AN EMISSION REINFORCED SCHEME FOR PIPELINE DEFENSE IN MICROPROCESSORS

S. CHRISTO JAIN
Assistant Professor, Dept. of Electronics and Communication,
K S Institute Of Technology, Bangalore-62
E-mail: s.christojain@gmail.com

Abstract - The hostile mounting of semiconductor technology has significantly increased the emission reinforced soft error rate in modern microprocessors. In the meantime, due to the increasing complexity of recent processor pipelines and the incomplete error-tolerance capabilities that former emission reinforced schemes can provide, the surviving pipeline protection schemes cannot achieve complete protection. This paper proposes a complete and cost-effective pipeline protection mechanism using a self-checking design. The emission reinforced pipeline is achieved by incorporating SETTOFF-based self-checking cells into the sequential cells of the pipeline. A replay recovery mechanism is also developed at the architectural level to recover the detected errors. The proposed pipeline protection scheme is implemented in an Open RISC microprocessor in 65nm technology. A gate-level transient fault injection and analysis technique is used to evaluate the error-tolerance capability of the proposed reinforced pipeline design. The results show that compared to techniques such as TMR, the SETTOFF-based self-checking technique requires over 30% less area and 80% less power overheads. In the meantime, the error-tolerant and self-checking capabilities of the register allow the proposed pipeline defense technique to provide an upper level of consistency for different parts of the pipeline compared to former schemes.

Key Terms- Fault-Tolerance, Consistency, Soft Errors, Error Tolerance, Timing Error, Fault Injection

I. INTRODUCTION

The consistency of modern integrated circuits is harshly dared by attacks from high energy particles. A particle attack can produce soft errors, which can be categorized into Single Event-Upsets (SEU) and Single Event-Transients (SET). SEUs are transient bit-flip errors that invert the state held in memories (DRAMs or SRAMs). SETs are transient voltage pulses occurring in combinational logic. An SET become an SEU if it is sampled by a storage element. Lower operating voltages reduce the energy required to induce soft errors. Increased operating speed also significantly increases the probability that SETs are captured. These trends suggest that a dramatic increase in the soft error rate is inevitable.

Usually, microprocessors used in safety-critical applications, such as space, can be protected by Triple Modular Redundancy (TMR). However, TMR is not a viable solution for less critical applications since its overhead (more a serious issue at current technology nodes. ECCs cannot address SETs, and are very expensive for addressing MBUs as they require a larger number of redundant bits. Other techniques have been proposed to mitigate either the SETs or the SEUs in general logic. However, few of these can efficiently provide full protection of the whole pipeline against both SETs and SEUs. They are either only suitable for protecting the pipeline registers (such as Razor II, SEM/STEM techniques), or only applicable in RFs (such as FERST). On the other hand, most of these techniques rely on hardware redundancies, but are not self-checking.

In this paper, we propose an innovative pipeline protection scheme based on a self-checking emission-reinforced register design. The technique is capable of providing cost-effective error-tolerance for a microprocessor pipeline. Our first contribution is the design of the self-checking register architecture, which is developed from the SETTOFF (Soft Error and Timing error Tolerant Flip-Flop), combined with a self-checker. Our second contribution is a pipeline protection technique which incorporates the emission-reinforced self-checking registers into the RF, and the registers between each stage of the pipeline. SEUs occurring during pipeline execution will be detected and corrected on the fly within the register architecture. SETs and Timing Errors (TE) occurring in the combinational gates will be detected if captured by the pipeline registers or the RF. These errors then trigger a pipeline replay which re-executes the operation and corrects the error (more than 200% area and power) is far too expensive. There is room, therefore, for compromise techniques that offer a little less protection than TMR, but with significantly lower overheads.

