

Validation of an SEU Simulation Technique for a Complex Processor: PowerPC7400

Sana Rezgui, Gary M. Swift, Raoul Velazco, Farhad F. Farmanesh

Abstract- Results from fault injection experiments on a modern, complex processor, the PPC7400, are combined with static register ground testing to predict SEU rates of several benchmark application programs. These results compare favorably with in-beam measurements on the same programs.

I. INTRODUCTION

Data on a processor's SEU (single event upset) sensitivities is generally obtained from radiation ground testing during which the program executed by the DUT (device under test) consists of inspecting of each of the processor memory cells accessible to the user, through the execution of a suitable instruction sequence. Such programs, so-called *static* or register tests, typically inspect memory cells such as general-purpose registers, special registers (program counter, stack pointer...) and internal memory. However, the duty cycle of register usage in an actual application will be very different, including using instructions not in the static tests and disturbing other potential SEU targets. The ideal case of radiation ground testing on the final application program is impractical for a variety of reasons including that this program is either unknown or unavailable when the qualification testing is performed on candidate circuits to space projects.

Some application test results have been published typically showing a dramatic difference in results [1]. To cope with this limitation, a *dynamic or application test strategy* can be applied. This consists of exposing the studied architecture to radiation while running simple benchmark programs. The use of such benchmarks relies on the assumption that the duty cycle of register use is not too different from that of a flight application. Unfortunately, this strategy is of only limited relevance for the usual complicated flight experiment.

Fault injection techniques have been explored in order to predict the processor error rate for a given program [2-3]. The

use of such fault injection techniques to simulate in-space or in-beam SEUs could allow testing of full flight applications, not just representative benchmarks. However, the benchmark's usefulness is in validating a particular fault injection methodology. The so-called CEU (Code Emulating an Upset) injection method has been validated for several simple microprocessors [4]. Its effectiveness was proven by comparing radiation data and CEU-based predictions [5-6]. The essence of this technique is to inject a bit flip randomly, that is into a random memory cell of the DUT at a random instant, and to observe the consequence on the operation of the studied application. A CEU experiment consists of repeatedly running the target program and injecting a pseudo-random fault each time in a Monte Carlo simulation of SEUs. When enough repetitions are done, a statistically valid result is obtained for the average number of injected faults needed to produce a given type of error in the program.

In previous works, we have shown that, by applying the CEU technique to different digital architectures based on several processors (the microprocessors 80C51 from Intel [4] and TS68332 [5] from Motorola and the digital signal processors TMS320C50 from Texas instruments and SHARC from Analog Devices [6]), this approach has lead to excellent results. In fact, due to the large percentage of accessible zones by the instruction set (for example none of these processors contains cache memory) those results were somewhat expected.

However, the CEU approach can only inject faults in those targets accessible to the processor's instruction set, that is, only bits that can be read and written. This intrinsic limitation causes a potentially serious impact on the accuracy of the error rate predictions. To investigate this issue for a very powerful and complex modern processor, CEU experiments were performed to predict the SEU application cross sections for several benchmark programs compiled for the PowerPC7400 microprocessor. For comparison, in-beam SEU experiments were conducted on the same benchmarks. Thus, the main goal of this paper is determining the accuracy of CEU-based error-rate predictions for the most difficult target to date, a processor that includes such advanced features as multiple simultaneous execution units, a high degree of pipelining, two-levels of cache control, and internal L1 instruction and data caches.

The research in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

Sana Rezgui and Raoul Velazco are with TIMA Laboratory, Grenoble, France (email: raoul.velazco@imag.fr)

Gary Swift and Farhad Farmanesh are with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, 91109, USA (email: Gary.M.Swift@jpl.nasa.gov)

II. TESTING METHODOLOGY

The CEU methodology does not replace SEU testing, rather its purpose is to leverage the basic register susceptibility determined by static testing and predict the rate of visible errors and malfunctions from upsets of an arbitrary application program. This is accomplished by performing CEU injection on the chosen application. Details of how this works are discussed in this section and experimental results validating the methodology's effectiveness are presented in Section III.

A. Error rate prediction methodology

The CEU injection technique measures the ratio of actual errors to injected faults (or CEUs) τ_{CEU} , for a specific processor running a given application. Multiplying the underlying SEU cross section, obtained from ground testing, by τ_{CEU} gives the application specific cross section:

$$\sigma_{CEU}(application) = \tau_{CEU} \times \sigma_{SEU} \quad (1)$$

In section III-C, this is called the predicted results. The measured (or in-beam) application cross section can be obtained as the number of errors detected divided by the number of particles (integrated flux) per unit area incident on the DUT (equation 2).

$$\sigma_{SEU}(application) = \frac{\text{Number of errors}}{\text{Particle fluence}} \quad (2)$$

When a CEU experiment has arrived at correct predictions, then $\sigma_{CEU}(application)$ will equal $\sigma_{SEU}(application)$.

