Single-Event Effects in Low-Cost, Low-Power Microprocessors

Heather Quinn, Tom Fairbanks, Justin L. Tripp, George Duran, Beatrice Lopez

***Abstract*— ARMs and microcontrollers are low-cost, low-power microprocessors that are frequently used in embedded computing. While not immune to the naturally occurring radiation environment in space, these microprocessors can be worthwhile replacements for space-grade microprocessors for non-mission-critical computational tasks. In this paper results from radiation testing for several available ARMs and microcontrollers are presented.**

I. Introduction

Most spacecraft designers remain steadfastly devoted to space-grade microprocessors, because the processing tasks are considered critical to spacecraft success. The most commonly used radiation-hardened microprocessor in satellites that is currently being used is the BAE Systems RAD750 [1]. The Atmel LEON, which is based on the SPARC-V8, has also been used in recent space missions [2]. There are also newer space-grade microprocessor that have been developed. Aeroflex has developed the LEON3 and LEON4 radiation-hardened microprocessors based on the SPARC-V8 architecture [3]. BAE Systems is developing the RAD5510 and RAD5545 [4], which is based on the Freescale PPC. All of these offerings are designed to be immune to the natural radiation environment, but are more expensive and consume more power than many commercial-off-the-shelf (COTs) microprocessors.

This work has been authored by an employee of the Los Alamos National Security, LLC (LANS), operator of the Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396 with the U.S. Department of Energy. The U.S. Government has rights to use, reproduce, and distribute this information. The public may copy and use this information without charge, provided that this Notice and any statement of authorship are reproduced on all copies. Neither the Government nor LANS makes any warranty, express or implied, or assumes any liability or responsibility for the use of this information. The Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; therefore, the Laboratory as an institution does not endorse the viewpoint of a publication or guarantee its technical correctness. This paper is published under LA-UR-14-26104.

H. Quinn, T. Fairbanks, J. Tripp, G. Duran and B. Lopez are with Los Alamos National Laboratory, Los Alamos, NM, 87545, USA (e-mail: hquinn@lanl.gov).

We are particularly interested in ARM microprocessors and microcontrollers. These microprocessors provide low-power, low-cost alternatives for general-purpose computing. Some of the computation done on the space-grade microprocessors is for configuration, monitoring and background tasks that do not need high-end, radiation-hardened microprocessors. By partitioning computation tasks onto another microprocessor, the space-grade microprocessor can be devoted to mission critical processing.

A number of researchers have looked at radiation effects in COTs microprocessors, including [5]–[8]. Recent publications have studied more modern microprocessors [9], [10] with reduced feature sizes and multiple processing cores. The authors have also previously published results for the NXP LPC2148FBD64 and the ST Microelectronics STM32F417IG [11]. Both the NXP LPC2148FBD64 and the ST Microelectronics STM32F417IG exhibited problems with high-current events and single-event latchup, so testing of other microprocessors was completed in 2013 and 2014.

The microprocessors that we are interested in must be capable of inputting both digital and analog data, outputting digital output and operating without external memory or power converters. In this manner, the microprocessor would be capable of being interfaced to state-of-health monitors so that the information can be converted to messages that can be transmitted to the ground station. Given the complexity of these microprocessors, we expect the components to be susceptible to a number of single-event effects (SEEs), including single-event upsets (SEUs), single-event transients (SETs), single-event functional interrupts (SEFIs), single-event latchup (SEL) and high-current events. In this paper, we will present results on SEE failure modes for several microprocessors.

This paper is organized as follows. The discussion of the components tested and their architectures is presented in Section II. Test setup is presented in Section III. SEE test results are presented in Section IV and a discussion of software tests is presented in Section V.

U.S. Government work not protected by U.S. copyright

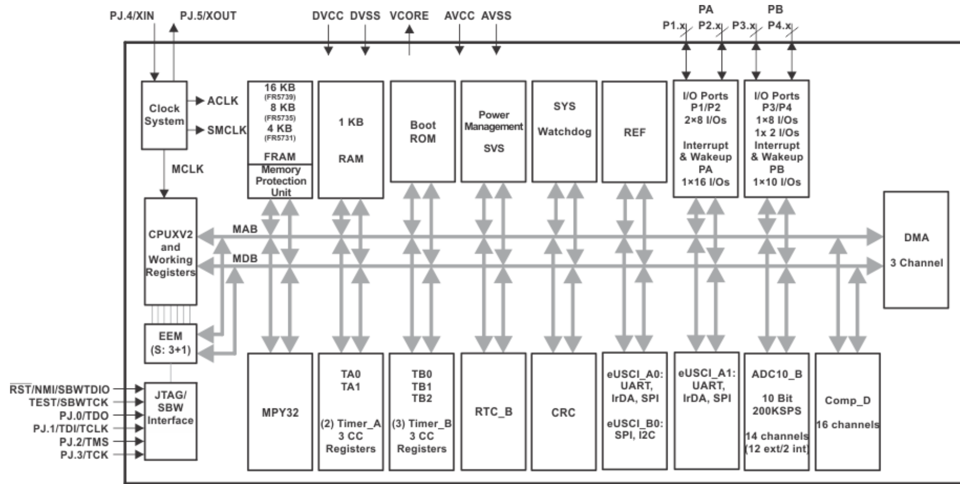

Fig. 1. Functional diagram of the FeRAM-based MSP430

TABLE I
LISTING OF POTENTIAL MICROPROCESSOR CANDIDATES.

