Crosstalk-Insensitive Via-Programming ROMs Using Content-Aware Design Framework

Meng-Fan Chang, Lih-Yih Chiou, and Kuei-Ann Wen, *Senior Member, IEEE*

***Abstract*—Various code patterns of a via-programming read only memory (ROM) cause significant fluctuations in coupling noise between bitlines (BLs). This crosstalk between BLs leads to read failure in high-speed via-programmable ROMs and limits the coverage of applicable code patterns. This work presents a content-aware design framework (CADF) for via-programming ROMs to overcome the crosstalk induced read failure. The CADF ROMs employ a content-aware structure and correspondent code-structure programming algorithm to reduce the amount of coupling noise source while maintaining nonminimal BL load for crosstalk reduction. A 256-Kb conventional ROM and a 256-Kb CADF ROM were fabricated using a 0.25-μm logic CMOS process. The measured results ascertain that the read induced read failure is suppressed significantly by CADF. The CADF ROM also reduced 86.2% and 94.5% in power consumption and standby current compared to the conventional ROM, respectively.**

***Index Terms*—Code patterns, crosstalk, read only memory (ROM).**

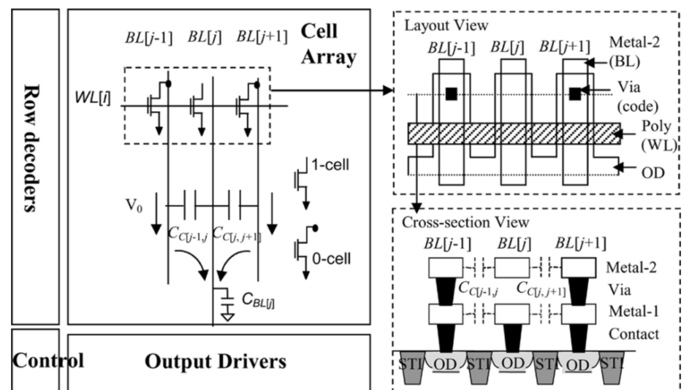

Fig. 1. Structure of conventional ROMs. The wordline (WL), ground line (VSS), code layer, and BL are implemented by poly, diffusion (OD), via, and metal-2, respectively.

I. Introduction

READ-ONLY memories (ROMs) are commonly embedded into system-on-chip (SoC) designs for storing programs and predefined data. Compared to those coded by diffusion and poly layers, via-programming ROMs (via-ROMs) shorten the turnaround time in manufacturing after code modification and is popular in today's designs. Unfortunately, various code patterns in via-ROMs or contact-programming ROMs produce large fluctuations in BL loading and coupling capacitances. To achieve a high speed, the wordline pulse width in a ROM is short, and the sensing margin becomes small and vulnerable to noise. This code-pattern-dependent crosstalk-induced read failure (CIRF) limits the speed and code-pattern coverage for via-ROMs.

To reduce power consumption and to improve speed performance of ROMs, previous works employed schemes, such as block/row inversion [1], precharge-discharge dynamic CMOS logic [2], charge recycling [3], and restricted bitline (BL) swing [4]–[6]. These studies have not addressed their solutions to the CIRF across different code patterns. In this brief, we investigate this problem and present content-aware design framework (CADF) to overcome CIRF for high-speed via-ROMs. The CADF ROMs reduce the coupling noise source and maintain nonminimum BL load to reduce the amount of corsstalk. Furthermore, the CADF ROMs consume lower power and higher speed than conventional via-ROMs. To our knowledge, this is the first to deal with the code-dependent read failure induced by crosstalk between BLs for via-ROMs.

The remainder of this brief is organized as follows. Section II analyzes the behavior of the code-dependent CIRF. Section III presents the proposed CADF. Section IV presents the experimental results. Section V draws conclusions.

Manuscript received June 22, 2005. This paper was recommended by Associate Editor R. Puri.

M.-F. Chang is with the Intellectual Property Library Company, Hsin Chu 300, Taiwan, R.O.C. (e-mail: mfchang.ee89g@nctu.edu.tw; mfchang@iplib.com).

