Timing Modeling Technology of DICE SRAM Based on SiliconSmart

Wu Yang, Yujing Li and Yi Tao

Sichuan Institute of Solid State Circuits, Chongqing, P.R. China.
Email: yangwu3101@163.com

Abstract. The rapid development of aerospace industry and semiconductor technology has put forward higher requirements for the radiation hardening of microelectronic devices. At the same time, with more and more applications of SRAM in SOC circuits, the anti-single event reversal effect ability of SRAM is more and more important. In this paper, the DICE structure is used to design SRAM to resist single event reversal effect. The simulation and test results show that the SRAM can resist single event reversal. In order to make SRAM with DICE structure be used in the digital design flow of SOC, this paper uses SiliconSmart to create timing model and successfully applies it to the design flow in the SOC project.

1. Introduction

With the development of integrated circuit technology to deep submicron, single event effect has put forward higher requirements for the radiation hardening of chip. As the process size decreases, the critical charge required for the inversion of the node level decreases, so the sensitivity of deep submicron integrated circuits to the single event effect will increase.

Single event effects mainly include single event transient effect (SET), single event latch-up effect (SEL) and single event reversal effect (SEU). The single event reversal effect (SEU) is the main source of single event effect in integrated circuits. Single event reversal (SEU) refers to the phenomenon that a single high-energy particle radiation causes the logical state of the circuit to be reversed, referred to as SEU. SEU effect will not cause irreversible damage to circuit function, and can restore its original function through system reset and re-charging, so SEU effect is also called soft error. Since there is no feedback signal in the combined circuit, when the duration of SEU effect is less than 1ns, most of the combined circuits can restore their original logical state after the high energy particle radiation is over. However, due to the feedback signal contained in the sequential circuit(such as SRAM), when the internal level of a node flips, the wrong data after the node flips may be locked, thus affecting the function of the circuit.

In this paper, we use DICE structure to design SRAM memory which can resist SEU effect. At the same time, SRAM with anti-SEU capability needs timing model to be integrated into the automatic place and routing flow of VLSI design. Therefore, this paper uses SiliconSmart to model the timing library of SRAM with DICE structure, so it can be used in the digital design flow.

2. Design of SRAM with DICE Structure

2.1. Theory Analysis of DICE SRAM

At present, the reinforcement technology against SEU effect is relatively mature, including Error Correction Codes, Triple Redundancy (TMR) and Double Interlocking Memory Cell (DICE) and other

reinforcement technologies. Considering the influence of area, speed and power consumption, SRAM with DICE structure is a widely used and dominant reinforcement technology in anti-SEU design. The DICE SRAM storage cell in this paper uses 12-tube DICE structure to reinforce the anti-SEU ability. The circuit structure is shown in Figure 1.
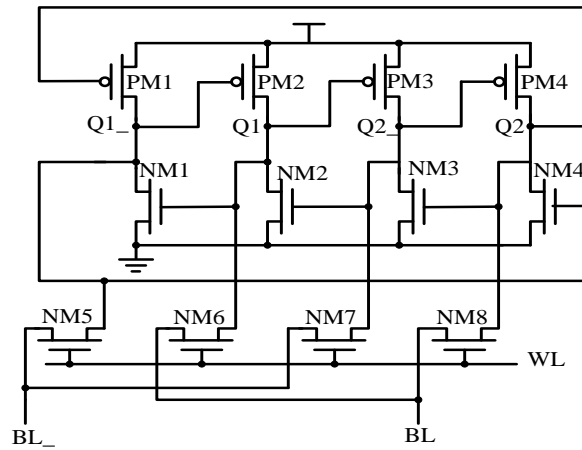

Figure 1. 12-tube DICE cell.

Compared with 6-tube element, 12-tube DICE structure has two-node feedback mechanism. During SEU effect, the destroyed node information is restored by the undestroyed node information, so as to achieve the anti-SEU effect. At the same time, the value of the corresponding node is determined only by two adjacent nodes, and there is no direct relationship between the non-adjacent nodes.

