Microelectronics Reliability 120 (2021) 114110

Contents lists available at ScienceDirect

Microelectronics Reliability

journal homepage: www.elsevier.com/locate/microrel

A fast simulation method for analysis of SEE in VLSI

Yufan Lu [a], Xin Chen [b], Xiaojun Zhai [a,*], Sangeet Saha [a], Shoaib Ehsan [a], Jinya Su [a], Klaus McDonald-Maier [a]

[a] School of Computer Science and Electronic Engineering, University of Essex Colchester, UK
[b] College of Electronic and information Engineering, Nanjing University of Aeronautics and Astronautics Nanjing, China

ARTICLE INFO

Keywords:
Single event effect
Fault injection
SEE models
SEE mitigation
HDL simulation
VLSI

ABSTRACT

The transistor simulation tools (e.g. TCAD and SPICE) are widely used to simulate single event effects (SEE) in industry. However, due to the variances of the physical parameters in practical design, e.g. the nature of the particle, linear energy transfer and circuit characteristics would have a large impacts on the final simulation accuracy, which will significantly increase the complexity and cost in the workflow of the transistor level simulation for large scale circuits. Therefore, a new SEE simulation scheme is proposed to offer a fast and cost-efficient method to evaluate and compare the performance of large scale circuits in the effects of radiation particles. In this work, we have combined both the advantages of transistor and hardware description language (HDL) simulations, and proposed accurate SEE digital error models for high-speed error analysis in the large scale circuits. The experimental results show that the proposed scheme is able to handle SEE simulations for more than 40 different circuits with the sizes varied from 100 transistors to 100 k transistors.

1. Introduction

Radiation exists throughout the solar system, and includes Solar flares, Solar wind, Galactic cosmic radiation and radiation emitting from nuclear reactions [1]. There are significant concerns regarding the safety of the crucial hardware circuits of the electronic systems in radiation environments (e.g., nuclear power plants or outer space). In radiation environments, there are a large number of particles with high energy. When charged particles travel through integrated circuits, semiconductor material will be ionised, which may disable the P—N junctions and cause errors. The transient and permanent effects caused by single particles are called singe event effects (SEE).

SEE can be divided into a number of effects, which include 1) Single Event Transient (SET), 2) Multiple-Bit Upset (MBU), 3) Single Event Functional interrupt (SEFI), 4) Single Event Latch-up (SEL), 5) Single Event Burnout (SEB), 6) Single Event Gate Rupture (SEGR) and 7) Single Event Upset (SEU). From the above list, SET and SEU are the most common effects which may cause soft errors (recoverable or transient errors) [2–5]. When embedded systems are operating in the environments with high radiation intensity, errors can occur frequently and may cause system failure. For example, in the GEO orbit environment, the SEU rate of the BRAM is 1.06×10^{-7} and the error rate of the register is 1.13×10^{-5} [6]. Therefore, it is important to evaluate the circuits and mitigate SEE in the radiation environments. This evaluation can be conducted either using real world experiments or the SEE simulation or both.

However, strategies to evaluate the effects of SETs and SEUs on the integrated circuits of different designs still have certain limitations. The real world radiation experiments (e.g. outer space experiments or nuclear radiation facility experiments) demand sophisticated mechanical setup and they are expensive as well. Indeed they can provide "high level" results like "error rates" and "sensitivity to radiation" of different chips, but the detailed radiation effects on various circuit modules are hard to find, due to the complexity of the integrated circuits.

Existing transistor simulation tools (e.g., TCAD, or SPICE) concentrate on the small devices with few transistors [7,8]. Software simulations consider physical parameters (e.g., capacitance, resistance, current, voltage, 3D structure) and thus, they can provide accurate "low level" simulation results including the duration of the pulses, the voltage changes and the probability of the bit-flips. However, the cost of computation for current or voltage increases along with the size of the circuits. Hence, a range of simplified methods have been proposed to conduct simulations. For instance, In [9], the authors presented a method using the simplified SPICE model in simulations to lower the

* Corresponding author.
E-mail addresses: yl19888@essex.ac.uk (Y. Lu), xin_chen@nuaa.edu.cn (X. Chen), xzhai@essex.ac.uk (X. Zhai), sangeet.saha@essex.ac.uk (S. Saha), sehsan@essex.ac.uk (S. Ehsan), j.su@essex.ac.uk (J. Su), kdm@essex.ac.uk (K. McDonald-Maier).

https://doi.org/10.1016/j.microrel.2021.114110
Received 17 August 2020; Received in revised form 19 February 2021; Accepted 21 March 2021
Available online 30 March 2021

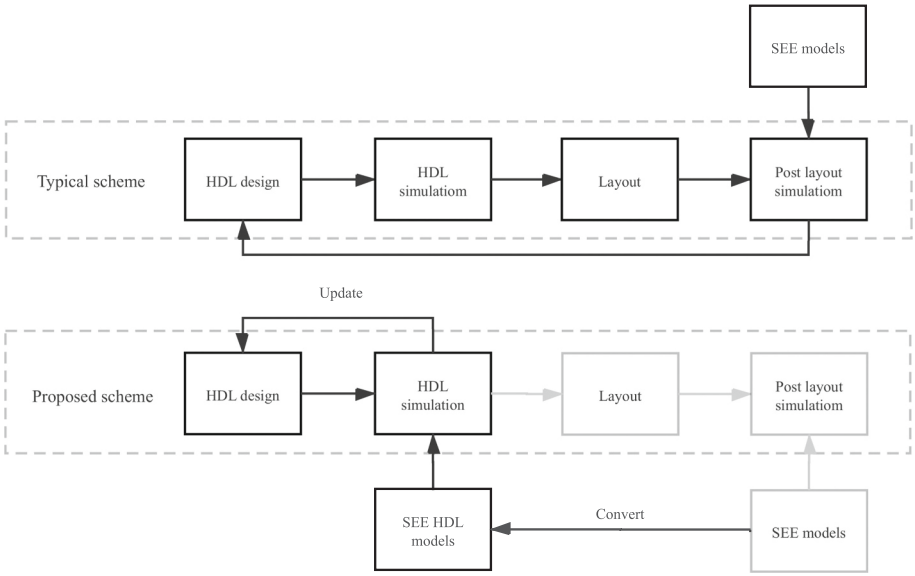

Fig. 1. The comparison between a typical SEE evaluation scheme and the proposed scheme.

cost of computation. Hubert et al. [10] presented a circuits-level layout-aware modeling method to avoid the complicated TCAD simulations. However, since these simulations are still based on the current and voltage, the simulation time is huge for larger circuit [11,12], and the comparison between different designs are also difficult.

System level simulation is another method to rapidly evaluate the SEE mitigation performance of the circuits. It carries out the error injections in the data stream or memory units based on the SEU rates [13,14]. However, this simulation approach lacks the analysis of the specific hardware modules in the design, and thus, cannot provide an accurate and detailed SEE simulation result.

In order to resolve these issues, in this paper, we propose a new SEE simulation scheme to offer a fast and cost-efficient method to evaluate and compare the performance of large scale circuits. The scheme consists of the following features: 1) building the SEE behaviour models based on the SPICE or TCAD, 2) generate the HDL netlists and injection scripts based on the HDL designs and 3) apply the SEE behaviour models in the HDL simulations to analyse and compare the performance of the circuit designs. 4) modify the hardware designs according to the results and repeat the simulation processes. Validation of the proposed scheme has been conducted using 180 nm logic gate library from the "Semiconductor Manufacturing International Corporation (SMIC)" [15] to build a set of SEE models for the simulation. The baseline circuits that we used in this paper, are a series of commonly used circuit designs from the ISCAS89 benchmark [16]. Experimental results exhibit that the proposed scheme is able to handle SEE simulations for more than 40 different circuits with the sizes varying from 100 transistors to 100 k transistors. Additionally, using low-level SEE behaviour models, the proposed simulation scheme is able to provide details of the error propagation and vulnerable logic design in the HDL simulation. As shown in the Fig. 1, compared to the typical SEE evaluation scheme, the proposed scheme provides a rapid solution to analyse the vulnerable modules in the logic designs before post layout simulations.

The main contributions of this article can be summarised as follows:

1) The proposed SEE simulation scheme provides a rapid, convenient and universal comparison method to evaluate the designs of circuits in context of SEEs. Due to various manufacturing processes, physical layouts and radiation environments, the simulation tools and simulation environments may also vary in different SEE research. It is difficult to repeat or compare those experiments directly. The proposed SEE models can be easily integrated into current circuit design work flow without significant cost. It can create a universal simulation environment to provide a quick analysis of the relative performance, which can significantly reduce the total simulation time for the time consuming back-end simulation.