L.-Y. Chiou is with the Electrical Engineering Department, National Cheng-Kung University, Tainan 701, Taiwan, R.O.C. (e-mail: lihyih@ee.ncku.edu.tw).

K.-A Wen is with the Institute of Electronics, National Chiao Tung University, Hsin Chu 300, Taiwan, R.O.C. (e-mail: stellawen@nctu.edu.tw).

Digital Object Identifier 10.1109/TCSII.2006.873640

II. Crosstalk Induced Read Failure

Crosstalk between bit lines erodes the sensing margin and leads to read failure in high-speed via-ROMs. The code-dependent CIRF limits the coverage of applicable code patterns in via-ROMs.

A. Sensing Margin

In conventional via-ROMs, as shown in Fig. 1, the value of the stored datum is determined by the connection to a via layer. A bit cell with a via layer that connects its nMOS transistor to its BL stored the datum 0 (0-cell), a cell without such a layer stored the datum 1 (1-cell). When a row is activated by the relevant wordline (WL), the 0-cell sinks current from the bit line, while the 1-cell does not sink any current. For a 0-cell, the parasitic capacitance on the drain side of the transistor is connected to its BL. For a 1-cell, the parasitic capacitance on the drain side of the transistor is not seen by its BL.

In ROMs, all BLs are precharged to a pre-determined voltage prior to the data-sensing phase of a cycle. In the data-sensing phase, a BL is discharged to develop a voltage drop (V_0) for

1057-7130/$20.00 © 2006 IEEE

reading a 0-cell or remains at the precharged voltage, $V_{\rm PRE}$, for reading a 1-cell. To differentiate a 0-cell from a 1-cell, the sense amplifier needs a reference voltage $V_{\rm REF}$ whose value must be set between $V_{\rm PRE} - V_0$ and $V_{\rm PRE}$. Clearly, the sensing margin $\rm SM_0$ for reading a 0-cell is $V_{\rm REF} - V_{\rm PRE} + V_0$ and the sensing margin $\rm SM_1$ for reading a 1-cell is $V_{\rm PRE} - V_{\rm REF}$. The fixed-value (FV) scheme [4] and the half-rate BL-tracking (HRBLT) schemes [5], [6] are the two most popular ways for providing $V_{\rm REF}$ in ROMs. The FV scheme, which provides a fixed value in $V_{\rm REF}$ during the entire data-sensing phase, is employed in this work since it is suitable for experiments. If a coupling noise drop (V_X) happens on a BL, it may reduce the value of $\rm SM_1$.

Various code patterns cause different amount of parasitic capacitance $(C_{\rm BL})$ on BLs in via-ROMs [7]. The BL whose bit cells are all 0-cells incurs the largest load effect in via-ROMs. Given wordline pulsewidth $T_{\rm WL}$ and cell current $I_{\rm CELL}$, the voltage drop $V_0[j]$ of a BL $\rm BL[\mathit{j}]$ can be derived as

$$V_0[j] = \frac{I_{\rm CELL} \cdot T_{\rm WL}}{C_{{\rm BL}[j]} + C_{C[j-1,j]} + C_{C[j,j+1]}}. \quad (1)$$

The $C_{C[j-1,j]}$ and $C_{C[j,j+1]}$, as shown in Fig. 1, are the coupling capacitance between $\rm BL[\mathit{j}]$ and its two adjacent BLs, $\rm BL[\mathit{j}-1]$ and $\rm BL[\mathit{j}+1]$. For a given $T_{\rm WL}$, the BLs with large $C_{\rm BL}$ have small V_0.

To detect all bit cells correctly, the minimum of V_0, denoted as $V_{0\,\rm min}$, and the maximum of V_X, denoted as $V_{X\,\rm max}$, deserve our attention. The voltage drops $V_{0\,\rm min}$ and $V_{X\,\rm max}$ across code patterns must satisfy the inequalities $V_{0\,\rm min} > V_{\rm PRE} - V_{\rm REF}$ and $V_{X\,\rm max} < V_{\rm PRE} - V_{\rm REF}$ simultaneously by noting that both $\rm SM_0$ and $\rm SM_1$ must be larger than zero. From (1), $T_{\rm WL}$ must be long enough to generate enough $V_{0\,\rm min}$ to satisfy $\rm SM_0$.