Manufacturer	Part	Processor Type	Clock Speed	Non-volatile Memory	On-Chip SRAM	ADC
Texas Instruments	MSP430	16-bit Microcontroller	16 MHz	120 KB Flash	4 KB	12-bit
Texas Instruments	MSP430	16-bit Microcontroller	24 MHz	16 KB FeRAM	1 KB	10-bit
Texas Instruments	Hercules	Dual redundant ARM Cortex-R4	80 MHz	384 KB Flash	32 KB	10/12-bit
Texas Instruments	Stellaris	ARM Cortex-M4F	80 MHz	256 KB Flash	32 KB	12-bit
Texas Instruments	Tiva	ARM Cortex-M4	80 MHz	64 KB Flash	24 KB	12-bit
Xilinx	Zynq	ARM Cortex-A9 MPCore	1 GHz	Off-chip	256 KB	12-bit

II. COMPONENTS ARCHITECTURES

In this paper, we will be presenting test results from the components listed in Tab I. The architectures for the microprocessors that we are interested are unlike standard general-purpose microprocessors. As these microprocessors are expected to work in embedded computing scenarios all of these components have internal analog-to-digital converters (ADCs), digital-to-analog converters (DACs), non-volatile memory and a number of peripherals. Unlike traditional microprocessors, many of these components do not have a cache for storing instructions or data. Most have scratchpad memory, such as on-chip SRAM, for storing data and for some instructions. In this section, we will provide a highlight of the MSP430, ARM and Hercules architectures.

Fig 1 shows a block diagram of the FeRAM-based MSP430 (MSP430FR5739) [12]. The architecture for this component is similar to the Flash-based MSP430 (MSP430F2619) that was tested, albeit the difference in non-volatile memory. Both MSP430 components have a 16-bit RISC-based processor, which executes register-to-register operations in a single clock cycle. These components use non-volatile Flash memory and on-chip SRAM for storing and executing software. While reading and writing to non-volatile memory is available during execution, in practice writing to Flash is too slow for normal execution, so all volatile data should be stored in SRAM on the Flash-based MSP430. For the FeRAM-based MSP430, reading and writing to FeRAM is practical, so the program can chose to put the stack and data into the FeRAM.

Fig 2 shows a block diagram of the Tiva, which is a 32-bit Cortex-M4 ARM [13]. While both ARMs, the Zynq architecture is not only a different version of ARM it also integrates to a reconfigurable fabric and is a dual-core architecture, as shown in Fig 3 [14]. Both ARMs are based on a Harvard architecture, which has separate buses for instructions and data. The Tiva has SRAM, flash, EEPROM and ROM available for instructions and data. The internal ROM is used for storing Tiva-based instructions, called TivaWare. The Zynq processing core includes both data and instruction L1 caches and a L2 cache. The Zynq has on-chip SRAM (OCM), as well as interfaces to external dynamic random access memory (DRAM).

Fig 4 shows a block diagram of the Hercules [15].

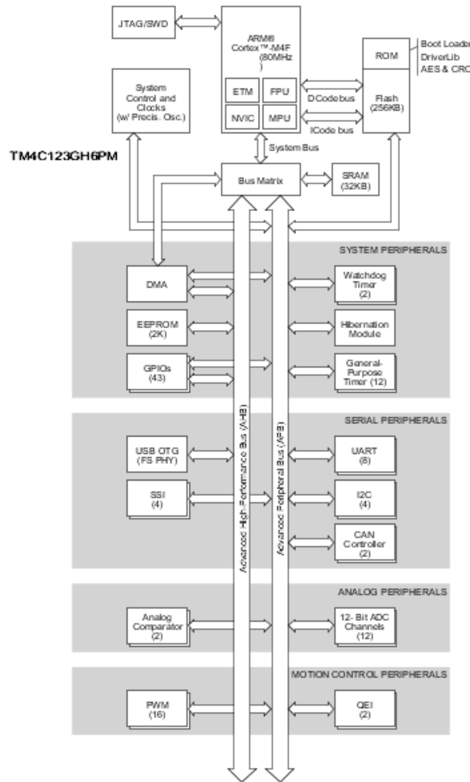

Fig. 2. Functional diagram of the Tiva Cortex-M4 ARM

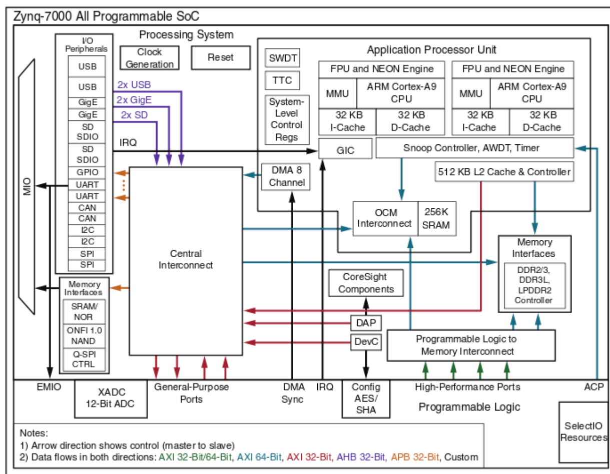

Fig. 3. Functional diagram of the Zynq Cortex-A9 ARM

The Hercules has two Cortex-R4 ARM processors that operate in lock step. The processors are deliberately staggered in execution such that one processor buffers execution for two clock cycles on input and the other one buffers execution for two clock cycles on output. If the outputs do not match, an error flag is raised. The component has on-chip flash and SRAM that are both protected with an error-correcting code (ECC). It should be noted that the two cores share the same SRAM for execution, so radiation-induced errors that defeat ECC would affect both cores.

