Producing a Bidirectional ATPG Compliant Verilog-HDL Memory Model of SRAM

Dorian Ronga, Xhesila Xhafa, Eric Faehn, Patrick Girard, Thibault Vayssade, Arnaud Virazel

▶ **To cite this version:**

Dorian Ronga, Xhesila Xhafa, Eric Faehn, Patrick Girard, Thibault Vayssade, et al.. Producing a Bidirectional ATPG Compliant Verilog-HDL Memory Model of SRAM. DTTIS 2024 - IEEE International Conference on Design, Test & Technology of Integrated Systems, Oct 2024, Aix-en-Provence, France. In press. lirmm-04738159

HAL Id: lirmm-04738159
https://hal-lirmm.ccsd.cnrs.fr/lirmm-04738159v1

Submitted on 15 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Producing a Bidirectional ATPG Compliant Verilog-HDL Memory Model of SRAM

D. Ronga[1], X. Xhafa[1], E. Faehn[2], P. Girard[1], T. Vayssade[1], A. Virazel[1]

[1]LIRMM - University of Montpellier/CNRS
Montpellier, France
dronga, xxhafa, girard, vayssade, virazel@lirmm.fr

[2]STMicroelectronics
Crolles, France
eric.faehn@st.com

***Abstract*—Memory components are increasingly used in modern systems such as System-on-Chip (SoC). Moreover, they are becoming more complex as the technology node shrinks, which makes them more prone to manufacturing defects. A large part of memory testing is based on functional tests, using March algorithms to target Functional Fault Models (FFMs). New test methodologies are developed to anticipate the growing complexity of memory components, such as the Cell-Aware (CA) test methodology which has been recently introduced in the field of memory testing. However, in order to apply the CA methodology, which introduces a structural consideration of the circuit to be tested, an accurate digital model of the SRAM has to be designed. This paper proposes a methodology to produce a digital Verilog-HDL netlist of an SRAM, based on an initial analog model (SPICE netlist). The resulting digital model considers the bidirectional nature of the memory, and it is ATPG-compliant, allowing test patterns generation and fault simulation as well.**

***Index Terms*—Memory testing, Automatic Test Pattern Generation (ATPG), Memory model, Cell-Aware test, SRAM.**

I. Introduction

Modern components such as System-on-Chip (SoC) have increasing performances requirements for computation, which consequently requires more memory capacity, hence leading to a significant increase of the embedded memory size and density. These increasing integration densities of memories are achieved by approaching the limits of the technology, which makes them more prone to defects that can impair their correct functioning [1]. It is therefore important to test memories to reduce the defective-parts-per-million (DPPM) measure.

Common test methods for testing memories are based on the use of March algorithms [2] targeting Functional Fault Models (FFMs) [3] [4]. However, these methods could become insufficient in the future, considering the increasing complexity, and the emerging technologies of memories such as the Magnetic Random Access Memory (MRAM) [5] [6] or Resistive Random Access Memory (RRAM) [7].

Novel test solutions for digital circuits testing have emerged to anticipate this problem, such as the Cell-Aware (CA) test method, which has been initially proposed to reduce DPPM in digital circuits. The CA test method relies on a post-layout transistor-level approach [8] [9] [10], and it is based on a structural consideration of the circuit to be tested.

Recently, the CA test methodology have been developed in the field of memory testing, and more specifically, for SRAM testing [11]. To apply this novel test method to SRAMs, CA models of memory modules such as the core-cell has been developed. In order to apply such test methodology to memory components, an accurate digital model of the memory that closely respects the structure and the behavior of its analog model has been produced [11].

The produced digital model must respect the functionality of the analog model, in particular for read and write operations. Moreover, in order to generate test patterns and to run a fault simulation, the digital model must comply with digital test tools such as Automatic Test Pattern Generator (ATPG). The digital model must also satisfy ATPG specificities, which will be discussed and resolved in this paper.

