electronics

MDPI

Article

A 0.94 µW 611 KHz In-Situ Logic Operation in Embedded DRAM Memory Arrays in 90 nm CMOS

Myeong-Eun Hwang [1] and Sungoh Kwon [2,*]

[1] SoC Development Group, SK Hynix Memory Solutions, San Jose, CA 95134, USA
[2] School of Electrical Engineering, University of Ulsan, Ulsan 44610, Korea
* Correspondence: sungoh@ulsan.ac.kr; Tel.:+82-52-259-1286

Received: 8 June 2019; Accepted: 30 July 2019; Published: 5 August 2019

Abstract: Conventional computers based on the Von Neumann architecture conduct computation with repeated data movements between their separate processing and memory units, where each movement takes time and energy. Unlike this approach, we experimentally study memory that can perform computation as well as store data within a generic memory array in a non-Von Neumann architecture way. Memory array can innately perform NOR operation that is functionally complete and thus realize any Boolean functions like inversion (NOT), disjunction (OR) and conjunction (AND) operations. With theoretical exploration of memory array performing Boolean computation along with storing data, we demonstrate another potential of memory array with a test chip fabricated in a 90 nm logic process. Measurement results confirm valid in-situ memory logic operations in a 32-kbit memory system that successfully operates down to 135 mV consuming 130 nW at 750 Hz, reducing power and data traffic between the units by five orders of magnitude at the sacrifice of performance.

Keywords: memory logic operation; embedded memory; eDRAM; SRAM; ultra-low power; subthreshold operation

1. Introduction

The rapidly growing volume of data processing and storage systems has constantly drawn attention to alternative approaches for higher memory bandwidth and lower-power data transaction between computational and storage units. Modern computers based on the Von Neumann architecture [1] have evolved with two main units: processing unit that computes logic and arithmetic functions, and a memory unit that stores data. Information moves back and forth between the processing and memory units while carrying out computation as well as storing intermediate data. These movements cause significant overhead in memory bandwidth and result in an inherent bottleneck in performance. The issue becomes even more serious as computation increasingly becomes data centric and heavier traffic associated.

Needless to say, memory is an essential and indispensable unit in modern controllers and processors. Need for larger memory and corresponding logic circuitry keeps growing up with technology scaling as well. There have been abundant and excellent researches especially on embedded memory systems exclusively dedicated to data storage [2–17]. The IBM research team demonstrated a paradigm-shifting approach of computational memory by using 1M phase change memory devices exploiting crystallization dynamics organized to perform massive data processing on temporally correlated data between event-based data streams [18]. To address the memory bandwidth or bottleneck issue, Shuangchen and Yuan Xie, et al. proposed a dynamic random access memory (DRAM)-based in-situ accelerator with extensive analysis and case study including neural networks [19]. Their experiment results show that in-situ processing units in memory arrays can

potentially provide significant gains in performance and energy efficiency over the conventional logic gate-based approach.

In this work, we for the first time propose a new approach toward the study of memory array for both performing computation and storing data, that we call in-situ logic operation in memory. Our approach is based on the innate ability of a memory cell array that can structurally perform wired-NOR operation. After exploring the organization of memory arrays for logic computation, we exemplify how the arrays can perform calculation and concurrently store in-between computational values and primary inputs. We also demonstrate in-memory logic operations with a test chip. Our scheme can be extended to any memory array with different types of memory cells like an one-transistor (1T) DRAM (dynamic random access memory) or 6T SRAM (static random access memory) cells as long as the bitlines are there.

Some highlights of this paper include: (1) we propose a different use of memory arrays, called as in-situ memory logic, based on an inherent feature of the arrays that can perform wired-NOR operation; and (2) with a test chip fabricated in 90 nm, we demonstrate the feasibility of in-situ memory logic for ultra-low voltage and power applications [20,21].

The rest of this paper is organized as follows. Section 2 reviews the general memory organization and several key components. We then examine the theoretical background of our proposed scheme in Section 3. After confirming the capability of memory arrays to provide a necessary and sufficient set of key logic operators in Section 4, Section 5 presents two examples for complex logical computation further. Measurement results for ultra-low voltage applications are discussed in Section 6, and conclusions drawn in Section 7.

2. General Memory Operations

In this work, we adopt conventional memory systems of that the details are available in Reference [22]. For the reader's convenience, we first discuss basic yet essential memory cells, components, organization and operations in modern DRAM devices.

2.1. Embedded Memory Gain Cells

Figure 1 shows the circuit diagrams of gain or bit cells used in modern embedded memory devices as storage units. The conventional 6T SRAM gain cell in Figure 1a has the write access transistor (XW), read access transistor (XR) and the cross-coupled inverters (the left PL-NL and right PR-NR inverter pair). Due to the positive feedback effect, the cross-coupled inverters immediately sense and amplify the value at *QL* and then drive another storage node *QR* with the opposite value at *QL*. Two data stored at *QL* and *QR* are always contrary to each other and remain static. Like 3T one, the 6T cell completely separates read and write accesses where the write wordline (WWL) and write bitline (WBL) are used for write access and the read wordline (RWL) and read bitline (RBL) are for read access [14]. The advantage associated with this separation will be discussed when exploring the 3T cell later. This 6T SRAM has been the first selection for the embedded cache memory in modern controllers and processors thanks to its static data retention (no refresh cycle required), logic compatibility and fast differential characteristics [8]. Large unit cell size and incompatible demands for read and write stabilities especially at low operating voltages however challenge the use of 6T SRAM as an embedded memory in scaled CMOS technologies.

Embedded DRAMs or eDRAMs have recently been attracting interest from the research community for their favorable aspects such as compact size, low leakage and non-ratioed circuit operation. A number of successful eDRAM designs are based on the one-transistor and one-capacitor (1T1C) DRAM cell as well as logic-harmonious bit cells [9–16].

In the traditional 1T1C cell illustrated in Figure 1b the data is stored as electric charge in its tiny storage capacitor. When the N-type access transistor (NX) is turned on by asserting the worldline (WL), the data value represented in the capacitor is pushed onto the bitline (BL) first and then used to charge back the capacitor later [9]. The capacitor holds the stored charge *Q* for a moment even after the

voltage on WL is de-asserted and NX is turned off. Because of leakage current through NX, the electric charge stored in the capacitor however gradually leaks away. Hence, before the charge stored in a DRAM cell diminishes indistinguishable, cell data should be periodically refreshed, that is, read out and written back from and to the cell.

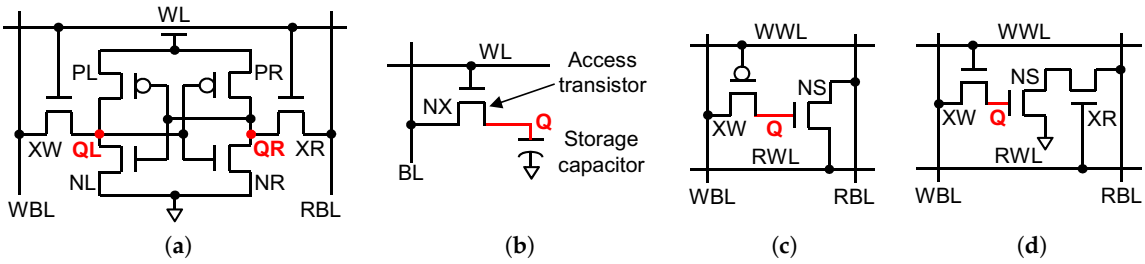


Figure 1. Logic-compatible memory gain cells. (**a**) 6T SRAM [8], (**b**) 1T1C eDRAM [9], (**c**) 2T eDRAM [16], and (**d**) 3T eDRAM [23].

Figure 1c shows an asymmetric 2T bit cell utilizing the gate and junction leakages of a PMOS write device (XW) to keep a logic high '1' voltage level enabling fast read access through an NMOS read device (NS) [16]. This gain cell designed for a long retention time without sacrificing read speed utilizes NS driven by RWL whose pre-charge level is a supply voltage (V_{DD}) for high current drive and XW to maintain the speed critical data '1' voltage close to V_{DD}. The storage device is usually turned off keeping its gate leakage marginal.