III. Components and Test Setups

We have conducted a total of seven tests on these components over the past two years: two neutron tests, four heavy ion tests, a single dose rate test and a single total ionizing dose test. A list of tests completed and presented in this paper is shown in Tab II. The Stellaris was replaced by the Tiva component at the end of 2013 and heavy ion testing was not completed on the Stellaris before we switched components. Evaluation of the Zynq started in late 2013 and we were unable to test the component further since the initial neutron test.

Neutron testing was completed at the Los Alamos Neutron Science Center (LANSCE) Irradiation of Chips and Electronics (ICE) House II flight path. A picture of the test setup at LANSCE is shown in Fig 5. Heavy ion testing was completed at the Berkeley Accelerator Space Effects (BASE) facility at Lawrence Berkeley National Laboratory using the 10 MeV heavy ion cocktail. A picture of the Tiva test setup at BASE are shown in Fig 6. All of the components were tested at nominal temperature, nominal voltages and normal incidence.

All of the initial hardware test fixtures leveraged commercially available evaluation boards connected to a host computer. Only some of these boards allow for the component to be independently biased so that SEL conditions can be monitored. Custom test cards were designed for both MSP430 components and the Tiva after initial testing was completed. One of the custom boards is shown in Fig 6. The custom test cards allow for independent biasing, as well as for controlling peripheral components and access to general purpose pins.

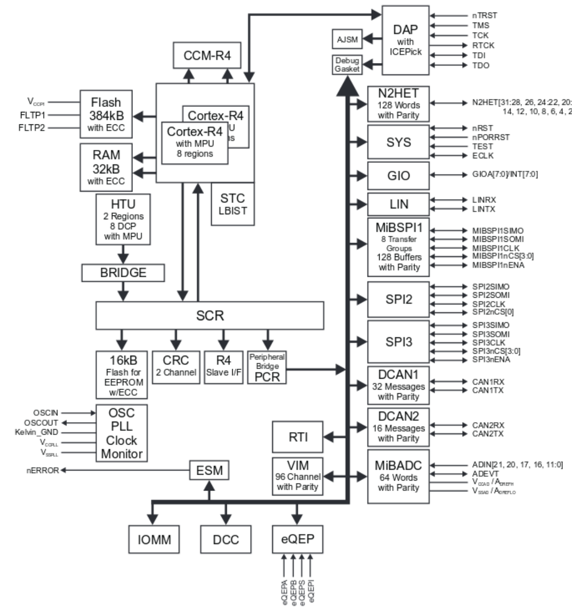

Fig. 4. Functional diagram of the Hercules

TABLE II
TESTS COMPLETED FOR CANDIDATE COMPONENTS

Part	SEL	SEU/SET	SEFI	ADC
Flash-based MSP430	heavy ion, neutron	heavy ion, neutron	heavy ion, neutron	heavy ion, neutron
FeRAM-based MSP430	heavy ion, neutron	heavy ion, neutron	heavy ion, neutron	heavy ion, neutron
Hercules	heavy ion, neutron	heavy ion, neutron	heavy ion, neutron	none
Stellaris	none	neutron	neutron	none
Tiva	heavy ion	heavy ion, neutron	heavy ion, neutron	none
Zynq	none	neutron	neutron	none

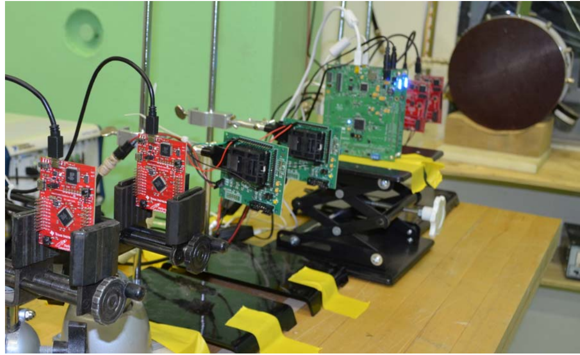

Fig. 5. Test setup at LANSCE with Stellaris, Flash-based MSP430, Hercules and FeRAM-based MSP430

To complete the hardware test setup, each board is interfaced to the host computer through the Universal Asynchronous Receiver/Transmitter (UART) and the Joint Test Action Group (JTAG) boundary scan port. While we have previously used the JTAG port to probe memory structures in components [16], for these tests the JTAG only loads the program into non-volatile memory when necessary. In Fig 6 two types of JTAG controllers are shown. The MSP430s use the gray MSP-FET430UIF JTAG box in the top of the figure, which is shown unconnected in Fig 6. The Tiva uses the Stellaris-based JTAG controller that is on the Stellaris Launchpad evaluation board. The two boards are connected using fly wires at the bottom of the image. Communication between the host computer and the boards is done through the UART using a terminal program. To minimize data movement on the UART, the software is instrumented to communicate only error conditions.