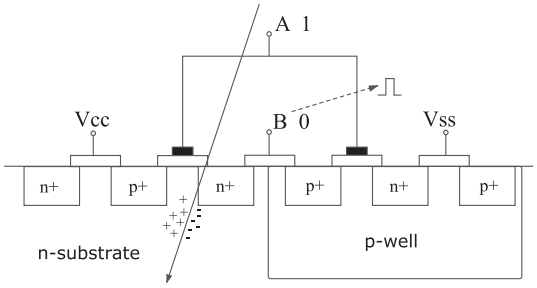

Fig. 2. The PMOS in the inverter is struck by the particle.

2) The proposed scheme introduces a range of new SEE behaviour models. Based on the transistor level simulations, the SEE behaviour models are firstly embedded into a range of digital functions in the HDL described circuits, the transient currents and voltages are then converted into the digital pulses and bit-flips. Unlike the typical transistor level based SEE behaviour models that fully rely on low-level currents and voltages simulation inputs, the proposed SEE models use only high level digital functions in HDL, therefore it can offer lightweight and fast simulations for large scale circuits.

3) The proposed scheme can offer a high level of flexibility in the design. All parts in this scheme including gate components, SPICE simulation and HDL simulation are decoupled. The gate components can be modified to adapt to different manufacturing processes, and the SEE spice model can be also modified to adapt to different radiation environments, as required. In this way, the proposed scheme can make full use of existing models to build simulation environments and be adapted for various requirements.

The remainder of this paper is organised as follows: Section 2 presents the related SEE simulation methods and their issues. Section 3 presents the design of the scheme including the strategies for basic logic components and large scale circuits. Section 4 presents the design of the SEE models. Section 5 presents the implementation of simulation for the large scale circuits. Section 6 presents the results and the analysis of the simulations. Finally, Section 7 concludes this paper.

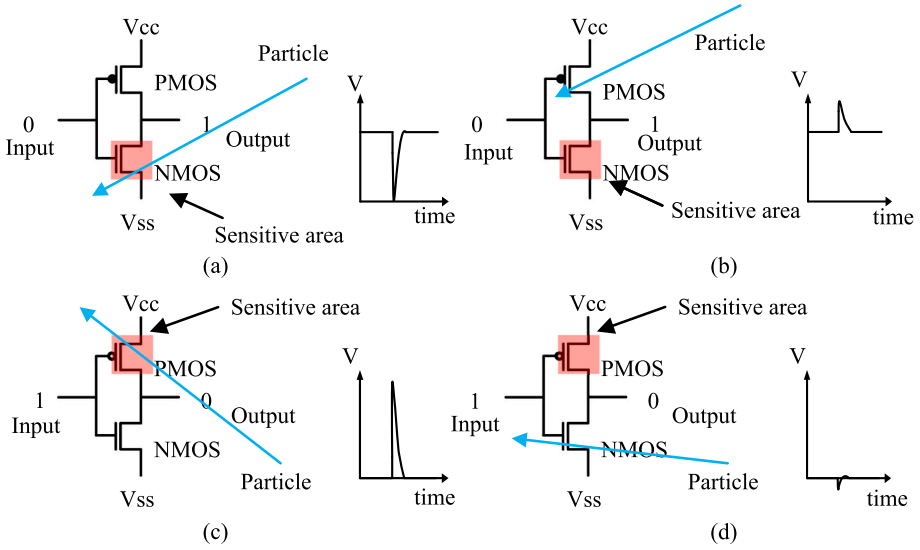

Fig. 3. The different possibilities of SEEs in the inverter circuit: (a) the input is "0" and the NMOS is struck, (b) the input is "0" and the PMOS is struck, (c) the input is "1" and the PMOS is struck and (d) the input is "1" and NMOS is struck.

2. Background

When circuits are struck by particles, the effects of SEE might be different according to affected areas and circuit states. The process of SEE in an inverter circuit is illustrated in Fig. 2. This inverter is made of one PMOS and one NMOS transistor. As can be observed in Fig. 2, current state of A (i.e. the input of the inverter) is "1". The state of B (i.e. the output of the inverter) is "0". In this state, the PMOS is closed but the NMOS is open. Therefore, there is a voltage difference between the Vcc and B. If one charged particle strikes the area between Vcc and B, the semiconductor material could be ionised, which will result in a large number of free electron-hole pairs. Due to this voltage difference between Vcc and B, free electrons will move towards the direction of Vcc and holes will move towards B. The moving electrons and holes will finally form a transient current at port B, which will cause a current pulse.

As per the energy released by the particles and the physical parameters of the semiconductor material, the magnitude of the current and voltage of the pulses might be different. In digital circuits, the magnitude of the current and voltage of the pulses will affect the width of the digital impulses. Therefore, in order to achieve highly accurate results, the transistor simulation tools (e.g. TCAD and SPICE) simulate the process of SEE with many physical parameters. Hence, demands massive calculation power for even single MOS component. However, the integrated circuits normally contain a large number of MOSFETs. It will be unrealistically expensive to do fault injections in large scale circuits by TCAD or SPICE in terms of time consumption.

A solution for simulation in large scale circuits is to utilise fault injection in HDL code [17]. In the HDL simulations, SETs and SEUs are considered as a digital pulses and bit-flips hence, there will be no complex calculation with physical parameters. SEEs are injected into the circuits by setting a transient digital pulse at random points. However, this approach ignores the architectures and the physical parameters of the circuits. According to the layouts and the states of the circuits, the waveform and error rates of the pulses at different circuit nodes may be different. The circuit nodes which can cause errors are called sensitive nodes.

Fig. 3 shows the four possibilities of SEEs in the inverter circuit. If the charged particle strikes the area between the Vss and B, the semiconductor material will still be ionised. However, there is no voltage difference between the Vss and B and hence, the electron-hole pairs will not move, as a result, there will be no pulse at port B. When the input of the inverter is "1", the area between the Vcc and B is the sensitive area. If the input of the inverter is "0", the sensitive area will be between Vss and B. Thus, the sensitive areas might be different in different circuit states. Hence, the outputs of SEE are determined by the affected areas and the inputs.

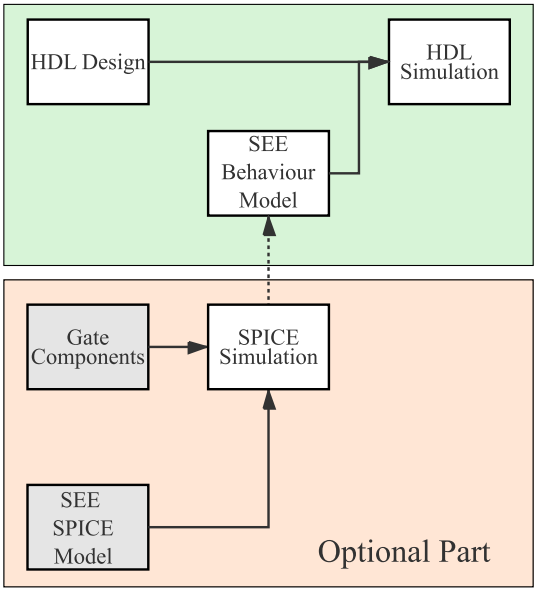

Fig. 4. The workflow of the proposed scheme. Optional part means that if SEE behaviour models has been generated, the SPICE simulation can be skipped.

Conventional HDL simulation cannot cover conditions, which are related to the physical structure of circuits and circuit states. Therefore, direct SEE injection in HDL code is not the correct way to simulate SEEs for integrated circuits. In this paper, the proposed SEE simulation scheme takes the advantage of both transistor simulation and HDL simulation to achieve high efficiency and accuracy.

3. The proposed SEE simulation scheme

The workflow of the proposed scheme is shown in the Fig. 4. Due to the complexity of the large scale circuits, it is very difficult to directly

undertake physical simulation for VLSI. Considering the fact that all digital circuits are made up of basic logic components, a set of SEE behaviour models for logic components will be generated in the proposed scheme. The SEE behaviour models are generated from SPICE simulations based on the gate libraries and SEE SPICE models. In this scheme, the SEE behaviour models can be reused for different HDL designs so that there is no need to re-conduct the generation of the SEE behaviour models. It also provides the flexibility to customize the gate libraries and SPICE models to achieve more accurate results.

3.1. Process of SEE simulation

The process of the SEE simulation for VLSI includes three steps: 1) generate the SEE models for basic circuit units, 2) build HDL netlists for the large scale circuits and 3) carry out simulation and analysis.