The critical effects to consider in device choice involve the charge injection and storage node coupling. An access to the write wordline (WWL) significantly damages the original voltage level of the storage node. In general, a PMOS transistor weekly transfers a data '0' (i.e., not full ground level) whereas an NMOS does a week '1' (not full V_{DD} level). Hence, an underdriven (PMOS) or boosted (NMOS) voltage on WWL is required to replenish the insufficient amount of voltage bias and thus to transmit a full level onto the storage node.

Three-transistor (3T) cell in Figure 1d especially designed for low-power applications consists of the write access device (XW), storage device (NS) and read access device (XR). Like 6T SRAM one, the 3T eDRAM cell is non-ratioed logic and separates read from write operation. This separation improves read and write stability margins and design flexibility. That is, read and write paths can be optimized individually enabling the cell to scale beneficially in future technologies. In data retention mode, XW and XR are off and the storage node is left floating. Careful engineering is required to maximize the retention time and performance by balancing the retention attributes of digital data '0' and '1'.

In this paper, we will explore our proposed technique by using the 3T eDRAM gain cell array just considered to involve moderate complexity and effort. Among gain cells, 1T1C cells are denser than others at the cost of a capacitor process and the noise margin is degraded substantially at low voltages as the read operation is based on the charge sharing principle. 3T gain cells are made of logic devices built in a standard logic CMOS process with minimal changes. Note that our proposed scheme is applicable to 1T, 2T and 6T cell-based memory arrays as well.

Figure 2a,b show the layouts of the 3T eDRAM and 6T SRAM memory gain cells in 90 nm process. Following a common Manhattan layout style, the 3T cell accounts for 54% of the area of its 6T counterpart. The main sources of degradation in the 3T cell are junction leakage and subthreshold current in the access device, charge sharing among adjoining signal lines and the accumulation of α-particle-induced charge on the cell capacitor. To lower the interference of voltage swings at the sensitive storage nodes, we use a ground network in the memory cell array. The gate terminal of the storage device (NS) is shielded with grounded metal 1 and metal 2 ground line electrically protects RBL from WBL.

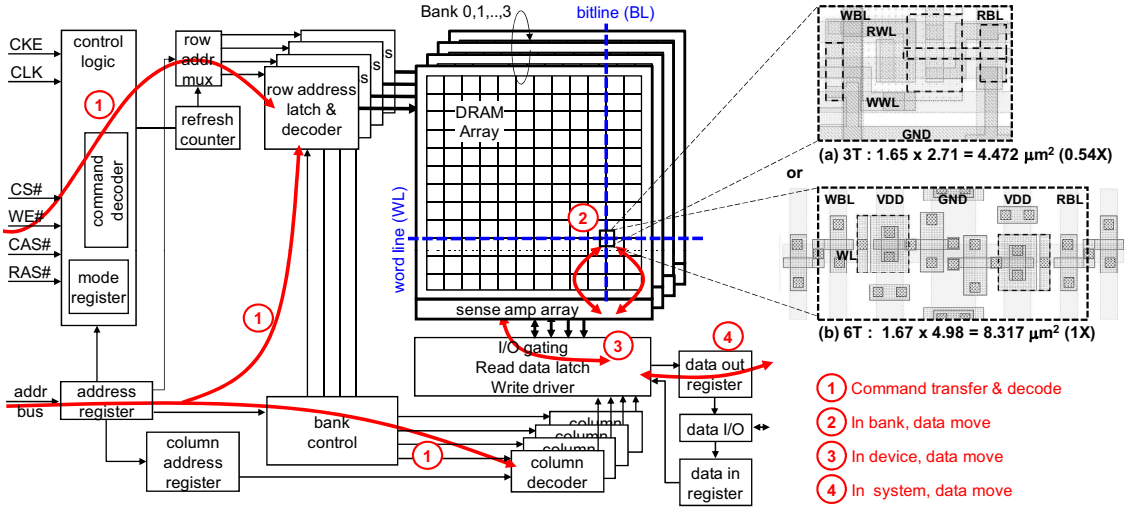


Figure 2. Generic memory architecture, array, cell layout and command and data movement. (**a**) 3T [23] and (**b**) 6T [8] gain cells.

2.2. Memory Array Structures

The device in Figure 2 has four memory banks of DRAM arrays and each bank has its own arrays of sense amplifiers and row and column address decoders. A single memory bank consists of plural arrays of storage cells where a row address is decoded to assert one or pair WLs driven jointly to activate one cell on each one of thousands of BLs. That is, target cell is selected by asserting the corresponding WL and then BL. Hundreds of cells may be connected to the same BL but only one cell puts charge stored at its storage capacitor onto BL at a time. The resulting voltage on BL is then developed into a digital value '1' or '0' by a differential sense amplifier. The storage cell selected by the forwarded row and column addresses in the chosen bank is marked with blue dash lines.

In today's DRAMs, the capacitance of a cell's capacitor is much smaller than that of BL. Typically, the capacitance of a storage capacitor is about 10% of that of BL connected to hundreds of other cells. This causes an issue that when small charge stored in a cell is discharged to BL, the resulting voltage variation on BL is too small to measure in an absolute sense. A differential amplifier address this voltage sensing issue by comparing the BL voltage against a reference voltage. The main functionality of the sense amplifier is examined later.

2.3. Generic DRAM Structure, Array and Operation

Figure 2 also shows a generic architecture of embedded DRAM devices. Like other modern eDRAMs, our device has split address registers controlling dataflow in the device. During row access process, the address register forwards an address to the row address latch and decoder activating the selected WL. Data, the entire page selected with WL, are then discharged onto BLs. The differential amplifier arrays then sense, amplify and hold the data until a successive column access command either reads out the data through the shared I/O gating channel to the data bus or takes write data from the data bus through the channel and overwrites data in the amplifier array and memory cells with new values.

Figure 2 further illustrates multiple phases of memory operation occurring in a DRAM device to perform data movement. Based on the generic DRAM access protocol are four pipelined phases of operation for a general access command: in "Command transfer & decode" phase, a command is transferred along the command and address buses and decoded by the device; in "Data move in Bank" phase, data is moved inside a bank either from the cells to the sense amplifiers or from the amplifiers back into the cell array; in "Data move in Device" phase, the data is moved along the shared I/O channel, read latches and write drivers; and finally in "Data move in System" phase, the data is placed onto the data bus by the DRAM device or the memory controller. As the data bus may be

connected to multiple ranks of memory, no two commands to different ranks can use the shared data bus simultaneously.

2.4. Differential Sense Amplifier

In DRAMs, a differential sense amplifier converts a small amount of electric charge stored in the storage capacitor into a digital value. The amplifier senses the voltage difference in between the BL pair and then amplifies or develops this tiny difference to a normal voltage level (logic "0" or "1") while rejecting any voltage common to the two inputs. The voltage value developed from the sense amplifier of the target cell is then latched into a buffer and finally read out as an output.

Sense amplifiers in the devices perform three different functions. First, they sense the minute change in voltage level that occurs when an access transistor is turned on and a storage capacitor discharges its charge onto the associated BL. The sense amplifier compares the voltage on that "selected" BL against to a reference voltage provided on the other separate BL and amplifies the small difference in voltage to the extreme so that the difference can be developed into a logic value "1" or "0". This is the primary function of the sense amplifier.

Second, they restore the value of a cell after the voltage on the corresponding BL is sensed and amplified. The act of turning on the access transistor allows a storage capacitor to share its stored charge with BL. This charge sharing however is a destructive process from the storage cell perspective. As a result, after every sensing and amplification operation, the sense amplifier must restore the amplified voltage value back to the storage cell whose voltage level became lowered or spoiled and cannot be used for further accesses.

