Design of Static Random Access Memory for Minimum Leakage using MTCMOS Technique

T. Esther Rani [1] and Dr. Rameshwar Rao [2]
[1] CVR College of Engineering, Dept of ECE, Hyderabad, India
estherlawrenc@gmail.com
[2] Hon'ble VC, JNT University, Hyderabad, India
Rameshwar_rao@hotmail.com

Abstract—**This work involves implementation of 8x8 SRAM using Multi Threshold CMOS (MTCMOS) technique which reduces the leakage power by placing sleep transistor either between ground and pull-down network or V_{DD} and pull up network. SRAM is designed using D-Latch. As reduction in the leakage power is more by using PMOS sleep transistor compared to NMOS sleep transistor, this design is implemented using MTCMOS technique with PMOS as sleep transistor. The design can be used where low power is the constraint, as it offers 86% of power savings.**

Index Terms—**SRAM, leakage power, MTCMOS, sleep transistor.**

I. INTRODUCTION

Memory refers to bits that are stored in a structured manner. In a two dimensional array, a row of bits are accessed at a time. Some kinds of memory may be integrated along with the logic in VLSI chips. Memory can be a primary memory or memory for other purposes in computers and digital electronic devices. Generally in CPU, Read Only Memories (ROM) are used to store complex instruction set; Static Memory is used to hold recently used instructions and data and Dynamic Memory for complete operating system, programs and data. DRAM is volatile and SRAM is fast cache memory used in CPU. SRAM is faster but energy-consuming and offers lower memory capacity per unit area than DRAM and non-volatile like flash memory, ROM, PROM, EPROM and EEPROM. SRAM does not require periodical refreshing and is faster than DRAM, the access time is very less and consumes less power compared to DRAM[1].

SRAM operates in three modes. They are standby mode or idle mode, Read mode and Write mode[2]. In idle mode, the SRAM is disabled. In read mode the data is read from a selected address location. In write mode the data is written into a particular address location. SRAM is used in many embedded applications. SRAM is also used in different industrial and scientific subsystems and automotive electronics. There are design tradeoffs like, density, speed, volatility, cost, and features. Suitable SRAM is selected based on these parameters. Applications of Static Random Access Memories include the revolutionary Quad Data Rate, NoBL and MoBL SRAMs from Cypress required for communications in the networks.

II. MTCMOS TECHNIQUE

In MTCMOS technique, low-V_t transistors are employed for the implementation of the logic block and high-V_t transistors for gating the logic block from the power supply. For this purpose, virtual power supply rails are placed rather than direct connection from the real power supply. By placing high-V_t transistors in this manner, leakage power is very much reduced. During active mode, the normal operation of the circuit is performed and during standby mode high-V_t transistors called sleep transistors are switched off by enabling an input to the sleep transistor. Sleep device can be a PMOS transistor which will be placed in between pull up network and V_{DD} or an NMOS transistor which will be placed in between pull down network and ground.

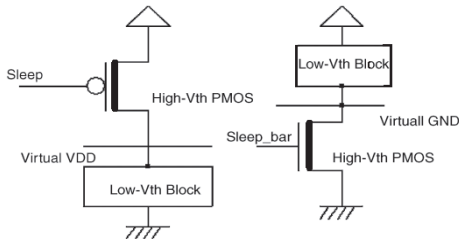


Figure 1: MTCMOS method

Effectiveness of MTCMOS in reducing power dissipation has been verified by implementing SRAM and the author observed great power savings as discussed.

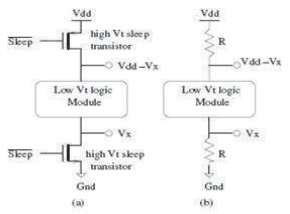


Figure2 : Sleep Transistors in MTCMOS circuits

MTCMOS circuit with both PMOS and NMOS as sleep devices is shown in figure 2.

Performance of a system can be improved by reducing leakage current. Sub-threshold leakage current is reduced by the sleep transistor with high Vt and performance loss is controlled. In MTCMOS Technique, there a disadvantage that, in sleep mode, the data that is stored in latches and flip-flops of logic blocks cannot be preserved when the power supply is turned off [3].