B. Crosstalk Between BLs

In nominal ROMs, all bit cells on the same row are controlled by the same wordline. Thus, there are V_0 on both the selected and unselected BLs during the data-sensing phase. The unselected BLs $\rm BL[\mathit{j}-1]$ and $\rm BL[\mathit{j}+1]$, which are the neighboring BLs of $\rm BL[\mathit{j}]$ as shown in Fig. 1, have ΔV_0 developed by 0-cells when $\rm WL[\mathit{i}]$ is on. The V_0 on $\rm BL[\mathit{j}-1]$ and $\rm BL[\mathit{j}+1]$ generate V_X through $C_{C[j-1,j]}$ and $C_{C[j,j+1]}$ onto the selected BLs, $\rm BL[\mathit{j}]$. Accordingly, the neighboring BLs are the aggressors and the selected BLs are the victims in crosstalk effects.

Fig. 2(a) shows the simulated waveform of BL without crosstalk effect with $V_{\rm PRE} = 2.5$ V and $V_{\rm REF} = 2.25$ V. The $\rm BL_\mathit{A}1$ and $\rm BL_\mathit{A}2$ read 0-cells but with light (with 511 1-cells) and heavy (with 512 0-cells) BL loads, respectively. The BL without any crosstalk $\rm BL_\mathit{D}$ retains the $V_{\rm PRE}$ during the data-sensing phase and has the correct sensing result. Fig. 2(b) shows the example of CIRF. The V_X causes the voltage on the victim BL, $\rm BL_\mathit{V}1$ (reads a 1-cell), being lower than the $V_{\rm REF}$. Then, 0-cell is mistakenly detected by a sense amplifier rather than the expected 1-cell for $\rm BL_\mathit{V}1$.

Coupling capacitance between BLs, BL load, and the amplitude of aggressor voltage are the key parameters for the crosstalk effect between BLs. Unfortunately, the coupling capacitance, which is determined by the spacing between BLs, becomes larger as the minimum-space between metal lines is smaller in advance technology nodes.

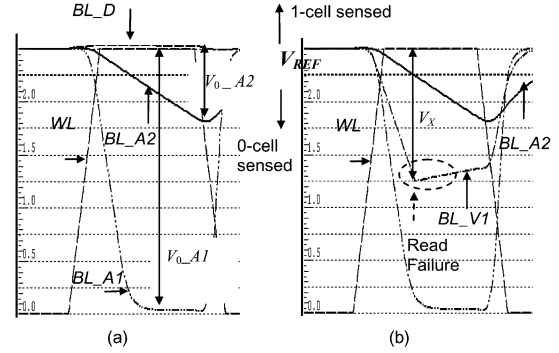

Fig. 2. Simulation waveforms of BLs (512 cells) in conventional ROMs: (a) without and (b) with crosstalk. $V_{\rm PRE} = 2.5$ V and $V_{\rm REF} = 2.25$ V.

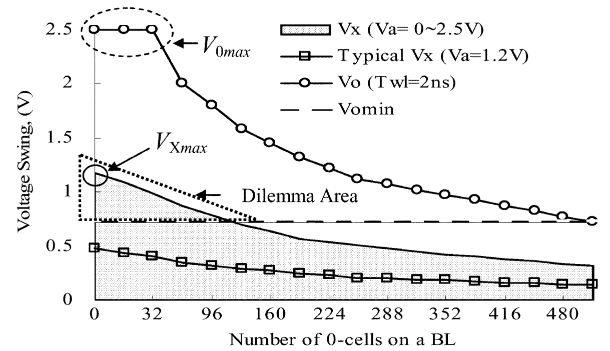

Fig. 3. Simulated V_0 and V_X versus the number of 0-cells on a BL. ($V_A = 0 \sim 2.5$ V, $V_{\rm PRE} = 2.5$ V).