Third, an array of the sense amplifiers effectively plays as a temporal storage buffer caching an entire row of data. The amplifier array keeps driving the sensed and amplified data values until the cells are precharged and ready for other accesses. When subsequent read accesses are made to the same row and different columns, these new requests can be replied immediately by reading from the amplifier array without repeated row accesses to the cells. This capability can improve performance especially when the memory access sequence has a high degree of temporal and spatial locality.

2.5. Sense Amplifier Operation

Figure 3 shows a schematic diagram of a basic sense amplifier with an open DRAM array layout, where two DRAM arrays are arranged next to each other. Bitlines (BLs) at the same height are paired and gated into a sense amplifier pulling up the voltage differential between the two BLs. Complex sense amplifiers further contain auxiliary components for delicate balance of the sense amplifier structure, array isolation and fast sensing speed, and so forth.

In typical DRAMs, any access operation directed toward memory cells begins with voltage equalization between the BL pair in the array. The voltage equalization circuit controlled by the equalization signal (EQ) in Figure 3a ensures that the voltages on the BL pair (RBL and RBLB) are as closely balanced to each other as possible. A precharging voltage of $V_{DD}/2$ drives the two BLs to have the same voltage. Since the differential sense amplifier is designed to amplify the voltage difference in between the BL pair, any voltage mismatch existing on the pair before the activation of the access transistors would debase the efficiency of the amplifier.

The core of the sense amplifier is the set of four cross-coupled transistors, labeled as sensing circuit in Figure 3a. The sensing circuit is basically a bi-stable circuit devised to drive the two BLs to complementary voltage extremes depending on the respective voltages on the BLs when the sensing signals SAN and SAP are asserted. Figure 3b shows a different view of the same circuit for a read operation with physical layout in mind. One bitline (RBL) delivers a voltage value read out from the target cell whereas the other bitline (RBLB) provides a reference voltage V_{REF} of $V_{DD}/2$.

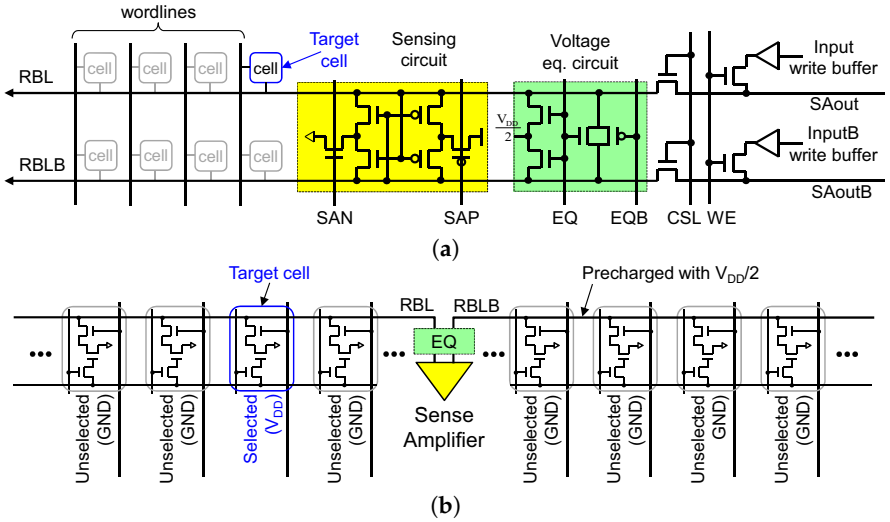


Figure 3. Basic sense amplifier. (**a**) Conceptual circuit diagram and (**b**) Circuit top view (read operation).

Figure 4 details four different phases. Circuit parts activated at each phase are marked in red. Precharge phase is a prerequisite for the next consecutive phases and typically considered separate from the row access operation whereas Access, Sense and Restore phases are automatically conducted in order for every row access operation.

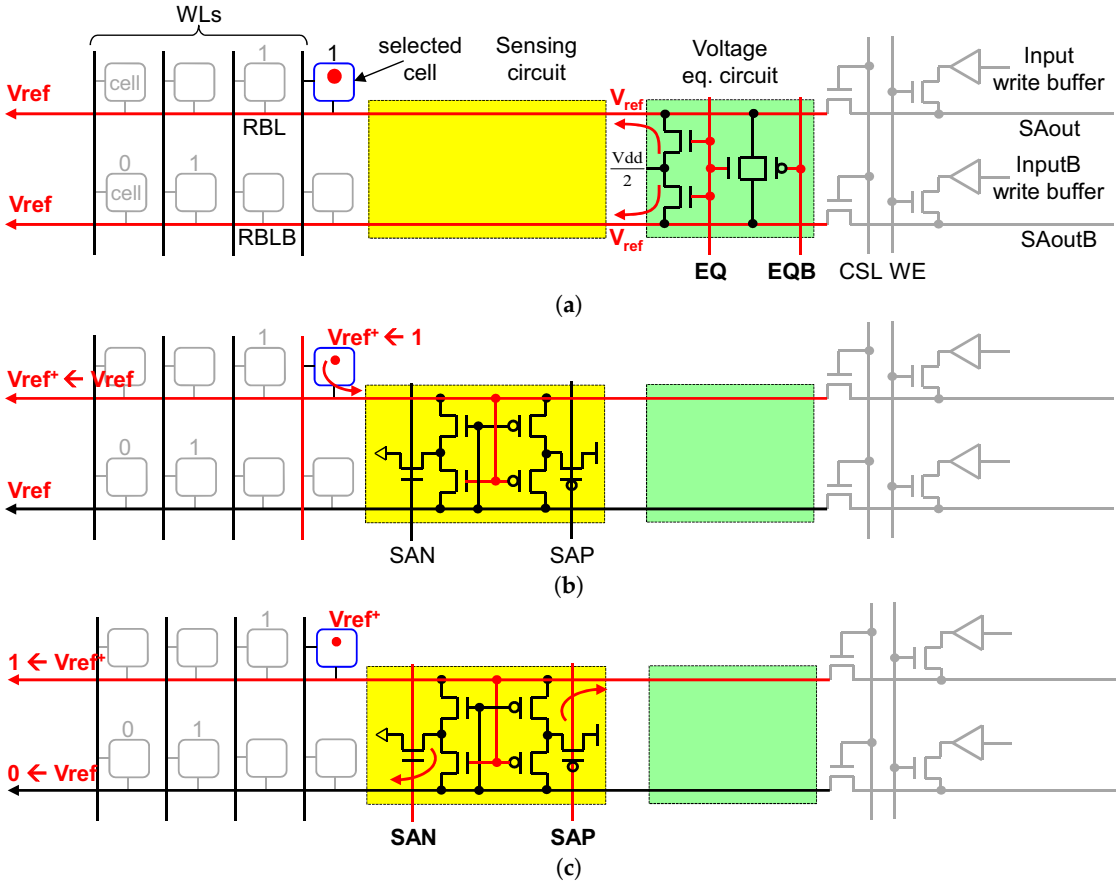
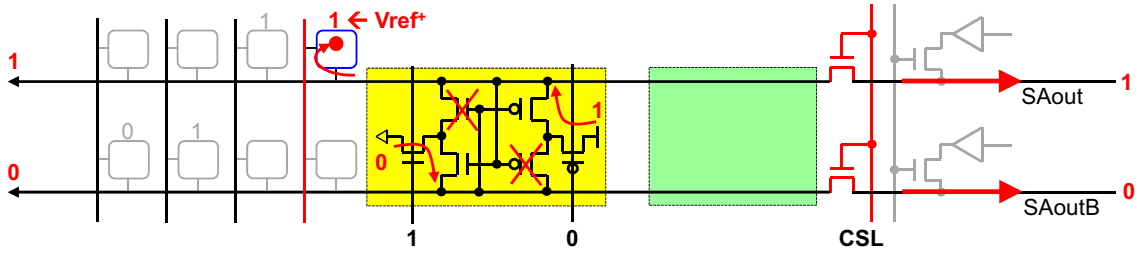


Figure 4. *Cont.*



(**d**)

Figure 4. Illustrative diagrams of sense amplifier operation (Read example). (**a**) Precharge, (**b**) Access, (**c**) Sense and (**d**) Restore phases.