Memory arrays and caches in microprocessors can be protected by conventional Error Correction Codes (ECC), which have acceptable overheads. Protecting the general logic in the pipeline of a microprocessor has always been a challenge as TMR or duplication are too expensive. Previous work has proposed using ECC to protect the Register File (RF), but the ECC bits need to be calculated and read during each operation. This paper is organized as follows: Section II introduces the literature related to the work. Section III presents the self-checking radiation hardened register architecture. The design and implementation of the radiation hardening pipeline protection technique is given in Section IV, and Section V presents the experimental methodology and evaluation results. Finally, the paper is concluded in Section VI.

II. BACKGROUND

Previous radiation hardening approaches fall into two main categories. A) Fault avoidance techniques aim to
reduce the probability that the system is affected by particle attacks. B) Fault correction techniques detect and correct faults occurring in the system.

Razor II is a pipeline protection technique proposed to tolerate both soft errors and timing errors within the pipeline. Razor II protection relies on error-detection latches in the pipeline registers. All the error correction is achieved by using an architectural replay which re-executes the faulty operation and overwrites the erroneous state. As a result of the replay recovery process, Razor II protection may incur large Instruction Per Cycle (IPC) overheads when the error rate is high, and may therefore impact the overall energy efficiency. In addition, Razor II is only suitable for protecting the pipeline registers, but cannot protect the registers that store the architectural state of the processor.

Soft Error Mitigation (SEM) and Soft and Timing Error Mitigation (STEM), both utilize a variant of TMR to mitigate SETs and SEUs. The STEM cell also adds timing error detection. One novelty of the techniques is that they remove error detection from the critical path, and therefore the delay overhead is reduced, compared to the TMR flip-flop. However, the area and power overhead of the SEM and STEM cells is approximately the same as or slightly bigger than the TMR flip-flop since additional recovery circuitry is added. The large overhead makes these techniques inapplicable for protecting the RF, and therefore they cannot achieve complete pipeline protection. Error detection and correction flip-flop structures to balance performance, power and reliability of pipeline architectures have been proposed. The four flip-flops provide different levels of error tolerance and overheads. By replacing the four FFs in the best storage locations of a pipeline, this approach can enhance circuit reliability with significantly lower over-heads compared to SEM and BISER. Similar to Razor, this approach focuses on protecting the pipeline registers rather than the RF, as the proposed FF architectures can tolerate SETs but do not provide full SEU-protection. A Confidence-Driven Computing (CDC) model is proposed adaptive protection against transient faults. The approach estimates the confidence in a computation and allows repeated computations (in time or in space) to increase confidence. CDC has low overheads compared to conventional error-tolerant techniques, but it requires a controller to trigger duplicate computations. It has a much larger error detection window (multiple clock cycles) than Razor or rollback recovery techniques. CDC is efficient for enhancing the reliability of combinational logic in each pipeline stage, but does not aim to protect dense memory blocks such as the RF. RF protection based on the FERST hardened cell has been proposed. FERST uses three C-elements to mitigate both SEUs and SETs at the input of the latch. FERST incurs nearly 100% area and power overheads. The main drawback of FERST is that a C-element is added in the signal path, which will induce a delay overhead of around 70% in 65nm technology. This makes FERST hard to use to protect the timing-critical pipeline registers. Even in the RF, the delay overhead of FERST can have big impact on the performance.

A. Self-checking capability

Most pipeline protection techniques do not have a self-checking capability and so are vulnerable to soft errors in the redundant, error-tolerance circuitry. The area and geometry of the redundancies determine the probability that the circuit is hit by particles, while the critical charge determines the vulnerability of the circuit to particle attacks.