This paper proposes an extension to the methodology presented in [11], to introduce bidirectionality in the digital modeling of the SRAM. The additional bidirectionality consideration allows all the considered modules of the memory model, such as the write driver and the sense amplifier, to be represented in a single digital model. The resulting digital memory model allows all of the considered modules to be processed by an ATPG and by a fault simulator as a single SRAM digital model. This extended representation of bidirectionality in the modeling strategy presented in [11] introduces a more complete representation of the SRAM model in the digital test environment. Using a complete memory model helps to improve the representation of the structural interactions between the SRAM modules, which helps to maintain a level of representation that remains as close as possible to the real component under test.

The paper is organised as follows. Section 2 presents the flow to produce a digital bidirectional ATPG-compatible Verilog-HDL netlist from a SPICE netlist. Section 3 presents the reduced analog model of the SRAM that has been designed as a case study. Section 4 provides details on the conversion from the analog netlist (SPICE) to a first digital netlist (Verilog-HDL). Section 5 discusses on specific modifications required to enable its compatibility with digital test tools, and presents the digital netlist's required modifications. Section 6 concludes the paper.

II. Proposed methodology

The proposed flow, depicted in Fig. 1., is based on [11], and proposes an extension to introduce bidirectionality in the digital model of the SRAM. This modeling methodology allows to load and process all of the considered memory modules in the digital test environment with a single SRAM digital model. This is achieved without dividing the memory model into multiple unidirectional sub-models.

The first part of the flow given in Fig. 1. consists in producing an analog model of the SRAM, which will serve as a case study. The produced SPICE netlist (i.e., Analog SPICE netlist) is organised hierarchically, as shown in Fig. 2. (a), so that the produced digital netlists retain this hierarchical construction. The analog model of the SRAM is simulated using an analog simulation tool (i.e., Analog simulation), to verify the correctness of the read and write operations.

The analog netlist is then converted to a Verilog-HDL digital equivalent netlist using a functional verification tool (i.e., ESP). The obtained digital netlist is then adapted to follow precisely the construction given by Fig. 2. (a), and can be simulated using any digital simulation tool (i.e., Digital Verilog-HDL netlist, and Digital simulation).

In the last step of the flow, the digital Verilog-HDL netlist is modified to become compatible with digital test tools (i.e., ATPG-aware modifications), while preserving the intrinsic bidirectional behavior of its modules. This is achieved while maintaining the SRAM top model's ports as input and output only. The obtained ATPG-aware digital Verilog-HDL netlist can also be simulated with any digital simulation tool by applying minor drive strength adaptations. The ATPG-aware digital Verilog-HDL netlist can be processed by an ATPG to generate test patterns, and to run any fault simulations.

III. Design of the analog model of an SRAM case study

To evaluate the proposed methodology, a reduced analog (SPICE) memory model of an SRAM that includes the necessary modules for its correct functioning has been designed. The case study model is depicted in Fig. 2., with a high-level, and a low-level representation, to identify the key modules of the SRAM. The model consists of the elementary modules that represent an SRAM: one core-cell (in 6-Transistors (6T) topology) to carry the information, one pre-charge circuit to set-up and equalize the memory bitlines for read and write operations, one sense amplifier to read the information from the core-cell, and one write driver to force a value in the core-cell.

In order to maximize the simplification of the analog memory model, only one core-cell is considered, allowing the memory to operate without any row and column decoders. Since the row decoder is a combinational circuit, it is already compliant with an ATPG, and therefore its removal is not a real concern for the purpose of the proposed methodology. Using one core-cell SRAM model also allows to remove the column decoder, which simplifies the model.

The reduced analog SRAM model is simulated using analog simulation tools. Read and write operations are performed to ensure its behavioral correctness. The results of the analog simulation are depicted in the waveform of Fig. 3. In order to simulate the memory model to verify its correct behavior, it is initialised with a high voltage at the core-cell's S node, representing a stored '1' value.

The first operation is *Write0*. The pre-charge is disabled (PRCH at VDD), and the word line is asserted (WL at VDD), connecting the core-cell to the bitlines. The write driver is also activated, and drives BL to low voltage while maintaining BLB to a high voltage (WR and W0 at VDD). The sizing of the transistors that composes the core-cell allows the write driver, through bitlines, to cause the inversion of the core-cell's inverters loop values. The resulting effect of the *Write0* operation is S node at low voltage, and SB node at high voltage, representing a stored '0' value in the core-cell. At the end of the *Write0* operation, the write driver is disconnected (WR at GROUND) and the core-cell is isolated from the bitlines (WL at GROUND). The pre-charge is re-enabled (PRCH at GROUND), to set-up bitlines at VDD to prepare for the next operation.