Precharge phase assures that before reading data from a DRAM array begins, BLs in the array are precharged to a reference voltage V_{REF} first. Many today's DRAM devices utilize the intermediate between the supply voltage and ground (i.e., $V_{DD}/2$) as V_{REF}. Figure 4a shows that as the voltage equalization circuit turns active, both BLs are set to V_{REF}.

Access phase represents that when RWL is overdriven with a higher-than-V_{DD} voltage, the storage capacitor gets charged to full voltage level (V_{DD}), where voltage overdrive on RWL is provided by an additional level-shifting voltage pumping circuit. The voltage-overdriven RWL activates the access transistors allowing the charge stored in the cell now discharged onto RBL. Since the voltage in the cell was high representing a digital value of '1', the voltage on RBL slightly increases from V_{REF} to V_{REF}^{+}. This increased voltage on RBL begins to affect operation of the bi-stable sensing circuit. As shown in Figure 4b, this slightly higher voltage on RBL triggers the lower transistor to be more conductive than the upper one. Subsequently, the small voltage difference makes the lower device less conductive the upper one further.

Sense phase confirms that as the tiny voltage difference drives a bias into the bi-stable sensing circuit. The N-sense and P-sense amplifier protocol signals, SAN and SAP, collectively force the sensing circuit to be driven to the respective extremity. Figure 4c exhibits that as SAN is asserted, the more conductive lower N-type transistor enables SAN to pull down the voltage on the lower bitline (RBLB) from V_{REF} to ground. At the same time, SAP now drives RBL to reach a fully recovered voltage level defining the digital data of '1'.

Finally, Restore phase in Figure 4d ensures that when BLs are driven to the extremes respectively, the overdriven WWL (not RWL) turns on and fully driven WBL (not RBL) recharges the cell capacitor via the access transistor while the voltage on RBL passes to WBL via a transmission gate (not shown in the figure). Meanwhile, the voltage values of BLs can be read from the amplifier to output the requested data. In such a way, the data of a row can be accessed and driven from a DRAM device in parallel with data recovery.

2.6. Writing into a DRAM Array

Figure 5 shows a simple timing sequence for a write operation. As described above, during the row activation process, data recovery is automatically performed from the sensing circuit to the memory cell. For a write command, however, data provided by the memory controller is buffered first by the input write buffer and then driven to overwrite the sensing circuit and the target cell. In case of write operation, Write Recovery phase may follow the Restore phase. Referring to the relative timing of Figure 6, adding a column write command just means that a precharge command is issuable only after data restoration process is complete correctly. The time required for write data to overwrite the sense amplifier and update the cell is named as the write recovery time t_{WR} in Figure 5.

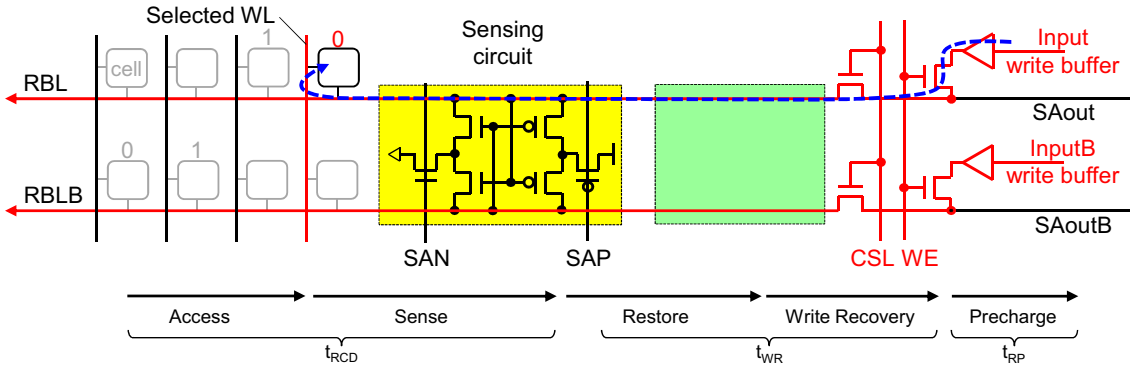


Figure 5. Row activation followed by column write into DRAM array.

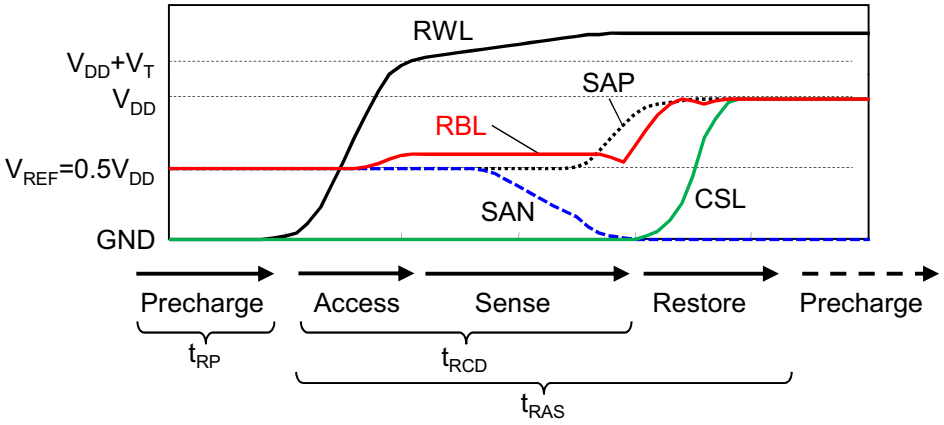


Figure 6. Sense amplifier voltage waveforms for read operation (Simulated).

3. Theoretical Background of Memory Logic Operation

Digital electronic circuits have been constructed by using logic gates only. In modern digital electronics on logic, generally taken as primitive is a subset of the connectives: inversion or negation (NOT), conjunction (AND) and disjunction (OR), where the names in parentheses are the corresponding logic gates. More connectives can be prescribed, if so wanted, by expressing them by means of these primitives. That is, we can build up arbitrarily complex combinational switching circuits by using an interconnection of a set of simpler combinational circuits of these primitives.

From the mathematical theory viewpoint, we can represent all possible truth tables with a functionally complete set of logical connectives or Boolean operators by combining elements of the set into a Boolean expression [24–26]. In electronic circuit design, Boolean functions are normally implemented by using logic gates. Hence, from the digital electronics perspective, functional completeness means that all possible logic gates can be expressed as a combination or network of logic primitives defined by the set as well.

There are again three basic logic operations in Boolean expression: negation (NOT), conjunction (AND) and disjunction (OR). In fact, this subset still involves some redundancy and is not a minimal set yet. Note that conjunction and disjunction can be realized with other connectives. For example, the conjunction of two inputs A and B can be implemented by negating disjunction of two negated inputs (i.e., AND (A,B) = NOT (OR (NOT A, NOT B))). The inverting gate however is irreplaceable since no others can produce a reversed value of the input(s). A well-known functionally complete set of connectives in binary system is {NOT, OR} or {NOT, AND}, comprising binary negation and either one between binary conjunction or disjunction primitive. In this context, the singleton set of {NOR} or {NAND} is also functionally complete since negation of one input can be performed by negating either one of these singleton sets with multiple inputs identical to each other. In other words, from the electronic circuit design standpoint, all logic gates can be represented by a combination of either only NOR gates or only NAND gates. This concept of a functionally complete set of logical connectives or

Boolean operators provides the theoretical ground for our scheme, in-situ memory logic operation, proposed in this paper.

In the next two sections, with several examples we will discuss how to practically achieve functional completeness of Boolean operations and thus carry out logic operations in memory cell arrays that are innately capable of "NOR" logic calculation thanks to their own generic structure.

4. Logic Computation in Memory Array

We here discuss the structure of memory cell arrays for logic operations or Boolean functions using the 3T memory cell [23]. To necessarily and sufficiently provide functional completeness of logic operations, at least two logic functions should be supported: negation and OR (or AND) operations.