III. IMPLEMENTATION OF SRAM USING MTCMOS DESIGN WITH PMOS AS SLEEP TRANSISTOR

A high V_{th} PMOS transistor placed between V_{dd} and the logic block results in the MTCMOS design with PMOS sleep control transistor as shown in figure3.

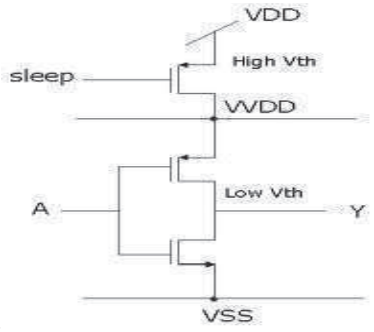


Figure 3 : MTCMOS design with PMOS sleep transistor for an inverter

The work involves in the implementation of SRAM using MTCMOS technique in Cadence Virtuoso Analog Design Environment. The design flow of SRAM is shown in figure 4[4,6].

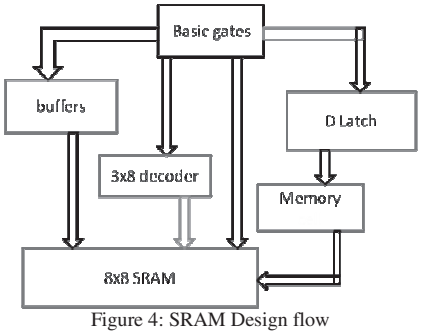


Figure 4: SRAM Design flow

A. D-LATCH WITH ENABLE

When an input of E is made '1', then the output of Q is same as the value on D. Now, the latch is said to be "open" and the path from D to Q is "transparent". So, this is called as a *transparent latch*. When the input E is made '0', then the latch "closes" and the output of Q retains as previous value and the changes made on D do not appear on Q. Figure [5] shows the circuit of D-Latch with enable.

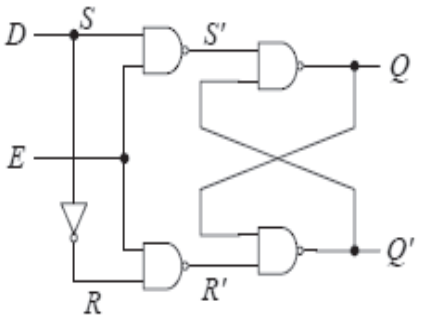


Figure5: D-Latch with enable

B. TRISTATE BUFFER

A tri-state buffer has three states. They are 0, 1, and *Z*. The value *Z* represents a high-impedance state, which acts like an open switch. The symbol, truth table and gate level circuit of tri state buffer is shown in figure 6 [5].

When *E* is made '0', the tri-state buffer is disabled, and the output *y* is high-impedance *Z* state. When *E* is made ' 1', the buffer is enabled, and the output *y* follows the input *d*.

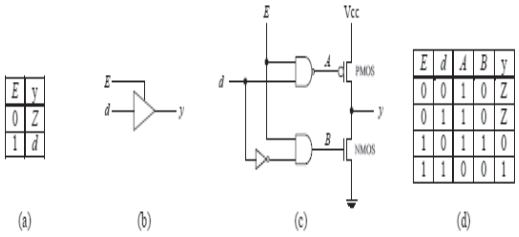


Figure 6: Tri-state buffer
(a) truth table (b) logic symbol (c) circuit (d) truth table

C. MEMORY CELL

A single bit memory cell of a SRAM chip is shown in Figure 7.

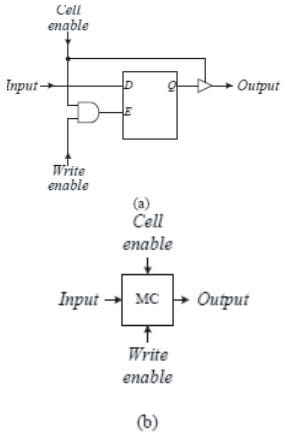


Figure 7: Memory cell (a) circuit; (b) logic symbol.

DOI: 10.32377/cvrjst0409

D latch with enable is used as a building block of SRAM. When read operation is selected, a tri-state buffer is connected to the output of the D latch. For reading, only the *Cell enable* signal is enabled. For writing, the *Cell enable along* with the *Write enable* signals are used to enable the D latch. Now, the data on the *Input* line is latched into the memory cell[1].