A number of different software codes have been used to test these components. Instead of using assembly codes, which have been used previously to test microprocessors [17], the test code was written in C. Initial tests focused on software codes that instrument the memory and basic processing. To instrument these memory structures, the software codes fill an array in the memory with known values. These values are checked

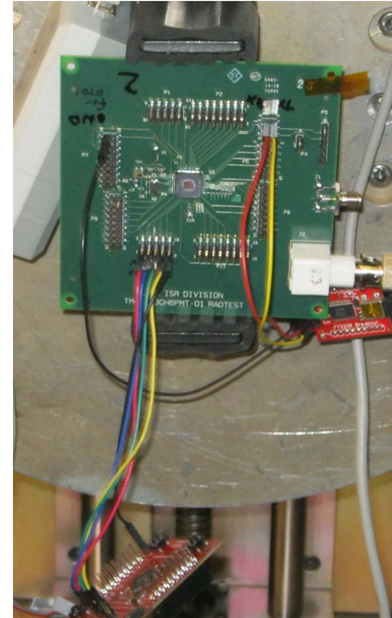

Fig. 6. Tiva test setup at BASE

for correctness while completing simple mathematical computations. The software instrumentation determines when the memory structures are affected by SEUs and whether computational failures occur.

Because some memory layouts have pattern-dependent SEUs, we use four different patterns:

- **Counter:** The memory array is filled with values from an incrementing counter starting at zero.
- **Inverted Counter:** The memory array is filled with values from a decrementing counter starting at the largest integer value.
- **Checkerboard:** The memory array is filled with A's and 5's in an alternating pattern.
- **Random:** The memory array is filled with values from an linear feedback shift array (LFSR) for creating predictable, pseudo-random numbers. This pattern represents real data.

These patterns highlight eccentricities with the layout, such as well sharing. When layout eccentricities exist,

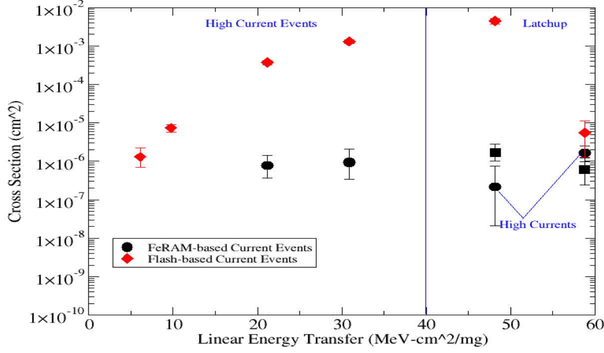

Fig. 7. High Current SEFI and SEL Cross Sections for the both MSP430s

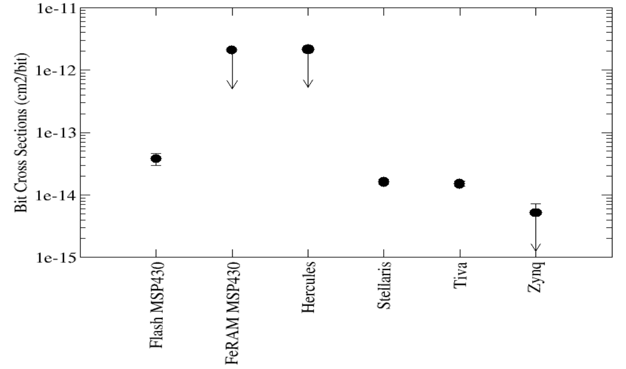

Fig. 8. SEU Cross Sections in Neutrons

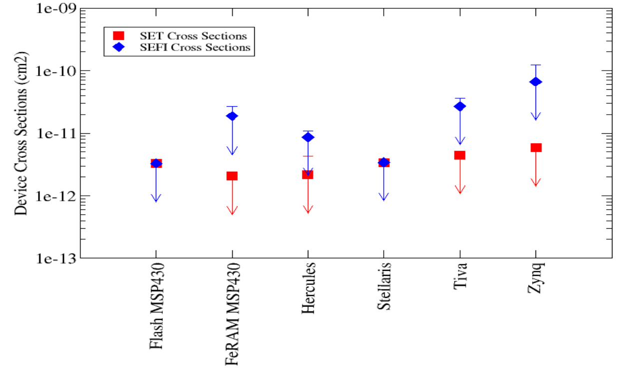

Fig. 9. SET and SEFI Cross Sections in Neutrons

there will be a pattern dependence to SEU cross sections.

In later tests we focused on more complex software test codes to get an understanding of how SEU and SET sensitivities translate into software failures. For these tests we focused on two software codes: CoreMark and Advanced Encryption Standard (AES). CoreMark is part of the Embedded Microprocessor Benchmark Consortium (EEMBC) benchmark, which is a set of benchmarks that have been designed specifically for embedded computing. Unlike other types of benchmarks, such as Linpack, CoreMark was designed to run on microcontrollers, which often have a limited amount of available memory [18]. A 128-bit AES code was also implemented.

It should be noted that the complexity of these components make them difficult to test with high-LET heavy ions. These problems increased as we started to look at more complex software codes. In an attempt to mitigate these problems, we also tested watchdogs and hard resets on some of the components in an attempt to stabilize the test fixture in heavy ion testing.

IV. Test Results

In this section we will cover the basic SEE characterization for each of the components. The results are separated by phenomenology. These results show that these components are sensitive to a number of different SEE phenomenons. It should be noted that the effect of many of the SEEs caused the component to be reset or to need resetting. Because of these issues, observability of the SEU, SET and software at higher LETs can be affected. We also modified the test fixture to measure the amount of time the component was operating when the beam was on so that we could adjust the fluence for observation time.

A. SEL, High-Current SEFIs and Destructive Events

As listed in Tab II, four components were tested for SEL and high-current events. In this paper, we define high-current events to be non-destructive events where the current is raised slightly. We define SEL as non-destructive and destructive events where the current is very high or limited only by the power supply. Both MSP430 components are sensitive to high-current events and SEL in heavy ion, although the neutron cross sections for both components are too small to measure. The Hercules and Tiva are not sensitive to SEL.