The first step is the most important step in the proposed scheme. The accuracy of the SEE models determines the accuracy of the simulation results. On the physical level, SEEs are transient currents caused by the particles, which can be affected by the capacitance between the transistors, the resistance of the circuit wire, the semiconductor material of the chips and the intensity of the radiation. In this step, we use transient current sources in the SPICE to simulate the SEEs. By injecting the transient current at the sensitive circuit nodes, the changes of voltage will be observed on the output ports.

The second step is to build HDL netlists of large scale circuits. Compared to the small circuits, the large scale circuits are much more complicated. In a circuit with thousands of transistors, there might be millions of possibilities which need to be simulated. It is not viable to do that in SPICE. We replace the basic logic components with the corresponding SEE models. Therefore, HDL netlists should correspond with the structure of the physical circuits. However, normal HDL designs do not indicate the physical circuits directly. Here, we generate the SPICE netlists by EDA tools and components library, which can be converted into HDL netlists. A script tool is also designed to replace the basic components and generate new HDL netlists automatically.

The third step is to undertake simulations and analysis. Considering that the large scale circuits may contain hundreds of inputs and outputs, the simulation bench files are also generated by scripts automatically.

In this final step, the simulation test-bench codes contain three parts: 1) input generation, 2) SEEs injection and 3) error detection. Input generation part is used to generate specific input data streams to represent different software programs. The injection part is used to control the SEE injection. The error detection part is used to monitor the output and analyse the errors. In this scheme, two identical designs are used as a reference module and an experiment module respectively. When the simulation starts, both modules are monitored. By comparing the outputs, the errors can be then detected. The number of the errors will be noted for subsequent analysis.

3.2. Circuit unit

In order to generate the SEE model of those basic logic components, a concept of the basic circuit unit is introduced in this paper. The basic circuit unit is a “black box”, with inputs and outputs and relationships between the inputs and outputs which can be represented by an equation. When SEEs occur in this unit, the effects of the SEE (i.e. digital pulses and bit-flips) can also be represented through this equation.

Therefore, we can build HDL SEE models of the integrated circuits by using equations. The model should include two parts: 1) the behaviors of the circuits without errors and 2) the effect of SEEs on operational circuits. In digital circuits (including combinational and sequential circuits), the output will only be affected by the current inputs and states. The equation of the normal behaviour of the circuits can be represented as follows:

$$O_n = f(S_n, I_n), \tag{1}$$

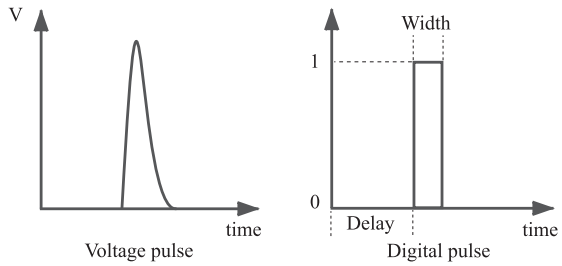

Fig. 5. The original transient SEE pulse and the generated SEE digital pulses.

where O_n represents the current output of the circuit, I_n represent the current inputs and S_n represents the current state of the circuit.

As indicated in Section 1, SETs and SEUs are two main types of soft errors in SEEs. SETs can generate digital pulses, which will propagate through the following circuits. SEUs can cause bit-flips and they will change the current state of the circuits. We represented SETs and SEUs in separate equations.

When the charged particles strike a sensitive area in the circuit unit, a transient current will be generated, which will cause a voltage pulse. If the peak voltage exceeds the threshold voltage of value “1”, a digital pulse will be generated. The waveform of the voltage pulse and the corresponding digital pulse is shown in Fig. 5. When one charged particle strikes a transistor in the circuit, the pulse will appear at output ports after a delay. The duration time of the signal at the output ports is the width of the pulse. According the different current states and the propagation paths, the output signal can be a positive pulse, a negative pulse or no pulse. For the output of one SET on transistor K, the equation can be then represented as follows:

$$O_{SET,k} = \begin{cases} 0 & 0 < t < T_d, \\ f_{SET,k}(S_n, I_n) & T_d \le t < T_d + T_w, \\ 0 & T_d + T_w \le t, \end{cases} \tag{2}$$

where $O_{SET,\,k}$ represents the output signal of SETs on one transistor, $f_{SET,\,k}(S_n, I_n)$ represents the outputs of pulses which can be positive pulse, negative pulse and no change, t represents the elapse time of the events, T_d represents the time for the pulses to propagate to the outputs from the strike point and T_w represents the width of the pulses.

The possibilities to trigger the $O_{SET,\,k}$ depends on the radiation cross section of the transistors. To simplify the process of the analysis, we assume that the size of the area under the radiation for each transistor is same. Therefore, if there are N transistors in the circuit unit, the probability of each transistor to trigger the $O_{SET,\,k}$ will be $1/N$. When a SET occurs in the circuit unit, the equation for this circuit unit can be then represented as follows:

$$O_{SET} = f_c\left(O_{SET,1}, O_{SET,2}, \ldots, O_{SET,N}\right), \tag{3}$$

where f_c is a choice function, which indicates the transistors struck by the particles, and O_{SET} represents the output of SETs on this circuit.

Similar to SETs, the effects of the SEUs can also be represented mathematically. When SEUs occur in the circuit unit, the state of the circuit will be changed, which may also change the output of the circuit unit. Therefore, the equation of the SEU on one transistor K can be then represented as follows:

$$S_{SEU,k} = \begin{cases} 0 & 0 < t < T_d, \\ f_{SEU,k}(S_n) & T_d \le t, \end{cases} \tag{4}$$

where $S_{SEU,\,k}$ represents the new state of the circuit unit after the SEUs on transistor k, T_d represents the propagation time and $f_{SEU,\,k}$ represents the effects of the SEU on transistor k.

Therefore, the effects of the SEUs on the circuit unit can be then represented as follows:

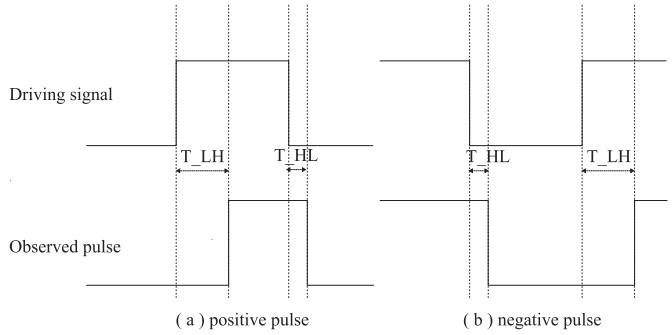

Fig. 6. The propagation effects on positive and negative pulses, which caused by the difference between the rising and the falling edge propagation time.

$$S_{SEU} = f_c\left(f_{SEU,1}, f_{SEU,2}, \ldots, f_{SEU,N}\right), \tag{5}$$

where S_{SEU} represents the new state of the circuit unit after the SEUs in this circuit unit, $f_{SEU,\,k}$ represents the effects of the SEU on transistor k and f_c represent a choice function, when SEUs occur, one of the $f_{SEU,\,k}$ functions will be triggered.

3.3. Propagation between multiple units

In physical circuits, the observed voltage changes depend on not only the struck units themselves, but also the propagation path. When SEEs occur in logic gate chains, the width of the transient pulses may significantly increase or decrease in the propagation, which is called "propagation-induced pulse broadening" (PIPB) effect [18].

As shown in [19], PIPB effect is induced by unbalanced propagation delay of the rising edge and the falling edge in logic gate chains, Fig. 6 shows an example of PIPB effect. In this figure, the 'Diving signal' is the output pulse of the previous circuit unit, assuming that there is a SET occurring in the previous unit. The observed pulse is then the input pulse of next circuit unit. T_{LH} represents the time for the voltage to change from low to high and T_{HL} represents the time for voltage to change from high to low, which include propagation time, charging time and discharging time. When the rising edge time (T_{LH}) is longer than the falling edge time (T_{HL}), the detected width of the positive pulse will reduce, while the negative pulse will widen. When the SEE occurs in the long logic gate chains, the propagation effect will be accumulated, which may further affect the transient pulses significantly.

The rising and the falling edge delays depend on the capacitance of the nodes, which are related to the number of driven gates [19]. In this paper, the detected propagation delay at input wires of unit K is represented as follows:

$$dt(I) = \begin{cases} T_{LH} & I = 0, \\ T_{HL} & I = 1, \end{cases} \tag{6}$$

where $dt(I)$ represents the detected propagation delay, I represents previous valid input values, T_{LH} represents rising edge delay and T_{HL} represents the falling edge delay.