The second operation consists of a *Read* operation of the core-cell's value, which now contains a representation of '0' value. The pre-charge is disabled, and the word line is asserted, connecting the core-cell to the bitlines. Bitlines voltages, which are floating at VDD, starts to discharge on the side which is connected to the low voltage node of the core-cell (S in that case). After a certain time, a sufficient differential voltage has developed between bitlines, and the sense amplifier is activated (SA_EN at VDD). The differential voltage from bitlines is locally amplified by the sense amplifier, resulting in a stable SA and SAB output voltages, which represents the core-cell's value. At the end of the *Read* operation, the sense amplifier is disabled (SA_EN at GROUND), and the core-cell is disconnected from the bitlines. The pre-charge is then re-enabled, to set-up bitlines at VDD to prepare for the next operation.

The third operation is *Write1*, which operates similarly to *Write0*, except that the W1 signal is set to VDD in order to force a high voltage in the core-cell's S node, which represents a stored '1' value.

The fourth operation is *Read*. It behaves similarly to the previous *Read* operation. However, the core-cell's S node voltage is now at a high voltage due to the previous *Write1* operation. As a result, the BLB voltage drops when the core-cell is connected to the bitlines, while the BL voltage remains stable. The *Read* operation outputs the expected value on the SA node, which now represents the value '1', contained in the S node of the core-cell.

The simulation results of these four operations demonstrate that the analog memory model behaves as it is intended to.

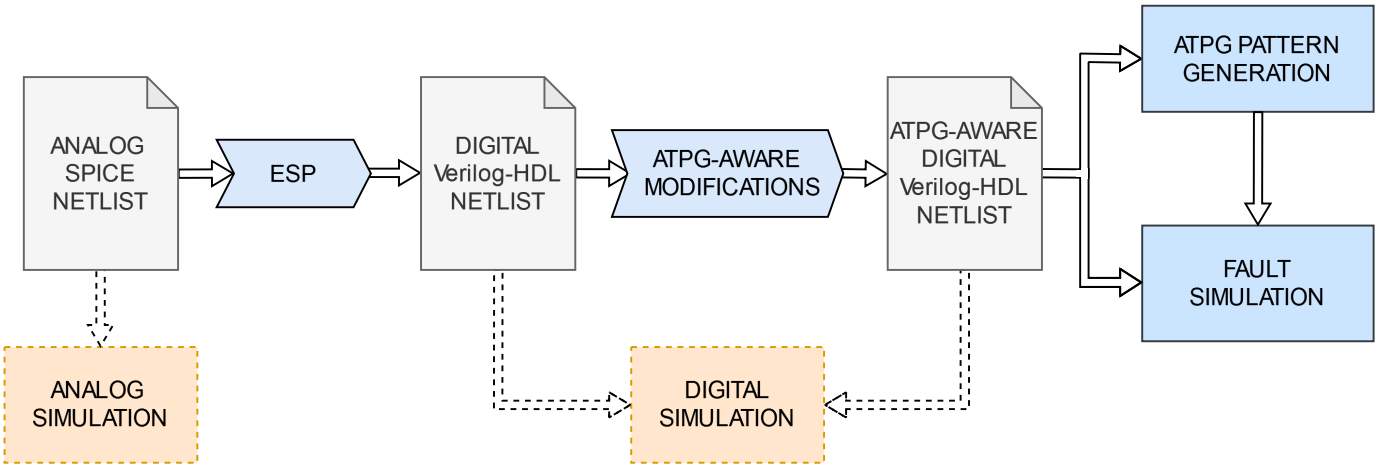

Fig. 1. Flow for producing a digital test tool compatible Verilog-HDL netlist from a SPICE netlist.