B. Time redundancy-based error detection

Because SETs occurring in the logic blocks only manifest themselves for a limited period of time, and will be recovered automatically, Time Redundancy-based Error Detection (TRD) moves duplication into the time-domain. The technique is illustrated in Fig. 1. With no hardware duplication, TRD can detect SETs that are manifest at the input of the flip-flop with a maximum pulse width of $D_{\text{TRD}} \leq \delta - D_{\text{setup}} - D_{\text{comp}}$, where $D_{\text{setup}}$ is the setup time of the error flip-flop and $D_{\text{comp}}$ is the delay of the comparator. Such SETs, if captured by the main flip-flop at $t_0$, will recover at $t_0 + \delta - D_{\text{setup}} - D_{\text{comp}}$, while the comparator will assert an error signal due to inconsistent inputs. Similarly, timing errors with a delay no greater than $D_{\text{TRD}}$ are also detected since the correct result will be presented at the input D when the comparison result is latched. This architecture can also detect SEUs in the main flip-flop from $t_0$ to $t_0 + \delta - D_{\text{setup}} - D_{\text{comp}}$, which is called the TRD interval. Although TRD is cost-efficient, it cannot correct. Moreover, SEUs occurring in the main flip-flop outside the TRD interval will escape detection. This is dangerous, as SEUs cannot be recovered until the flip-flop is overwritten by the next input.

III. SELF-CHECKING EMISSION REINFORCED REGISTER ARCHITECTURE

To overcome the drawbacks of the previous techniques, we propose an innovative full pipeline protection scheme. The technique is based on a self-checking radiation hardened register architecture that can address both SEUs occurring inside the register, and the SETs and TEs that are captured by the register. It has a self-checking capability that can, complementarily, protect the redundancies added for error-tolerance. This section presents the circuit-level approach of the register architecture, which is developed from the SETTOFF emission reinforced flip-flop.
A. SETTOFF ARCHITECTURE

SETTOFF is a Soft Error and Timing error Tolerant Flip-Flop, which can recover the SEUs occurring inside the flip-flop on the fly, and can detect the captured SETs and TEs that originate from the preceding combinational logic gates.

The architecture of SETTOFF is shown in Fig. 2. The main flip-flop is a conventional flip-flop. For clarity, only the last storage unit (the inverter pair) is shown. Node N holds the state of the inverter pair. Q is the inverted value of node N in normal operation. The error-tolerant circuitry is divided into two parts, which work in turn during two intervals within a clock cycle. Part I is a TRD architecture adapted. The TRD part works during the TRD interval which is equal to the high clock phase. It detects errors occurring during the write cycle, which include captured SETs and TEs at the input, and the SEUs that flip node N during the TRD interval. On detection of an error, Part I asserts the Error SET signal which can be used to trigger a replay mechanism to re-execute the erroneous write operation and overwrite the errors in SETTOFF. Part II is the Transition Detector (TD) architecture which works during the TD interval (the low clock phase). SEUs that flip node N during the TD interval are interpreted as illegal transitions and are detected by the TD. A correction XOR-gate is used to replace the inverter driving the output Q of a conventional flip-flop. In normal operation, the Error SET bar signal stays high, such that the correction XOR-gate acts as a normal inverter. When an illegal transition (an SEU) is detected by the TD, it assigns 0 to Error SET bar. The correction XOR-gate will then propagate N to Q to correct the SEU on the fly. A correction glitch is generated upon correction of the SEU due to the delay of the TD. The glitch is not a threat because, if captured by the SETTOFF in the following stage, it will be detected by the TRD part as an SET pulse. Notice that only the SEUs that corrupt node N are considered, others are masked.

B. CIRCUIT-LEVEL EVALUATION FOR SETTOFF

The SETTOFF circuit architecture was implemented in a 65nm technology. The proposed error-tolerant architecture in SETTOFF was modeled in SPICE. A conventional D-type flip-flop is used for the main flip-flop in SETTOFF. The power consumption and performance (Clock-to-Q delay and setup time) of SETTOFF was then measured by SPICE simulations, with 1.2V supply voltage, and a 185MHz clock. The power consumption is measured with 10% activity rate, and the overhead is relative to a conventional flip-flop with the same drive strength. The reliability of SETTOFF is also evaluated using fault-injection and simulation in SPICE. Current sources are used to simulate the collected charge induced by particle attacks at circuit nodes. When sufficient charge is injected into a node, it will produce an SET or an SEU, depending on whether the node belongs to a combinational circuit or a storage element. During the simulation, SEUs are injected into both the master and slave latches of the main flip-flop in SETTOFF, and SETs are injected at the input logic. The faults are injected at time instances distributed across the entire clock period.