Hence, the input of the unit can be represented as follows:

$$I_n' = I_{(t_n - dt(I))} \tag{7}$$

where I_n' represents the input value of current state for operations, t_n represents the current moment and $dt(I)$ represents the detected propagation delay.

3.4. Large circuits

Large circuits are composed of many small circuits or circuit units. When the circuits are struck by the particles, SEEs may occur in any of the small circuits. The probability of an SEE occurring in the specific units corresponds to the SEE cross section of the units and the size of the circuit units. The SEE cross section represents the number of events per unit fluence. For the circuits in one chip, the SEE cross section should be identical. Therefore, the probability of the SEEs for each unit should correspond to the size of circuits. In other words, the larger circuits may have higher chances to capture particles and generate more SEEs.

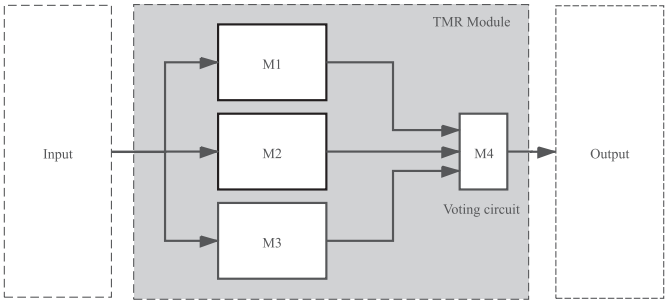

Fig. 7. A TMR module for SEE injection. M1, M2 and M3 are three identical modules. M4 is the voting module. The size of M1, M2 and M3 is bigger than M4.

Fig. 7 shows an example of SEE injection in large circuit, which is a TMR module with 4 sub-modules. There are 3 identical big functional modules, M1 M2 and M3, and a small voting module M4. If a SEE occurs in the TMR module, the transient pulse should occur in one of those modules. However, considering that the size of the functional module and the voting module are very different, it is more likely that the pulse occurs in the bigger sub-modules: M1, M2 or M3. The size of the unit is relative to the probability the SEE function of the unit that is triggered.

Therefore, when one SEE occurs in one complex circuit with M circuit units, the probability (P_i) that the unit i is struck can be represented as follows:

$$P_i = \frac{s_i}{s_1 + s_2 + \ldots + s_M} = \frac{s_i}{S}, \tag{8}$$

where s_1, s_2..s_M represents the sizes of each circuit unit respectively in this large circuit, S represents the total size of the circuits and s_i represents the size of the unit i.

Integrated circuits consist of PMOS and NMOS transistors. The size of the circuits can be represented by the number of the PMOS and NMOS. The probability of the occurrence of SEEs can be represented as follows:

$$P_i = \frac{s_i}{S} = \frac{N_{pmos,i} + \lambda N_{nmos,i}}{N_{pmos} + \lambda N_{nmos}}, \tag{9}$$

where $N_{pmos,\,i}$ represents the number of the PMOS in unit i, $N_{nmos,\,i}$ represents the number of the NMOS in unit i, N_{pmos} represents the number of all PMOS in this complex circuit, N_{nmos} represents the number of all NMOS and λ represents the ratio of size of the PMOS and NMOS.

We assume that the PMOS and NMOS transistors have an identical sensitive size that can catch particles to ease calculations. In that case, the λ is equal to 1 and the probability can be represented as follows:

$$P_i = \frac{s_i}{S} = \frac{N_{mos,i}}{N_{mos}}, \tag{10}$$

where $N_{mos,\,i}$ and $N_{mos,\,i}$ represents the numbers of the MOS in unit i and the number of all MOS in the large circuit respectively. By using the SEE models and the probability for each unit, the SEE simulation scheme for large circuits can be then implemented.

4. Implementation of SEE models

The SEE models of the circuit units are the foundation of the predication scheme. To obtain the accurate models, the physical parameters are included in the circuit units simulations. In this paper, the circuit

Table 1
The physical netlist of the S27 circuits.

```
.subckt S27 GND VDD CK G0 G1 G17 G2 G3
XDFF2 G7 CK G13 DFFQNX1M
XDFF0 G5 CK G10 DFFQNX1M
XDFF1 G6 CK G11 DFFQNX1M
XU10 G14 G0 INVX2M
XU11 G17 G11 INVX2M
XU12 G8 G14 G6 AND2X1M
XU13 G15 G12 G8 OR2X1M
XU14 G16 G3 G8 OR2X1M
XU15 G9 G16 G15 NAND2X1M
XU16 G10 G14 G11 NOR2X1M
XU17 G11 G5 G9 NOR2X1M
XU18 G12 G1 G7 NOR2X1M
XU19 G13 G2 G12 NOR2X1M
```

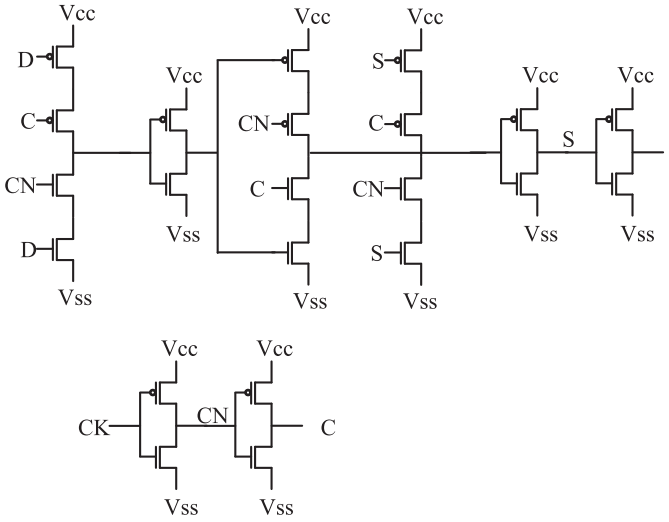

Fig. 8. The circuit of the QFFQX1M in SMIC gate library.

units are simulated in HSPICE [20], a Synopsys circuit simulator for accurate circuit simulation.

4.1. Selection of basic components

In this scheme, the basic units to generate SEE models can be a set of gate components or other circuit modules in the targeted designs. The SEE models will be more accurate, if more transistors are covered in the HSPICE simulations. However, the larger the units are, the more difficult the transistor simulation will be. Considering the complexity of the integrated circuits, the basic units with smaller size and higher reusability can ease both transistor level simulations and system level simulations. Thus, the gate components are better options.

For instance, Table 1 shows the HSPCE netlist generated from the S27 circuit, which is one of the ISCAS89 benchmark circuits. QFFQX1M, INVX2M, AND2X1M, OR2X1M, NAND2X1M and NOR2X1M in the HSPICE code indicates different gates used in physical circuits, respectively. Those components are all units required for S27 circuit. It is possible to combine parts of the components to a bigger units. However, it will significantly increase the complexity of following simulations. Considering the reusability and small size of those components, it will be suitable to choose those components as basic circuit units in the following simulations. In this paper, there are 14 components from the gates library are used in simulation for circuits in ISCAS89 from the smallest one to the largest one.

Table 2
Time required of the SPICE simulation to build SEE models.

Unit	Size of the unit	Number of probabilities	Time required
INVX2M	2	4	1 s
OR2X1M	6	24	16 s
OR3X1M	8	32	31 s
OR4X1M	10	160	238 s
NOR2X1M	4	16	6 s
NOR3X1M	6	48	32 s
NOR4X1M	8	128	121 s
AND2X1M	6	24	16 s
AND3X1M	8	32	31 s
AND4X1M	10	160	238 s
NAND2X1M	8	32	6 s
NAND3X1M	10	160	32 s
NAND4X1M	12	192	480 s
QFFQX1M	25	176	196 min

4.2. Architecture of circuit units

The ISCAS89 benchmark circuits are coded in HDL. The physical parameters (e.g., capacitance, resistance) are not included in the ISCAS89 files. In order to obtain the physical architecture of the basic units, HDL designs need to be firstly compiled and implemented. After the implementation, the SPICE netlists are generated and the logic devices in the HDL codes are replaced with the logic components in the physical gate libraries. In this paper, a 180 nm gate library [15] from the SMIC is used to build SEE models.