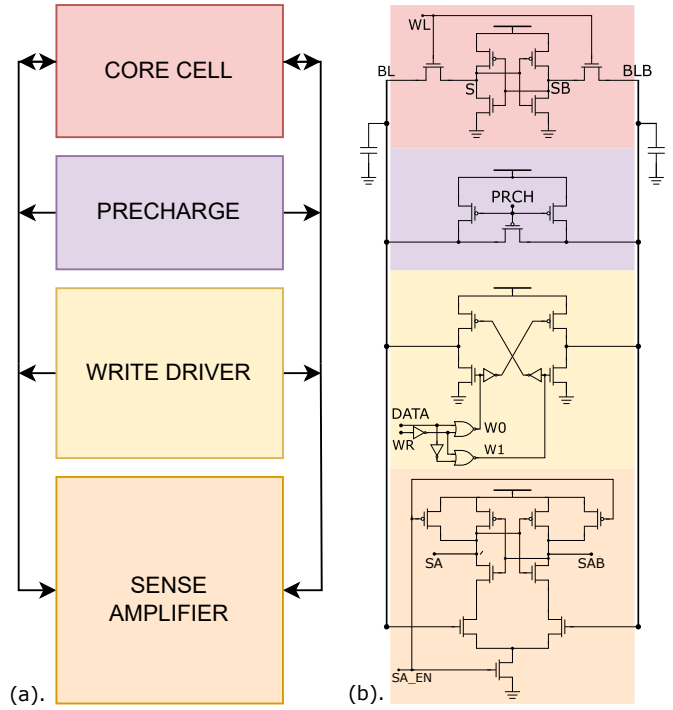

Fig. 2. (a) High level schematic representation of the reduced SRAM model. (b) Transistor and gate level representation of the analog reduced SRAM model.

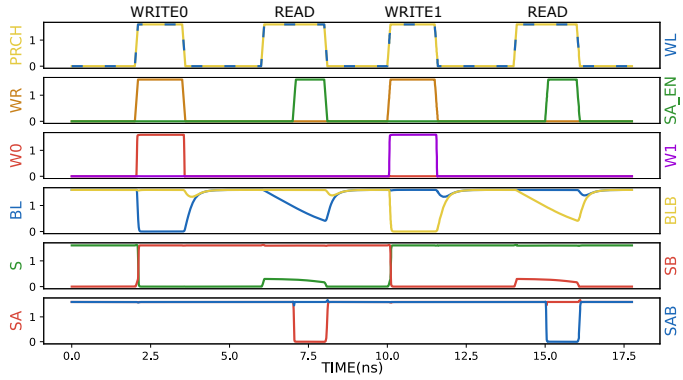

Fig. 3. Analog simulation of the reduced SPICE model of SRAM.

IV. Producing a Digital Netlist from the Analog Memory Model

The hierarchical SPICE transistor-level description of the memory model can be converted to a first hierarchical Verilog-HDL netlist. The conversion is processed using a functional verification tool named ESP from Synopsys [12], which produces an equivalent transistor-level, module-based, digital Verilog-HDL netlist, for which digital structure related information is retained.

Several modifications are operated on the resulting digital netlist in order to match with the hierarchical inter-module interactions given in Fig. 2. (a). (represented by arrows). These modifications are detailed in the following sub-sections and are related to modules ports direction, primitive drive strength, digital sense amplifier simplification, and charge storage representation.

A. Modules ports direction

Some of the performed modifications are related to modules' ports direction, for which unidirectional configuration (i.e., input or output) should be set where it respects Fig. 2. (a) construction. Other module ports, which are bidirectional by nature, such as the core-cell, can be set to inout type. Moreover, reducing the amount of bidirectional paths in the memory model can simplify the resulting representation of the model that is considered by the ATPG.

B. Primitive drive strength

In addition with ports direction modifications, digital primitive's drive strengths are also adjusted using the resistive versions of MOS primitives, in order to represent the intra-module and inter-modules interactions of the analog model. The usage of a resistive MOS primitive reduces the drive strength of the signal that passes through it [13]. Drive

strength modifications are performed in the core-cell module, by replacing the inverters MOSs primitives by their resistive version, in order to obtain a complete functional equivalence in simulation with the analog model for read and write operations (see Fig. 3.).

