FINAL NAG9-831

NASA/CR-97- 206025

[illegible]
IN-61-CR
OCIT
C.0427

TEST REPORT

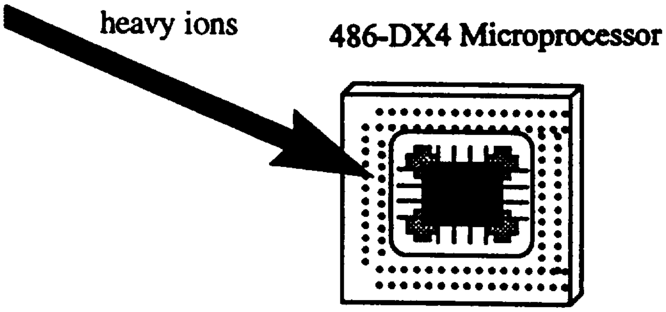
THE SINGLE EVENT EFFECT CHARACTERISTICS OF THE 486-DX4 MICROPROCESSOR

prepared by:

Coy Kouba
Dr. Gwan Choi

Department of Electrical Engineering
Texas A&M University
College Station, TX 77843

December 30, 1996



TEST REPORT

THE SINGLE EVENT EFFECT CHARACTERISTICS OF THE 486-DX4 MICROPROCESSOR

prepared by:

Coy Kouba
Dr. Gwan Choi

Department of Electrical Engineering
Texas A&M University
College Station, TX 77843

December 30, 1996

Coy Kouba

Coy K. Kouba

Dr. Gwan S. Choi

TABLE OF CONTENTS

Section page

Appendix

ABSTRACT

This research describes the development of an experimental radiation testing environment to investigate the *single event effect* (SEE) susceptibility of the 486-DX4 microprocessor. SEE effects are caused by radiation particles that disrupt the logic state of an operating semiconductor, and include *single event upsets* (SEU) and *single event latchup* (SEL).

The relevance of this work can be applied directly to digital devices that are used in spaceflight computer systems. The 486-DX4 is a powerful commercial microprocessor that is currently under consideration for use in several spaceflight systems. As part of its selection process, it must be rigorously tested to determine its overall reliability in the space environment, including its radiation susceptibility.

The goal of this research is to experimentally test and characterize the single event effects of the 486-DX4 microprocessor using a cyclotron facility as the fault-injection source. The test philosophy is to focus on the "operational susceptibility," by executing real software and monitoring for errors while the device is under irradiation. This research encompasses both experimental and analytical techniques, and yields a characterization of the 486-DX4's behavior for different operating modes. Additionally, the test methodology can accommodate a wide range of digital devices, such as microprocessors, microcontrollers, ASICS, and memory modules, for future testing.

The goals were achieved by testing with three heavy-ion species to provide different linear energy transfer rates, and a total of six microprocessor parts were tested from two different vendors. A consistent set of error modes were identified that indicate the manner in which the errors were detected in the processor. The upset cross-section curves were calculated for each error mode, and the SEU threshold and saturation levels were identified for each processor. Results show a distinct difference in the upset rate for different configurations of the on-chip cache, as well as proving that one vendor is superior to the other in terms of latchup susceptibility. Results from this testing were also used to provide a mean-time-between-failure estimate of the 486-DX4 operating in the radiation environment for the International Space Station.

ACKNOWLEDGMENTS

We would like to express our gratitude and appreciation to the Texas A&M Electrical Engineering Department, and to the faculty who participated in this research: Dr. Karan Watson, Dr. Mark Weichold, Dr. Thomas Fogarty, and Dr. Aaron Cohen. We are also very thankful for the collaborative efforts of Mr. Zhong You and the Prairie View A&M Center for Applied Radiation Research.

We would like to extend our appreciation to the TAMU Cyclotron Institute SEE research group, especially Bob Rogers and Robert Gutierrez. Their continuous support of our work greatly helped in completing the cyclotron experiments.

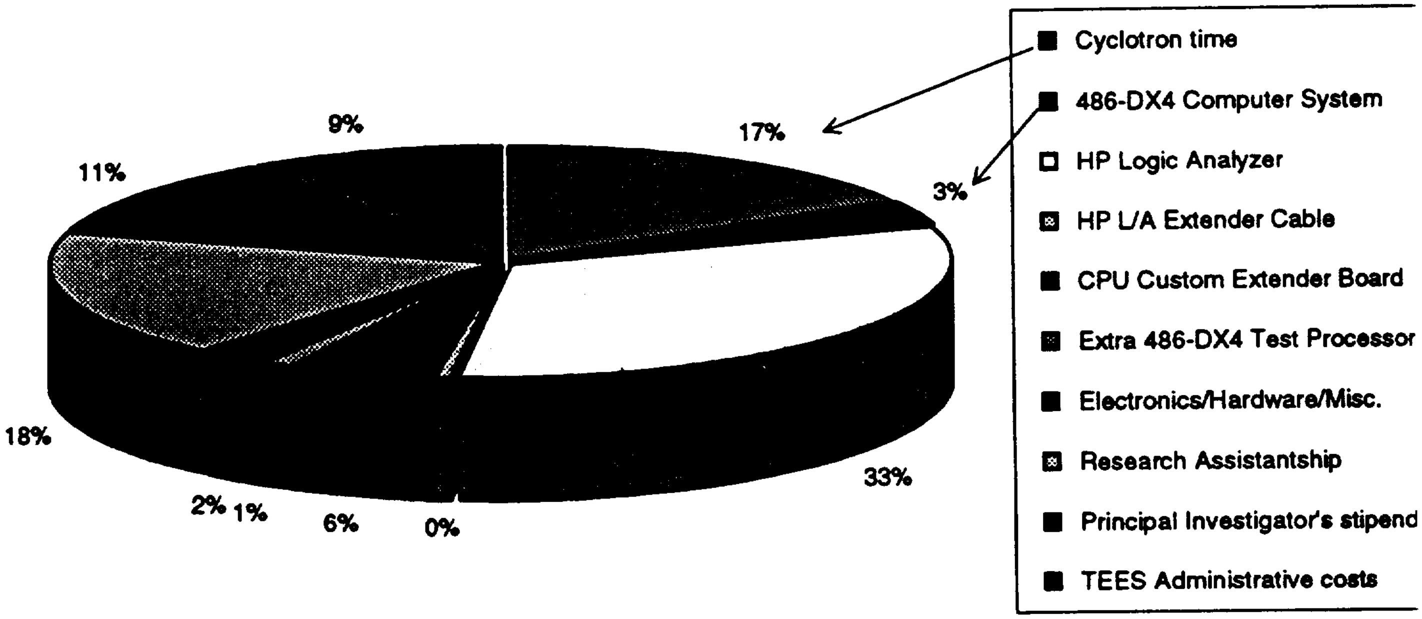
This work would not have been possible without the strong support and guidance from Mr. Mike Gaudiano and Dr. Pat O'Neill, of the NASA–Johnson Space Center. They provided much of the motivation and theoretical assistance needed to make this project a reality. We would also like to thank all those who helped out from the Avionic Systems Division and from the NASA Regional University Grant Office; the resources they provided proved so helpful.

In addition, we would like to acknowledge Dave Beverly from NASA–JSC for his expertise in de-lidding the microprocessors, Alex Ekrot from Compaq Computers for his advice on the CPU extender board, Charles Kouba & Hal Reagan from the Aluminum Company of America (ALCOA) for their donation of the aluminum plates, and Ray Dunkle for his help in the machine shop. We would also like to thank the Computer Engineering secretaries for all their dedication and assistance; Lisa Poole & Carolyn Warzon.