Fig. 8 shows the architecture of the DFFQNX1M, which is a flip-flop in the SIMIC 180 nm gate library. This is sequential circuit which consists of 22 transistors. As mentioned in Section 3, SEEs are affected by the current inputs and states. In order to build accurate models, it is necessary to cover all circumstances in the SPICE simulation. The number of the possibilities corresponds with the number of inputs and the number of the states. Therefore, The workload for building the SEE models can be estimated by the number of the inputs and states. For a K-size circuit with N_i changeable inputs and N_s changeable states. The possibilities for the circuit can be represented as follows:

$$S_t = 2^{N_i} \times 2^{N_s} \times K, \tag{11}$$

where the S_t represents the number of the probabilities which need to be simulated and the K represents the number of the transistors in this circuit.

The number of the probabilities can also be used to evaluate the time required to build SEE models. For example, QFFQX1M has 1 circuit state and 5 inputs including D, C, CK, VDD and VSS. When the circuit is working, the VDD and VSS are constant and the D, C and CK are changeable. According to Eq. (11), the number of circumstances for all transistors is equal to 176, which is also the number of the simulation rounds required for building the SEE model.

The time required can be predicted by the probabilities and the time required for each simulation round. Table 2 shows the time required for the SPICE simulation for all units used in this paper. The SEE models of small units can be generated in several minutes, while the SEE models of big circuits (e.g. QFFQX1M) will take several hours. In general, the SEE models for all units could be generated in hours.

4.3. Injection currents

Both SETs and SEUs are caused by the transient currents. At present, the most common methods for simulating single-particle effects are to inject the transient currents to the target node, where the location is struck by high-energy particles. The current sources are used in this paper to generate the transient currents to simulate single-particle effects. The intensity of the current reflects the intensity of radiation and the capability of energy absorption.

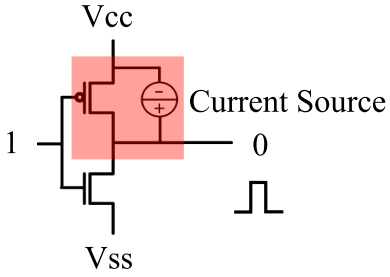

Fig. 9. The location of injection current source in the inverter circuit, when the PMOS in the inverter is struck by the particle.

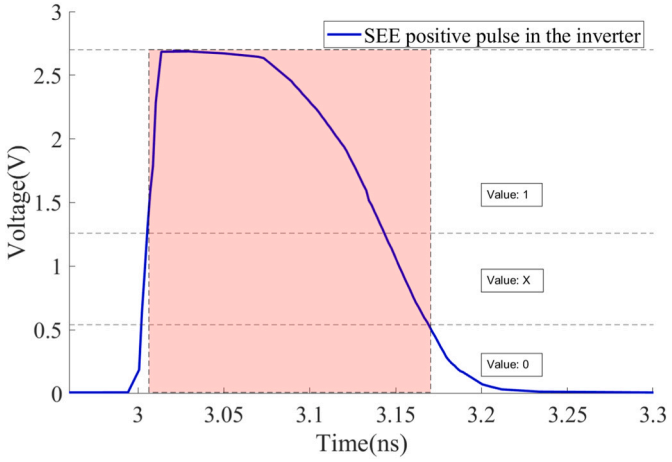

Fig. 10. The simulation results of SEE positive pulse in the inverter circuit. The input of the inverter is "1" and the PMOS is struck by the particle.

In order to improve accuracy of the SEE model, the injection currents should follow the trends of the physical effects. There are some current models for SEE simulation (e.g., the rectangular pulse model, the double exponential pulse model and the transient pulse model [21,22]).

However, for the rectangular and the double exponential pulse, it has been proven that the peak of the currents could be 20% lower than real transient currents [21]. Here, we use a transient current model which is based on the quantity of the electricity generated by the ionization effects. Compared to the other models, the current waves generated by this model are closer to the real currents of the SEEs [23].

The equation of the transient pulse model can be represented as follows:

$$I(t) = \frac{2Q}{\tau\sqrt{\pi}}\sqrt{\frac{t}{\tau}}exp\frac{-t}{\tau}, \tag{12}$$

where the Q represents the quantity of the free electricity generated by the ionization effects, which is also the quantity of free electricity in the injection, τ represents the physical parameters related to the electricity absorption, which is affected by the materials of the semiconductors, physical shapes and architecture of the transistors. The higher value of τ means the slower current changes in the circuits. Normally, Q is between 100 fC (femto-coulomb) and 150 fC [24] and τ is between 25 ps and 35 ps (picosecond) [25] for the circuits between 100 nm and 200 nm. We considered the circuits are from the 180 nm gate library, Q is set at 100 fC and τ is set at 25 ps.

4.4. Simulation of SEE models

4.4.1. Implementation of current sources

The direction of the SEE transient currents are affected by the electric field direction. Therefore, the injection current source should correspond to the electric field. Fig. 9 shows the examples of the injection current source in the SMIC 180 nm inverter circuit. If the output the inverter is "0", the PMOS will be the sensitive gate. In this case, the SEE transient current flows from Vcc to output port, which will cause a positive pulse. If the output is "1", the direction of the transient current is from output port to Vss causing a negative pulse. Therefore, the current source will be connected to the sensitive gates to generate pulses in the simulation.

When transient currents are injected, corresponding voltage changes will be observed at output nodes. Fig. 10 shows the voltage change of the inverter output in the SPICE simulation. The peak of the voltage pulse is about 2.74 V and the time duration is about 0.2 *ns*. When it is converted to the digital model, the time duration is the width of the digital pulses.

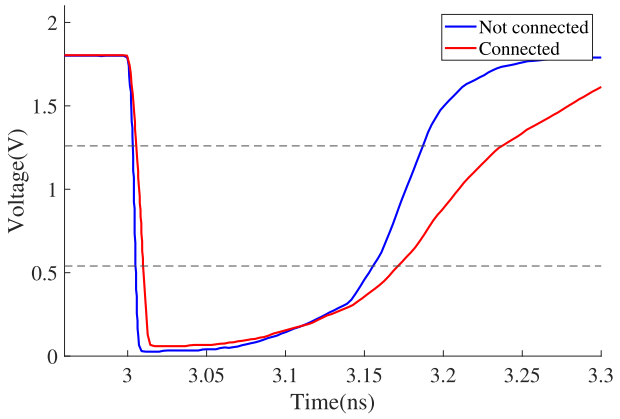

Fig. 11. The comparison of the output voltages of the inverter with and without following circuits.

Table 3
Lookup table for initial parameters.

Number of units	Specifications (ps)	
	Rising edge delay	Falling edge delay
1	79	64
2	97	80
3	112	92
4	127	104
5	141	115
6	159	131
7	178	146

4.4.2. Propagation effects

Due to the different connections of the circuits, the voltage curve of transient pulses are not constant. Fig. 11 shows the voltage curve of transient pulses in the inverter circuit with one and without following circuits. Compared to the separate inverter without following circuits, the output voltage curve in inverter with following circuits is smoother, which causes propagation effects in the circuits.

As mentioned in Section 3, Increasing number of logic gates increases the capacitance of the connection node, which will cause more rising edge and falling edge delay. In this paper, a lookup table is built to represent Eq. (7). Considering that the number of driven gates can be obtained by simply counting how many times the output wires are used in the netlist, T_{HL} and T_{LH} parameters can be initialized, when final netlists are generated. Table 3 shows the pre-built lookup table used in the simulation. For example, if a previous units is driving 3 following units, the rising edge from the previous output should be detected by the following units 112 ps later.

Table 4
the SEE model of INVX2M.

	Inputs	probability	Width (ns)	Delay (ns)
SET	0	50%	0.22	0.0
	1	50%	0.22	0.0
SEU	X	0%	/	/

Table 5
the SEE model of DFFQNX1M.

	Inputs-states[a]	Probability	Width (ns)	Delay (ns)
SET	XXX	8%	0.22	0.0
	0×0	4%	/	0.10
SEU	0×1	8%	/	0.12
	1×0	8%	/	0.11
	1×1	4%	/	0.10

[a] The data input, CLK and Stored bit.

4.4.3. The width of the digital pulses

The width of the digital pulses represents the time duration of the voltages, which can be sampled as incorrect values. In CMOS devices, the threshold voltage of logic "0" is normally $0.3(Vcc - Vss)$. When voltages are lower than $0.3(Vcc - Vss)$, the sampled values will be logic "0". The threshold voltage of logic "1" is $0.7(Vcc - Vss)$. When voltages are higher than $0.7(Vcc - Vss)$, the sampled values are "1". In this paper, the threshold voltages are represented as follows:

$$V_H = \frac{Vcc - Vss}{\sqrt{2}} + Vss, V_L = \frac{Vcc - Vss}{2 + \sqrt{2}} + Vss, \tag{13}$$

where the V_H represents the threshold of the high voltages and the V_L represents the threshold of the low voltages. The width of the positive pulse is the duration of the voltage peaks which are higher than V_H. The width of the negative pulse is the duration of the voltage peaks which are lower than V_L.