C. Digital sense amplifier simplification

In the analog domain, charges and discharges of bitlines are continuous in time. This property enables the analog sense amplifier to capture and amplify a differential voltage that develops between the bitlines. However, considering the digital domain imposes the use of discrete values such as '1' and '0' and also a discrete time consideration to represent the final stable states of the values. The amplification role of the digital sense amplifier module is simplified with a set of two *bufif1* primitives, as illustrated in Fig. 4. These primitives can be used to modulate the drive strength of their input signal [13] and to transmit the bitlines' digital values to the output ports of the digital sense amplifier module, depending on the input control value.

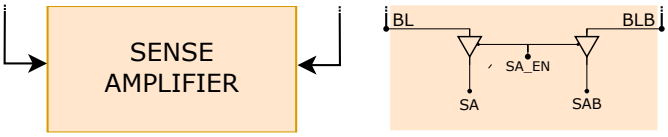

Fig. 4. Simplified digital sense amplifier.

D. Bitlines charge strength representation

In addition, in the digital domain, bitlines may be represented using *trireg* net instead of *wire* net. The *trireg* net allows to represent the charge storage effect [13] of the analog bitlines, which can prevent a floating 'Z' value if they are left undriven. However, a low charge strength should be affected to the *trireg* net (such as "small") in order to do not disturb the normal SRAM model operating.

The obtained digital model of the SRAM can then be simulated using a digital simulation tool, to verify its functional equivalence with the analog model.

V. Modifications to design an ATPG-aware Verilog-HDL netlist

At this step of the methodology, *(R)tran(IF)(0/1)* primitives are used in the digital domain to preserve the bidirectional nature of specific SRAM modules, such as the core-cell's access transistors. However, these digital primitives are not defined for the considered ATPG. In order to preserve the bidirectional nature of the SRAM modules, and to produce an ATPG compatible netlist, substitution modules are introduced to replace ATPG undefined primitives such as *(R)tran(IF)(0/1)*. In this section, the substitution modules that replace the mentioned primitives are presented, and are then integrated to the digital netlist at specific locations of the SRAM model.

A. Substitution modules for Rtranif(0/1) primitives

The substitution module interface has two inout ports (inout1, inout2), and one input port (control). Moreover, it uses a set of two ATPG-defined digital resistive transistors to represent the *rtranif(0/1)* primitives. The first module uses two digital resistive *RpMOS* to represent the *rtranif0* primitive as depicted in Fig. 5. (a). The second substitution module uses a set of two digital resistive *RnMOS* transistor to represent the *rtranif1* primitive (see Fig. 5. (b)). For both modules, the two digital resistive MOSs are controlled by a shared control input port, and each of the transistors are mapped in a complementary unidirectional manner, connecting together the two inout ports of their respective substitution module.

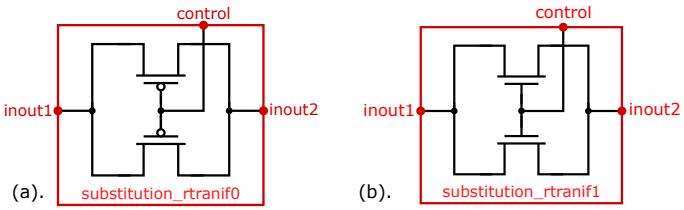

Fig. 5. Modules to substitute the Verilog-HDL (a) *rtranif0* and (b) *rtranif1* primitives.

These substitution modules, are evaluated in simulation using a Verilog-HDL simulator, to ensure that their behavior matches with the primitives they replace. The digital test-bench consists of an exhaustive test sequence, applied to every input and inout ports of the modules and the primitives. A combination of all the ['0', '1', 'X', 'Z'] digital discrete values is applied. The 64 resulting combinations of values for the two inout ports of the substitution module are compared with the resulting values of the two inout ports of the primitive.