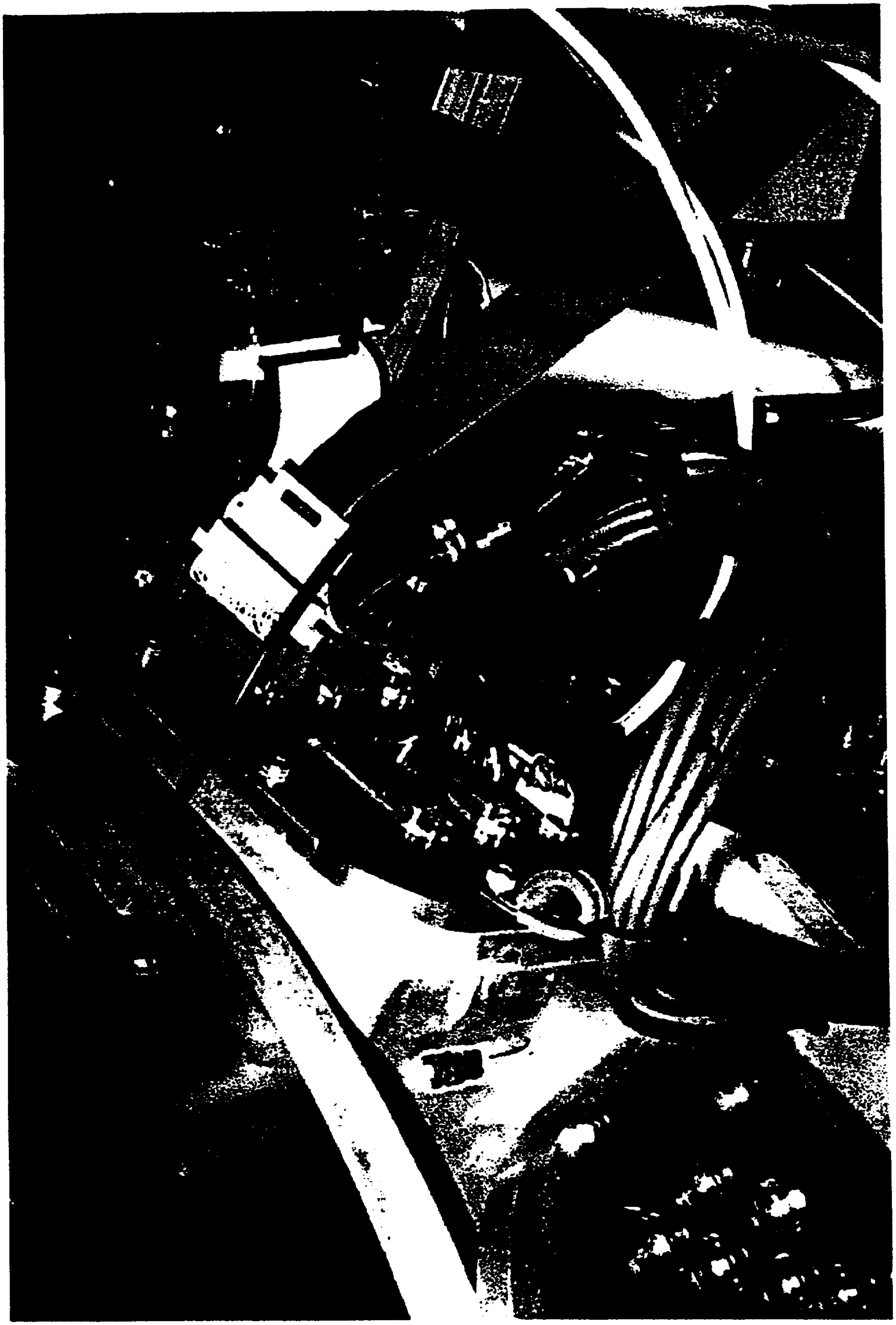
This research was funded under a NASA-University Regional Grant, TEES contract #95-299

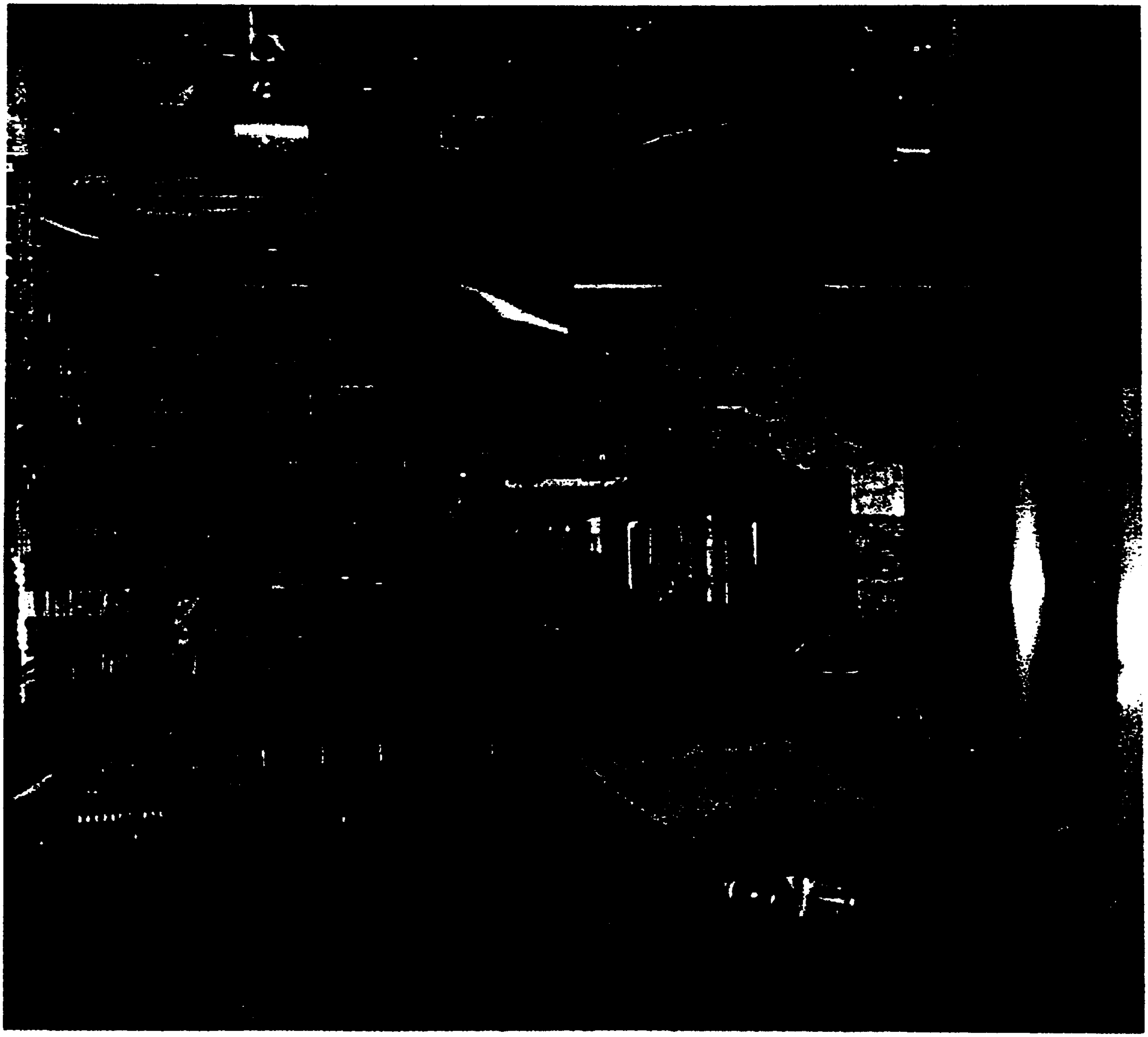
Expenditure Description	Est. Cost
Cyclotron time	7,500
486-DX4 Computer System	1,470
HP Logic Analyzer	14,000
HP L/A Extender Cable	140
CPU Custom Extender Board	2,440
Extra 486-DX4 Test Processors	485
Electronics/Hardware/Misc.	1,100
Research Assistantship	8,000
Principal Investigator's stipend	5,000
TEES Administrative costs	4,000
TOTAL	**$ 44,135**

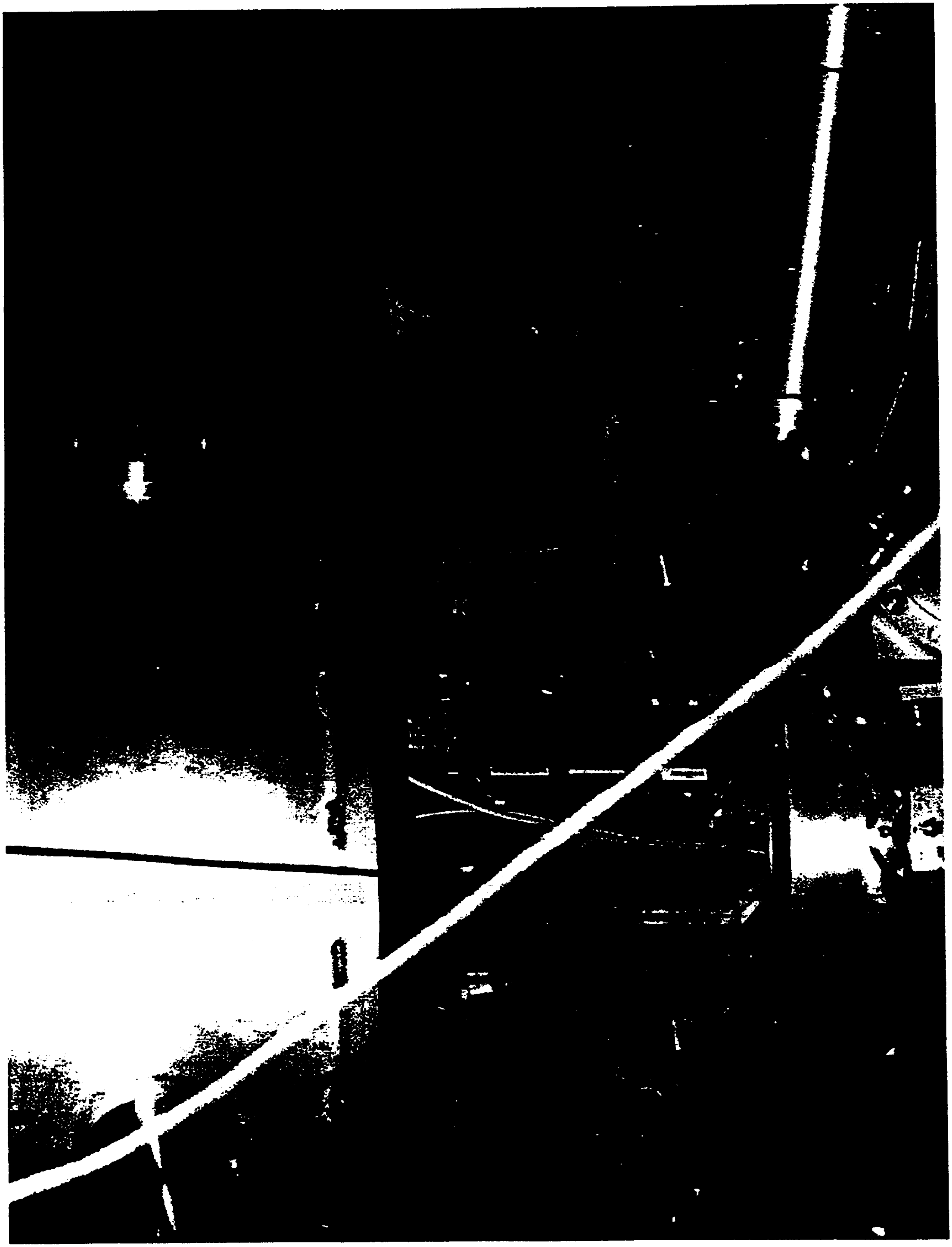
Cost Breakdown of the 486-DX4 RADIATION TEST PROJECT