Table 3 shows the specifications of the two pulses. The width of the pulse in the inverter with following circuits is 15% wider than the pulse in separate circuit.

Finally, The digital SEE models can be generated by using the SPICE simulation results. The digital SEE models of the INVX2M and DFFQNX1M are shown in Table 4 and Table 5, which include the trigger conditions, possibilities, the width of the pulse in different circumstances and the output delay.

5. SEE simulation of large scale circuits

In order to carry out simulations for large scale circuits on the system level, the SEE models generated from the SPICE simulation need to be integrated into the HDL circuit units. Therefore, the original logic components (e.g. OR gates and AND gates) are replaced with the circuit unit modules that contains the digital SEE models. Considering the large number of the logic components, Python script tools are developed to generated HDL codes, automatically.

5.1. Implementation of HDL circuit units

Circuit units are used to replace the original logic components, therefore, the function of original logic components should also be covered by the circuit unit. The circuit units will include the function of the original logic components and the SEE models. There are three parts in the circuit unit models: 1) original logic, 2) configurable parameters and 3) the functions of SEE models to generate the bit-flips and digital pulses.

The SEE model is implemented as the task functions in Verilog. When the SEEs are triggered, the task functions will be called by the injection scripts. The task functions simulate the pulse caused by forcing signals on the output wire to be incorrect for a certain time and simulate the bit-flips by changing the circuit states.

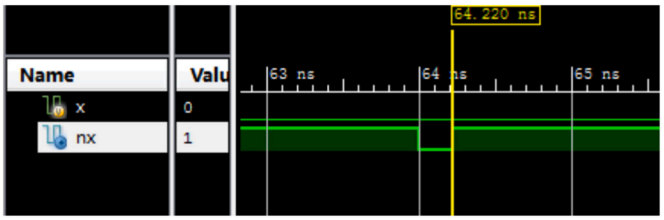

Fig. 12. The transient pulse in HDL simulation. "x" is the input of the inverter and "nx" is the output of the inveter. The pulse is injected at the moment of 64 ns in the inverter unit.

Fig. 12 shows the example of the SEE output in the inverter in the HDL simulation. The "x" is the input of the inverter and "nx" is the output. The SEE occurs at 64 ns and the function of the SEE model is triggered at the same time. SPICE simulation also shows a negative pulse lasting for 0.22 ns.

In the HDL simulations, the SEEs are represented by injection functions in the units. Considering that there is only one unit struck by the particle in the single event, therefore only one of the injection functions should be triggered. We marked all circuit units and injection functions with unique IDs to indicate the circuit units. In this way, only one function will be accessed during SEE injections. The IDs can be calculated by the following equation:

$$ID_k = \sum_{i=1}^{k-1} s_i \tag{14}$$

where s_i represents the size of the marked circuit units, the number of the MOSFETs in the circuit unit. During SEE injections, the scripts will generate a random number under the range of the IDs. If the number is between the ID_k and $ID_k + s_k$, the SEE function in the unit k will be triggered. The probability will be calculated using the Eq. (10).

5.2. Script tools for injection

The large scale HDL designs contain a large number of circuit units. For example, S38584, the largest circuits in ISCAS89, contains 11,448 logic units. The test-bench codes include three parts: 1) input generation, 2) SEE injection and 3) output analysing.

In the HDL simulation, there are two types of inputs generated by the scripts, which are random inputs and specific inputs. In this scheme, the random inputs are generated based on the given switch probabilities of input signals. If the HDL bench codes just drive the target design with the random inputs, the difference of the data stream will not be observed. If the test-bench codes drive the circuits with specific input data streams, the SEE mitigation performance of the input streams can also be

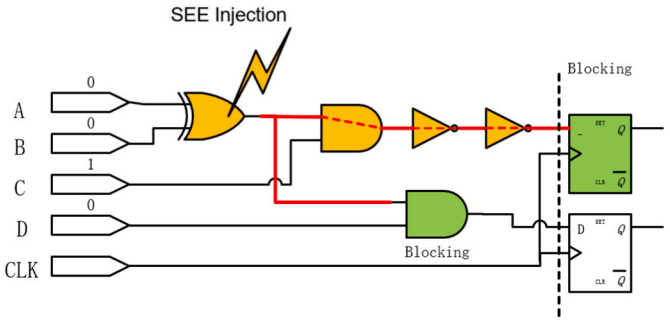

Fig. 13. Errors blocked by logic gates and registers. SEE occurs in the XOR gate. If C or D is "0", the pulses will be blocked by AND gates. If CLK is not flipping, the pluses will be blocked by registers.

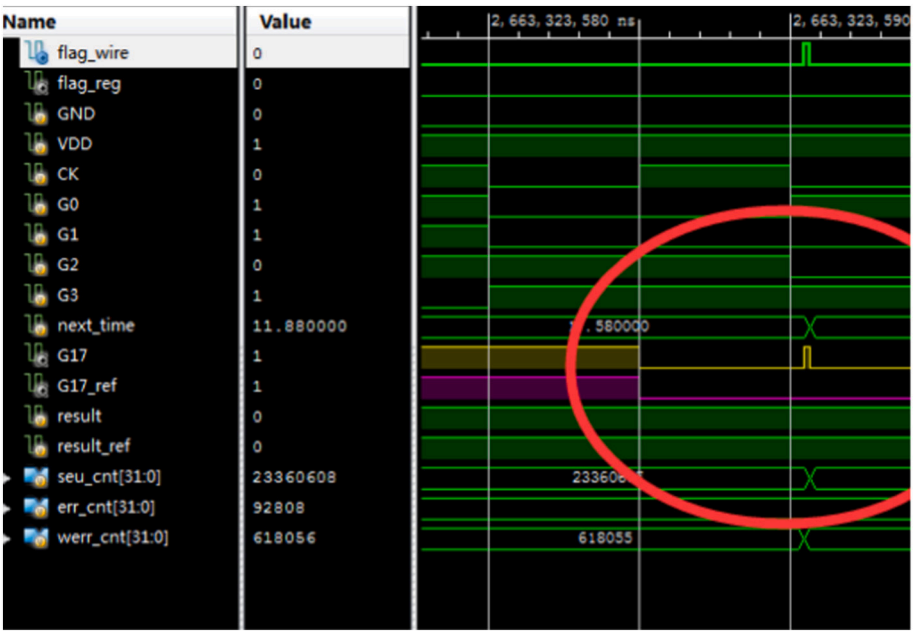

Fig. 14. The outputs of the S27 in HDL simulation. The signal "G17" is the original output of S27. "G17_ref" is the reference output of S27. A transient pulse can be observed in "G17".

evaluated.

The test-bench codes call the functions in the SEE models to carry out injections. In the simulation, there are two identical HDL designs that will be instantiated. The first one is for SEE injection, the other one is used as a reference. When SEEs occur, one of the functions with unique IDs will be triggered and HDL scripts monitor the outputs of both designs. By comparing the outputs, the errors can be detected and recorded. The script will calculate the number of errors, the number of SEEs and the error rates of the design.

5.3. *Propagation of SEE induced errors*

When SEEs occur in the large scale circuits, some error may not be visible at output ports of the circuits. The pulses may be blocked in the propagation path by the logic gates and registers.

Fig. 13 shows the scenario where SEE errors are blocked by logic gates and registers. There are 2 different propagation paths of the errors in this figure. In the first propagation path, since the current input 'C' is "1", the error generated in the OR gate can reach the register. However, the registers only sample the input signals at the positive edge of the clock. There is a high chance that the pulse will not be sampled, which means that the error is blocked. In the second path, since the input signal 'D' is "0", the error cannot go through the AND gate, which means that the errors are blocked by the logic gate.

Considering the above cases, the scripts will also monitor the signal after buffers. When generating bench codes, scripts will automatically add registers connected to the output wires. The original outputs will be compared to the outputs of the buffers, which is used to analyse the effects of the errors on the following circuits.

6. Simulation results of large scale circuits

We used the same generated SEE behaviour models to analyse the performance of more than 40 circuits from ISCAS89 benchmark circuits, which represents that the circuits are under same circumstances. In addition, we also compared the SEE mitigation performance of the same circuits hardened by the triple modular redundancy (TMR) technology and register space-time redundancy (STR) technology.