D. MEMORY TIMING DIAGRAM

During write operation, a suitable address on the address lines and suitable data on the data lines is specified and *CE* is enabled. When the *WR* line is asserted, the data here on the data lines is written into the memory location pointed by the address lines. Memory Timing Diagram with read and write operations is in Figure 8.

Suitable address on the address lines and *CE* to high make read operation begins. The *WR* is made low and suitable data from the addressed memory location is accessible on the data lines.

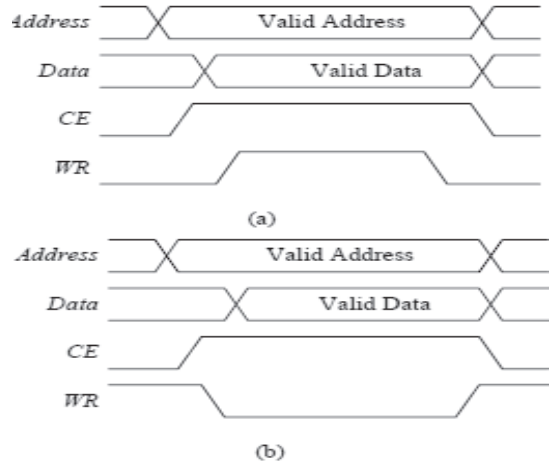
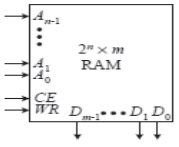


Figure 8: Memory Timing Diagram (a) read operation (b) write operation.

E. STATIC RANDOM ACCESS MEMORY

SRAM has data lines, *Di*, and address lines, *Ai*. Address lines specify the address and corresponding data on either input or output can be read or written into. Depending on the number of bits in the data, number of data lines are decided. Depending on the number of locations, the number of address lines are decided.



(a)

Figure 9: A 2^n x *m* RAM chip: (a) logic symbol

CE	*WR*	Operation
0	×	None
1	0	Read from memory location selected by address lines
1	1	Write to memory location selected by address lines

(b)

Figure 9: A 2^n x *m* RAM chip: (b) operation table.

Along with the data and address lines, two control lines called chip enable (*CE*) and write enable (*WR*) are also used as shown in figure9. For accessing memory of a microprocessor, either with read or write operation, the active-high *CE* line is enabled. The entire memory chip can be enabled by making CE high. The active-high *WR* line selects memory operation to be performed. For read operation, 0 is applied to *WR*, and data from the memory is retrieved. For write operation, 1 is applied to *WR* and microprocessor data is written into the memory. Some memory chips have separate read enable and write enable. The memory location in which the read and write operations are to be done, is selected by the corresponding address. The RAM chip designed in this work does not require a clock signal and is shown in figure 10. Here, the data operations are synchronized to the two control lines, CE and WR [5].

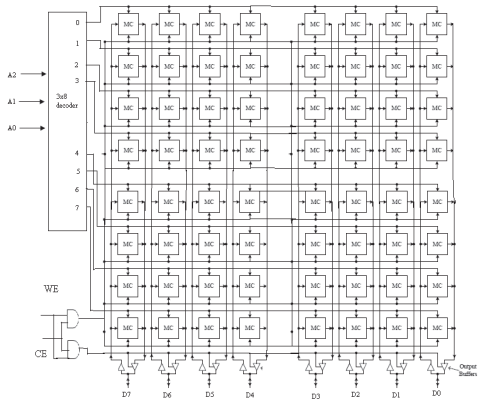


Figure 10: A 8X8 SRAM chip circuit.

To design 8X8 static RAM chip, 64 memory cells are required to form a 8X8 grid, as shown in Figure10.

Each row of the memory represents a single storage location. Number of bits in the data represents the number of memory cells in a row. By using the same address, all the memory cells of a row are enabled. A decoder is used to decode the address lines. Here, a 3 to 8 decoder is used to point to eight address locations. The *CE* signal enables the chip, and specific enabling of the read and write functions is through the two AND gates.