In Figure 1, when the DICE cell is logical "1", that is Q1 and Q2 are "1", and Q1_and Q2_are "0". Four storage nodes have two pairs of storage information, read and write operations is done through the transmission gate NM5-NM8. NM1, PM2, NM3 and PM4 are turned on and combine to form two latches: NM1 and PM2, NM3 and PM4. PM1, NM2, PM3 and NM4 are cut off and these four transistors constitute two bidirectional feedback circuits of NM2 and PM3, NM2 and PM3,so that they are separated from the latches. Eight single transistor inverters form two feedback loops, PMOS and NMOS. This feedback mechanism enables the correct node information to correct the wrong node information.

The 12-tube DICE structure has the same read-write reliability as the 6-Tube structure. In order to ensure the read-write reliability of the circuit, the width-length ratio of the DICE cell designed in this paper is shown in Table 1.

Table 1. Width-Length Ratio of DICE cells.

MOS tube	Width-Length Ratio
NM1~NM4	250n/190n
PM1~PM4	220n/190n
NM5~NM8	220n/290n

2.2. Radiation Simulation of DICE SRAM
Initial storage nodes store information as Q1 = Q2 = "1", Q1 = Q2 = "0", so NM1 and NM3 are turned on, NM2 and NM4 are cut off, so sensitive nodes are Q1 and Q2.

In Q1 node, we added double exponential current source, the simulation results are shown in Figure 2. Only Q1 flips during the current period. After the current is over, the positive feedback of the other three nodes restores Q1 to its original value. Q2 node is also a sensitive point. The effect of adding double exponential current source in Q2 node is the same as that of adding current source in Q1 node. It can be seen that the structure of DICE has the function of hardening single event reversal.
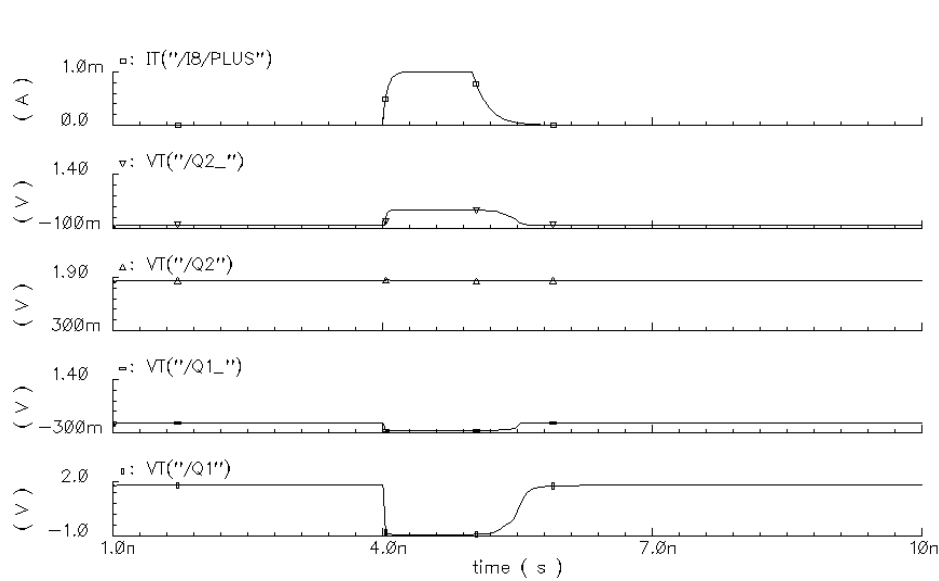
Figure 2. Simulation results of Q1 plus current source.

3. Create Timing Model with SiliconSmart

SiliconSmart is an EDA tool provided by Synopsys for creating timing model. SiliconSmart is the leading characterization and modeling solution tool focused on creating high quality cell libraries for today's advanced nanometer processes. It has been designed from the ground up to handle a full range of cells, from high-performance standard cells to advanced I/O cells implementing the latest communications protocols, and embedded memories, including RAMs, ROMs, and register files. It is the mainstream tool that can simulate SRAM with high speed and high precision and build the timing model. Using advanced algorithms, SiliconSmart can automatically identify the functions of standard cells and complex circuits and model them to generate effective simulation excitation sets for the timing model.