C. REGISTER ARCHITECTURE

The TRD and TD-based parts in SETTOFF provide both SEU- and SET-tolerance for the main flip-flop. However, the added error-tolerant architecture can itself be struck by radiation particles and hence introduces extra vulnerability. For the TRD part, radiation particle attacks can induce SEUs in the error flip-flop, or SETs in the preceding comparator in the TRD part. Such SEUs or SETs, if captured by the error flip-flop, can generate a false Error SET signal at the output of the TRD part. The false Error SET signal invokes an unnecessary replay execution, but cannot corrupt the system. The unnecessary replay operations only incur an IPC overhead. Similarly, particle attacks in the TD-based part may propagate to its output and hence induce an erroneous Error SET bar signal, which can then propagate through the correction XOR-gate and corrupt the output of the flip-flop. This section introduces a self-checking mechanism to address this problem. The architecture of an n-bit self-checking register is shown in Fig. 3. It is constructed from n SETTOFFs and a self-checker adapted from that in [4]. The self-checker can detect soft errors that corrupt the TDs in each SETTOFF, by monitoring the register output during the interval when TD is enabled. One self-checker is used to monitor all the outputs of all the SETTOFFs in the register through a parity checker. The parity checker is constructed as an n-input XOR-tree. Any illegal transitions occurring at the output of any single SETTOFF will change the parity, and therefore will be detected. Upon detection, the parity checker generates a transition at its output. Such transitions are then captured by the self-checker, which asserts the Error TD final signal.

The transistor-level design of the TD-checker, Fig. 4, is adapted from the transition detector built into SETTOFF. The two delay chains of inverters and transmission gates remain unchanged. The dynamic OR-gate for capturing the implicit pulses generated by the delay chains is separated into two branches, both driven by the system clock. During the TRD interval, when the clock is high, nodes M1 and M2 are charged, the TD-checker is disabled, and both of the outputs, rising tran and falling tran, stay low. During the TD interval, when the clock is
low, the TD-based part of SETTOFF is vulnerable to soft errors. Therefore, the two branches are both enabled, to capture the pulses generated by the delay chain for the rising and falling transitions, respectively. Any rising transitions at the input of the TD-checker will discharge node M1 through transistors d1 and d3, and thus will be signaled at the output rising tran. Similarly, any falling transitions will assert falling tran through the respective branch.

The TD-checker can distinguish correction glitches from transitions caused by errors in the TD. A transition can only assert one of the two outputs of the TD-checker. However, a glitch consists of both a falling and a rising transition, and thus will assert both outputs of the TD-checker. The two outputs, rising tran and falling tran, are then XOR-ed to generate a valid error signal, which is asserted when only one of its input is high. The error signal will stay at 0 when both inputs are 0, or when both inputs are asserted due to a correction glitch.

Notice that there is a possibility that correction glitches at the output of the register will propagate through the TD-checker. This can be caused by the rising and falling transitions not asserting the rising tran and falling tran signals at exactly the same time. The time difference may lead to a positive glitch appearing at the output of the XOR-gate. A Glitch Filter (GF), [4], is used to filter out these glitches and generate the final error signal Error TD final.