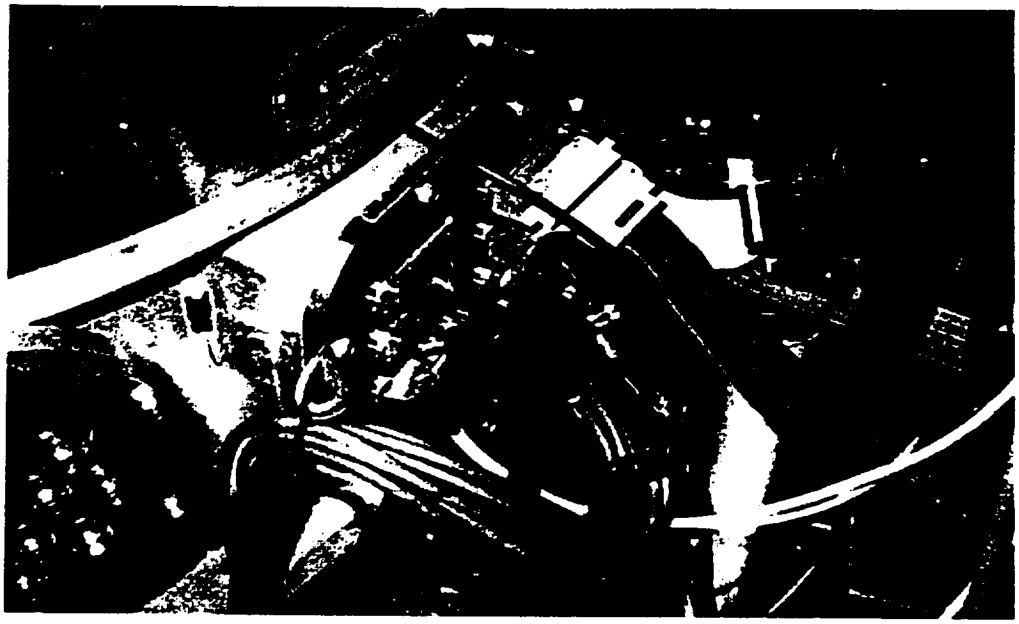


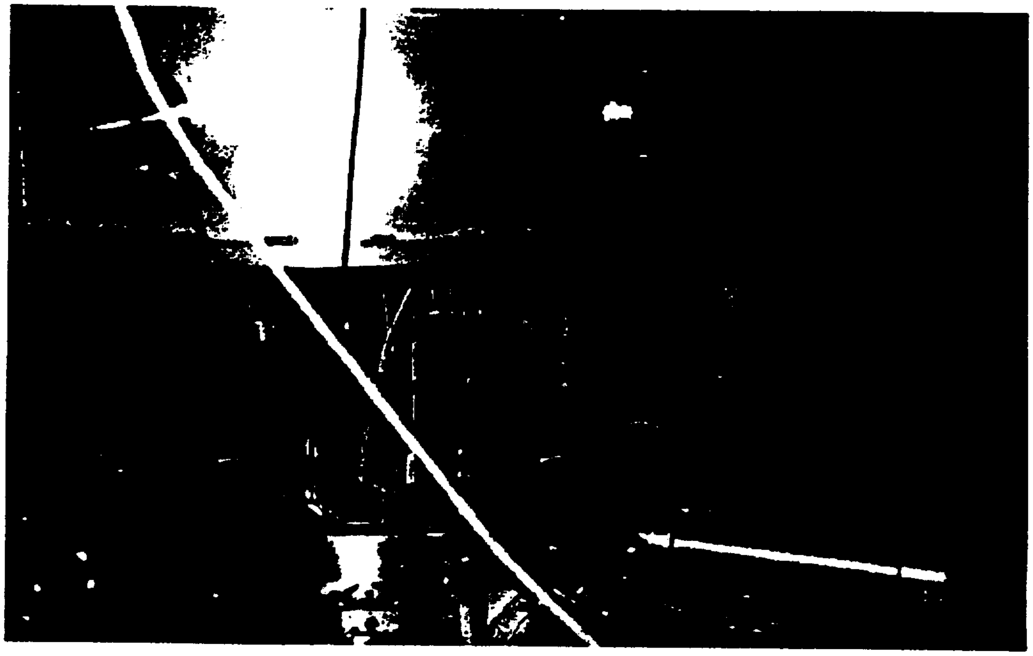
* The following data reports the expenditures of the 486-DX4 SEU Radiation Test Project, performed by Texas A&M University.











The Single Event Effect Characteristics of the 486-DX4 Microprocessor

C.Kouba & G.Choi
Dept. of Electrical Engineering
Texas A&M University
College Station, TX 77843

1.0 INTRODUCTION & MOTIVATION

As our space agency's budget continues to downsize, NASA is using more and more commercial, off-the-shelf technology to carry out its scientific missions & objectives. The 486-DX4 microprocessor is one of those commercial devices chosen for use in several spaceflight computer systems, including at least one used onboard the Space Shuttle. The *Payload & General Support Computer* (PGSC) is a laptop PC computer that astronauts use during Space Shuttle flights. It is a 486-DX4-based system that is primarily used for non-critical mission functions such as data collection, environmental control & observation, and personal management. However, in order to use the PGSC to a greater extent, such as to control the Orbiter's functions, its reliability must be accurately evaluated and predicted for critical spaceflight operations. This includes ground-based radiation testing of the PGSC's components, especially the microprocessor.

To this date, the only known radiation database for the 486-DX4 consists of *total dose* testing. SEU test data exists for the DX2, but not for the DX4; thus the premise for this research was to investigate the SEU susceptibility of the 486-DX4 microprocessor using a cyclotron facility. The results from this experimental testing can then be used to help determine the overall performance and reliability of any 486-based computer, including the PGSC, in a space radiation environment.

2.0 TEST PHILOSOPHY & OBJECTIVES

In the past, much of the SEU testing efforts have focused solely at the device level, with an emphasis on register-level or bit-level susceptibility. The approach in this research however, was to concentrate on the *"operational susceptibility"* of the device. The premise is that many potential error states induced by SEU may not be detected by register-level testing alone [8]. An observed error may be the result of multiple upsets that have manifested in separate locations or circuits. While their individual effects alone may not cause an error, their combined efforts may trigger an observable functional error. The 486-DX4 is a very complex microprocessor with many interleaved operations occurring during each clock cycle, therefore the interaction of multiple SEU errors presents important *control-flow* issues to consider. While this method may not give as much insight as to *where* something happened, it will give a good estimate of the *operational* upset rate for the microprocessor as a whole.

This approach then, requires the device to be tested in a manner that is consistent with the actual application environment in which it is to be used. Our philosophy was to test the processor in an operational state while executing real code at the device's rated speed. For the 486-DX4 testing, this included using a PC-based system board with all associated peripherals to host the processor, albeit only the processor was exposed to the radiation beam. This method allows the CPU (or any other system component) to be tested in an integrated fashion, just as it would be configured for operation in its intended environment. This standardized setup was developed to accommodate a wide range of digital devices, such as memory modules or ASICS, for future testing. This research did not try to identify or improve weak areas of the chip design, instead it measured its performance and susceptibility in a heavy-ion environment.