The modules that substitutes *rtranif0* and *rtranif1* primitives produces similar results with their respective primitive, except for three cases for each substitution modules. These cases return different values which are illustrated in Table I and Table II, which shares the same table structure. The first column of the tables gives the name of the concerned primitive, and the case number for which the primitive and the substitution module have returned different values. The second column gives the names of the ports that are concerned by the corresponding cases number. The third column gives the values that are applied by the test-bench for the given case. The fourth column gives the values read from the inout ports of the primitive. The last column gives the values read from the inout ports of the substitution module.

For simulations of both substitution modules, "Case #1" and "Case #4" correspond with their respective active state (control='0' for *rtranif0*, and control='1' for *rtranif1*), when the test-bench inout values are ['Z', 'Z']. Simulation differences denoted by "Case #2" and "Case #5" are produced by similar test-bench values for both of the substitution modules, when the control value is 'X'. As well for "Case #3" and "Case #6" which are produced by the same test-bench values, when the control value is 'Z'. For each of the six different cases, the

Rtranif0	Port name	Testbench values	Primitive values	Substitution module values
Case #1	control	0	0	0
	inout1	Z	Z	*
	inout2	Z	Z	*
Case #2	control	X	X	X
	inout1	Z	Z	*
	inout2	Z	Z	*
Case #3	control	Z	Z	Z
	inout1	Z	Z	*
	inout2	Z	Z	*

TABLE I
Rtranif0 SUBSTITUTION MODULE AND PRIMITIVE VALUES DIFFERENCES IN DIGITAL SIMULATION.

Rtranif1	Port name	Testbench values	Primitive values	Substitution module values
Case #4	control	1	1	1
	inout1	Z	Z	*
	inout2	Z	Z	*
Case #5	control	X	X	X
	inout1	Z	Z	*
	inout2	Z	Z	*
Case #6	control	Z	Z	Z
	inout1	Z	Z	*
	inout2	Z	Z	*

TABLE II
Rtranif1 SUBSTITUTION MODULE AND PRIMITIVE VALUES DIFFERENCES IN DIGITAL SIMULATION.

primitive's inout ports values that can be read are ['Z', 'Z'], which are supposed to be the correct values. However the substitution modules' inout ports values that can be read may differ with the primitive's values for the given cases, based on the previous values that were driven before ['Z', 'Z'] is set. These possible output values are denoted by '*' in both tables. The differences of values must be considered when using these substitution modules in the SRAM model.

B. Pre-charge equalizer substitution

Consider the analog pre-charge module, for which two *pMOS* transistors are used to pull-up the bitlines, and one *pMOS* is used to equalize both bitlines, (see Fig. 2. (b)). The equivalent digital model should maintain the bidirectional nature of the equalizer MOS, using an unidirectional primitive such as *pMOS* can break a structural part of the equalization. The first obtained digital netlist (i.e., Fig. 1., Digital Verilog-HDL netlist) already solves this issue by using a *tranif0* primitive. However, even if using the *tranif0* primitive complies with the simulated analog results given in Fig. 3., the *tranif0* is not a defined primitive in the scope of the considered ATPG tool, which prevents the generation of test patterns. In order to allow the digital model to be processed by an ATPG, the *tranif0* primitive is replaced by the *rtranif0* substitution module (see Fig. 6.).

The *rtranif0* substitution module is used in the pre-charge module to replace the equalizer transistor. This modification is performed considering the results given in Table I which discuss the value deviation of the substitution module for three specific cases. However, considering the normal operating cases of the SRAM model, a digital situation where both bitlines values are at floating 'Z' is not happening. Considering this, the mentioned three cases are not interfering

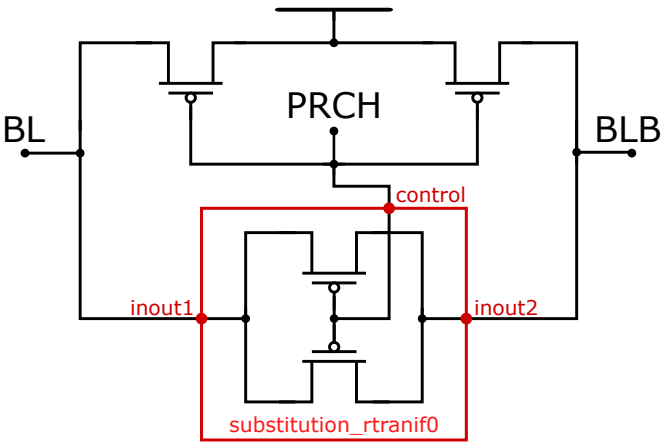

Fig. 6. Pre-charge module using the *rtranif0* substitution module.

with the normal operating SRAM model (i.e., Fig. 3.), and the *rtranif0* substitution module is used.