As discussed earlier, various code patterns on a BL result in various $C_{\rm BL}$ and V_0 on BLs for a given $T_{\rm WL}$. The maximum of V_0, denoted as $V_{0\,\rm max}$, could be a full swing of the $V_{\rm PRE}$ (e.g., V_0_A1) for a BL with minimum load. The $V_{0\,\rm min}$ is only a few hundred millivolts for a BL with maximum load (e.g., V_0_A2). These various V_0 on unselected BLs, acting as the aggressor voltages (V_A) in crosstalk, generate various level of V_X on their neighboring BLs. Moreover, long $T_{\rm WL}$ results in large V_0 for a BL. Hence, both $T_{\rm WL}$ and code patterns affect the V_A and V_X.

Furthermore, the crosstalk between BLs is also dependent on the $C_{\rm BL}$ of the victim BLs. The coupled voltage on a victim BL is derived in (2). The V_X on victim BLs is large with small intrinsic BL load (less number of 0-cells) across various code patterns

$$V_X = V_{A[j-1]} \times \frac{C_{C[j-1,j]}}{C_{{\rm BL}[j]} + C_{C[j-1,j]}} + V_{A[j+1]} \times \frac{C_{C[j,j+1]}}{C_{{\rm BL}[j]} + C_{C[j,j+1]}}. \quad (2)$$

Fig. 3 shows the simulated V_X (with $V_A = 0 \sim 2.5$ V) and V_0 on a BL (with 512 cells) versus various BL loads. Since both $\rm SM_0$ and $\rm SM_1$ must be larger than zero for correct sensing, the $V_{0\,\rm min}$ and V_X determine the applicable values of $V_{\rm REF}$. The $T_{\rm WL}$ employed in Fig. 3 is 2 ns. The $V_{0\,\rm min}$ is the voltage swing of the BL has maximum $C_{\rm BL}$ (512 0-cells). The V_X on victim BLs can be smaller than $V_{0\,\rm min}$ if their V_A is small (e.g.,

$V_A = 1.2$ V or smaller). For those victim BLs whose V_X are larger than $V_{0\,\min}$ (see the dilemma area in Fig. 3), there is no $V_{\rm REF}$ can be found to correctly differentiate 0-cells from 1-cells for those BLs. In this example, the victim BLs with less than 96 0-cells suffer read 1-cell failure if $V_{\rm REF}$ is equal to $V_{0\,\min}$.

In summary, a BL has small $C_{\rm BL}$ and large V_A suffer large V_X and CIRF. Various code patterns generate large fluctuation in BL loads. The $V_{0\,\max}$ is large when $V_{0\,\min}$ is required to satisfy a given SM_0 because of the large fluctuation in BL loads in via-ROMs. Thus, CIRF is dependent on the data patterns on BLs, and limits the coverage of applicable code patterns on a via-ROM.

III. Content-Aware Design Framework

To reduce the aggressor voltage and avoid minimum $C_{\rm BL}$ for crosstalk suppression in via-ROMs, the fluctuation in BL load must be further reduced. The CADF optimized the structure and data patterns of via-ROMs to achieve high uniformity in parasitic capacitance and voltage swing on BLs across various code patterns. The CADF consists of the content-aware structure (CAS) and code-structure programming algorithm (CSPA). The CAS provides the infrastructure for CSPA.

A. CAS

The CAS comprises hybrid-segmented BLs (HSB), flag tables and dual-path output drivers.