B. Experimental set-up

The hardware/software set up needed to perform both fault injection and radiation testing experiments was based on a dedicated system, the THESIC (Testbed for Harsh Environment Studies of Integrated Circuits), developed by TIMA laboratory [7]. The architecture of THESIC consists of three components:

- a motherboard built around a microcontroller (the 87C52 from Intel). The motherboard controls and monitors the DUT. It commands power on/off, monitors DUT current, downloads test programs and starts and stops test cycles. In addition, it receives the data, performs pre-processing and transmits the data to the user interface computer via a serial link (RS232).
- a daughterboard designed and developed for each particular DUT.
- a computer for displaying the user interface. The latter accepts commands for the motherboard, allows on-line monitoring of test execution, displays results in "understandable" format, and stores experiment history for later analysis.

A custom daughterboard was built to support the PowerPC family with only the most basic components necessary to run an application program. The Motorola PowerPC7400 microprocessor (also known as the G4) is a low-power 32-bit implementation of the PowerPC Reduced Instruction Set Computer (RISC) architecture. It contains two L1 caches (32K bytes each) and a controller for an external cache

memory L2 (1 Mbyte). It has a 3.3V I/O voltage and a 2.2V core voltage. The daughterboard clocks the processor's external input at 40 MHz but the internal core frequency is multiplied to 260 MHz. This last feature makes the execution of a small program running within the L1 caches 8-50 times faster than when the caches are disabled. None of the processors previously successfully tested with the CEU technique have the complicated feature set of the PPC7400 including internal caches, cache controllers, and extensive pipelining.

As can be seen in Fig.1, the PowerPC daughterboard includes SRAMs, EEPROMs memories (for data and program storage), the Memory Mapped Interface MMI (shared memory between the two boards), 5V compatible buffers (to provide an interface between the PPC7400 and the other memory devices), power supply regulators and a clock circuit. Two ALTERA FPGAs (EPM7128SLC84) are glue logic to implement the interface between the PowerPC7400 and the memory devices and to control the operation of the processor. FPGA1 controls the chip select signals, the MMI control signals, data transfer control, reset and interrupt signals to the PowerPC7400. FPGA2 controls the address bus transfer and the read/write operations of the PowerPC7400 signals.

The THESIC motherboard controls and monitors the daughterboard in both CEU and SEU experiments. After some daughterboard initializations (mostly loading the selected application), the DUT is reset to start the application running. When the DUT completes the application, it signals "FINIT" to the motherboard. Then the motherboard counts the error, if any, and reports to the user interface where some information about errors is logged for further analysis later. Finally, the motherboard resets the DUT and thus starts another iteration.

For SEU experiments, the beam flux is first adjusted to provoke, on average, one error after several iterations through the program. Irradiation runs consist of several error free cycles before the beam shutter is opened and after it is closed. In contrast, CEU experiments are conducted by injecting *one* bit flip into each program cycle at a random location and instant.

III. EXPERIMENTAL RESULTS

CEU targets include only those memory elements directly accessible to the instruction set. In the case of the PPC7400, the CEU target set comprises only its internal registers (approximately 9k bits) excluding the large number (~256k) of SEU sensitive bits in the L1 data and instruction caches. The internal registers that are CEU targets are the General Purpose Registers (GPR), the Floating Point Registers (FPR), the Special Purpose Registers (SPR) and the AltiVec (AVR) registers. The AVRs are not used by any of the benchmarks tested and, thus, upsets there are irrelevant to their correct operation. Consequently, the CEU experiments only inject bit flips into GPR, FPR and SPR bits, which are 55% of the PowerPC internal register bits (the AltiVec's are the remaining 45%). Of course, the in-beam experiments have SEUs

occurring in all susceptible bits, but upsets in the AVRs are irrelevant to the present benchmarks, as are upsets in the caches for the caches OFF tests. When testing the caches ON case, some fraction of cache upsets (corresponding to the fraction of the caches used) is relevant; thus, it is expected that the inherent limitation of the CEU technique would cause an under-prediction of the in-beam results.

A. Results of the CEU injection sessions

CEU injection sessions have been performed on the sensitive zones of the PowerPC7400 running three benchmark programs: matrix multiplication, bubble sort and Fast Fourier Transform. Only the FFT benchmark uses the floating point registers. Random CEU sessions were conducted in which up to ~10,000 faults were injected randomly in all accessible targets while running each benchmark program in two different PPC L1 cache configurations.

Injected faults cause three types of results: tolerated faults, result errors, sequence-loss errors. The first group, *tolerated faults*, corresponds to injected CEUs that had no effect on the known-good results. Some faults are tolerated, that is they do not provoke errors, because the contents of many memory elements are not relevant at the time the CEU occurs (for instance, a register which will be written after the bit flip occurrence, thus “erasing” the fault). Some injected CEUs lead to *result errors* for which the obtained answer is different from that expected from correct program operation. In some iterations, the CEU prevents the PowerPC from ever asserting the “FINIT” signal; these errors are classified as a *sequence loss*.