Fig 7 shows the high current and SEL cross sections for both MSP430 components in heavy ions. High-current events dominate the lower LET portion of the graph, but at the highest LET ions the events switch to SEL. Both components are nearly immune to SEL and only have issues at high LETs. In both cases, the component will current limit based on the power supply settings and in many cases is still functional during these events. The destructive events only occurred when the components were allowed to current limit at 1A for several minutes while irradiating the parts with Xenon. When the components are allowed to current limit at 0.1A for several minutes while irradiating the parts with Xenon, the component is able to recover from the effect

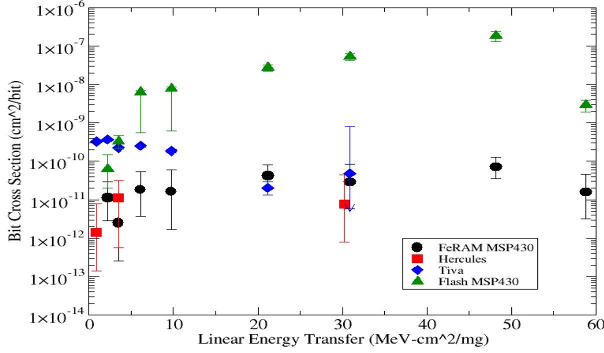

Fig. 10. SEU Cross Sections in Heavy Ions

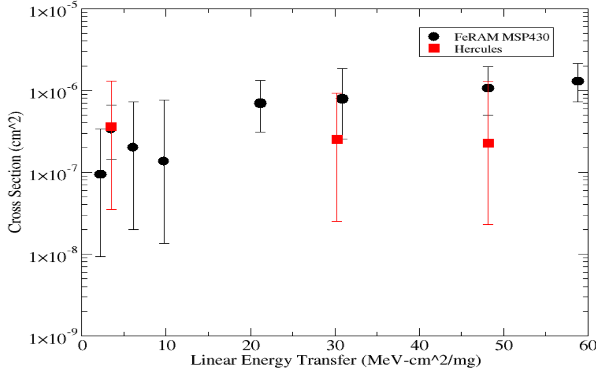

Fig. 11. SET Cross Sections in Heavy Ions

with a power cycle, so the potential issue is likely sustaining the higher current for that length of time.

The Tiva did not exhibit high-current events or latchup, but did have a destructive failure in heavy ions at 30.86 $\frac{MeV-cm^2}{mg}$. Due to the lack of preceding high current condition, we are uncertain of what happened to the component. After the event, the component's current consumption was near zero and it was not possible to connect to the component through JTAG to reprogram the component. The cross section for this event is $2.15 \times 10^{-7} cm^2$ with a 95% confidence interval of $(2.15 \times 10^{-8} cm^2, 7.73 \times 10^{-7} cm^2)$.

B. SEU, Multiple-Cell Upsets and SETs

The neutron cross sections for SEUs are shown in Fig 8 and heavy ion cross sections are shown in Fig 10. The only component with a traditional SEU response was the Flash-based MSP430. The FeRAM-based MSP430 exhibited some SEUs in the FeRAM, but sporadically and with a sensitivity almost seven orders of magnitude smaller than SRAM structures. The Tiva also had an unusually low sensitivity to SEUs likely due to problems with observability issues. The Hercules had three SEUs for which one definitively looks like a multiple-bit upset (MBU).

TABLE III
MCU PERCENTAGES IN NEUTRON WITH 95% CONFIDENCE INTERVALS

Part	Pattern	MCU%
Tiva	Checkerboard Left	6.78% (1.45%, 26.67%)
	Checkerboard Right	11.33% (4.15%, 29.32%)
	Counter	11.76% (3.60%, 36.49%)
	LFSR	7.79% (1.45%, 37.55%)
Stellaris	Checkerboard Right	25.77% (13.86%, 47.57%)
	Counter	16.87% (5.69%, 48.12)%
	LFSR	90.48% (6.85%, 6.85%)

The Flash-based MSP430, Stellaris, Tiva and Zynq have large blocks of SRAM and experience SEUs during testing that affect computation. We were able to test all of these components with our four patterns during neutron testing and did not observe much difference in bit cross sections. The cross sections for all of these components is similar to many other SRAM technologies. Furthermore, the SRAM structures in these microprocessors are smaller in comparison to many other microprocessors. Therefore, we believe the deployed SEU rate for these components should be reasonably low. If lower SEU rates are desired, masking SEUs through software mitigation is also possible [16].

We are particularly interested in the breakdown of SEUs into single-bit upsets (SBUs), multiple-cell upsets (MCUs) or multiple-bit upsets (MBUs). In this paper, we will define MCUs as the case when an SEU affects multiple bits and MBUs as an MCU that is confined to a word. As MBUs in microprocessors could affect error-correcting codes applied to SRAMs, it is important to determine whether the components will be affected by MBUs or MCUs. One of the advantages of testing in neutron is being able to explore MCUs and MBUs at very low fluxes, such that the probability of constructing MCUs through coincident single-bit upsets is low.