In HSB, each column (BL) has a base segment (base-BL) and numerous segments (sub-BLs). When a column is accessed, its base-BL and only one of its sub- BLs are accessed. There are n bit cells in a base-BL. A column, excluding its base-BL, is physically divided into numerous segments by local BL switches (LBSs) according to the to-be-stored ROM code and the CSPA. The maximum number of divided segments on each column, k, is the predefined parameters based on the memory configuration, area limitation and crosstalk consideration. The value of k is equal to 2, 4, or 8 in CADF. Each sub-BL originally has 2 m bit cells with the exception of the bottom sub-BL. The bottom sub-BL has $(2\,m{-}n)$ bit cells and is next to the base-BL. Fig. 4 illustrates the CAS for a ROM macro with x rows and y columns. The m value is defined as

$$m = \frac{y}{2\times k}\times\left(\mathrm{INT}\left[\frac{y}{k\times \mathrm{rs}}\right]+1\right),\quad \text{if MOD}\left[\frac{y}{k\times \mathrm{rs}}\right]\neq 0$$

or

$$m = \frac{y}{2\times k},\quad \text{if MOD}\left[\frac{y}{k\times rs}\right]=0. \quad (3)$$

Herein, the row step (rs) specifies the minimum number of steps increased on the number of rows due to the structure of leaf-cells in ROMs. The value of rs is equal to 4 in this brief. The number of sub-BL on a BL is derived by CSPA and varies from column to column if the code patterns on BLs are different. Namely, the cell array consists of BLs that are hybrid segmented.

An LBS, controlled by segment selection signals (SS), comprises two nMOS transistors and one shared contact that connects its top and bottom sub-BLs to a BL. The dummy LBS

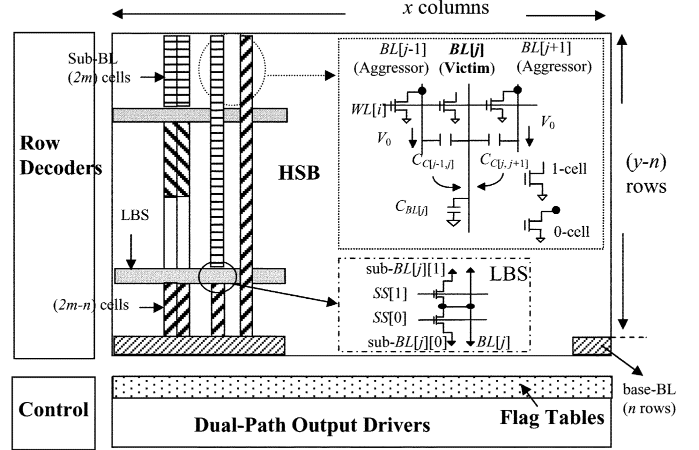

Fig. 4. CAS of CADF ROM (with $k = 4$). A simplified array is included with selected/unselected BL.

(DLBS), which has the same width and height as the LBS, has no device but a vertical metal layer to connect its top and bottom sub-BLs. The DLBS is used to replace a LBS in CSPA.

For each BL, a $(k+1)$-bit table stores the flags that indicate the data-inversion status of each sub-BL and base-BL. Dual-path output drivers utilized the output from flag table to select the normal or inverted output path. If a flag signifies true for an accessed sub-BL, then the output driver select the inversion path from the output of the sense amplifier, and the codes read from the CADF ROM are still correct.

B. CSPA

A four-step algorithm, CSPA, was developed for code programming and sub-BL structuring in CADF. The CSPA reduces fluctuations but maintain the minimum amount in BL load. The procedure of the proposed CSPA is explained as follows.

- Step 1) The input ROM code is initially programmed into HSB based on user defined parameters k and n.
- Step 2) If the number of bit cells with code 0 on a sub-BL exceeds m, then the CSPA assigns 1-cells for the bit cells with code 0 on this sub-BL. The correspondent flag bit of this sub-BL is set to 1. Otherwise, the CSPA assigns 0-cells for bit cells with code 0. The correspondent flag bit of this sub-BL is set to 0.
- Step 3) If the number of 0-cells on consecutive sub-BL exceeds m, these sub-BLs are merged. The LBSs assigned at step 1 for these merged sub-BLs are replaced by DLBSs.
- Step 4) For each column, if the number of 0-cells on a sub-BLs AND its base-BL are both smaller than $n/2$, then the CSPA assigns 1-cells for the bit cells with code 0 on the base-BL of this column. The flag bit of this base-BL is set to 1.