Tables 1 and 2 summarize the CEU results for the benchmark. Note that for the Power PC, “Caches ON” means that both L1 Instruction and Data caches are enabled.

The main result of these experiments is the percentage of injected CEUs that result in observable program execution errors. Indeed, this figure combined with the measured register cross sections yields the predicted error rate for each studied application. For the selected benchmarks, the result-error component of the σ 's are fairly low from ~0.5 % to ~8 %. A rough estimate of the register usage is 20% for the matrix multiplication and the bubble sort programs, although the fraction of registers used is somewhat larger for the FFT application due to the use of floating point registers.

It is clear that the sequence-loss error is a very important upset error mode of the PPC. Compared with result errors, sequence losses are between 1.1 and 6 times more frequent for all cases except one: only in the Matrix benchmark with the caches OFF are result errors more frequent than the sequence losses. This is likely due to the number of critical registers (Program Counter (PC), Link Register (LR), Machine State Register (MSR), etc.) available in the PowerPC architecture and their relatively high duty cycle.

For the three benchmark programs, the number of detected errors when the L1 caches are OFF is significantly (around 50%) greater than when the caches are ON. Indeed, since the CEUs can only be injected in the internal registers and not in

the memory, it is expected that the caches ON results would have fewer result errors and sequence-loss errors. In this case, the operating frequency of the PPC runs many times faster (260 MHz instead of the 40 MHz memory bus speed). As the resolution of CEU occurrence instant (issued from an asynchronous interrupt generated from the Intel 87C52) equals 1 microsecond, the probability of missing sensitive instants will significantly increase. The fraction of errors that are sequence losses increases with the caches ON.

B. Radiation testing results

In order to qualify the PowerPC7400 under heavy ion beams and to evaluate the effectiveness of the CEU injection technique, a radiation testing campaign has been carried out 4-6 November 2001 at the Texas A&M cyclotron facility. For error rate estimation, the target of this study, the first step was to perform the register testing in which all the load/store registers were initialized with a pattern. The patterns used for 32 bits registers contained equal numbers of 0s and 1s (55AA00FF in hex), while for 64 bits registers this pattern was repeated.

In the static tests, only approximately half of the sensitive bits in the PowerPC7400 processor's register set were set to the test pattern. There are two types of bits left out intentionally. First, for the contents of the critical registers such as the Program Counter and the Machine State Register cannot be set to arbitrary patterns (since they vary and are critical to correct operation). Second, the AltiVec register (AVR) set was not measured because the current benchmarks do not take advantage of the AVRs. Planned for future testing is a matrix benchmark that uses the AVRs.

Table 3 summarizes the features of the beams used, the breakdown of errors by category (upsets and sequence loss), and the underlying SEU cross section derived from this standard static register test where the L1 caches have been disabled. This static test program involves only the registers accessed directly by PowerPC load and store instructions, excluding the AltiVec registers. Hence, only about half of the PPC internal registers were checked during the radiation testing. These partial device cross sections are given in the last column of the Table 3.

Note that the results show a low register upset sensitivity (saturation cross section), but also a low LET threshold. It can be concluded from the results given in Table 3 and represented in Fig. 2 (where the points are plotted at twice the cross section from the last column of Table 3), that the SEU threshold is below 1.24 MeV per mg/cm² and that a near saturation cross section of about 10⁻⁴ cm² per device is reached at an LET of only 10 MeV per mg/cm², consistent with the results presented in Ref. 8, where the authors applied static test strategies using a commercially available PowerPC7400 evaluation board (called Yellowknife).

C. Error rate estimation and comparison of predicted and measured error rates

The PowerPC7400 benchmark programs were exposed to

the set of beams while running the benchmark programs in both L1 caches ON and caches OFF configurations. High fluxes were used, often having magnitudes of the order of 10^5 particles/sec. Measured and predicted error rates (using equation 1) for each of the benchmark programs, are given in Tables 4 and 5 and represented in Figures 3 and 4.

As shown in Table 4 and Figure 3, the agreement is fairly good between predictions and measurements derived from the experiments performed with the benchmark programs where the L1 caches are disabled. For the L1 enabled, results shown in Table 5 and given in Figure 4, the predictions are not as close but they are still within an order of magnitude. Indeed, if statistical uncertainties are considered (see the two-sigma error bars given in Tables 3 and 4), predictions and measures may be in agreement. This achievement is more particularly remarkable according to the large number of sequence-loss errors, as stated before, cannot be simulated by the CEU injection approach in an accurate way. In addition, although a higher SEU sensitivity when enabling the caches was expected, the results show that the sensitivities of all of the three benchmark programs are approximately the same in both cache configurations. Such an unexpected result may come from the short cache duty cycle due to the high operating frequency of the PowerPC when enabling the caches (260 MHz), which makes the registers duty cycles remain approximately the same in both cache configurations.