The Stellaris and Tiva components were the most sensitive to MCUs and MBUs. There were interesting trends for both components, especially as they compare to each other. The Stellaris is an older Texas Instruments component that is fabricated in a larger feature size than the newer Tiva. Despite the larger feature size, the Stellaris MCU rates are higher than the Tiva, especially in the LFSR pattern. Both components are sensitive to MCUs but not MBUs, except in the LFSR pattern. All of the SEUs that occur in the LFSR pattern for both components are MBUs. The MCU percentages for this pattern show the difference between MCUs that span

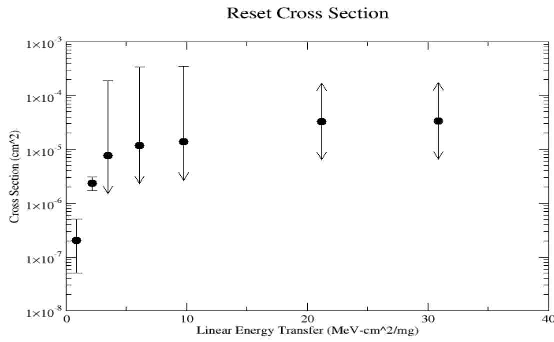

Fig. 12. Tiva Reset SEFI Cross Sections in Heavy Ions

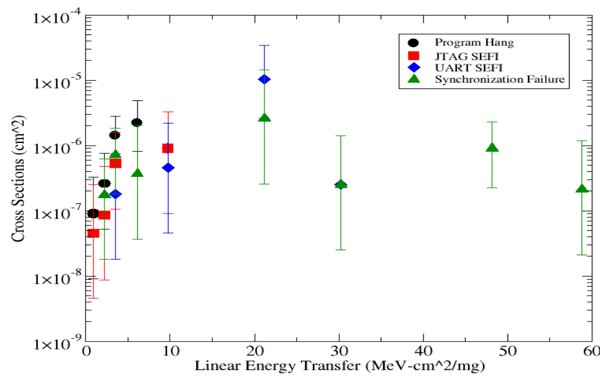

Fig. 13. Hercules SEFI Cross Sections in Heavy Ions

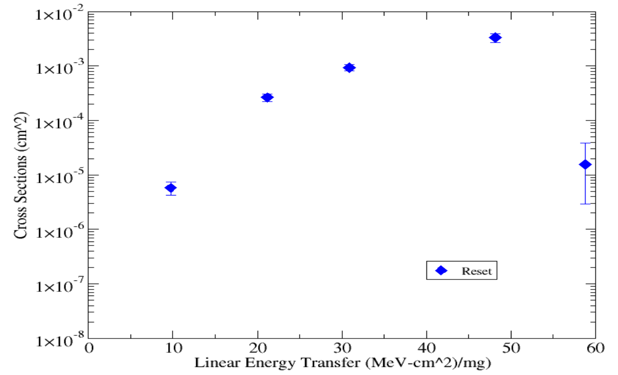

Fig. 14. Flash-based MSP430 SEFI Cross Sections in Heavy Ions

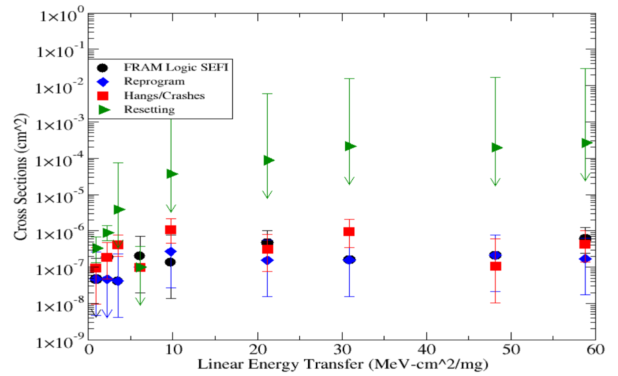

Fig. 15. FeRAM-based MSP430 SEFI Cross Sections in Heavy Ions

multiple words versus MBUs that affect only one word.

As shown in Fig 11 two of the components — the Hercules and the FeRAM-based MSP430 — exhibited SETs. These two components exhibited very few SEUs, which seems to be common amongst other processing elements with SETs. In each case, the SET caused an error in computation that did not coincide with an SEU. For the other components, computation errors were coincident with SEUs that corrupted the computation.

C. SEFIs

Many of the components were sensitive to a number of different SEFI modes. Figs 12– 15 show the heavy ion cross sections for the components. Neutron cross sections are shown in Fig 9. The types of SEFIs can be roughly categorized as such:

- **Resets:** Programs will self-reset in the beam, even without the use of watchdogs.
- **Program Hanging/Crashing:** The software execution halts and is unable to restart without a power cycle or external reset.
- **Peripheral Failures:** Memory-mapped functionality is altered by radiation or the peripheral functionally fails until the component is power cycled or reset.
- **Reprogram:** Component is functionally interrupted and the only the intervention that restores functionality is reloading the program into non-volatile memory.

Many components are sensitive to a number or all of these SEFI modes.

The most common SEFI mode was the Reset SEFI. The Tiva and both MSP430s are sensitive to Reset SEFIs. It is possible that this SEFI mode is caused by power-on-reset circuitry in the components. For these components, the sensitivity is great enough that testing in heavy ions is difficult. For example, the Reset SEFI for the FeRAM-bsaed MSP430 is $2.7 \times 10^{-4} cm^2$ in the saturation region in heavy ions. Even if the beam was running at $1 \times 10^3 \frac{ions}{cm^2}$, the Reset SEFI would occur every 3.7 seconds on average. Therefore, getting usable computation completed with high LET ions with these components can be challenging.

The Hercules and FeRAM-based MSP430 are both sensitive to program hangs or crashes in heavy ions and neutrons. The Zynq was sensitive to this SEFI mode in neutrons. These SEFIs can also be difficult to manage in heavy ion testing, because the part will need to be power cycled to recover functionality.