486-DX4 SEU Test Goals

The primary goal of this research was to develop an evaluation platform to experimentally measure the SEU and SEL susceptibility of the 486-DX4 microprocessor. Specifically, the following objectives were sought:

(1) To perform the 486-DX4 test with 3 different heavy-ion species;

- **Xenon** (LET = 43.1 MeV-cm^2/mg)
- **Krypton** (LET = 25.1 MeV-cm^2/mg)
- **Argon** (LET = 7.7 MeV-cm^2/mg)

(2) Identify the observable *error modes* in the microprocessor

(3) Calculate the *upset cross-sections* of the device using the error modes, fluence, and error rate

(4) Compare the upset rate for different operating configurations of the device, specifically with the internal L1-cache enabled vs. disabled

(5) Determine the SEU and SEL threshold levels

(6) Test different chip implementations of the 486-DX4 using parts from two different vendors: *Intel Corporation* and *Advanced Micro Devices* (AMD)

(7) Use these test results to predict the reliability of the 486-DX4 in the radiation environment associated with the International Space Station orbit

3.0 THE 486-DX4 MICROPROCESSOR

The 486-DX4 is a widely used 32-bit microprocessor that has an extensive software base and is utilized on a wide variety of platforms. The device's popularity is due to its advanced design, low power requirements, low cost, mass production, and strong compatibility with different PC systems. It is also binary compatible with all previous versions of the 80x86 processor family, thus any 80x86 software can be supported by the 486-DX4.

The DX4 contains a 32-bit pipelined integer arithmetic logic unit (ALU), as well as an on-chip floating point unit. The instruction set microarchitecture is implemented using RISC design techniques such that frequently-used instructions are executed in one clock cycle. The DX4 supports a segmented addressing scheme which includes byte-level addressing. Code and data for each task are stored and managed in "segments" which can be up to four gigabytes (2^{32} bytes) in size. Each task can have a maximum of 16,381 segments, thus each task can address a maximum of 64 terabytes (trillion bytes) of virtual memory.

The memory management unit consists of a segmentation unit and a paging unit. The segmentation unit allows management of the logical address space by keeping code and data for each task organized in certain portions of memory. The segmentation unit also implements a four-level protection scheme on all data structures. The paging unit provides access to data structures larger than the available physical memory by swapping the current data into memory and retaining the unused part in mass storage.

The 486-DX4 has an internal L1-cache that is used to hold instructions and data of a currently executing program inside the processor, thus saving time and speeding up memory operations by an order of magnitude. It is a 4-way set associative cache that supports a write-through policy. The DX4 also contains built-in self-test circuitry, test registers, and a debug mode to aid programmers and system designers in software development. A functional block diagram of the 486-DX4 microprocessor is given in Fig. 1.

The 486-DX4 has four modes of operation that determine which instructions and processor features are accessible, and are called: (1) Real Mode, (2) Protected Mode, (3) Virtual 8086 Mode, and (4) System Management Mode. In Real Mode the processor acts as a fast 8086 and it is normally used to set up the processor for protected mode operation. The Protected Mode allows execution of all the 32-bit instructions and sophisticated privileges of the 486-DX4, as well as use of the memory management paging. The Virtual 8086 Mode is a sub-mode of the protected mode, allowing the execution of multiple 8086 tasks within the bounds of the protected, multitasking environment of the processor. Finally, the System Management Mode is used by system designers to add new software-controlled features to the system.

The application register set of the 486-DX4 is a group of sixteen registers that may be used by the programmer to support program execution, and are grouped into three categories: (1) General registers, (2) Segment registers, and (3) Status & Control registers. The 32-bit General registers are used to hold operands and results for logical and arithmetic operations, as well as operands for address calculations. They are byte-accessible to provide flexibility for the programmer and to remain compatible with earlier 80x86 families. The 16-bit Segment registers are used to support the memory organization of the processor, and contain values which index into tables in memory. They are used to keep track of where the data, code, and stack is of each process. The Status & Control registers hold the results and condition codes of each instruction, as well as control certain operations and indicate the status of the processor. The application register set of the 486-DX4 is illustrated in Fig. 2.

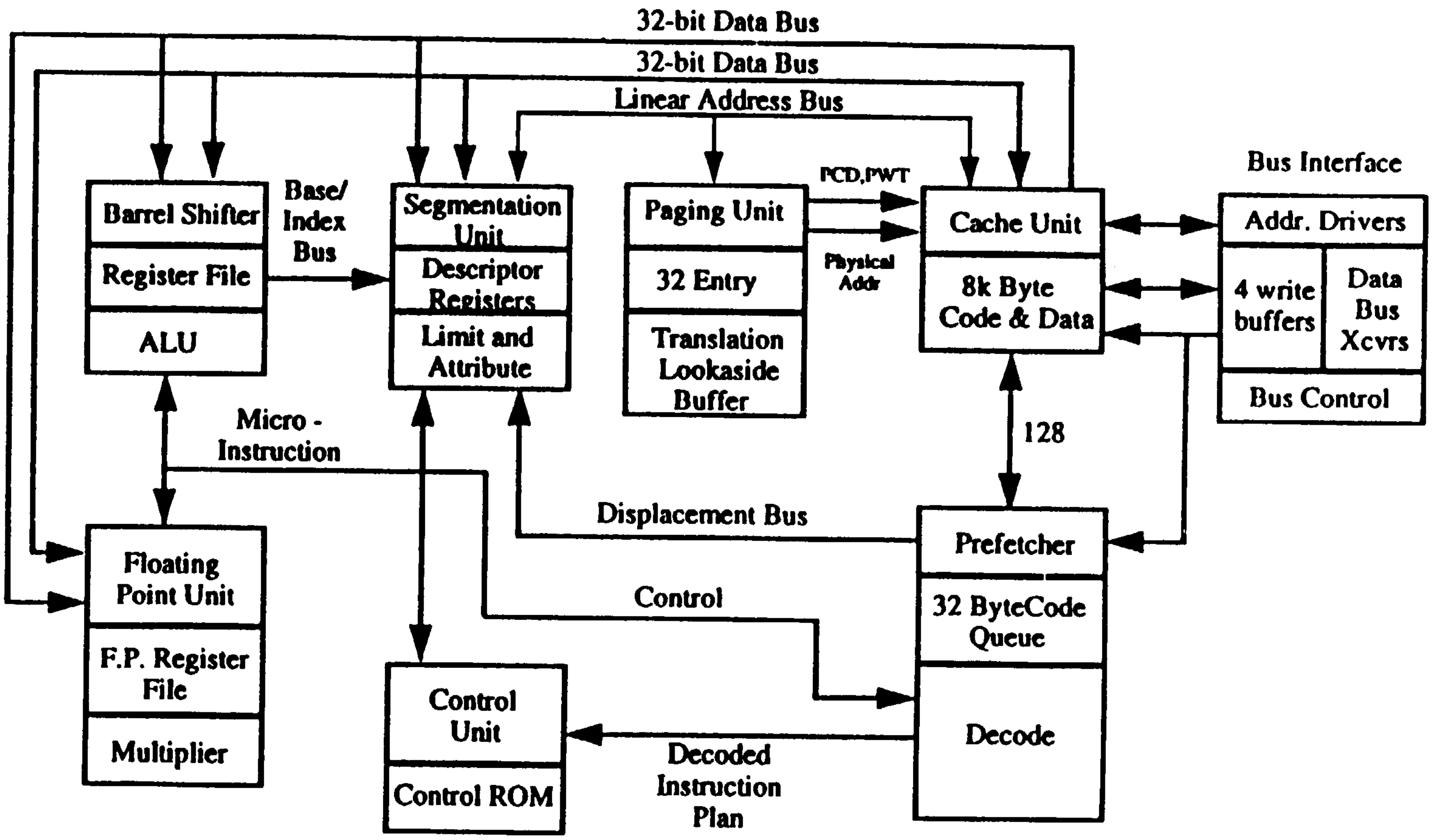


Figure 1: A functional block diagram of the 486-DX4 microprocessor.