The first mandatory operation for functional completeness, negation is generally performed with a NOT gate commonly known as an inverter that produces an inverted version of the input at its output. Figure 7 shows a 3T memory cell that performs an inversion or NOT operation where the output is HIGH or LOW as the input A is LOW or HIGH, respectively. The atomic unit in the array is a memory cell that is composed of one or multiple transistors with one or two bitlines for inputs and output. The procedure of in-situ memory NOT operation is as follows: (1) pre-charge the read bitline (RBL); (2) write the value driven on the write bitline (WBL) into node A by asserting WA to HIGH (or '1'); and (3) read out the cell value stored at node A by asserting RA into RBL, where WA is the 'write enable A' and RA is the 'read enable A' signal and both are wordlines. The cell value at node A is stored at the gate capacitance C_A of the storage transistor SA in the cell. We use an arrow(s) to express in-situ memory logic operation as shown in the symbol.

The second mandatory operation for functional completeness, OR or AND operation, can be performed with negation of NOR or NAND function, respectively. Figure 8 now presents a two-input wired-NOR operation with two inputs A and B and output storing cell Z, where a tri-state write driver TZ (transfer enable Z) connecting RBL to WBL enables writing the in-situ memory logic operation result (RBL) back to the same array (WBL) externally as well as internally or other arrays if necessary. Note that rather than to the cell Z, the output can be stored back to one of input cells A or B if required as well. The truth table confirms correct NOR operation where the output of the memory cell array becomes LOW only when inputs A and/or B are HIGH. The sequence of in-memory NOR operation is similar to that of NOT operation.

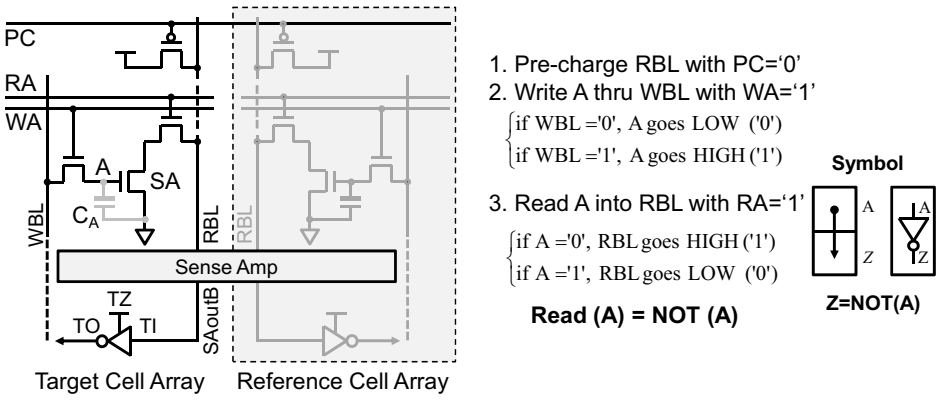


Figure 7. 3T memory cell for negation (NOT) operation (inverter).

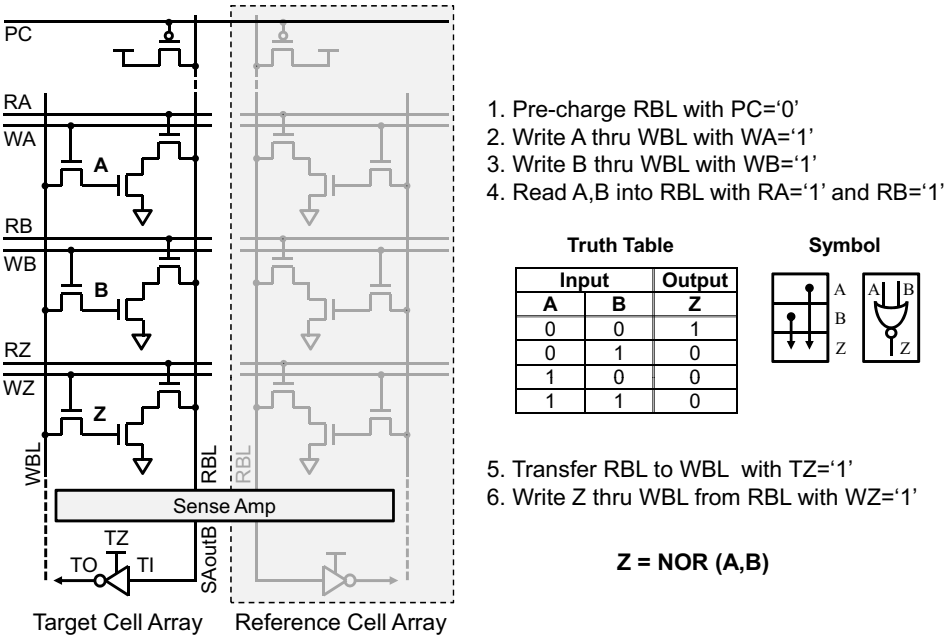


Figure 8. 3T memory cell array for two-input NOR operation.

It is worthwhile to mention that in-situ memory logic operation can be extended and performed with large number of inputs in a more effective way compared to the conventional design where logic operation is done by combinational random logic gates. For example, Figure 9 depicts a three-input wired-NOR logic operation with inputs, A, B and C and output storing cell Z. Note that the fan-in overhead of our design increases linearly with the number of inputs whereas that of the conventional design with logic gates increases more than in a quadratic way. This can help lower complexity and power consumption further in signal processing or ECC (error control code) applications where computation with many inputs is often desired.

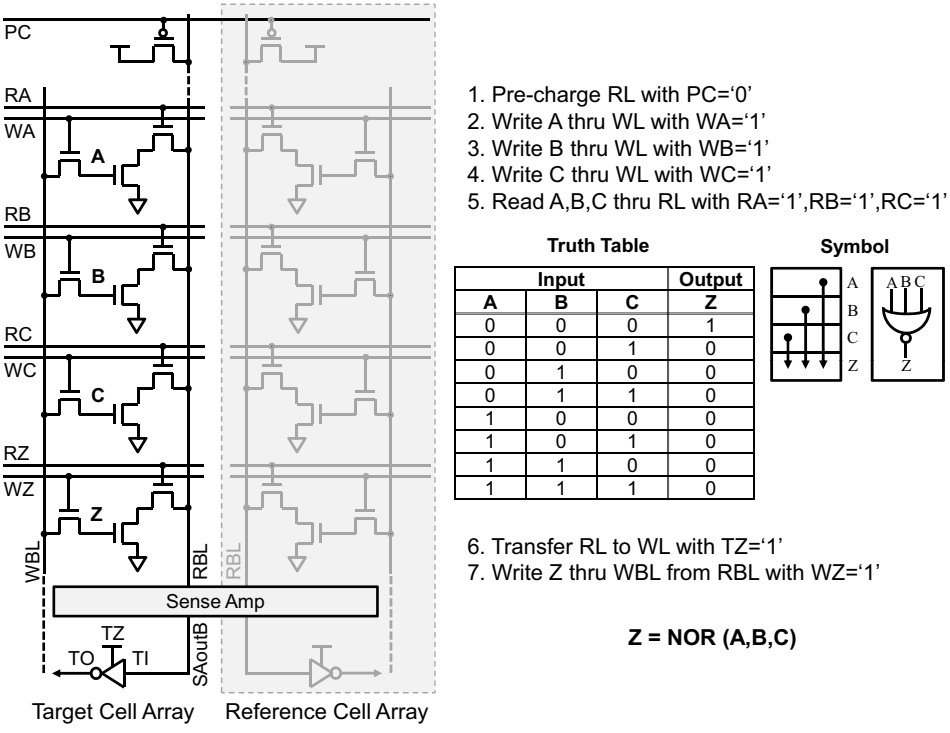


Figure 9. 3T memory cell array for three-input NOR operation.