Using SiliconSmart to creating timing model of SRAM, the input data and the flow are shown in figure 3.
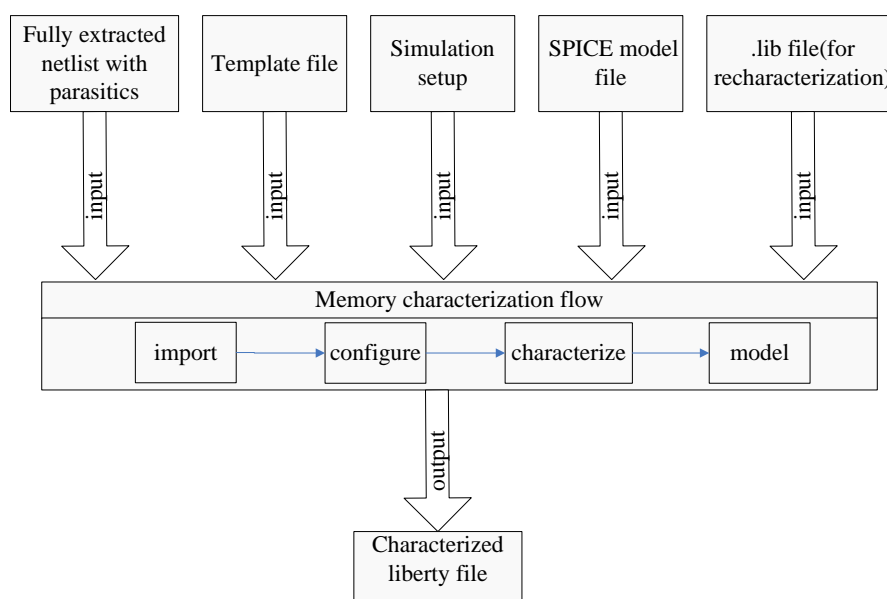

Figure 3. The input data and flow of SilconSmart.

3.1. Input Data

3.1.1. Spice model file. The spice model parameters of basic circuit components (such as NMOS and PMOS) provided by foundry manufacturer. It includes device models of various process and different temperature.

3.1.2. Spice netlist of DICE SRAM. The spice netlist with parasitic parameters is extracted from the layout of DICE SRAM. According to the accuracy of simulation, the parasitic parameter categories that need to be extracted are selected. The parasitic parameter accuracy of R+C is relatively low, but the simulation speed to create timing model is fast. The parasitic parameter netlist of R+C+CC has high accuracy, but the simulation speed is slow.

3.1.3. Template file. The template file contains basic information of the memory. It includes the type of memory, whether there is BIST mode, the function definition of each pin and so on. The structure and pin definition of a dual-port SRAM are shown in Figure 4, and the corresponding template file is shown in Figure 5.
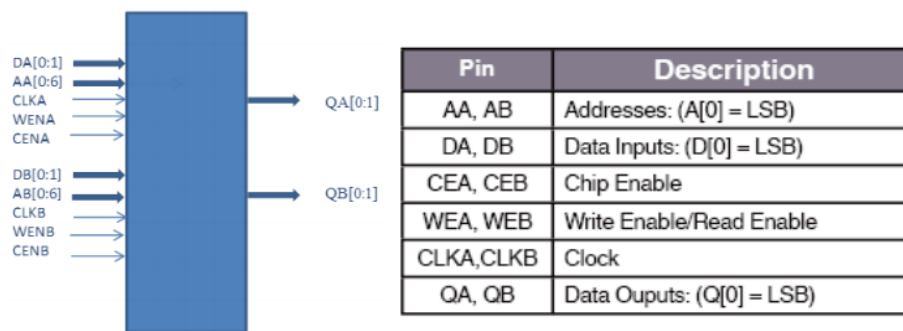

Figure 4. Structure and pin definition of dual-port SRAM.

```
set_memory_type multi_port_ram
set_memory_name dpram018
set_bypass_mode OFF
set_pipeline_mode OFF
set_writethrough_mode ON
set_bist_mode OFF
set_extramargin_adjustment_mode OFF

create_readwrite_port A
set_clock CLKA -active r -port A
set_address_bus AA -width 9 -port A
set_data_bus DA -width 16 -port A
set_read_enable WENA -active H -port A
set_write_enable WENA -active L -port A
set_output_enable  OENA  -active L -port   A
set_chip_enable CENA -active L -port A
set_data_output QA -width 16 -port A

create_readwrite_port B
set_clock CLKB -active r -port B
set_address_bus AB -width 9 -port B
set_data_bus DB -width 16 -port B
set_read_enable WENB -active H -port B
set_write_enable WENB -active L -port B
set_output_enable  OENB  -active L -port  B
set_chip_enable CENB -active L -port B
set_data_output QB -width 16 -port B
```

Figure 5. Template file example of SRAM.