D. REGISTRATION EVALUATION

To verify and evaluate the method at sub-system level, a 32-bit self-checking register based on SETTOFF2 was synthesized in 65nm technology and simulated in SPICE. The supply voltage was 1.2V. A 185MHz symmetric clock was used to drive both the register (positive edge), and the error flip-flop (negative edge) of the TRD part. The power consumption of the self-checking register was compared to a conventional register with the same operating conditions and drive strength. With a 10% activity rate for a single bit, the average power overhead of the proposed register is 33%, which is only a 5% increase over SETTOFF2 without the self-checking capability (Section III-B). In terms of area, a single SETTOFF2 requires 30 extra transistors. The proposed register only adds one self-checker, shared between bits, thus the area overhead increase is insignificant. Compared to a conventional register constructed from flip-flops with 32 transistors each, the area overhead of the 32-bit self-checking register is 136%. Since the self-checker is not added to the signal path of the register, the delay overhead of the register is comparable to that of a single SETTOFF2, with an average value of 16.5%. Compared to a technique such as TMR which induces over 200% power and area overhead, the self-checking register requires over 30% and 80% less area and power overhead, respectively A current source based fault-injection mechanism was used to verify the reliability of the self-checking register. The redundancies in the register are separated into 3 parts: the TRD part, the TD-based part, and the self-checker.

V. EXPERIMENTAL SETUP AND RESULTS

A. Implementation Details and Error-tolerance Overheads

The Open RISC processor with the proposed radiation hardened pipeline architecture was synthesized to a 65nm technology, with 1.2V supply voltage and 185MHz clock frequency. The transistor level implementation of SETTOFF and the self-checker which construct the radiation hardened register architecture, were characterized as new cells using Synopsys Liberty NCX. The behavioral model of the OpenRISC 1200 processor was re-designed to incorporate the new pipeline architecture and was then synthesized to cell level using the characterized technology library. The 32-bit pipeline registers and the registers in RF were replaced by the 32-bit proposed radiation hardened registers. Gate-level simulation was carried out, based on the ORPSoc platform which provides the smallest-possible reference system for testing the processor. Three programs, quick sort, tak and a matrix multiplication program, were used for the fault simulation. The evaluation results are compared with the reliability results of an unprotected OpenRISC processor. 4050 faults (including both SETs and SEUs) were injected into the original cells of each of the two processors. In order to carry out a comprehensive fault analysis and reliability evaluation, the locations of these faults cover all the vulnerable registers (the ones identified in Section IV-A) inside the pipeline. Also, the occurrence times of the injected faults are randomly distributed across the entire clock cycle, and both SETs and SEUs were simulated for each bit of the vulnerable registers. All the injected faults incur 306, 305, and 354 errors in the original processor for the quick sort, tak, and matrix multiplication programs, respectively. All these errors however, are mitigated in the protected OpenRISC processor. Specifically, 287 errors of the 306 errors occurring in the quick sort program were recovered on the fly, while the rest (19 errors) were recovered by replay operations. Of the 305 errors occurring in the tak program, 290 were recovered on the fly and 15 errors were recovered through the architectural replay. 344 of the 354 error occurring in the matrix multiplication were
recovered on the fly and the other 10 errors were recovered by the replay operations. In addition, the transient faults injected into the redundant circuitry in the protected processor were also tolerated and did not induce any soft errors into the outputs of the programs.

CONCLUSIONS AND FUTURE WORK

In this paper, we present a complete pipeline protection mechanism realized on an OpenRISC microprocessor. The pipeline protection is achieved by incorporating SETTOFF-based self-checking cells into the most vulnerable sequential cells of the pipeline. A pipeline replay recovery mechanism is also developed at the architectural level to recover the errors detected by the hardened cells. The entire pipeline is protected, both SETs and SEUs occurring in each stage of the pipeline are detected by fault-tolerant cells in the corresponding stages. The proposed robust OpenRISC microprocessor was implemented in 65nm technology for evaluation. A cell-level transient fault injection and simulation technique was used to automatically inject SETs and SEUs into different parts of the pipeline and to record errors caused by the injected faults. The fault simulation results show that the proposed processor pipeline is robust to both SEUs and SETs occurring in different pipeline stages. The overheads of proposed technique for protecting the pipeline registers are smaller than or comparable to the previous low-cost techniques, while the power and performance overhead for protecting the RF is noticeably smaller than conventional ECC. Future work will focus on developing reliable systems which can satisfy both aggressive power and performance requirements.

REFERENCES