5. Complex Logic Computation

Multiple arrays of memory cells enable many in-memory logic operations to be performed in parallel for more complex Boolean functions. Figure 10 shows one possible implementation of the 1-bit full adder where X, Y and Z are the two input and carry-in signals and C and S are the carry-out and sum outputs, respectively. In proposed in-situ memory logic, complex logic computation is performed through iterative wired-NOR operations along the bitlines whereas in conventional logic, logic operation is carried out sequentially throughout the random logic gates connected in cascading order.

The timing diagram of a sequence to compute full addition in a memory array is shown with arrows in Figure 10. During computation, the cell values including intermediate results are notated at the right side at each cycle. Assuming two inputs X and Y and carry-in Z are stored in their corresponding memory cells, the sum S and carry-out C outputs are calculated at the 11th and 12th cycle, respectively.

As another example of complex logic, straightforward implementation of the two-input exclusive-OR operation is shown in Figure 11, where interim values are calculated by following the De-Morgan Theorem. Note that even though there are many ways to build an arbitrary function with various logic gates, in-situ memory logic uses only NOR and NOT (or inverter) gates as basic building components in order to directly leverage the innate wired-NOR feature of memory arrays, which is the key to our approach proposed in this paper.

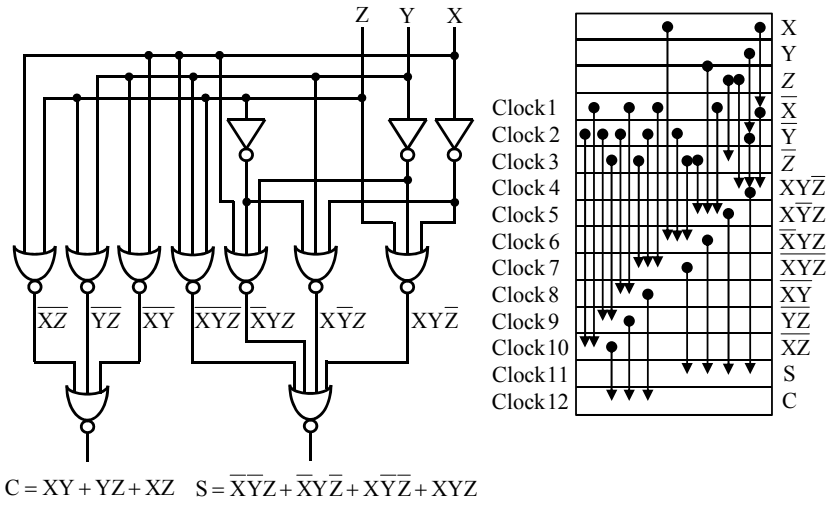


Figure 10. Implementation of an 1-bit full adder.

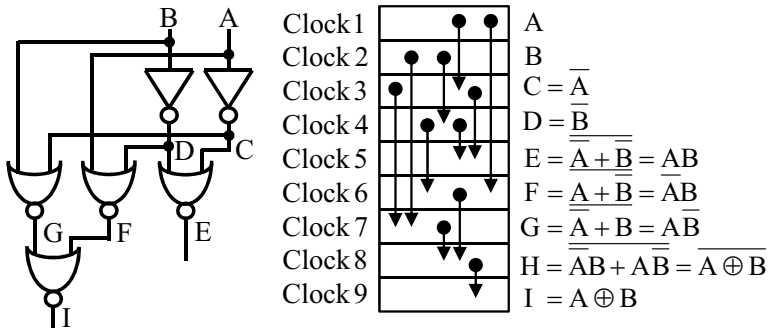
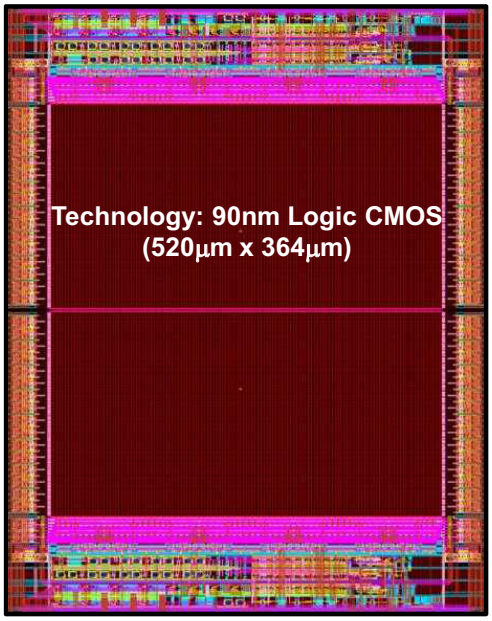


Figure 11. De-Morgan Theorem.

6. Results and Discussions

To demonstrate in-situ memory logic operation, we choose a two-bank 32-kbit memory array system. Figure 12 shows the GDS (graphic database system) view and summary of a test memory chip fabricated in 90 nm 8-metal layer logic CMOS technology.



Technology (Metal Levels)	90nm logic process 8-metal CMOS
Core Size (Density)	520 × 364 μm^2 (16kbit/bank x 2 banks)
Gain Cell Area (3T/6T conv. cell)	1.65 x 2.71 = 4.472 μm^2 (0.54X)
Supply Voltage	135 mV~1.2 V (Sub-V_T to Super-V_T)
Retention Time	Target 10 μs
Threshold Voltage	PMOS: 0.27 V NMOS: 0.23 V
Power Dissipation	130 nW @V_{dd}=135 mV 0.94 μW @V_{dd}=0.3 V 367μW @V_{dd}=1.2 V
Operating Frequency	750 Hz @V_{dd}=135 mV 611 KHz @V_{dd}=0.3 V 980 MHz @V_{dd}=1.2 V

Figure 12. GDS layout view of a test chip in 90 nm technology.

Data in a memory chip is stored in tiny circuits called memory cells. Sense amplifiers are key components for valid and reliable in-situ memory logic operations as for conventional memory read and write operations. As part of core memory circuitry, sense amplifiers are primarily applied in volatile memory cells. As discussed in Section 2.5, their role is to sense the low power signals from a bitline that represents a data bit ("1" or "0") stored in a memory cell and amplify the small voltage swing to recognizable logic levels so that the data can be interpreted properly by logic outside the memory. Simulations are performed for the memory read cycle at the normal operating voltage of 1.2 V in a 90 nm logic-compatible CMOS process. Figure 6 presents the voltage waveforms of the bitlines and selected control signals in one sense amplifier implemented as arrays in this work. Note that before a row access operation gets started, RBL (and RBLB not shown in the figure) is precharged and the voltage on RBL is set to V_{REF}. At first, the wordline RWL is overdriven to at least V_T above V_{DD}, where V_T is the threshold voltage of an access transistor and the storage cell discharges its content onto RBL raising the RBL voltage from V_{REF} to V_{REF}^{+}. Then, the asserted SAN and SAP quickly drive RBL to reach a full voltage level V_{DD} and finally restore charge onto the cell capacitor.

Figure 6 also shows the relationship among timing parameters of interest for a read command. After the delay t_{RCD} from the initiation of row access operation, sensing and amplification are performed and the data is available and ready to be read from the sense amplifiers onto the device data bus through the column access process. Due to the volatile feature of DRAM memory cells, however, the data has yet to be restored to the cells. Hence, assuming that the data restoration is made complete after t_{RAS}, the DRAM device gets ready to take a precharge command that concludes the entire row access operation after t_{RP}.

The sense amplifier operation in SRAM is quite similar to that in DRAM but the sense amplifier in DRAM performs an additional function. The data in DRAM memory chips is stored as electric charge in tiny capacitors in the memory cells. The read operation depletes the electric charge stored in a cell, destroying the representative data, so after the data is read out the sense amplifier must immediately write it back in the cell by applying a voltage to it, recharging the capacitor, which is called memory refresh.