The merge activity in Step 3) makes the number of 0-cells on each sub-BL close to m as possible to reduce the fluctuation in accessed BL load across cycles. Step 4) ensures the minimum load of a BL exceeds $n/2$ to reduce the crosstalk between BLs. After Step 4), the maximum and minimum number of 0-cells on a BL in CADF ROM is $(m+n)$ and $n/2$, respectively. In conventional ROMs and the inverted ROM [1], the maximum

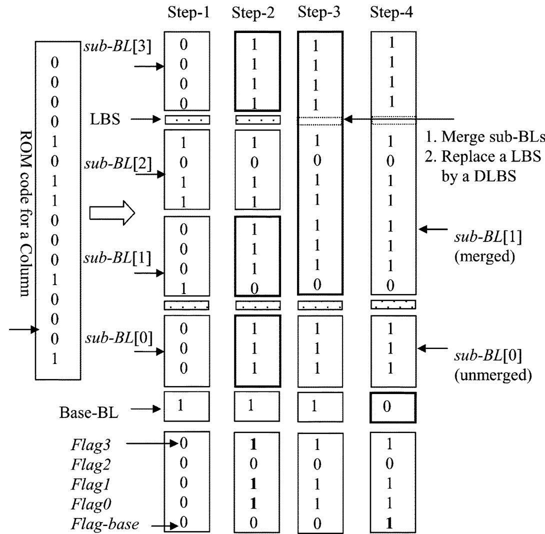

Fig. 5. Example of CSPA, where $k = 4, m = 2$, and $n = 1$.

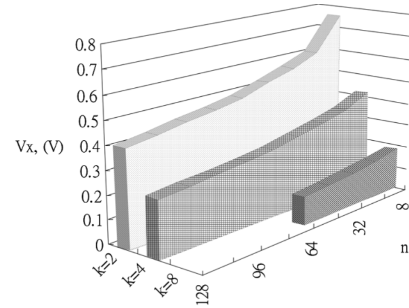

Fig. 6. Simulated maximum coupled voltage drops (V_X) on a victim BL (512 cells) with various values for k and n.

Fig. 7. Die photos of a 256-Kb conventional (Std) ROM with X pattern and a 256-Kb CADF ROM ($k = 8, n = 16$, and $m = 32$) with X pattern.

and minimum number of 0-cells on a BL in CADF ROM is y and zero, respectively.

In the example of CSPA depicted in Fig. 5, a BL was initially divided into a base-BL $(n = 1)$ and four $(k = 4)$ sub-BLs by two LBSs. In Step 2), the data in sub-BL[0], sub-BL[1], and sub-BL[3] were inverted because their number of code 0 were both larger than m $(m = 2)$. The Flag0, Flag1, and Flag3 are set to 1. In Step 3), since the total number of 0-cells on sub-BL[3]-sub-BL[1] did not exceed m, these three sub-BLs were merged and only one LBS was utilized for this column instead of two. The removed LBS is replaced by a DLBS. Since both the number of 0-cells on the sub-BL[0] and base-BL did not exceed $n/2$ [in Step 4)], the data on the base-BL was inverted and *Flag-base* is set to 1. Hence, the load on a CADF BL was one or three 0-cells across cycles and significantly smaller than the original code (eleven 0-cells) in conventional ROMs.

Therefore, the data patterns and the structure of HSB are "smartly" programmed based on the ROM code. Despite the small load on a BL, the fluctuations in BL load for various code patterns, $(m + n/2)$ 0-cells, in CADF ROM is much smaller than that (y 0-cells) in conventional ROM.

C. Crosstalk Suppression by CADF

Since CADF ROMs achieve small BL loads, short T_{WL} is needed to generate required $V_{0\,\mathrm{min}}$ for a given SM_0. Generally, $V_{0\,\mathrm{min}}$ is equal to a few hundred millivolts in high-speed ROMs. On the other hands, the difference between $V_{0\,\mathrm{min}}$ and $V_{0\,\mathrm{max}}$ in CADF ROMs is small due to the high uniformity feature of BL loads across various code patterns. The smaller the fluctuation of the load on a BL, the smaller the $V_{0\,\mathrm{max}}$ can be achieved in the CADF ROM. Thus, $V_{0\,\mathrm{max}}$ can be limited to a few hundred millivolts in CADF ROMs, rather than the full swing of V_{PRE} as in the conventional ROMs. A large value of k results in a small value of m, as derived in (1), and small $V_{0\,\mathrm{max}}$ across code patterns. Therefore, the maximum value of V_A, which equals to $V_{0\,\mathrm{max}}$, is significantly reduced in CADF ROMs when k is large.