The Hercules was sensitive to SEFI modes that affected both peripheral ports that we were using to communicate to the component – the JTAG and the UART. In many of these components, peripherals like the UART have memory-mapped functionality, where the peripherals are configured using a number of registers. For the UART, these registers would set the baud rate and the communication settings. In the Hercules we could detect SEUs in the UART that would change the timing and functionality of the communication to the host. We did not see similar behavior in the other components, even though the other components often use the same amount of memory to configure the peripherals.

Some of the components were sensitive to failure modes that were specific to the hardware or only occurred on that component. The Zynq was susceptible a SEFI failure mode that caused multiple SEUs across several words that could only be cleared with a power cycle. The FeRAM-based MSP430 also had a FeRAM-specific SEFI mode that is similar to the DRAM SEFI mode, where SETs in peripheral logic causes SEU-like errors to manifest. Many of the FeRAM Logic SEFIs were transient, but occasionally the component needed power cycling to clear the SEFI. Synchronization failures on the Hercules were also categorized as a SEFI mode, as we currently do not know if there is a software dependence on these failures.

While many of these SEFI modes make testing in heavy ions challenging, we believe that some of the SEFI modes could possibly be mitigated through software mitigation. The synchronization failures on the Hercules component could possibly be mitigated with a high-performance computing software practice called "checkpoint/restart." In checkpoint/restart, the state of the computation is stored periodically. In this case, if the synchronization error line indicates that the processors are no longer in synch, the computation could be restarted using the state in the checkpoint. We believe that peripheral SEFI modes could be mitigated through scrubbing the peripheral registers at regular intervals. In this manner, any SEUs that affect the peripherals would be overwritten on a regular basis, which will restore correct behavior to the peripheral.

V. Tests on Software and Watchdogs

We are interested in determining how the SEE cross sections affect software codes that use realistic software structures and do realistic computation on the component. We used CoreMark on the Tiva and the flash-based MSP430 and AES on the FeRAM-based MSP430 for this purpose. Because we were having problems stabilizing the components at high LETs, we also tested these codes with watchdogs and hard resets. In this section, we will present these results.

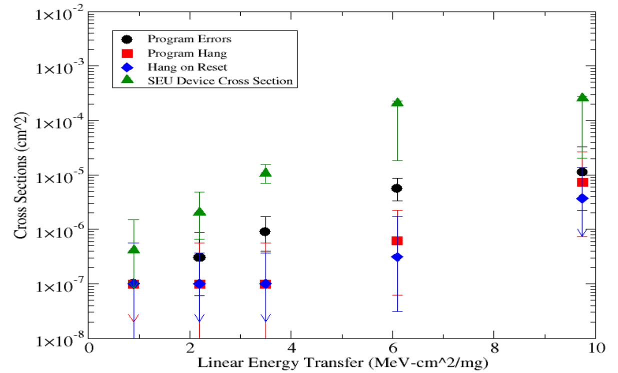

Fig. 16. CoreMark Cross Sections in Heavy Ions for the Flash-based MSP430

CoreMark was reasonably stable on the Flash-based MSP430 at low LETs. Only 1-7% of the executions of the code had errors. The cross section for program errors in shown in Fig 16. This figure shows that the program error cross section is between 4-25 times smaller than the device cross section for SEUs, even though nearly all of the SRAM was used for program execution. At low LETs, the SEFI modes are reasonably low, such that it is possible to test for several minutes without hanging.

Unlike the other software codes, CoreMark on the Flash-based MSP430 had problems with the Reset SEFI. Because the software barely fit in the available memory, we instrumented the code with a reset at the end of each loop, instead of a watchdog. This approach was not successful, as two of the eight program crashes occurred before the hard reset. The other six crashes occurred while executing the reset routine.

CoreMark was less stable on the Tiva. At 0.89 $\frac{MeV-cm^2}{mg}$ the software could only execute for 60-90 seconds without crashing. For this ion, the cross section for execution errors is $2.22 \times 10^{-7} cm^2$ with 95% confidence intervals of $(2.22 \times 10^{-8} cm^2, 7.99 \times 10^{-7} cm^2)$; the cross section for program hangs is $3.33 \times 10^{-7} cm^2$ with 95% confidence intervals of $(6.66 \times 10^{-8} cm^2, 9.77 \times 10^{-7} cm^2)$. At 2.19 $\frac{MeV-cm^2}{mg}$ the software this time period decreased to 25-35 seconds. For this ion, the cross section for execution errors is $8.06 \times 10^{-6} cm^2$ with 95% confidence intervals of $(4.65 \times 10^{-8} cm^2, 2.98 \times 10^{-5} cm^2)$; the cross section for program hangs is $8.06 \times 10^{-7} cm^2$ with 95% confidence intervals of $(8.08 \times 10^{-7} cm^2, 4.52 \times 10^{-5} cm^2)$. Software watchdogs were not helpful, because the watchdog would not execute properly after the program crashed.

A 128-bit AES code was implemented on the FeRAM-based MSP430. The initial implementation of the code

was very unstable, because we were storing the encryption and decryption keys in SRAM. We moved these keys to FeRAM and did not have any other software failures even at the highest LETs. At high LETs the component is affected by the reset SEFI, but not by SEUs or SETs. While this component has many SEFI modes that could make it a challenge to use in spacecraft, the ability to move the stack and data portions of the software into FeRAM reduces problems with SRAM-based SEUs in software.