C. Core-cell access transistors substitution

The memory core-cell's access transistors transmit information into the core-cell during write operations, and also route the information out of the core-cell during read operations. They can therefore be considered as bidirectional elements of the model. The first obtained digital netlist (i.e., Fig. 1., Digital Verilog-HDL netlist) represented these access transistors with *tranif1* primitives. However, in order to comply with digital test tools, and to allow the SRAM model to be processed, the *tranif1* primitives are replaced by *rtranif1* substitution modules (see Fig. 7.), which preserves their bidirectional structure.

The *rtranif1* substitution modules are used in the core-cell module to replace both access transistors. These structural modifications are performed considering the results given by Table II, which presents the value deviation of the *rtranif1* substitution module, for three specific cases. However, considering the substitution modules to be connected to one bitline and to one core-cell node, situations where both their values are floating 'Z' is not encountered in the normal operating SRAM model (i.e., Fig. 3.). The *rtranif1* substitution modules are then used to replace both access transistors of the core-cell module.

D. Strength values adjustment

Replacing *tranif(0/1)* primitives with *rtranif(0/1)* substitution modules changes the drive strength configuration of the digital model by applying a drive strength reduction on the node where the substitution module is placed. In order to restore the drive strength balance between the SRAM modules, drive strength adaptations are performed in the core-cell and in the pre-charge modules, as follow :

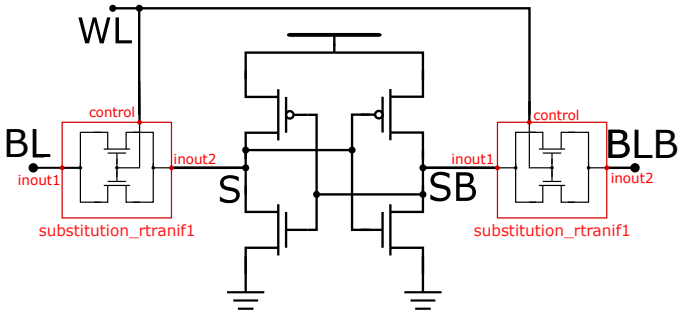

Fig. 7. Core-cell module using the *rtranif1* substitution modules.

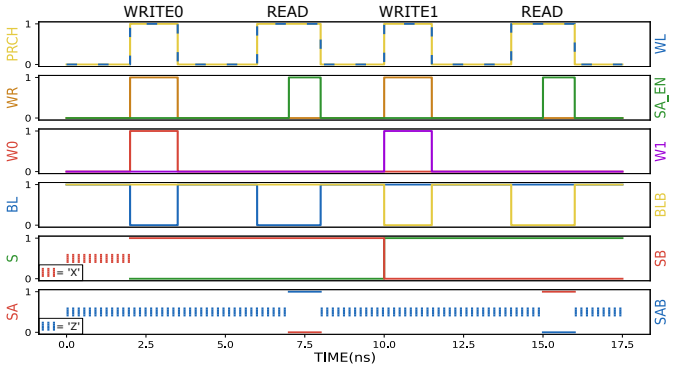

Fig. 8. Digital simulation of the reduced ATPG-aware Verilog-HDL model of SRAM.

1) Drive strength adjustment in the pre-charge module: The pre-charge equalizer pMOS transistor, which has been replaced by the *rtranif0* substitution module, introduces a drive strength reduction in the pre-charge module. In order to counterbalance this effect, the two pre-charge's pMOSs pull-up transistors must be replaced by their resistive equivalent. This modification restore the strength balance within the pre-charge module, so that the primitives and the equalizer interactions are correctly represented.