The external data bus, *Di*, gives data through the input buffer to the *Input* line of each memory cell. Depending on the given address, the data is written in to the memory cells. For read operation to be performed, *CE is* enabled and *WR is* disabled. This enables the internal *RE* signal, which in turn enables the eight output tri-state buffers. The location that is read from is selected by the address lines. Figure 11 and 12 shows the schematic of the

SRAM with PMOS MTCMOS cell and without the cell respectively.

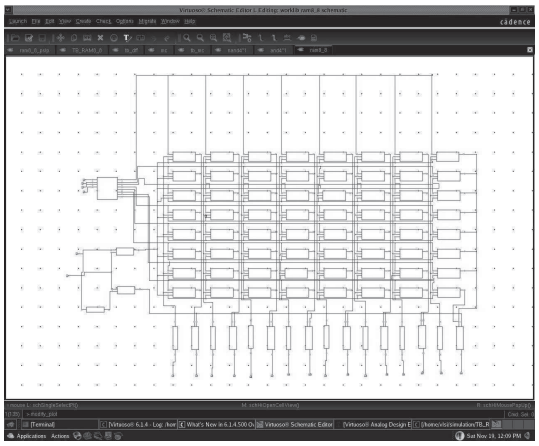


Figure 11:SRAM without MTCMOS

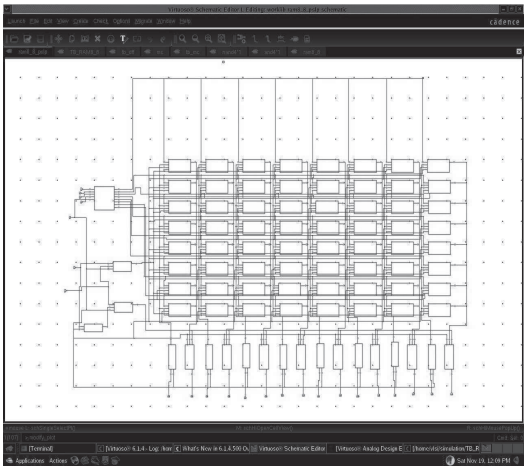


Figure 12: SRAM with MTCMOS

V. RESULTS

The operation of SRAM is verified and the average power is also calculated. Data is written into SRAM by using chip enable and write enable. The written data is read from SRAM chip. The results of read operation and write operation are shown in figure 13 and 14 with and without MTCMOS cell respectively. Power savings is observed to be 86% with the PMOS transistor used as sleep transistor.

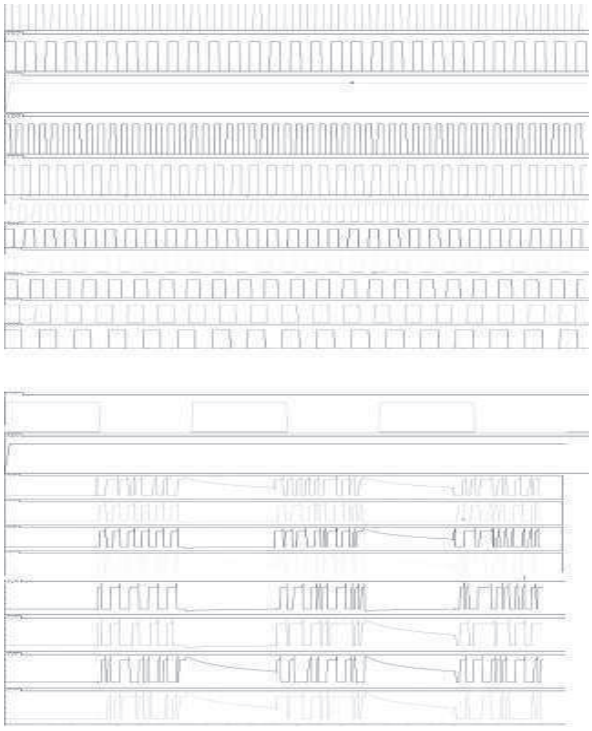
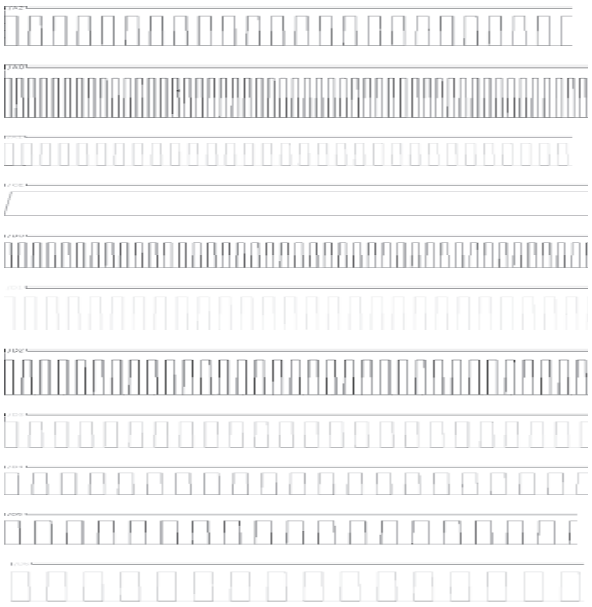