3.1.4. Configure file. The file for configuration of simulation parameters include the selection and configuration of simulators and the definition of the header of timing model. In addition to selecting

and configuring the simulator, the slew and load lookup tables to creating timing model must be specified in the configure.tcl file for simulator.

The example of slew values is shown below (in s).

```
set numsteps_slew 5
set constraint_numsteps_slew 5
set smallest_slew 20e-12
set default_slew 200e-12
set largest_slew 3e-9
```

The example of the load value is shown below (in f).

```
set autorange_load 0
set numsteps_load 5
set smallest_load 20e-15
set default_load 400e-15
set largest_load 1e-12
```

3.2. The Flow of Creating Timing Model

The flow using SiliconSmart to create a timing model for DICE SRAM is shown in the figure 6.

```
set charpoint test_sram_1022
create $charpoint
set_log_file $charpoint/sis.log
set_location $charpoint
set cells {dpram018_512x16}
exec cp configure.tcl $charpoint/config/configure.tcl
exec ln -sf [pwd]/netlists/${cells}.pex.netlist.pex ${charpoint}/netlists/.
exec ln -sf [pwd]/netlists/${cells}.pex.netlist.${cells}.pxi ${charpoint}/netlists/.

set_config_opt auto_fix 0
set_config_opt char_engine_max_lifespan 12000
set_config_opt cdpl_task_max_lifespan 12000

set_config_opt -cell dpram018_512x16 -from OENB -reference CLKB state_partitions none

import -template ./dpram512x16_template.tcl -netlist_dir ./netlists -ext .pex.netlist $cells -overwrite

set_config_opt -cell dpram018_512x16 -from OENA -reference CLKA state_partitions none
set_config_opt -cell dpram018_512x16 -from OENB -reference CLKB state_partitions none

find_internal_nodes_for_constraint
configure -timing -power $cells
characterize $cells
model -output dpram018_512x16 $cells
```

Figure 6. The flow of creating timing for DICE SRAM.

3.2.1. Simulation option setting. Configure some tool parameters and timing modeling conditions.

3.2.2. Import. Import spice netlist to generate cell.inst file. SiliconSmart can identify the functions of the SRAM from the spice netlist and the template file.

3.2.3. Configure. Configure to build SPICE simulation files for each arc. Add input slew and output load to the SRAM spice netlist to generate simulation files.

3.2.4. Characterize. SiliconSmart calls HSPICE to simulate for each arc of the timing model, and determines whether the arc is successful simulated.

3.2.5. Model. Generates NLDM or CCS timing/power lib libraries as needed which can used in the design flow of SOC.

4. Conclusion
In this paper, the timing model of DICE SRAM is created by SiliconSmart, which is successfully used in the SOC project design flow. Functional tests show that the timing model can ensure the timing correctness of SOC.

Under the $LET_{TH} \geq 75 MeV \cdot cm^2/mg$ ion source condition, the SEU effect experiment is performed on the SOC with DICE SRAM. The result of the experiment is shown in table 2.

Table 2. A table with headings spanning two columns and containing notesa.

Sample Num.	Fluence rate(ions/cm^2.s)	Fluence(ions /cm^2)	Phenomenon description
1	1.05×10^4	10^7	Output flips 13 times
2	1.05×10^4	10^7	Output flips 15 times
3	1.05×10^4	10^7	Output flips 8 times

From the experimental result of table 2, all the three circuits do not flip more than 15 times, it indicates that these circuits is not sensitive for SEU effect at this LET_{TH} value. The SEU irradiation experiment shows that SRAM with DICE structure has the function of anti-SEU.

5. References
[1]   Lingyu Zhang, Yuming Jia, Lei Li. Radiation Resistance SRAM Design Based on DICE Structure. Microelectronics, 2011.4
[2]   Jing Shen, Haiwei Xue. Radiation hardening design of SRAM based on DICE structure. ELECTRONICS & PACKAGING, 2016.3
[3]   SiliconSmart ACE User Guide [M]. SYNOPSYS, 2014.09