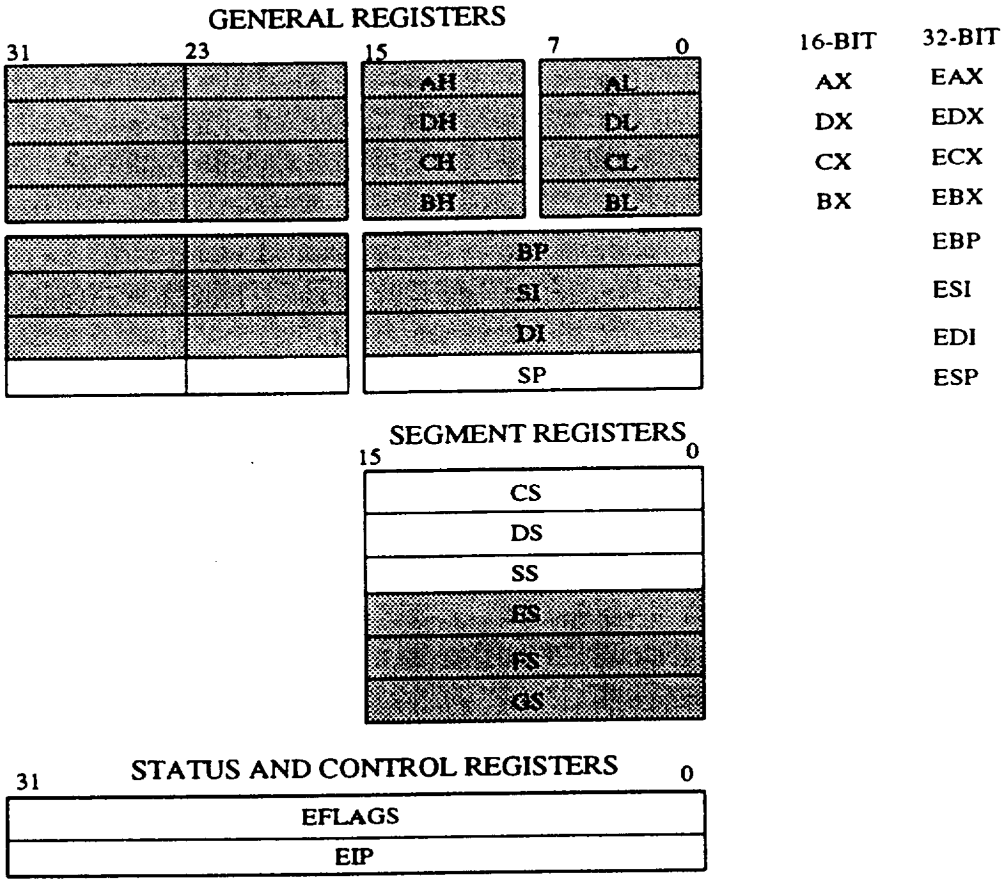


Figure 2: The *Application Register Set* of the 486-DX4, where the shaded regions represent those registers directly testing in the SEU experiment (see section 5.2).

One important aspect of this SEU experiment is to test the susceptibility, or "radiation hardness," of the application register set. This was achieved by loading them with test patterns and checking to see if and when they change under the influence of radiation. The shaded areas in Fig. 2 indicate the registers directly tested in the SEU experiment, giving a test coverage of 76.9% of the entire set. Some registers could not be directly tested due to the fact that they were needed to control the test program execution, such as the instruction pointer (EIP), but they were always indirectly tested by observing errors in the program flow. The details of this algorithm will be discussed in the Test Software section.

When the L1-cache was disabled all memory references have to go off-chip to the system DRAM, thereby reducing system performance. During the radiation test, the processor was tested in both cache configurations, and it was expected that higher upset rates would occur when the L1-cache is enabled. This is due to the fact that when instructions and data are held in the cache they are vulnerable to being upset, even before they are executed. With the cache disabled, all code and data reside in system DRAM and are thus protected from radiation.

While both AMD and Intel have produced pin-to-pin equivalent DX4 designs, there are several major differences between them. Intel's design possess 16KB of internal L1-cache (code & data), while AMD's design contains 8KB. The biggest difference between the two is in the fabrication process and die size. Intel's DX4 design is fabricated on a 3.3 volt 0.6 micron BiCMOS process, contains 1.8 million transistors, and the die is approximately 0.7225 cm^2. AMD's design is produced on a 0.5 micron 3LM CMOS process, contains about 1.6 million transistors, and is approximately 0.49 cm^2. The 486-DX4 is available in either a 168-pin Pin Grid Array (PGA) ceramic package, or a 208-pin Plastic Quad Flatpack (PQF). The devices used in this testing were of the *PGA* type. The process feature differences (that could be obtained) for each vendor are summarized in Table 1.

Table 1: 486-DX4 Device and Process Characteristics.

Characteristic	Intel	AMD
Fab process	0.6 µm BiCMOS	0.5 µm 3LM CMOS
Operating voltage	3.3 Volts	3.45 Volts
Transistor count	1.8M	1.6M
Die size	0.7225 cm^2	0.4900 cm^2
Package type	168-pin PGA	168-pin PGA
Internal L1-cache	16KB	8KB
Wafer type	P-EPI on P	n/a
Well type	dual well	n/a
Gate Oxide thickness	80 Angstroms	100 Angstroms
Min feature size	0.57 µm	n/a
Min channel length	0.30 µm	n/a
Diffusion depth	n/a	4-layer/0.5 µm
Passivation thickness	4500 Angstroms	n/a
Die overcoat material	Polyimide	n/a
Die overcoat thickness	3.1 µm	n/a

4.0 DESCRIPTION OF THE TEST ENVIRONMENT

The radiation testing was performed in the *Single Event Effects Facility* at the Texas A&M Cyclotron Institute in College Station, Texas. Their K500 superconducting cyclotron was used to provide the heavy-ion radiation source needed for SEU testing. The K500 is a physically compact, low energy consumption, high energy machine capable of generating a diverse range of particles and energies to support many different atomic and nuclear physics experiments [21]. For the TAMU cyclotron, seven beam lines exist including the Single Event Effects line.

The SEE Line

As the radiation beam is sent down to the SEE Line, magnets help control, direct, and shape the beam to help provide for uniformity and to reduce attenuation. The beam enters a test chamber where the target device will be exposed to radiation. A shutter at the chamber entrance gives the experimenter precise control over the application of the beam to the test system. The SEE Line consists of the following components:

(1) Target chamber
(2) Positioning mechanism
(3) Imaging system
(4) Test Control and Monitoring Station
(5) Radiation safety
(6) User systems

The target chamber is a large aluminum enclosure with inside dimensions of 30 inches diameter by 30 inches high. In the middle of the chamber is a mounting bracket where the test system is attached. The experimenter has real-time control over the X, Y, and Z axes, plus angular control in reference to the beam arrival. The electrical interface to the test chamber is via a block of six 50-pin male IDC connectors.

The test chamber must be closed and in a vacuum during any radiation testing. Two mechanical fore pumps and one turbomolecular pump are used to bring the chamber down to an operating pressure in the low 10^{-5} torr range. De-pressurization can take as little as 15 minutes (depending on the test system), and venting takes about three minutes.

Once the test system is installed, a position check is made to ensure the target is in the center of the beamline. This is achieved by coupling a high brightness phosphor to a sensitive CCD camera, as well as using a laser for visual confirmation of the beam's center.

The experiment is controlled and monitored from the Test Control Station (TCS), located approximately 2 meters from the test chamber, separated by a thick lead shielding wall. All user equipment and personnel are stationed at the TCS during testing. Instrumentation for the SEE Line is also performed at the TCS, and includes the positioning and imaging systems, beam integrity monitoring, and operation of the beam shutter.

Beam diagnostics are performed during testing which give the experimenter an accurate count of the beam uniformity and flux. A faraday cup in the SEE Line provides the first measurement of the total beam intensity, and continuous real-time monitoring of the particle beam is provided by an array of four plastic scintillator and photomultiplier tube (PMT) assemblies. These are arranged on a three inch diameter circle around the beam axis. A fifth PMT assembly is mounted on a moveable arm that allows it to be placed on the beam axis [21]. Closer to the test chamber is a silicon surface barrier detector mounted on a movable arm. This detector measures the total energy of the particle beam and thus provides information on beam purity.

Ion Beam Selection

It is up to the experimenter to select the heavy-ions to be used in the cyclotron test. The most important beam parameters to consider are the linear energy transfer (LET), range in silicon, and ion charge state. For the first beam that was used, it was desired to test the processor in its saturated upset rate, thus Xenon (Xe) was selected with an LET = 43.7 MeV-cm^2/mg. The second beam selected was Krypton (Kr), to provide intermediate data points (LET = 25.1 MeV-cm^2/mg). The third beam was used to try and capture the device's upset threshold, thus Argon (Ar), with an LET = 7.7 MeV-cm^2/mg was used. A conservative measure was used in selecting this beam, so that its LET was slightly higher than the actual expected threshold. This was to ensure that the threshold was not missed altogether by undershooting and producing no data points at all. Currently, eleven particle beams are available for SEE testing at the TAMU facility, and are summarized in Table 2.

Table 2: Ion beams qualified for SEE Testing at the TAMU Cyclotron Facility [21]. The shaded entries comprised the beams used in the 486-DX4 experiment.

TAMU CYCLOTRON - HEAVY-ION BEAMS AVAILABLE FOR SEE TESTING							
Ion	Q/A	Energy (MeV)	E/A (MeV/A)	LET (MeV-cm^2/mg)	Range (µm)	LET$_{max}$	Penetration LET$_{max}$
$^{12}C^{2+}$	0.167	125	10.4	1.3	250.9	5.3	246.2
$^{16}O^{3+}$	0.188	210	13.2	1.9	307.0	7.4	302.0
$^{20}Ne^{4+}$	0.200	298	14.9	12.5	315.9	9.6	308.4
$^{40}Ar^{8+}$	0.200	599	15.0	7.7	228.9	19.9	220.2
$^{63}Cu^{13+}$	0.207	1003	16.0	17.2	185.8	33.9	169.6
$^{84}Kr^{17+}$	0.203	1284	15.3	25.1	177.6	41.3	156.4
$^{84}Kr^{16+}$	0.191	1141	13.6	26.6	136.0	41.3	114.8
$^{84}Kr^{15+}$	0.179	1002	12.0	28.2	128.9	41.3	107.7
$^{93}Nb^{16+}$	0.172	1030	11.1	34.5	120.0	47.9	95.8
$^{129}Xe^{26+}$	0.202	1960	15.2	43.7	162.4	63.4	131.3
$^{197}Au^{33+}$	0.168	2068	10.5	87.1	105.6	93.4	52.5

5.0 TEST PROCEDURES

5.1 Hardware Configuration

A commercial, off-the-shelf PC computer system was chosen and modified to host the test. The PC motherboard was extracted and attached to a bracket that fit the inside dimensions of the test chamber. The motherboard & test processors were the only hardware inside the test chamber, with the remaining hardware residing at the Test Control Station. Interface cables 2.8 meters in length were required to connect the motherboard to the rest of the system. These cables interfaced to the test chamber through the block of 50-pin IDC connectors.