Figure 13 now shows the waveforms of the write driver signals measured from a memory array in the middle of 1-bit full adder operation at a supply voltage of 300 mV. We employ direct probe capability for the input and output signals of the write driver of the array with the ESD (electrostatic discharge) protection in the pads. The input signal TI of the driver is the complementary output

(SAoutB) of the sense amplifier that amplifies the result of wired-NOR operation, that is, the read bitline (RBL). Note again that while the transfer enable signal TZ is activated the output TO of the write driver drives WBL which can be fed as an input to the same array internally as well as externally or to other arrays for the next in-memory logic operations.

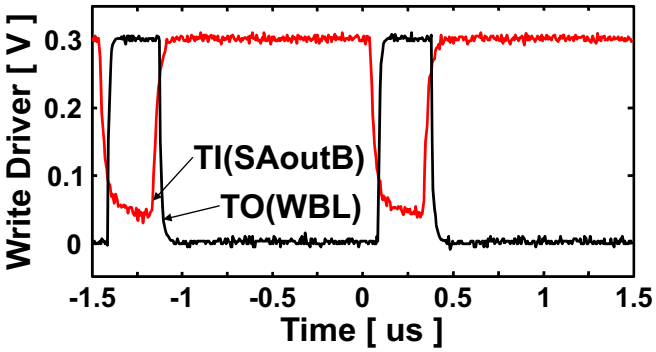


Figure 13. Waveforms of write driver (Measured).

We employ a level-down shifter using a simple voltage divider scheme in the work. A Tetronix pattern generator is used to generate input and clock signals. This generator has a 2 V output swing and thus level down converters are embedded to adjust the signals' voltage level. On the other hand, to get the boost effect on the wordline drivers, the power supply pads of the row and column address decoders are placed separately from others in the test chip layout. Not only sensing speed improvement could be achieved but also signal reliability. In the proposed sense amplifier, one shut wordline boost is used to the simulation and silicon. By wordline boost, the control gate of memory cell could be higher than V_{DD}, improving memory cell current significantly and thus sensing speed. This boosting scheme, however, results in more power consumption in wordline circuitry and boosting read and write wordlines requires careful design of the embedded charge pumps and level shifters.

Figure 14 shows a schematic of an operational amplifier style level-up shifter. This level shifter is used for communication with the outside and internal signal boosting. The aim of level shifting for external interface is obvious whereas its justification for internal boosting is as follows.

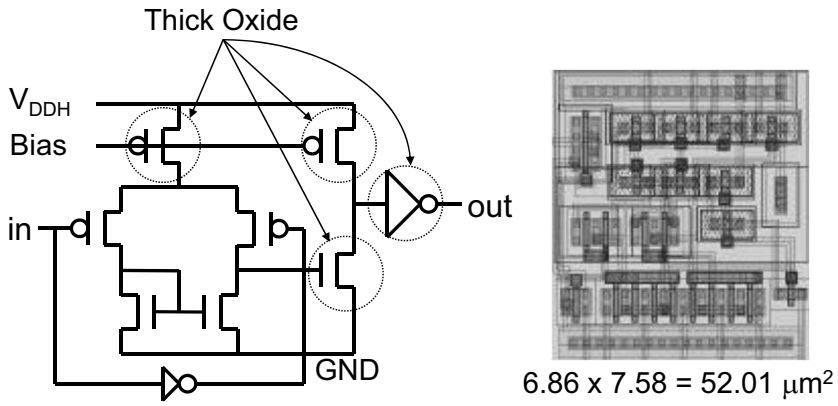


Figure 14. Schematic of a level-up shifter.

The clock and control signals propagate into the memory core block and their voltage level is adjusted through level shifters according to the region of operation. Memory cell accesses begin by asserting the associated wordline that turns on the access transistor. As discussed in Section 2.1, however, an NMOS transmits a weak '0' (not full ground) and a PMOS does a weak '1'(not full V_{DD}), which can directly degrade writability and read stability of a memory cell. Moreover, their degradation is getting more serious with V_{DD} scaling.

To compensate this degradation caused during memory cell accesses, we adopt a boosting scheme that provides strong writability and read stability even in the worst-case process corner (slow NMOS and fast PMOS). The wordline drivers for both write and read accesses are boosted by 25% of a supply voltage of the memory core. Near and in the subthreshold region of operation (say, $V_{DD} \leq 270$ mV), more aggressive boosting is required. Due to voltage level boosting applied to WWL, degradation of writability resulting from voltage-drop across the NMOS structure in the write driver is marginal. Similar to the read stability with RWL during read accesses.

This technique enables the minimum operating supply voltage $V_{DD,min}$ of the test chip to be lowered all the way down to 135 mV. $V_{DD,min}$ can further scale down with more aggressive boosting (e.g., 50% boosting for more supply voltage scaling) at the expense of performance.

In ultra-low voltage mode, the test chip operates at 300 mV supply with 25% boosted wordlines, consuming dynamic switching power of 0.94 µW at a frequency of 611 KHz. With more aggressive wordline boosting of 50 mV, the design operates down to 135 mV. At this minimum voltage $V_{DD,min}$, the operating frequency and read power measured are 750 Hz and 130 nW, respectively.

On the other hand, our other experiment with simulations shows that power consumption of the proposed in-situ memory logic operation is six times smaller than that of conventional logic gate operation at the same operating frequency and five orders of magnitude smaller compared to normal voltage memory operation at the sacrifice of the maximum operating frequency, that is, performance.

One may argue that the proposed in-situ memory logic would take more power and computational time than the conventional gate-based logic to perform logic operation, because modern memory systems typically consist of various components such as BLs, WLs, SAs and column and row decoders, and so forth, while the conventional approach just requires set of logic gates. In this work, we implement the fine-grained memory architecture inside an eDRAM bank. A bank vertically consists of multiple sub-arrays. Each sub-array is further horizontally split into many memory cell matrices (MATs). A MAT has 512 $\times$ 512 storage cells in row (WL) and column (BL) dimensions with a local row decoder and a local SA array (i.e., row buffer). There are dedicated latches in each MAT to further hold the selected data and then relay them onto the internal data bus.

Note that like in the modern DRAM architectures, a MAT in our work is an atomic access unit for memory operation. For a single logic function, our proposed scheme consumes more energy by three order of magnitude and takes more computational time by two order of magnitude compared to the conventional gate-based style.

However, innate capability of parallel computation enables the in-situ memory logic to operate with marginal increase in power or energy consumption when processing additional hundreds or thousands of logic functions concurrently. With a finer-grain architecture, for example, 256 $\times$ 256 MAT size, energy efficiency would be increased in a quadratic way and become comparable to that of the conventional approach. The reduced MAT size will also help to improve the memory cell access time by reducing the capacitive load associated with BLs and WLs at the cost of parallel computation in memory arrays. Optimization of the MAT size would be required for low power yet high parallelism in-situ memory logic design according to target applications.

Note that one possible big advantage in the proposed scheme may be reduced overhead in memory bandwidth or bottleneck. In general, modern computer systems have suffered from the insufficient memory bandwidth. The proposed scheme can reduce power consumption by eliminating data movement back and forth between the processing unit and the storage unit. To our knowledge, this is one of main reasons why like AMD, Intel has implemented eDRAMs as cache memory inside their mainstream microprocessors since the first generation of Haswell and IBM's POWER7 processor applied an eDRAM technology for L3 cache memory. Further, unlike the gate-based topology, in-situ memory logic is is less sensitive to the fan-in and fan-out issues and reconfigurable.

7. Conclusions

Besides generic storage capability, memory arrays can mathematically provide a functionally complete set of Boolean operations and thus perform all logic computation as well. Measurements from a test chip fabricated in 90nm demonstrated our proposed in-memory logic operation for ultra-low power applications while performing correct logic calculation. Low-power would be more important than performance in some isolated and extreme environments. We hope that our approach contributes to mission-critical yet battery-operated and low-power applications like space or deep ocean explorations.

Author Contributions: Conceptualization, M.-E.H. and S.K.; methodology, M.-E.H.; validation, M.-E.H. and S.K.; formal analysis, M.-E.H.; resources, M.-E.H.; writing—original draft preparation, M.-E.H.; writing—review and editing, M.-E.H. and S.K.; supervision, S.K.