As discussed in Section II-B, the minimum number of 0-cells on a BL in CADF ROM is $n/2$ rather than zero 0-cells as in conventional ROMs. This feature, nonzero 0-cells on a BL, increase the minimum BL load and enhance the immunity of coupling noise in CADF ROMs. The optimized value of n is dependent on the technology node, predefined V_{REF} and the value of k.

As both the values of n and k increase, the crosstalk reduction becomes more significant. Fig. 6 shows the simulated maximum V_X on a victim BL (512 bit cells) with various values for k and n. For a CADF ROM with small k value, increasing the n value effectively reduced the coupled voltage on a victim BL. However, the trend of V_X reduction is saturated when n is large. Moreover, employing large k values enable a CADF ROM can have a small n value for a given tolerated V_X.

With appropriate value of n and k, which result in small V_A and nonminimum C_{BL}, the maximum V_X on a victim BL can be always smaller than $V_{0\,\mathrm{min}}$ in CADF ROMs. A V_{REF} can be found without the dilemma area depicted in Fig. 3. Thus, the pattern-dependent CIRF can be avoided and the code-pattern coverage is increased by CADF.

IV. Experimental Results

A 256-Kb CADF ROM ($k = 8, n = 16, m = 32$, and $x = y = 512$) and 256-Kb conventional ROM were fabricated in 1P5M 0.25-μm CMOS technology, as shown in Fig. 7. The X pattern was applied on both the experimental ROMs for investigating the CIRF.

Since the T_{WL} in CADF ROM were much smaller than that in conventional ROM, the $V_{0\,\mathrm{max}}$ is about 500 mV rather than 2.5 V as in conventional ROM. Both the access time and the cycle time of the CADF ROM were also shorter than those of the conventional ROM thanks to the small precharge time and T_{WL}. The access time of a 256-Kb CADF ROM was only 55.6% of the conventional ROM.

TABLE I
FUNCTIONALITY TEST OF FABRICATED ROMs WITH X PATTERN

ROM Type			Conventional		CADF	
Capacity (bits)			256K		256K	
Area (mm^2)			0.53		0.57*	
Standby Current (uA)			0.36		0.02	
X-Pattern	Number of code 0		V_{REF} (V) at V_{DD} = 2.5V			
	Aggressor	Victim	2.15	2.25	2.15	2.25
X1	512	0	Fail	Fail	Pass	Pass
X2	512	256	Pass	Pass	Pass	Pass
X3	512	511	Pass	Pass	Pass	Pass
X4	256	0	Fail	Fail	Pass	Pass
X5	256	256	Pass	Pass	Pass	Pass
X6	256	511	Pass	Pass	Pass	Pass
X7	1	0	Fail	Fail	Pass	Pass
X8	1	256	Fail	Fail	Pass	Pass
X9	1	511	Pass	Fail	Pass	Pass

*: The optimized area of CADF ROM is 0.54 mm^2.

The suppression of crosstalk effects by CADF was demonstrated through functional verification with the X pattern. Nine levels of crosstalk effect (including the minimum and maximum cases) are included in the X pattern, X1–X9. The aggressor BLs read 0-cells and the victim BLs read 1-cell during the testing. The measured results of the nine test patterns for the fabricated conventional (Std) and the CADF ROMs are shown in Table I. The conventional ROM failed to sense the X1, X4, X7, X8, and X9 while the fabricated CADF ROM passed the X pattern.