One conclusion from these tests is that methods for testing software in heavy ions needs to be explored. Many of the software tests were unstable in heavy ions. In particular the SEFI rates at higher LETs makes it difficult to finish computations. Some of the methods needed to reset the component take many seconds-minutes, which is not recommended during testing unless the beam can be "paused" during the test. It is also possible that completing the entire sweep of ions is not reasonable when testing software. Similar testing in FPGAs is often handled using the JPL "Tri-flux Test" methodology, where a relatively light ion is used to test the component at three fluxes that are meant to mimic rates seen in space during normal and flare conditions.

VI. Conclusions

In this paper we presented SEE results for six microprocessors. Many of these components were sensitive to a number of different SEE modes. The Flash-based MSP430 and FeRAM-based MSP430 were both sensitive to high-current events and SEL, although destructive events were rare. These results also show that most of the components are sensitive to SEUs and SEFIs, but that fewer are sensitive to SETs. Because these microprocessors have less SRAM that traditional microprocessors, we believe the deployed SEE rates for these components could be low and can be masked through the use of software modifications.

References

[1] BAE Systems, on Web at http://www.baesystems.com/.

[2] H. Quinn, P. Graham, K. Morgan, Z. Baker, M. Caffrey, D. Smith, M. Wirthlin, and R. Bell, "Flight experience of the Xilinx Virtex-4," *IEEE Transactions on Nuclear Science*, Vol. 60, No. 4, pp. 2682–2690, 2013.

[3] J. Gaisler, "A portable and fault-tolerant microprocessor based on the sparc v8 architecture." in *DSN*, 2002, pp. 409–415.

[4] J. Marshall, "Spacewire satellite usage," BAE Systems, Tech. Rep., 2013, Available on web at:http://www.dtic.mil/dtic/tr/fulltext/u2/a579510.pdf, last accessed 2/2014.

[5] D. Hiemstra, A. Baril, and M. Dettwiler, "Single event upset characterization of the Pentium (R) MMX and Pentium (R) II microprocessors using proton irradiation," *IEEE Transactions on Nuclear Science*, Vol. 46, No. 6, pp. 1453–1460, December 2009.

[6] D. Hiemstra and A. Baril, "Single event upset characterization of the Pentium (R) MMX and Celeron (R) microprocessors using proton irradiation," in *IEEE Radiation Effects Data Workshop*, 2000, pp. 39–44.

[7] D. Hiemstra, S. Yu, and M. Pop, "Single event upset characterization of the Pentium (R) MMX and Low Power Pentium (R) MMX microprocessors using proton irradiation," in *IEEE Transactions on Nuclear Science*, 2001, pp. 32–37.

[8] G. Swift, F. Farmanesh, F. Guertin, F. Irom, and D. Millward, "Single-event upset in the PowerPC750 microprocessor," *IEEE Transactions on Nuclear Science*, Vol. 48, No. 6, pp. 1822–1827, December 2001.

[9] M. Cabanas-Holmen, E. Cannon, A. Kleinosowski, J. Ballast, J. Killens, and J. Socha, "Clock and reset transients in a 90 nm RHBD single-core Tilera processor," *IEEE Transactions on Nuclear Science*, Vol. 56, No. 6, pp. 3505–3510, 2009.

[10] X. Li and J.-L. Gaudiot, "Tolerating radiation-induced transient faults in modern processors," *International Journal of Parallel Programming*, Vol. 38, No. 2, pp. 85–116, 2010.

[11] T. Fairbanks, H. Quinn, J. Tripp, J. Michel, A. Warniment, and N. Dallmann, "Compendium of TID, neutron, proton and heavy ion testing of satellite electronics for Los Alamos National Laboratory," in *2013 IEEE Radiation Effects Data Workshop (REDW)*, 2013, 10.1109/REDW.2013.6658191.

[12] "MSP430FR573x mixed-signal microcontrollers," Texas Instruments, Tech. Rep. SLAS639J, 2014, Available on web at: http://www.ti.com/lit/ds/symlink/msp430fr5739.pdf, last accessed 7/2014.

[13] "Tiva TM4C123GH6PM microcontroller data sheet," Texas Instruments, Tech. Rep. SPMS376E, 2014, Available on web at: http://www.ti.com/lit/ds/symlink/tm4c123gh6pm.pdf, last accessed 7/2014.

[14] "Zynq-7000 all programmable SoC overview," Xilinx, Tech. Rep. DS190 (v1.6), 2013, Available on web at: http://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf, last accessed 7/2014.

[15] "Rm42l432 16- and 32-bit risc flash microcontroller," Texas Instruments, Tech. Rep. SPNS180A, 2013, Available on web at: http://www.ti.com/lit/ds/spns180a/spns180a.pdf, last accessed 7/2014.

[16] H. Quinn, T. Fairbanks, J. L. Tripp, and A. Manuzzato, "The reliability of software algorithms and software-based mitigation techniques in digital signal processors," in *2013 IEEE Radiation Effects Data Workshop (REDW)*, 2013, 10.1109/REDW.2013.6658218.

[17] F. Irom, "Guideline for ground radiation testing of microprocessors in the space radiation environment," Jet Propulsion Laboratory, Tech. Rep. 13, 2008.

[18] S. Gal-On and M. Levy, "Exploring coremark^{TM} — a benchmark maximizing simplicity and efficacy," The Embedded Microprocessor Benchmark Consortium, Tech. Rep., Available on web at: http://www.eembc.org/techlit/coremark-whitepaper.pdf, last accessed 7/2014.