6.1. *The simulation of the ISCA89 circuits*

There are more than 40 circuits in ISCA89, among those, the S27

Fig. 15. The outputs of the S27 with buffers in HDL simulation. The signal "result" and "result_ref" are buffered "G17" and buffered "G17_ref".

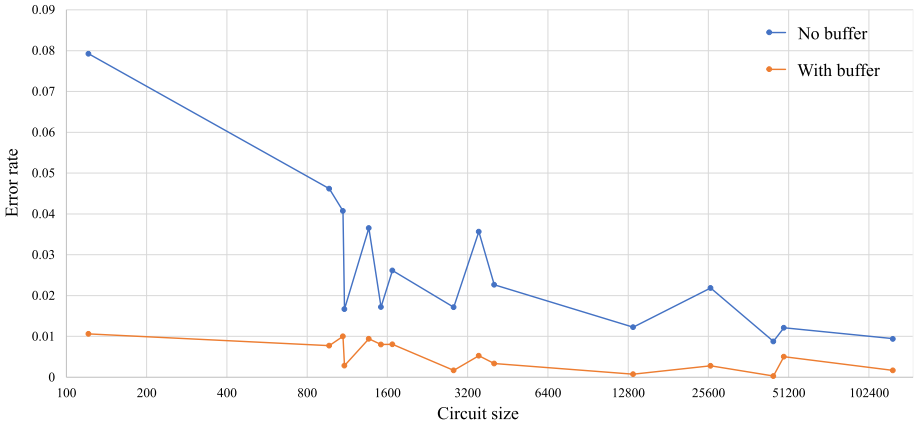

Fig. 16. The error rates of the circuits in ISCAS89 without buffers and the circuits with buffers.

circuit is the smallest. It contains 13 circuit units. S28584 circuit is the largest one. It contains 11,448 circuit units. The simulation results in the circuits indicate the effects of SET and SEU effects between small scale and large scale circuits.

Fig. 14 shows the SEE injection in the S27 circuits, where the G0, G1, G2 and G3 are inputs of the S27 circuits. The G17 is the output of the S27, where the scripts do the SEE injection. The G17_ref is the output wire of the reference unit. The flag_wire indicates the SEE injection. At the positive edges of the flag_wire, the SEEs are injected. In this figure, there is a positive pulse at the G17 wire, which is different from the signal at G17_ref, and werr_cnt is the number of the observed errors.

Fig. 15 shows the comparison of the circuits with and without output buffers. "G17" is the output of the injection circuit and "G17_ref" is the output of the reference circuit. "result" is the output of the buffer which is connected to the "G17" and "result_ref" is the output of the buffer which is connected to the "G17_ref". In the simulation, the output buffers are used to filter out digital pulses to evaluate bit flip rates.

Fig. 16 shows the error rates of the ISCAS89 circuits with and without buffers. This result indicates the bit-flips rates and the total error rates, respectively. In this figure, the error rates of the small circuits without buffers drop significantly, while the error rates of the circuits with output buffers remain stable. This can be attributed to the fact that the SEE induced pulses are more likely to be blocked by the registers and logic designs, while the bit-flips cannot be blocked by the register. Considering there are more registers in larger circuits, there could be a high chance that the pulses are blocked by the registers in the propagation path. Therefore, the error rates of circuits without buffer decreases along with the decrease of the circuit size.

On the other hand, when the circuits are getting larger, there will be more unused propagation paths and invalid states, where the errors cannot affect the following circuits. That is why the observed error rates in the simulation decrease slightly with the circuit size. It is possible to utilise the redundant states and path to improve the SEE mitigation performance.

6.2. Simulation of the hardened circuits

The simulation results of the unhardened circuits show that there are a large number of error bits caused by the pulses. Therefore, we evaluated performance of the circuits hardened by STR and TMR methods.

The SRT method only hardens registers. With extra registers as redundancy, SRT can fix the one-bit errors by itself. In addition, the registers will sample input signals at different moments, which means that it has the capability to filter out digital pulses. TMR is a popular method for hardened systems. It triples the modules for redundancy and uses an extra voting module to select the correct outputs.

It has been proven that TMR has the best error mitigation performance with the highest hardware costs and the space-time redundancy technology has a balance between performance and costs [26–28]. Therefore, we validated the proposed scheme by comparing the performance of both methods in the proposed scheme.

Table 6
The error rates of the S27 circuits with different hardening technologies.

	Original design	TMR	STR[a]
Inputs	4		
Outputs	1		
Transistors	121	389	184
Error rates without buffers	0.0792	0.0195	0.0620
Error rates with buffers	0.0106	0.00130	0.00131

[a] Space-time redundancy.

Table 7
The error rates of the S1423 circuits with different hardening technologies.

	Original design	TMR	STR
Inputs	17		
Outputs	5		
Transistors	5102	15,436	8609
Error rates without buffers	0.0226	0.0024	0.0091
Error rates with buffers	0.0033	0.00016	0.00027

Table 8
The error rates of the S38584 circuits with different hardening technologies.

	Original design	TMR	STR
Inputs	38		
Outputs	304		
Transistors	125,940	385,724	155,886
Error rates without buffers	0.0094	0.0040	0.0065
Error rates with buffers	0.0017	0.00027	0.00035

The Tables 6, 7 and 8 shows the simulation results of S27, S1423 and S38584 with different hardening technologies. S27 represents the small circuits, S1423 represents the medium sized circuits and the S38584 represents the large circuits.

The simulation results show that the proposed scheme provides details analysis to evaluate and compare the SEE mitigation performance of the hardened and unhardened circuits. Firstly, the proposed scheme provides a general analysis includes 1) SET rates, 2) SEU rates, 3) changes of the circuit size and 4) error rates with buffers. It helps to exclude unnecessary hardening methods. For example, compared to TMR, STR could be a better option to harden the S27 circuit with the same performance and much less costs. Secondly, with HDL simulations, the proposed scheme provides the waveform analysis, which can be used

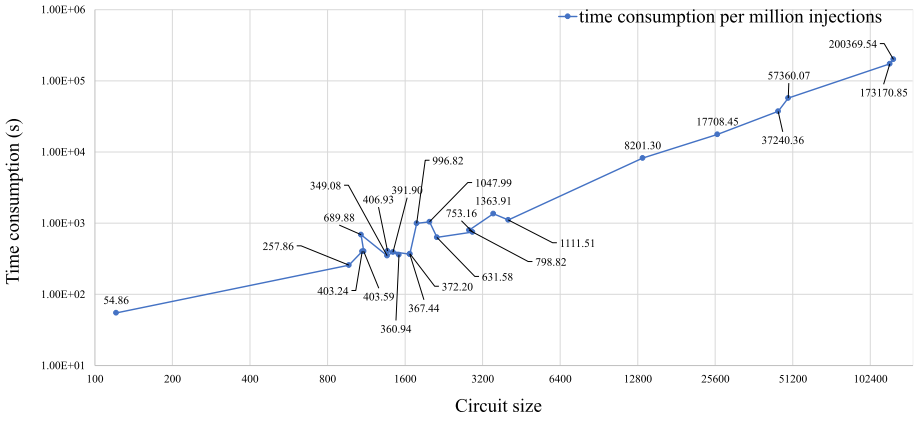

Fig. 17. The time required to SEE HDL simulations for ISCAS89 benchmark circuits.

Table 9
The average time required to run S27 circuit for 1 ms in different simulation environments.

	Average time cost
HDL simulation without injection	0.473 s
The proposed simulation with SEE injection	0.676 s
SPICE simulation with SEE injection	67,000 s

to address the vulnerable parts and the error propagation paths.

6.3. Time required of the simulation

In this paper, the time cost of HDL simulation is elaborated. We can evaluate the time required for the HDL simulation by averaging time costs for each SEE injection. Fig. 17 shows the time required for the HDL simulation with one million SEE injections in the ISCAS89 circuits. By using HDL models, one million SEEs can be injected into S27 circuit (121 transistors) in just 55 s, while it will take 55 h in S38584 (125,940 transistors). The time required for SEE simulation increases nearly linearly with the scale of the circuits.

In this paper, the time requirements in different simulation environments are also tested. Due to the long time required for large circuits to run simulations in SPICE, S27, the smallest circuit in the benchmark circuits, is used for the comparison. There are three simulations including HDL simulation without SEE injection, SEE HDL simulation and SEE SPICE simulation.

