researched on compiler optimization. From September 2003, he was at KAIST as a Brain Korea 21 (BK21) Visiting Professor. In March 2004, he joined Division of Electronics & Information Engineering at Chonbuk National University, where he is now an Associate Professor. His research interests include SoC design methodology, algorithms for compiler optimization, and VLSI design.