The hardware at the Test Control Station included a digital logic analyzer, two power supplies, a hard disk drive, video monitor, keyboard, and a temperature monitor. An illustration of the hardware setup is given in Fig. 3, and a brief description of the required modifications are presented in the following subsections.

De-lidding the Test Processors

For each test device, the metal lid of the PGA package had to be removed in order to allow sufficient energy to be deposited onto the silicon die. The de-lidding was a fairly trivial but delicate task, and was achieved by using a hefty Exacto knife and several razor blades to cut away at the solder that secures each lid. When the heel of the blade could be slid under a corner of the lid, pressure was used to pry it away. The lids were then carefully taped back in place to protect the die until the experiment, and a thorough functional check-out was performed to ensure no damage occurred in the removal process.

Design of the CPU Extender Socket

The PGA package presented a problem since the lid to the die was also on the same side as the pins. Thus, when the processor was plugged into its socket, the lid was "sandwiched" between the CPU and the motherboard. Since the only way to expose the die was through this lid, provisions had to be taken to alter the orientation of the processor to provide a direct path for the beam.

A *CPU extender socket* was designed and built so that it flipped the processor 180°. This was achieved by soldering together two 168-pin PGA sockets with 30-guage wire. One of these sockets plugged into the motherboard socket and the other was a zero-insertion force (ZIF) socket that the CPU plugged directly into. The length of the connecting wires was kept as short as possible, approximately 14.5 cm each. This approach was successful and allowed the processor to boot-up normally, however the processor's clock speed had to be reduced from 100-MHz down to 75-MHz to run correctly. The increased signal lengths on the CPU bus had introduced timing delays that could only be corrected by slowing down the bus speed. Refer to Fig. 4 for a picture of the CPU extender socket, as well as the heat sink hardware that is described in the subsection below.

CPU Thermal Management

One important parameter that had to be monitored during the experiment was the temperature of the processor. The first sign of overheating would be in the form of data errors, thus steps were needed to eliminate this threat. With conduction as the only means of removing heat, a 14" x 4" x 0.75" inch aluminum plate was used to provide a massive heat sink. This plate was attached to the test bracket, which was then attached to the top of each test processor with a thermal adhesive pad. A small hole was drilled into the center of this plate where a thermocouple was inserted to allow contact with the top of the processor. The thermocouple wire was routed outside the chamber to the Test Control Station, where

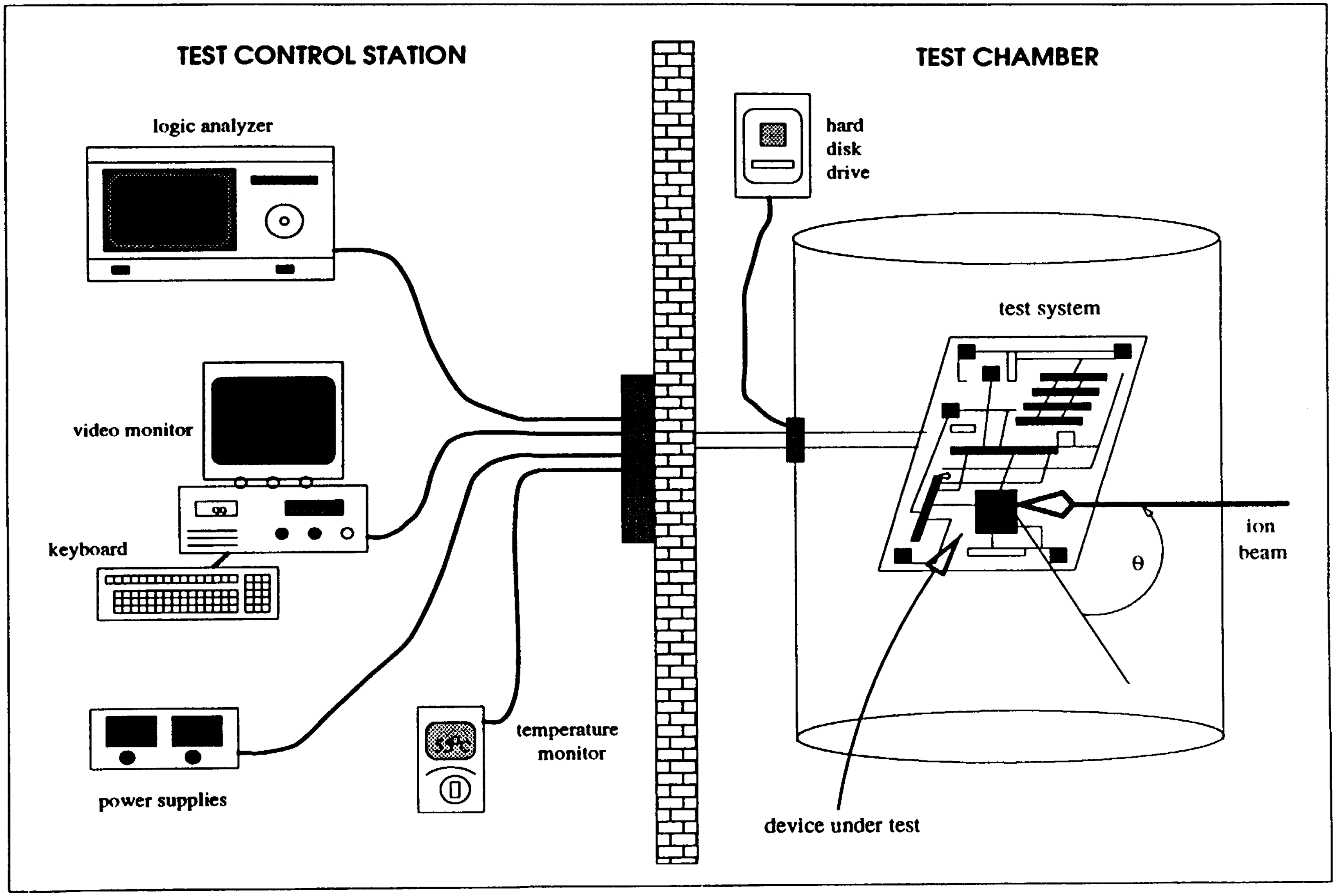


Figure 3: The cyclotron configuration for SEU testing on the 486-DX4. Note that only the microprocessor was exposed to the radiation beam.

it was connected to a Wavetek multimeter. The maximum operating temperature limit for the 486-DX4 was 85° Celsius (measured on the package surface), as defined by the data book. If a device was ever to violate this limit, the system would be powered down and allowed to cool before resuming the test. However, throughout all tests, the maximum temperature observed was only 63° Celsius.

Power Supplies

Two separate power supplies were needed for the cyclotron experiment. The goal was to isolate the CPU power from the rest of the system, in order to monitor its power consumption for detecting latchup conditions. The motherboard requires voltages of ±12V, ±5V, and the CPU requires +3.3V which it receives by stepping-down the +5V source via onboard voltage regulators.

The PC system power supply was used to drive the ±12 and −5 voltages to the motherboard, as well as the hard disk and floppy disk drives. A separate, current-limiting DC power supply was used to independently drive the +5V voltages to the motherboard. By monitoring the +5V motherboard source, we could indirectly monitor the CPU current draw, since any significant change in the CPU current would be positively reflected at the power supply. The nominal current draw using AMD's CPU was 2100mA, and with Intel's CPU it was 2360mA. The current limit was set at the nominal operating value plus 100mA. If this limit was ever breached, the power supply would enter a constant-current mode, where the supply voltage would drop proportionally to maintain the current limit.

During the cyclotron experiment, a latchup condition would be detected by a sudden increase in the supply current (due to the "virtual short"), and the current-limiting feature would help protect the device from permanent latchup damage. The most accurate means of detecting latchup, however, would be to monitor the current draw at the V_{CC} and V_{SS} pins on the CPU. But there are 51 power pins on the 486-DX4, and due to cabling constraints this option was not pursued.