2) Drive strength adjustment in the core-cell module: Using *rtranif1* substitution modules in the core-cell module to replace the *tranif1* primitives introduces a drive strength reduction in the core-cell module. In the same way as for the pre-charge module, the inverter's drive strength of the core-cell are reduced by an additional level, in order to achieve a double drive strength reduction in the inverters loop. As with the pre-charge module, these modifications restore the strength balance within the core-cell module, so that the inverters and the access transistors interactions are correctly represented.

E. Results of the ATPG-aware digital netlist of SRAM

This design methodology, using primitive substitutions, coupled with drive strength adaptations, allows the ATPG-aware digital SRAM model (that embeds a write driver and a sense amplifier) to be simulated with a digital simulation tool. The obtained results (see Fig. 8.) from the digital simulation matches with the obtained results of the analog simulation (see Fig. 3.), considering the differences between the analog and the digital domains regarding the representation of time and values, discussed in IV. (C.).

VI. Conclusion

This paper has presented a methodology to produce a bidirectional ATPG-compatible digital memory model, based on an initial SPICE model of an SRAM. The resulting digital model, which uses primitives substitutions and drive strength adaptations, can be simulated to ensure its functional equivalence with the analog model. The resulting digital model can also be build in the digital test environment, in order to be processed by an ATPG, and to run fault simulations.

An ideal application of this proposed methodology would be to produce digital bidirectional ATPG-compatible models for other components, or other types of memory such as Magnetic Random Access Memories (MRAMs), in order to apply the CA methodology to emerging technologies.

Acknowledgement

This work has been funded by the French National Research Agency (ANR) under the framework of the ANR-22-CE24-0014 QUALMEM (Quality Assurance of Advanced and Emerging Memory Technologies by Using Machine Learning) project.

References

[1] S. Borkar *et al.*, "Microarchitecture and design challenges for gigascale integration," in *MICRO*, vol. 37, pp. 3–3, 2004.
[2] A. J. Van de Goor, *Testing semiconductor memories: theory and practice*. John Wiley & Sons, Inc., 1991.
[3] A. J. Van de Goor and Z. Al-Ars, "Functional memory faults: a formal notation and a taxonomy," in *Proceedings 18th IEEE VLSI test symposium*, pp. 281–289, IEEE, 2000.
[4] A. Bosio, L. Dilillo, P. Girard, S. Pravossoudovitch, and A. Virazel, *Advanced test methods for SRAMs: effective solutions for dynamic fault detection in nanoscaled technologies*. Springer Science & Business Media, 2009.
[5] K. Lee *et al.*, "22-nm fd-soi embedded mram technology for low-power automotive-grade-l mcu applications," in *2018 IEEE International Electron Devices Meeting (IEDM)*, pp. 27.1.1–27.1.4, 2018.
[6] Y. Song *et al.*, "Demonstration of highly manufacturable stt-mram embedded in 28nm logic," in *2018 IEEE International Electron Devices Meeting (IEDM)*, pp. 18.2.1–18.2.4, 2018.
[7] H.-S. P. Wong *et al.*, "Metal–oxide rram," *Proceedings of the IEEE*, vol. 100, no. 6, pp. 1951–1970, 2012.
[8] F. Hapke *et al.*, "Cell-aware test," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1396–1409, 2014.
[9] F. Hapke and J. Schloeffel, "Introduction to the defect-oriented cell-aware test methodology for significant reduction of dppm rates," in *2012 17th IEEE European Test Symposium (ETS)*, pp. 1–6, 2012.
[10] Z. Gao *et al.*, "Application of cell-aware test on an advanced 3nm cmos technology library," in *2019 IEEE International Test Conference (ITC)*, pp. 1–6, 2019.
[11] X. Xhafa, A. Ladhar, E. Faehn, L. Anghel, G. Di Pendina, P. Girard, and A. Virazel, "On using cell-aware methodology for sram bit cell testing," in *2023 IEEE European Test Symposium (ETS)*, pp. 1–4, 2023.
[12] Synopsys, "Esp user manual," 2021.
[13] "Ieee standard for systemverilog–unified hardware design, specification, and verification language," *IEEE Std 1800-2023 (Revision of IEEE Std 1800-2017)*, pp. 1–1354, 2024.