Exception occurrence (illegal instruction, floating point unavailable, data alignment...) was observed many times during radiation testing but never because of CEU injection. Upsets causing exceptions may result either in tolerated faults or result errors or sequence-loss errors. The latter can result from errors in critical control registers (for instance in the Hardware Implementation-Dependent register 0 (HID0)) which are accessible to CEUs or in the processor's underlying blocks (such as the pipelines and state sequencers) which are inaccessible by means of the instruction set and thus by the CEU injection technique. Details of the sequence-loss errors caused by beam-induced exceptions for the three benchmark programs are given in Tables 6 and 7 for the caches OFF and ON, respectively.

The high rate of exception occurrence during radiation testing shows that a significant portion of sequence-loss errors might be recoverable using suitable exception handling routines. From the analysis of the results, it is observed that, with the caches OFF, the percentage of sequence-loss errors detectable by exception handlers could be 50% or more. With the L1 data caches ON, in the best case (for the bubble sort program), this percentage is only 20%. Note that, for a given program and cache configuration, the analysis of exception detection for all the beams shows that the ratio of sequence-loss errors detectable by exception is approximately the same for all the beams. Thus, it appears that the fraction of sequence-loss errors that arise from SEE-caused exceptions is not a function of LET.

IV. CONCLUSION AND FUTURE WORK

This paper presents the results of performing fault injection and radiation testing using the THESIC PowerPC daughterboard. Static register cross sections and the measured benchmark cross sections were obtained from radiation testing at Texas A&M cyclotron facility. For the fault injection experiments, the CEU technique was relatively easy to implement for the register set of Power PC, but CEUs could not be injected in the internal cache memories and the current time-granularity constraints may be skewing the types of the resultant errors.

Predicted error rates for benchmark programs were derived according to the CEU injection methodology. For both cache configurations, the predictions were in reasonably good agreement with the measurements despite the contribution of the bits inaccessible to CEU injection. Although some consistent under-prediction is evident when the number of inaccessible bits increases, i.e., in the caches ON cases. This successful result can be explained by the fact that although the speed of execution of the programs in caches ON and caches OFF modes is very different, the registers duty cycles remain the same. Thus, it is concluded that CEU experiments make predictions that agree with measurements, at least for the programs used in this study. These experiments bring new evidence of the effectiveness of the error rate prediction approach based on the CEU injection technique. Further studies that include fault injection into the caches are needed.

V. ACKNOWLEDGEMENTS

The authors would like to thank Farokh Irom of the Radiation Group of the Jet Propulsion Laboratories, California Institute of Technology, for his help during preparations for the radiation testing and Bogdan Nicolescu of the TIMA Laboratory Qualification Group for his help with and insights into the benchmark software.

REFERENCES

- [1] F. Bezerra, et al., "Commercial Processor Single Event Tests", 1997 *RADECS Conference Data Workshop Record*, pp. 41-46.
- [2] J. Carreira, H. Madeira, J.G. Silva, "Xception : A Technique for the Experimental Evaluation of Dependability in Modern Computers", *IEEE Transactions in Software Engineering*, Vol. 24, No. 2, Feb. 1988, pp. 125-136.
- [3] T.A. Delong, B.W. Johnson, J.A. Profeta, "A fault injection technique for VHDL behavioral-level models", *IEEE Design & Test of Computers*, 1996, pp. 24-33.
- [4] R. Velazco, S. Rezgui, R. Ecoffet, "Predicting Error Rate for Microprocessor-Based Digital Architectures through C.E.U. (Code Emulating Upsets) Injection", *IEEE Trans. Nucl. Sci.*, Vol. 47, No. 6, Dec. 2000, pp. 2405-2411.
- [5] S. Rezgui, R. Velazco, R. Ecoffet, S. Rodriguez, J.R. Mingo, "Estimating Error Rates in Processor -Based Architectures", *IEEE Trans. Nucl. Sci.*, Vol. 48, No. 5, Oct. 2001, pp. 1680-1687.
- [6] S. Rezgui, R. Velazco, R. Ecoffet, S. Rodriguez, "A New Methodology for the Simulation of Soft Errors on Microprocessors : A Case Study", Presented at MAPLD 2000 Military and Aerospace of Programmable Devices and Technologies, Laurel, Maryland (USA), Vol. 1, Session B, 26-28 September 2000, in press for J.S.R. (Journal of Space Rockets).
- [7] R. Velazco, P. Cheynet, A. Bofill, R. Ecoffet, "THESIC: A Testbed Suitable for the Qualification of Integrated Circuits Devoted to Operate

in Harsh Environment”, *IEEE European Test Workshop (ETW’98)*, Sitges, Spain, pp. 89-90, May 1998.

[8] G.M. Swift, S.M. Guertin, F.F. Farmanesh, F. Irom, D. Millward, “Single event Upset in the Power PC750 Microprocessor”, *IEEE Trans. Nucl. Sci.*, Vol 50, No. 6, pp. 1822-1827, December 2001.