Logic Analyzer Interface

A Hewlett-Packard 16500B digital logic analyzer was used in the experiment to acquire CPU data for post-test analysis. The lower 16-bits of the processor's data bus were connected to the analyzer's signal pods, and these were then routed to the logic analyzer at the Test Control Station. The logic analyzer pod cable had to be modified to reach the inside of the test chamber. An HP extender cable was used to increase its length, and two short (5 cm) adapters had to be built to connect the HP cables to the IDC connectors on the chamber interface. The goal was to use the analyzer to store the contents of a corrupted register when an SEU was detected. This data was then stored on the analyzer's hard drive, where it would be processed off-line to determine which particular bit(s) of the register were hit. The purpose of using the logic analyzer was to explore the effect of radiation on the control-flow issues of the processor. For details of the logic analyzer's triggering mechanism, please see the discussion in the Test Software section.

Hard Disk Drive

A 560MB hard disk drive (HDD) was used to boot the system, load the test code, and record the data files. The HDD was attached to the outside of the test chamber via a 50-pin IDC connector. The cable length was 114 cm, and no delay or timing problems were observed. A similar cable was built for the floppy disk drive, to provide file transfer capability and to serve as an emergency backup in case the HDD failed.

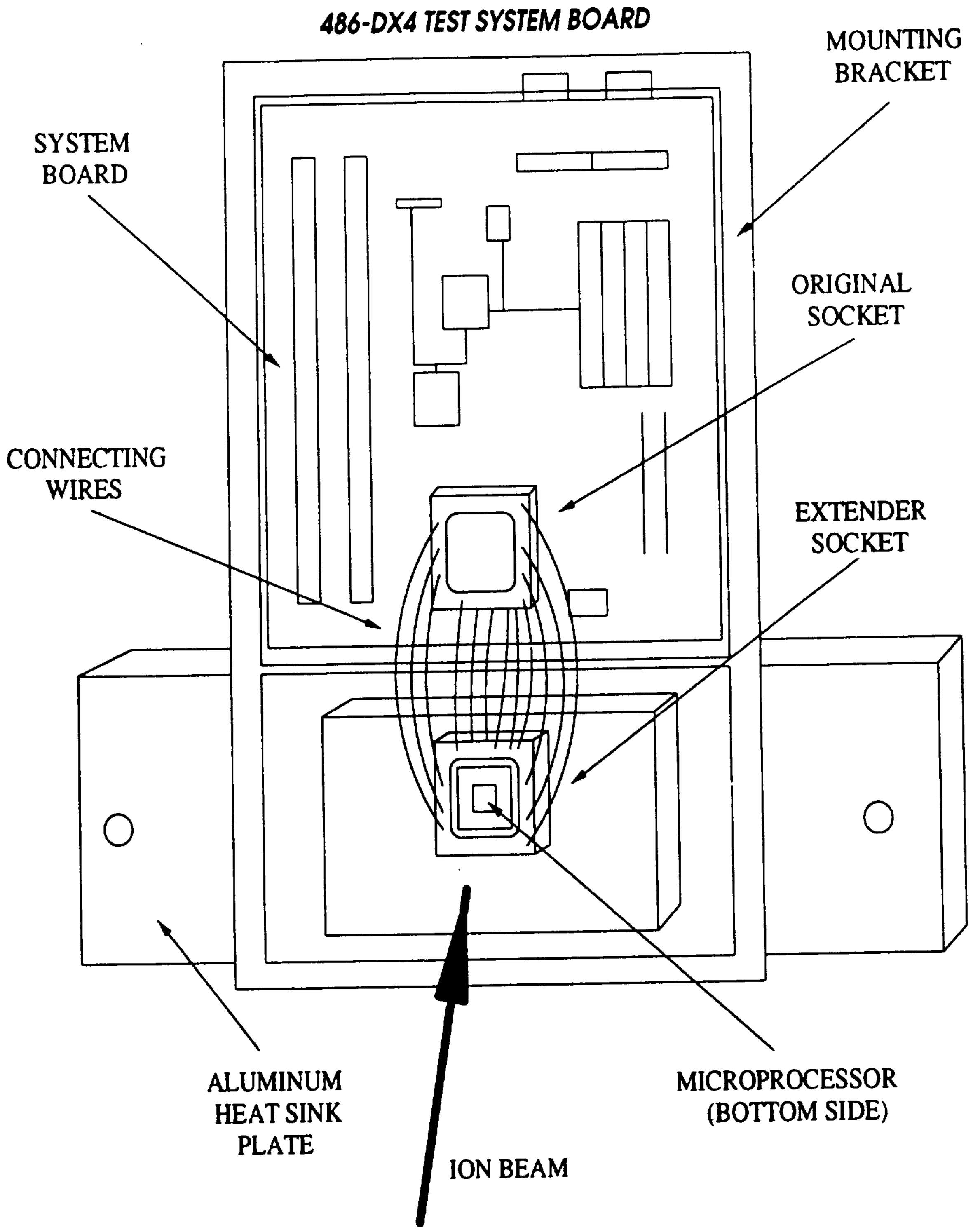


Figure 4: An illustration of the test bracket, CPU extender socket, and heat sink mount.

Monitor Keyboard
A video monitor and keyboard were required during the experiment, and were connected to the system by extension cables that were modified with 50-pin IDC connectors. These cables were 2.8 meters in length.

5.2 Software Requirements

The purpose of the test software was to dynamically exercise the 486-DX4 in a variety of ways to get a broad response of the device under the influence of radiation. A suite of test programs were chosen to provide different workloads and instruction mixes that exercised different functional aspects of the device. The software also provided a means to test the processor in different operating configurations, such as enabling or disabling the internal L1-cache.

Three types of programs were used in the SEU experiment, that comprised a total of eleven individual programs. The first program type, called REGTEST, tested the application register set (refer to section 3.0) of the 486-DX4 by loading the registers with one of four test patterns and then continuously checked them for errors. The second program type, ALUTEST, performed a series of ALU-intensive operations and wrote the results to a file for off-line analysis. The third program type, MCPDIAG, tested the floating-point unit and reported the errors in real-time. While each program type attempted to focus on one functional unit of the device, the entire processor was always vulnerable to upsets. All programs were written in assembly language, with the exception of the MCPDIAG program, which was obtained from Intel.

The REGTEST Program
The goal of the REGTEST program was to directly test the application register set of the 486-DX4 for SEU upsets. REGTEST had eight different versions; four to be used when the internal L1-cache was enabled and four for when it was disabled. The only difference between these two groups is in a loop delay constant, which was decreased when the cache was disabled to keep the loop overhead and time durations the same. The programs within each group were logically the same except for the test pattern used to fill the registers, either an $FFFF, $0000, $1010, or $0101.

When a REGTEST program was executed, the registers EAX, EBX, ECX, EDX, EBP, EDI, ESI, ES, FS, and GS were loaded with one of the four test patterns before the device was exposed to the beam. After the initialization and loop overhead was performed, the program entered an endless loop where each register was continuously compared to the test pattern. If at any time a mismatch was detected in a register, a call would be made to an error-handling routine. The first action in this routine was to display a "*Halt Beam, Error in Register....*" message, at which time the experimenter would shut off the beam. The next step for the error-handling routine was to re-compare the suspect register again to make sure the error was still present. If not, the error was assumed to be transient and the routine ends by displaying a "*Prepare to Resume Beam....*" message. The experimenter then turns the beam back on and the program resumes checking all the registers for upsets again.

However, if the re-compare still did not agree with the test pattern, the error had latched and an SEU had occurred. The error-handling routine then compared each byte of the 32-bit register to determine which byte was corrupted. This information was then displayed on the experimenter's video monitor and was recorded in the data log. The next step was to determine if SEL had occurred by attempting to write the correct test pattern back to the register and immediately read it back again. If SEL had occurred, the corrupted location would still remain latched and the correct pattern could not be written. If SEL occurred, it would be logged and the power to the system would be recycled to clear the latch and a new test would then be started. If the correct test pattern was written and read back successfully, thus only an SEU occurred, the error-handling routine would end by displaying the *"Prepare to Resume Beam...."* message and the ion beam would be turned on and the program would resume checking all registers again.

This endless loop would continue until either an SEU interrupted the program-flow, or the experimenter terminated the program. The test program spent the majority of its execution time in this compare loop. Flowcharts for the REGTEST algorithm are given in Figs. 5 and 6, and the source code may be found in Appendix C.