Table 9 shows the average time required for HDL simulations and SPICE simulation to run for 1 ms. The time precision of HDL simulation and SPICE simulation is set to 1 ps, so that the time required could be compared directly. The HDL simulation with SEE injection for S27 take 0.676 s to run for 1 ms. The time required increases slightly from 0.473 s to conduct SEE injections. Compared to SPICE simulation, which costs 67,000 s (i.e. ≈18.6 h) to run the simulation for 1 ms, the HDL SEE simulation shows great advantage in term of simulation efficiency. Due to high complexity and huge number of probabilities, it is unlikely to run HSPICE simulations to analyse SEE error rates of large integrated circuits.

7. Conclusion and future work

We have proposed a fast and cost-efficient method to evaluate and compare the performance of large scale circuits under the effect of radiation particles. We used SPICE simulations to build a set of general SEE models to simplify the processes of the evaluation of SEE effects on large scale circuits. The proposed scheme has been evaluated using 40 different circuits from the ISCAS89 benchmark circuits. It is shown that the scheme is able to analyse the circuits with the sizes varying from 100 transistors to 100 k transistors. We have also compared and evaluated the simulation results of the circuits that are hardened by TMR and space-time redundancy technology. The results we obtained proves the correctness of the scheme.

There are still some limitations for the combined research activities. Firstly, the injection currents utilised in this paper are constant values, however, these may be affected by the intensity of the radiations the angles of the beams and the types of the particles. Future research should consider those factors in the SEU models. Secondly, future researches need to explore the automation and optimisation of the HDL SEE model generation. Thirdly, in order to simplify the HDL simulation, the driven circuits are considered here as inverters. In future work, we will consider complex circuits.

CRediT authorship contribution statement

Yufan Lu: Conceptualization, Methodology, Formal analysis, Visualization Writing - Original Draft. **Xin Chen**: Resources, Conceptualization. **Xiaojun Zhai:** Writing - Review & Editing, Supervision, Investigation. **Sangeet Saha:** Data Curation. **Shoaib Ehsan**: Validation. **Jinya Su:** Writing- Reviewing and Editing, **Klaus McDonald-Maier:** Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the UK Engineering and Physical Sciences Research Council through grants EP/R02572X/1 and EP/P017487/1.

References

[1] G. Messenger, M. Ash, The Effects of Radiation on Electronic Systems, 1986, pp. 216–266.
[2] E. Stassinopoulos, J. Raymond, The space radiation environment for electronics, Proc. IEEE 76 (11) (1988) 1423–1442, https://doi.org/10.1109/5.90113.
[3] G.C. Messenger, M.S. Ash, The Effects of Radiation on Electronic Systems, Van Nostrand Reinhold Co, 1986.
[4] S. Saha, S. Ehsan, A. Stoica, R. Stolkin, K. McDonald-Maier, Real-time application processing for FPGA-based resilient embedded systems in harsh environments, in: 2018 NASA/ESA Conference on Adaptive Hardware and Systems, AHS 2018, Institute of Electrical and Electronics Engineers Inc, 2018, pp. 299–304, https://doi.org/10.1109/AHS.2018.8541449.
[5] S. Li, Z. Huang, L. Han, C. Jiang, A genetic algorithm enhanced automatic data flow management solution for facilitating data intensive applications in the cloud, in:

Concurrency Computation Vol. 30, John Wiley and Sons Ltd, 2018, https://doi.org/10.1002/cpe.4844.
[6] X. Li, H. Zhang, B. Mei, P. Li, L. Luo, Q. Yu, The experiment and simulation calculation study on seu hardened effect of flash fpga, in: 2019 3rd International Conference on Circuits, System and Simulation (ICCSS), IEEE, 2019 (pp. 133–136).
[7] X. Cao, L. Xiao, J. Li, R. Zhang, S. Liu, J. Wang, A layout-based soft error vulnerability estimation approach for combinational circuits considering single event multiple transients (semts), IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 38 (6) (2018) 1109–1122.
[8] M. Ebrahimi, H. Asadi, R. Bishnoi, M.B. Tahoori, Layout-based modeling and mitigation of multiple event transients, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 35 (3) (2015) 367–379.
[9] K.O. Petrosyants, L.M. Sambursky, I.A. Kharitonov, B.G. Lvov, Fault simulation in radiation-hardened soi cmos vlsis using universal compact mosfet model, in: 2016 17th Latin-American Test Symposium (LATS), IEEE, 2016, pp. 117–122.
[10] G. Hubert, D. Truyen, L. Artola, M. Briet, C. Heng, Y. Lakys, E. Leduc, Set and seu analyses based on experiments and multi-physics modeling applied to the atmel cmos library in 180 and 90-nm technological nodes, IEEE Trans. Nucl. Sci. 61 (6) (2014) 3178–3186.
[11] C. Bernardeschi, L. Cassano, A. Domenici, L. Sterpone, Assess: a simulator of soft errors in the configuration memory of sram-based fpgas, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 33 (9) (2014) 1342–1355.
[12] S. Golshan, H. Kooti, E. Bozorgzadeh, Seu-aware high-level data path synthesis and layout generation on sram-based fpgas, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 30 (6) (2011) 829–840.
[13] Y. Sun, H.-W. Zhang, Z.-C. Wei, Q.-K. Yu, M. Tang, C. Shen, D. Gong, Heavy ion- and proton-induced seu simulation and error rates calculation in 0.15 um sram-based fpga, in: 2019 3rd International Conference on Circuits, System and Simulation (ICCSS), IEEE, 2019, pp. 84–88.
[14] B. Du, J.E.R. Condia, M.S. Reorda, L. Sterpone, On the evaluation of seu effects in gpgpus, in: 2019 IEEE Latin American Test Symposium (LATS), IEEE, 2019, pp. 1–6.
[15] Smic foundry solutions 90nm,130/110nm,150nm,180nm,250nm,350nm, [EB/OL]. https://www.smics.com/en/site/mature_logic. (Accessed 4 June 2020).
[16] M.S. Hossain, I. Savidis, Reusing leakage current for improved energy efficiency of multi-voltage systems, in: 2019 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE, 2019, pp. 1–5.
[17] A.M. Tosson, M. Anis, L. Wei, RRAM refresh circuit, in: Proceedings of the 26th Edition on Great Lakes Symposium on VLSI - GLSVLSI '16, ACM Press, New York, New York, USA, 2016, pp. 227–232, https://doi.org/10.1145/2902961.2903017.
[18] V. Ferlet-Cavrois, P. Paillet, D. McMorrow, N. Fel, J. Baggio, S. Girard, O. Duhamel, J. Melinger, M. Gaillardin, J. Schwank, et al., New insights into single event transient propagation in chains of inverters—evidence for propagation-induced pulse broadening, IEEE Trans. Nucl. Sci. 54 (6) (2007) 2338–2346.
[19] G. Wirth, F.L. Kastensmidt, I. Ribeiro, Single event transients in logic circuits—load and propagation induced pulse broadening, IEEE Trans. Nucl. Sci. 55 (6) (2008) 2928–2935.
[20] M. Karimian, M. Dousti, M. Pouyan, R. Faez, An improved macro-model for simulation of single electron transistor (set) using hspice, in: 2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH), IEEE, 2009, pp. 1000–1004.
[21] A. Kasnavi, J.W. Wang, M. Shahram, J. Zejda, Analytical modeling of crosstalk noise waveforms using weibull function, in: IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004, IEEE, 2004, pp. 141–146.
[22] L. Rockett, An seu-hardened cmos data latch design, IEEE Trans. Nucl. Sci. 35 (6) (1988) 1682–1687.
[23] L.B. Freeman, Critical charge calculations for a bipolar sram array, IBM J. Res. Dev. 40 (1) (1996) 119–129.
[24] Q. Zhou, K. Mohanram, Cost-effective radiation hardening technique for combinational logic, in: IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004, IEEE, 2004, pp. 100–106.
[25] D. Munteanu, J.-L. Autran, Modeling and simulation of single-event effects in digital devices and ics, IEEE Trans. Nucl. Sci. 55 (4) (2008) 1854–1878.
[26] L. Sterpone, M. Violante, Analysis of the robustness of the tmr architecture in sram-based fpgas, IEEE Trans. Nucl. Sci. 52 (5) (2005) 1545–1549.
[27] R. Oliveira, A. Jagirdar, T.J. Chakraborty, A tmr scheme for seu mitigation in scan flip-flops, in: 8th International Symposium on Quality Electronic Design (ISQED'07), IEEE, 2007, pp. 905–910.
[28] L. Anghel, D. Alexandrescu, M. Nicolaidis, Evaluation of a soft error tolerance technique based on time and/or space redundancy, in: Proceedings 13th Symposium on Integrated Circuits and Systems Design (Cat. No. PR00843), IEEE, 2000, pp. 237–242.