Table 1: CEU Results with L1 Caches OFF

	Matrix (1000)	Bubble (10000)	F.F.T. (1000)
Tolerated faults	(90.8 ± 6.96)%	(98.0 ± 1.98)%	(94.9 ± 6.16)%
Result errors	(7.70 ± 1.75)%	(0.55 ± 0.14)%	(2.40 ± 0.98)%
Sequence loss	(1.50 ± 0.77)%	(1.46 ± 0.24)%	(2.70 ± 1.04)%
τ_{CEU}	(9.20 ± 1.92)%	(2.01 ± 0.57)%	(5.10 ± 1.42)%

Table 2: CEU Results with L1 Caches ON

	Matrix (9606)	Bubble (10646)	F.F.T. (3055)
Tolerated faults	(97.1 ± 2)%	(97.8 ± 1.91)%	(97.2 ± 3.56)%
Result errors	(0.87 ± 0.19)%	(0.34 ± 0.11)%	(0.73 ± 0.3)%
Sequence loss	(1.96 ± 0.28)%	(1.82 ± 0.26)%	(2.07 ± 0.5)%
τ_{CEU}	(2.83 ± 0.96)%	(2.16 ± 0.28)%	(2.80 ± 0.6)%

Table 3: Measured Static Register PPC G4 SEU Cross sections (Caches OFF, not including AVRs)

Heavy Ions	Effective LET (MeV/mg/cm ²)	Energy (GeV)	Range (µm)	Fluence (Particles)	Upsets	Sequence-loss errors	Cross section for tested registers (cm ² / device)
Neon	1.24	0.763	1525	5.00 10 ⁷	3	3	(1.2 ± 0.97) 10 ⁻⁷
Neon	2.28	0.348	403	2.00 10 ⁷	2	36	(1.9 ± 0.62) 10 ⁻⁶
Argon	4.04	1.48	948	9.99 10 ⁶	52	2	(5.4 ± 1.47) 10 ⁻⁶
Argon	9.3	0.442	149	2.00 10 ⁷	957	58	(5.07 ± 0.31) 10 ⁻⁵
Argon	13.2	0.215	58	5.00 10 ⁷	5210	292	(1.10 ± 0.03) 10 ⁻⁴
Krypton	15.8	2.668	491	9.99 10 ⁶	682	52	(7.34 ± 0.54) 10 ⁻⁵
Krypton	23.4	1.368	194	1.00 10 ⁷	1384	65	(1.45 ± 0.08) 10 ⁻⁴
Xenon	47.4	1.922	155	9.98 10 ⁵	175	10	(1.85 ± 0.27) 10 ⁻⁴

Table 4: Predicted and measured cross sections for the PPC – G4, Caches OFF

Heavy Ions	Effective LET (MeV/mg/cm ²)	Matrix Multiplication Cross-section (cm ² / device)		Bubble Sort Cross section (cm ² / device)		FFT Cross section (cm ² / device)	
		Measured	Predicted	Measured	Predicted	Measured	Predicted
Neon	2.28		(1.75 ± 1.32) 10 ⁻⁷	< 2 10 ⁻⁸	(3.8 ± 6) 10 ⁻⁸	(2 ± 4) 10 ⁻⁸	(7.95 ± 8.9) 10 ⁻⁸
Argon	4.04	< 10 ⁻⁷	(4.78 ± 3.15) 10 ⁻⁷	(6 ± 4.89) 10 ⁻⁷	(1.08 ± 1.4) 10 ⁻⁷	(3.8 ± 1.74) 10 ⁻⁷	(2.38 ± 2.1) 10 ⁻⁷
Argon	9.3	(4.96 ± 0.63) 10 ⁻⁶	(4.65 ± 0.68) 10 ⁻⁶	(3.06 ± 0.49) 10 ⁻⁶	(1.01 ± 0.31) 10 ⁻⁶	(5.17 ± 1.44) 10 ⁻⁶	(2.23 ± 0.46) 10 ⁻⁶
Krypton	15.8	(5.8 ± 1.52) 10 ⁻⁶	(6.76 ± 1.16) 10 ⁻⁶	(5.8 ± 1.52) 10 ⁻⁶	(1.47 ± 0.52) 10 ⁻⁶	(7.4 ± 1.72) 10 ⁻⁶	(3.25 ± 0.78) 10 ⁻⁶
Xenon	47.4	(5.8 ± 1.52) 10 ⁻⁶	(1.7 ± 0.58) 10 ⁻⁵	(6 ± 1.57) 10 ⁻⁶	(3.70 ± 2.65) 10 ⁻⁶	(8.41 ± 0.21) 10 ⁻⁶	(8.18 ± 3.94) 10 ⁻⁶