Figure 13: Output waveforms of SRAM without MTCMOS



DOI: 10.32377/cvrjst0409

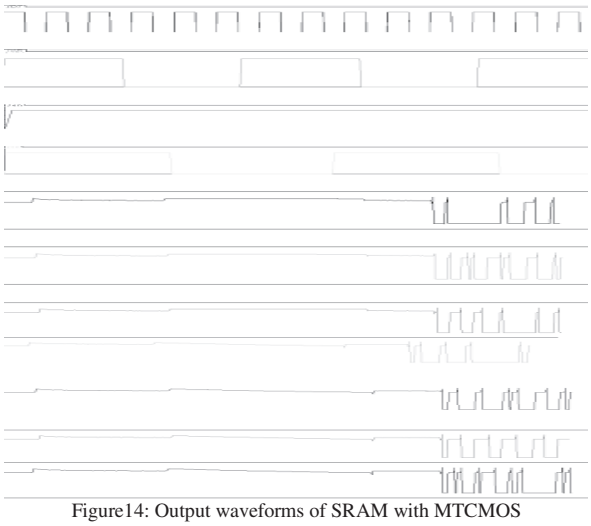


Figure14: Output waveforms of SRAM with MTCMOS

Table1
Comparison of power consumed with and without MTCMOS

S.NO	COMPONENT	WITHOUT MTCMOS	WITH MTCMOS
1	INVERTER	117.5nW	71.53nW
2	NAND GATE	248.9nw	98.12nW
3	D-LATCH	1.507uW	0.6268uW
4	TRI-STATE BUFFER	1.216uW	0.801uW
5	3x8 DECODER	9.47uW	2.113uW
6	MEMORY CELL	1.559uW	0.368uW
7	8x8 SRAM	97.92uW	13.11uW

CONCLUSIONS

In this work, SRAM is implemented for minimum leakage power dissipation with overall 86% of power savings. The SRAM operating in read mode and write mode should have "readability" and "write stability" respectively. SRAM has been verified for *reading* the data that is requested and *writing* the data or updating the contents. When the sleep enable is not active, SRAM will be in standby mode. There is tradeoff between area and power as PMOS takes more space on silicon but with much better savings in power dissipation.

REFERENCES

[1] Digital Design Principles & Practices- John F. Wakerly, PHI/ Pearson Education Asia, 3rd Ed., 2005

[2] Essentials of VLSI circuits and systems by Kamran Eshraghain, Eshraghian Dougles and A. Pucknell, PHI, 2005 Edition

[3] M. Anis, S. Areibi, and M. Elmasry, "Design and Optimization of Multithreshold CMOS (MTCMOS) Circuits," *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*,vol. 22, no. 10, pp. 1324- 1342, Oct. 2003.

[4] Cadence, "Cadence Low Power Design Flow" Cadence, "Low Power Application Note for RC 4.1 and SoCE 4.1 USR3", Version 1.0,1/14/2005

[5] Digital Logic and Microprocessor Design With VHDL by Enoch O. Hwang

[6] S. Sirichotiyakul and et al., "Stand-by Power Minimization through Simultaneous Threshold Voltage Selection and Circuit Sizing," *Proc. of the DAC*, pp. 436-441, 1999.