Funding: This work was supported by the 2019 Research Fund of University of Ulsan.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Von Neumann, J. *First Draft of a Report on the EDVAC*; Moore School of Electrical Engineering, University of Pennsylvania: Philadelphia, PA, USA, June 1945.
2. Somasekhar, D.; Ye, Y.; Aseron, P. 2 GHz 2 Mb 2T gain cell memory macro with 128 GBytes/s bandwidth in a 65 nm logic process technology. *IEEE J. Solid State Circuits* **2009**, *44*, 174–185. [CrossRef]
3. Luk, W.K.; Dennard, R.H. A novel dynamic memory cell with internal voltage gain. *IEEE J. Solid State Circuits* **2005**, *40*, 884–894. [CrossRef]
4. Chun, K.; Jain, P.; Lee, J.; Kim, C.H. A sub-0.9 V logic-compatible embedded DRAM with boosted 3T gain cell, regulated bit-line write scheme and PVT-tracking read reference bias. In Proceedings of the Symposium on VLSI Circuits, Kyoto, Japan, 16–18 June 2009; pp. 134–135.
5. Teman, A.; Meinerzhagen, P.; Burg, A. Review and classification of gain cell eDRAM implementations. In Proceedings of the 2012 IEEE 27th Convention of Electrical & Electronics Engineers in Israel (IEEEI), Eilat, Israel , 14–17 November 2012.
6. Meinerzhagen, P.; Teman, A.; Giterman, R.; Burg, A.; Fish, A. Exploration of Sub-V_T and Near-V_T 2T Gain-Cell Memories for Ultra-Low Power Applications under Technology Scaling. *J. Low Power Electron. Appl.* **2013**, *3*, 54–72. [CrossRef]
7. Lee, J.; Ahn, Y.; Park, Y.; Kim, M.; Lee, D.; Lee, K.; Cho, C.; Chung, T.; Kim, K. Robust Memory Cell Capacitor using Multi-Stack Storage Node for High performance in 90 nm Technology and Beyond. In Proceedings of the Symposium on VLSI Circuits, Kyoto, Japan, 10–12 June 2003; pp. 57–58.
8. Zhang, K.; Bhattacharya, U.; Chen, Z.; Hamzaoglu, F.; Murray, D.; Vallepalli, N.; Wang, Y.; Bohr, B.Z.M. SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction. *IEEE J. Solid State Circuits* **2005**, *40*, 895–901. [CrossRef]
9. Barth, J.; Reohr, W.R.; Parries, P.; Fredeman, G.; Golz, J.; Schuster, S.E.; Matick, R.E.; Hunter, H.; Tanner, C.C.; Harig, J.; et al. A 500 MHz random cycle, 1.5 ns latency, SOI embedded DRAM macro featuring a three-transistor micro sense amplifier. *IEEE J. Solid State Circuits* **2008**, *43*, 86–95. [CrossRef]
10. Romanovsky, S.; Katoch, A.; Achyuthan, A.; O'Connell, C.; Natarajan, S.; Huang, C.; Wu, C.; Wang, M.; Wang, C.J.; Chen, P.; et al. A 500 MHz random-access embedded 1Mb DRAM macro in bulk CMOS. In Proceedings of the 2008 IEEE International Solid-State Circuits Conference, San Francisco, CA, USA, 3–7 February 2008; pp. 270–271.
11. Klim, P.; Barth, J.; Reohr, W.; Dick, D.; Fredeman, G.; Koch, G.; Le, H.M.; Khargonekar, A.; Wilcox, P.A.; Golz, J.; et al. A 1 MB cache subsystem prototype with 1.8 ns embedded DRAMs in 45 nm SOI CMOS. *IEEE J. Solid State Circuits* **2009**, *44*, 1216–1226. [CrossRef]
12. Barth, J.; Plass, D.; Nelson, E.; Hwang, C.; Fredeman, G.; Sperling, M.A.; Mathews, A.; Kirihata, T.; Reohr, W.R.; Nair, K.; et al. A 45 nm SOI embedded DRAM macro for the POWER processor 32 MByte on-chip L3 cache. *IEEE J. Solid-State Circuits* **2011**, *46*, 64–75. [CrossRef]

13. Luk, W.K.; Cai, J.; Dennard, R.H.; Immediato, M.J.; Kosonocky, S.V. A 3-transistor DRAM cell with gated diode for enhanced speed and retention time. In Proceedings of the 2006 Symposium on VLSI Circuits, Honolulu, HI, USA, 15–17 June 2006; pp. 184–185.
14. Chun, K.; Jain, P.; Lee, J.; Kim, C.H. A 3T gain cell embedded DRAM utilizing preferential boosting for high density and low power on-die caches. *IEEE J. Solid State Circuits* **2011**, *46*, 1495–1505. [CrossRef]
15. Chun, K.; Jain, P.; Kim, T.; Kim, C.H. A 1.1 V, 667 MHz random cycle, asymmetric 2T gain cell embedded DRAM with a 99.9 percentile retention time of 110 s. In Proceedings of the 2010 Symposium on VLSI Circuits, Honolulu, HI, USA, 16–18 June 2010; pp. 191–192.
16. Chun, K.C.; Jain, P.; Kim, T.; Kim, C.H. A 667 MHz Logic-Compatible Embedded DRAM Featuring an Asymmetric 2T Gain Cell for High Speed On-Die Caches. *IEEE J. Solid State Circuits* **2012**, *47*, 547–559. [CrossRef]
17. Ichihashi, M.; Toda, H.; Itoh, Y.; Ishibashi, K. 0.5 V asymmetric three-Tr. cell (ATC) DRAM using 90 nm generic CMOS logic process. In Proceedings of the Digest of Technical Papers. 2005 Symposium on VLSI Circuits, Kyoto, Japan, 16–18 June 2005; pp. 366–369.
18. Sebastian, A.; Sebastian, A.; Tuma, T.; Papandreou, N.; Gallo, M.L.; Kull, L.; Parnell, T.; Eleftheriou, E. Temporal correlation detection using computational phase-change memory. *Nat. Commun.* **2017**, *8*, 1115. [CrossRef] [PubMed]
19. Li, S.; Niu, D.; Malladi, K.T.; Zheng, H.; Brennan, B.; Xie, Y. DRISA: A DRAM-based Reconfigurable In-Situ Accelerator. In Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture, Cambridge, MA, USA, 14–18 October 2017; pp. 288–301.
20. Hwang, M.E. Supply-Voltage Scaling Close to the Fundamental Limit Under Process Variations in Nanometer Technologies. *IEEE Trans. Electron. Devices* **2011**, *58*, 2808–2813. [CrossRef]
21. Hwang, M.E.; Roy, K. ABRM: Adaptive β-Ratio Modulation for Process-Tolerant Ultradynamic Voltage Scaling. *IEEE Trans. Very Large Scale Integr. (VLSI) Systs.* **2010**, *18*, 281–290. [CrossRef]
22. Wang, D.T. Modern DRAM Memory Systems: Performance Analysis and a High Performance, Power-Constrained DRAM Scheduling Algorithm. Ph.D. Thesis, University of Maryland, College Park, MD, USA, 2005.
23. Heshami, M.; Wooley, B. A 250-MHz skewed-clock pipelined data buffer. *IEEE J. Solid State Circuits* **1996**, *31*, 376–383. [CrossRef]
24. Enderton, H. *A Mathematical Introduction to Logic*, 2nd ed.; Academic Press: Boston, MA, USA, 2001; ISBN 978-0-12-238452-3.
25. Nolt, J.; Rohatyn, D.; Varzi, A. *Schaum's Outline of Theory and Problems of Logic*, 2nd ed.; McGraw-Hill: New York, NY, USA, 1998; ISBN 978-0-07-046649-4.
26. Smith, P. *An Introduction to Formal Logic*; Cambridge University Press: Cambridge, UK, 2003; ISBN 978-0-521-00804-4.