Table 5: Predicted and measured cross sections for the PPC – G4, Caches ON

Heavy Ions	Effective LET [MeV/mg/cm ²]	Matrix Multiplication Cross section (cm ² / device)		Bubble Sort Cross section (cm ² / device) (1000)		FFT Cross section (cm ² / device) (3055)	
		Measured	Predicted	Measured	Predicted	Measured	Predicted
Argon	4.04	(0.8 ± 0.8) 10 ⁻⁷	(1.53 ± 1.75) 10 ⁻⁷	(1.4 ± 1.06) 10 ⁻⁷	(1.16 ± 1.52) 10 ⁻⁷	(1.86 ± 0.38) 10 ⁻⁶	(1.51 ± 1.7) 10 ⁻⁷
Argon	9.3	(3 ± 1.55) 10 ⁻⁷	(1.43 ± 0.37) 10 ⁻⁶	(2.54 ± 0.45) 10 ⁻⁶	(1.09 ± 0.33) 10 ⁻⁶	(7.6 ± 2.46) 10 ⁻⁷	(1.41 ± 0.37) 10 ⁻⁶
Argon	13.2		(3.12 ± 0.33) 10 ⁻⁶		(2.37 ± 0.30) 10 ⁻⁶	(9.7 ± 0.77) 10 ⁻⁶	(3.08 ± 0.35) 10 ⁻⁶
Krypton	15.8	(4.8 ± 1.38) 10 ⁻⁶	(2.08 ± 0.64) 10 ⁻⁶	(5.8 ± 1.52) 10 ⁻⁶	(1.58 ± 0.56) 10 ⁻⁶	(8.9 ± 1.88) 10 ⁻⁶	(2.06 ± 0.64) 10 ⁻⁶
Krypton	23.4		(4.12 ± 0.9) 10 ⁻⁶		(3.13 ± 0.79) 10 ⁻⁶	(1.27 ± 0.22) 10 ⁻⁵	(4.06 ± 0.9) 10 ⁻⁶
Xenon	47.4	(7.7 ± 1.75) 10 ⁻⁶	(5.25 ± 3.2) 10 ⁻⁶	(5.8 ± 1.52) 10 ⁻⁶	(3.99 ± 2.83) 10 ⁻⁶	(6.07 ± 1.28) 10 ⁻⁶	(5.18 ± 3.22) 10 ⁻⁶

Table 6: Breakdown of caches OFF sequence-loss errors detected under Argon irradiation (LET = 9.3 MeV per mg/cm²)

Exceptions	Bubble	Matrix	FFT
SR: 0x100		2	
MC: 0x200		1	2
DSI: 0x300			
ISI: 0x400	2	1	
Align.: 0x600	2	1	
Ill. In: 0x700	14	17	8
UFP: 0x800	2	12	
Dec.: 0x900			
Trace: 0xD00			1
PM:0xF00	1		
TMI: 0x1700			
Total of sequence-loss errors detected by exceptions	21	34	11
Sequence-loss errors detected by THESIC watch dog	54	56	53

Table 7: Breakdown of caches ON sequence-loss errors detected under Argon irradiation (LET = 9.3 MeV per mg/cm²)

Exceptions	Bubble	Matrix	FFT
SR: 0x100			
MC: 0x200			1
DSI: 0x300			
ISI: 0x400			2
Align.: 0x600			
Ill. In: 0x700	17	1	
UFP: 0x800	7		
Dec.: 0x900			
Trace: 0xD00			
PM:0xF00			
TMI: 0x1700			
Total of sequence-loss errors detected by exceptions	24	1	3
Sequence-loss errors detected by THESIC watch dog	121	14	26