The measured standby current for a fabricated 256-Kb CADF ROM was only 0.02 μA, compared to 0.36 μA for the 256-Kb conventional ROM with X pattern, a reduction of 94.5%. Therefore, CADF is also good for nanometer technology to resolve the issue on subthreshold leakage due to its less number of leakage paths (0-cells in a activated column) in cell arrays than the one in conventional ROMs.

The comparison of the performance of a 256-Kb CADF ROM to other low-power approach is shown in Table II. Since different works had various speed and power performance for various sizes of memory capacity, we adopt the power-delay product (PDP) per bit (PDP-bit) for fair comparison. The maximum PDP-bit of CADF ROM, 0.06 pico-Joule per bit, was much smaller than those obtained in other reports.

V. CONCLUSION

The code-pattern-dependent crosstalk induced read failure have been investigated. A CADF is proposed to overcome the CIRF and improve the power and speed performance for via-ROMs. Fabricated 256-Kb CADF ROM demonstrated its effectiveness and achieved 100% code coverage under high-speed operation. The fabricated CADF ROM also reduces 86.2% in power consumption and 94.5% in standby current with 2.7% area penalty compared to the fabricated conventional ROM. The CADF ROM had improved 55.6% in access time. Furthermore, the CADF had a smaller power-delay product to memory capacity ratio than previous techniques.

TABLE II
COMPARISONS OF EXPERIMENTAL ROMs AND PREVIOUS REPORTS

	NHS-PD [2]	CRCS [3]	HSCSS [8]	CADF ROM	Std ROM
Capacity (bits)	16K	128K	32K	256K	256K
Process (um)	0.35	0.35	0.25	0.25	0.25
VDD (V)	3.3	3.3	1.0/1.7	2.5	2.5
Access Time (ns)	3.8	8.4	5.7[1]/3[1]	2.0	3.6
Power (mW @100Mhz)	6.09[1,U]	8.63[U]	2.2[1]/8[1,U]	8.03[1,X]	63.5[1,X]
PDP-bit[2]	2.82	0.55	0.38/0.73[U]	0.06[X]	0.87[X]

[1] Silicon result
[2] Power-Delay Product per bit (pico-Joule/bit)
[X] X-pattern for crosstalk analysis
[U] Applied code-pattern was not specified.

REFERENCES

[1] E. de Angel and E. E. Swartzlander, Jr., "Survey of low-power techniques for ROMs," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, Aug. 1997, pp. 20–20.

[2] C.-R. Chang, J.-S. Wang, and C.-H. Yang, "Low-power and high-speed ROM modules for ASIC applications," *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1523–1523, Oct. 2001.

[3] B. D. Yang and L. S. Kim, "A low-power ROM using charge recycling and charge sharing technique," *IEEE J. Solid-State Circuits*, vol. 38, no. 4, pp. 653–653, Apr. 2003.

[4] A. Tuminaro, "A 400 Mhz, 144-Kb CMOS ROM macro for an IBM S/390-class microprocessor," in *Proc. IEEE Int. Conf. Comp. Des.*, Oct. 1997, pp. 255–255.

[5] M.-F. Chang, K.-A. Wen, and D.-M. Kwai, "Supply and substrate noise tolerance using dynamic tracking clusters in configurable memory designs," in *Proc. IEEE Int. Symp. Qual. Electron. Des.*, Mar. 2004, pp. 302–302.

[6] T. Tsang, "A compilable read-only-memory library for ASIC deep sub-micron applications," in *Proc. IEEE 11th Int. Conf. VLSI Design*, Chennai, India, Jan. 1998, pp. 494–494.

[7] M.-F. Chang, L.-Y. Chiou, and K.-A. Wen, "A low supply noise content-sensitive ROM architecture for SOC," in *Proc. IEEE Asia-Pacific Conf. Circuits Syst.*, Dec. 2004, pp. 1024–1024.

[8] R. Sasagawa, I. Fukushi, M. Hamaminato, and S. Kawashima, "High-speed cascode sensing scheme for 1.0 V contact-programming mask ROM," in *Proc. IEEE Symp. VLSI Circuits*, 1999, pp. 96–96.