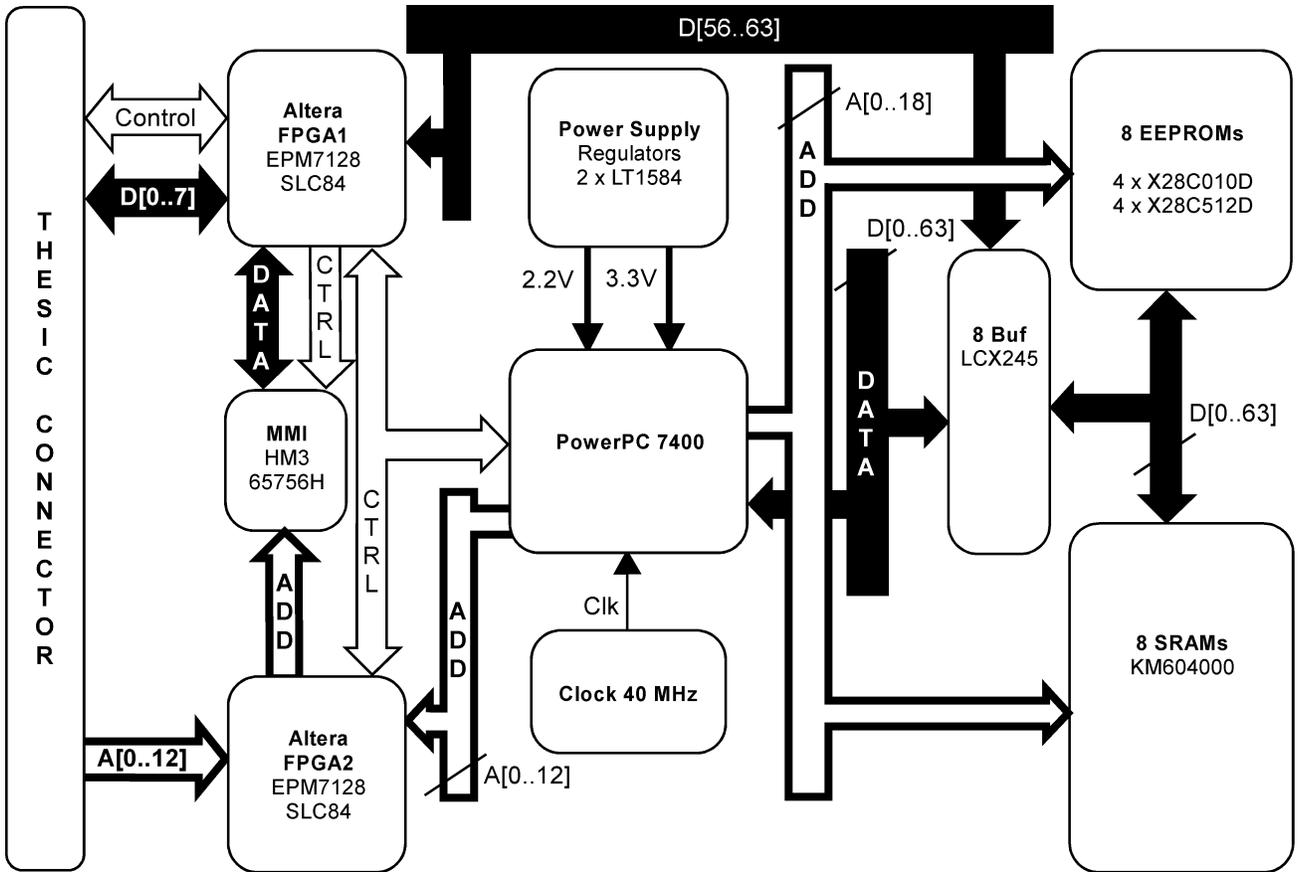


Figure 1: Block diagram of PPC 7400 THESIC daughterboard

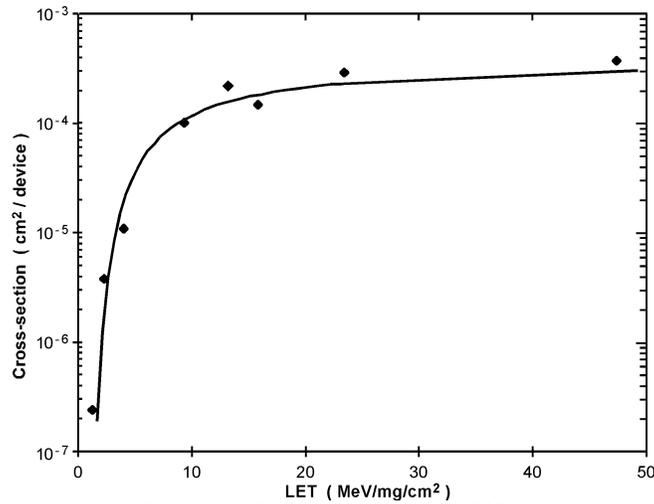


Figure 2: Measured SEU cross section of G4 PPC registers

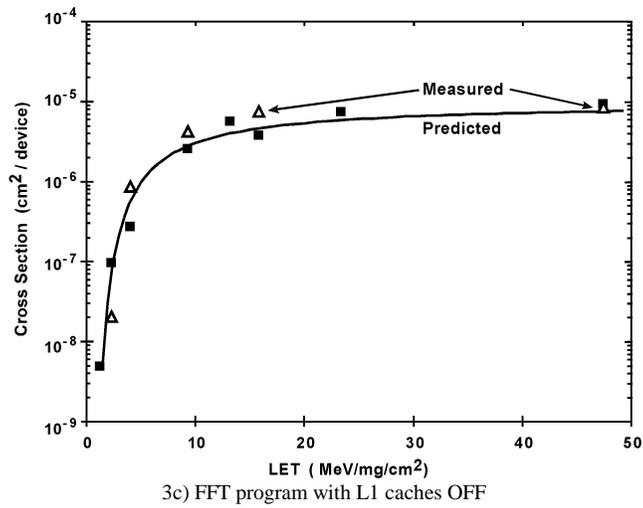
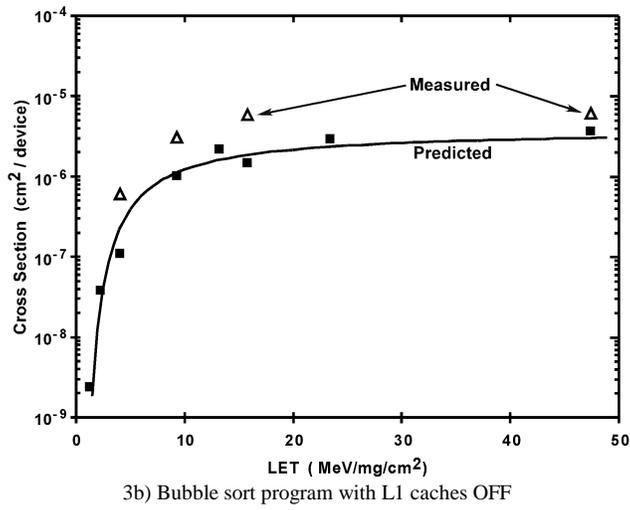
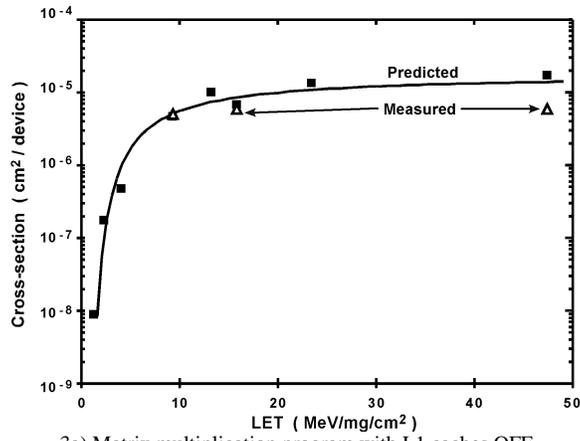
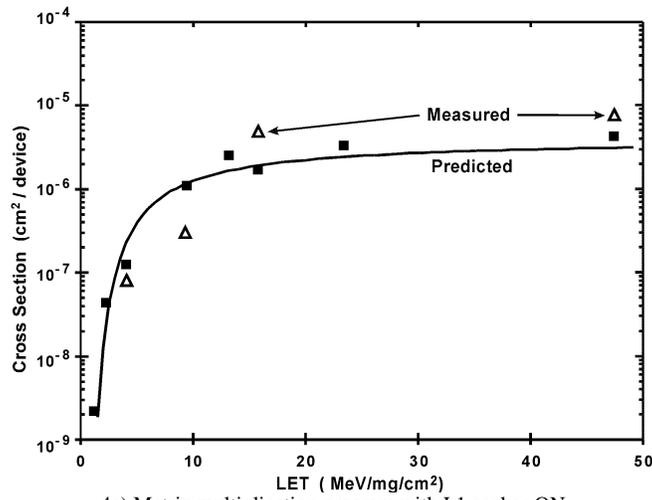
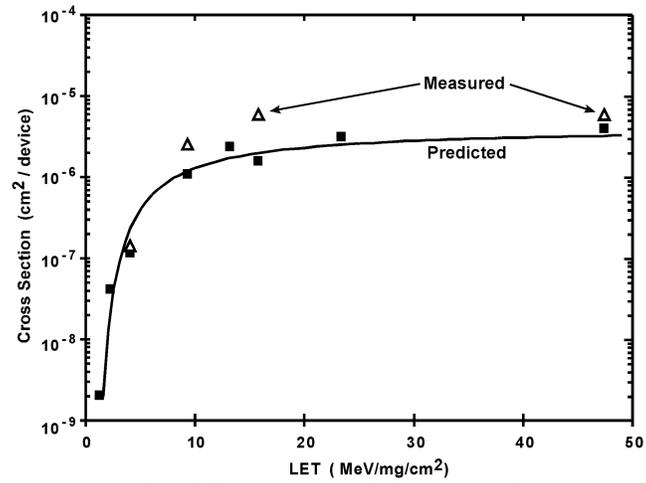


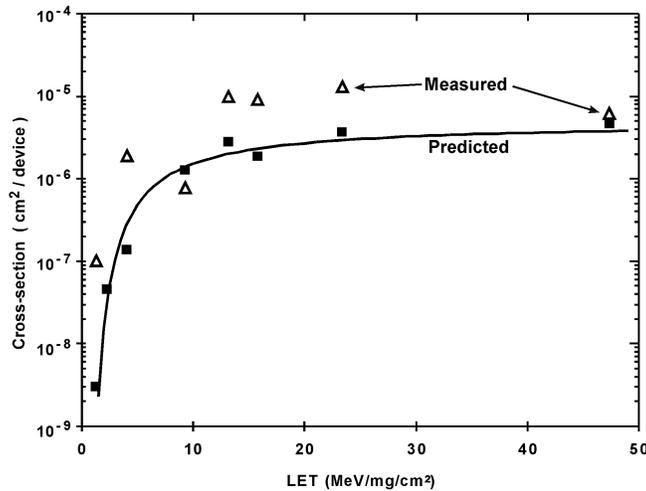
Figure 3: G4 PPC predicted and measured cross sections for SEU of the three benchmarks programs with L1 Caches OFF



4a) Matrix multiplication program with L1 caches ON



4b) Bubble sort program with L1 caches ON



4c) FFT program with L1 caches ON

Figure 4: G4 PPC predicted and measured cross sections for SEU of the three benchmarks programs with L1 Caches ON