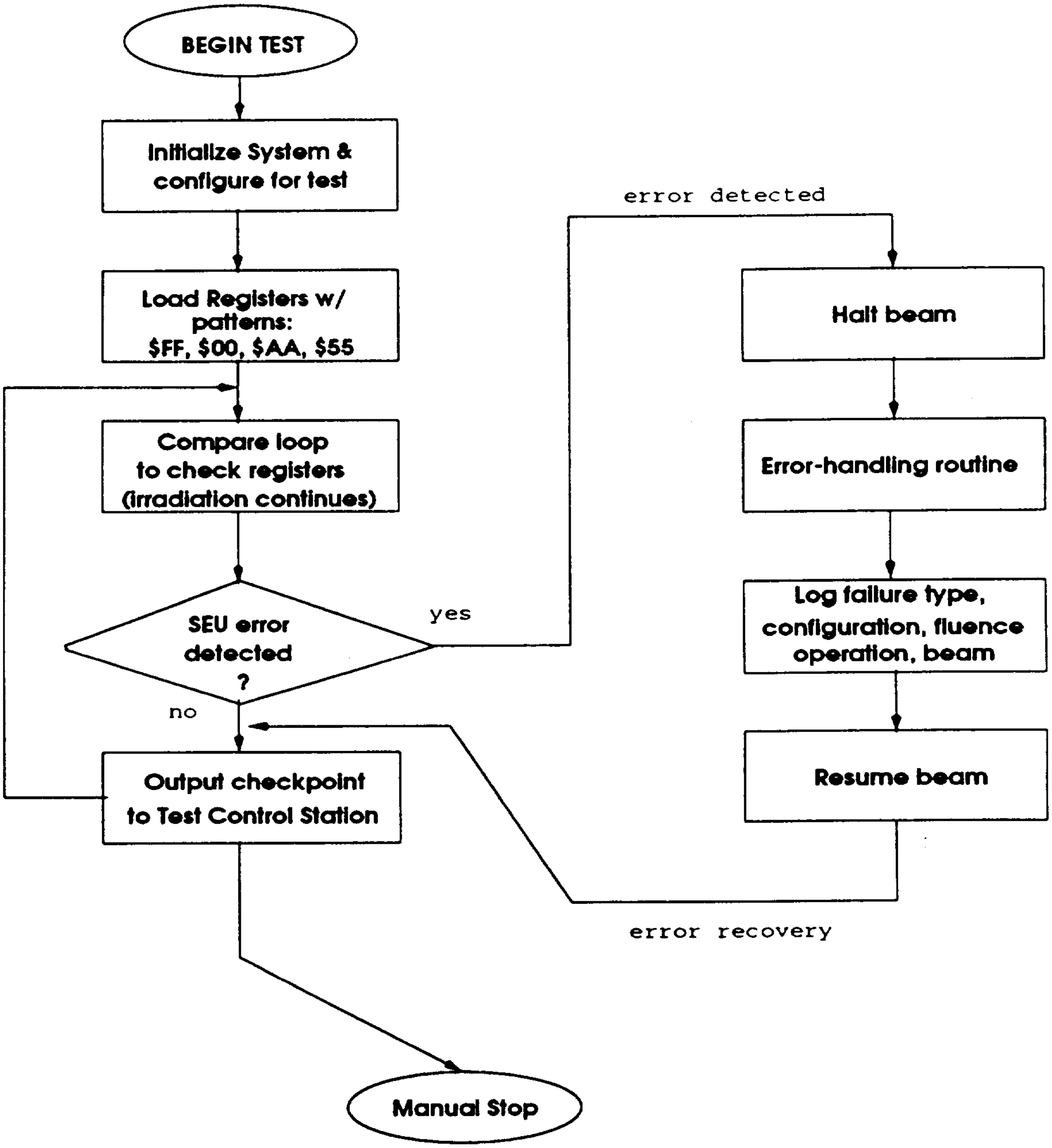

BEGIN TEST
Initialize System & configure for test
Load Registers w/ patterns: $FF, $00, $AA, $55
Compare loop to check registers (irradiation continues)
SEU error detected ?
yes
no
Output checkpoint to Test Control Station
error detected
Halt beam
Error-handling routine
Log failure type, configuration, fluence operation, beam
Resume beam
error recovery
Manual Stop

Figure 5: The flowchart for the *REGTEST* program.

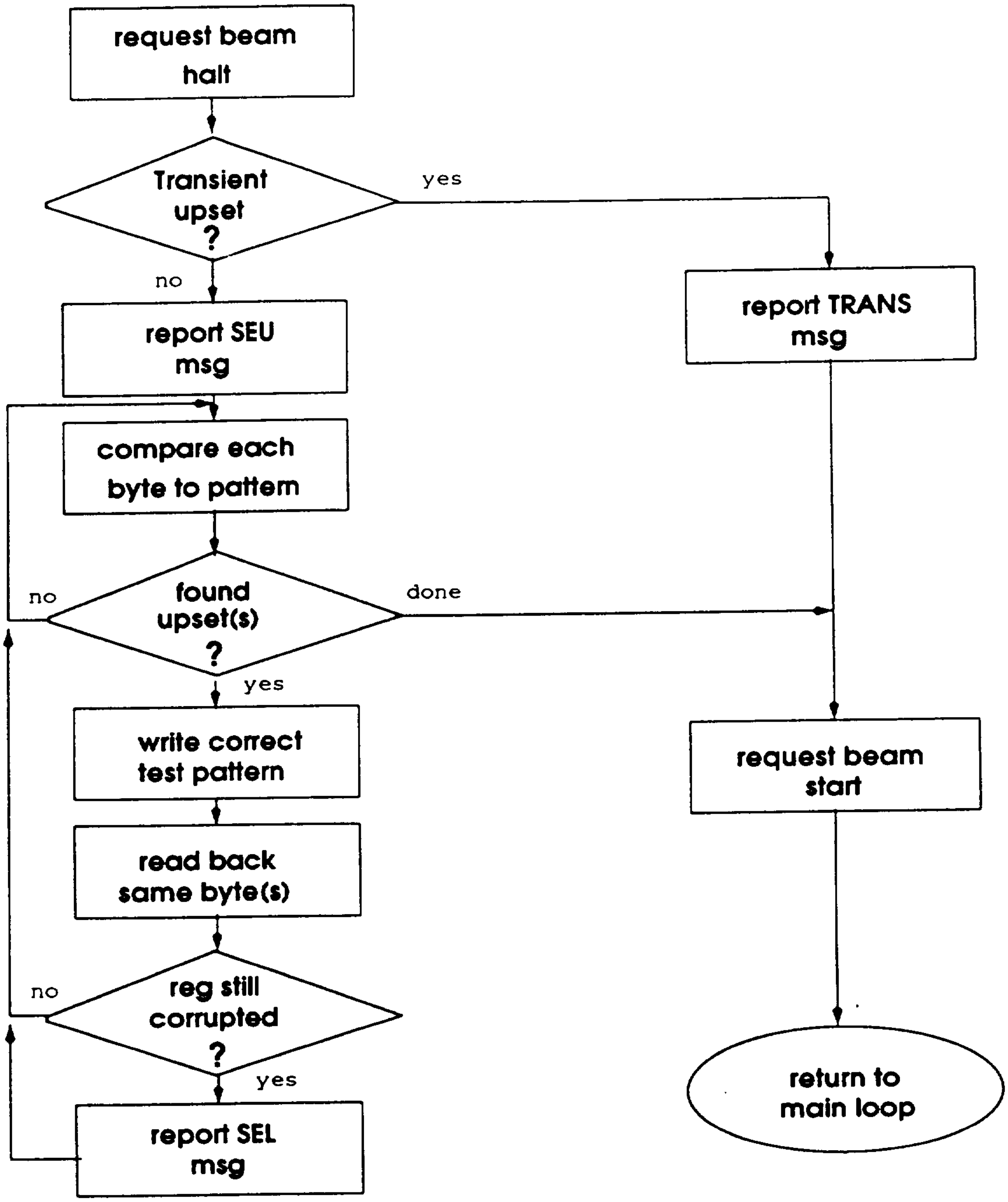


Figure 6: The error-handling routine for the *REGTEST* program.

As stated back in the Hardware Configuration section, the purpose of the logic analyzer was to determine the corrupted bit locations of an upset register, and it was used in conjunction with the REGTEST program. When an SEU was detected, the error-handling routine was invoked as described above. When it was determined that an SEU had occurred, the routine would send out a "trigger signal" to the logic analyzer, by writing a specific data flag to a memory location. This value would be sent out over the data pins which the analyzer was sampling. Immediately after sending this trigger, the routine would then send the contents of the corrupted register to the same memory location. When the analyzer saw the trigger, it would begin storing state information off the data pins, thus capturing the trigger and the corrupted register. The memory depth of the analyzer was 1MB, and after it was full the data was saved to disk for post-test analysis. Since the beam was off during the error-handling routine, there were no hard time constraints that had to be obeyed. Thus the experimenter would command the logic analyzer to start looking for this trigger once inside the routine, and the analyzer did not have to be synchronized with the test software. It is important to note that this method could only be used with confidence when the L1-cache was disabled. This is because the data flag's memory location could be assured of residing in system DRAM (as opposed to onboard the chip's cache), and thus be picked off the CPU's pins by the analyzer at the correct time.

The ALUTEST Program

The arithmetic logic unit (ALU) and the input/output (I/O) functional units of the 486-DX4 were exercised with the ALUTEST test program. This program was written in assembly language and its purpose was to repeat a series of ALU-intensive operations and write the results to a file for post-test analysis. The ALU operations performed were integer ADD, SUB, MULTIPLY, and DIVIDE. The program began by first opening an output data file and then entering a loop that continuously repeated the series of four integer operations. Each operation started with an initial value that was recursively manipulated for a set number of operations. The results were written to the output file so the answers could be checked for errors post-test. This test lasted for approximately 18 seconds, at which time the output file was closed and the program terminated normally.

There were two versions of this program, one for each configuration of the L1-cache. The two versions were identical except for the program variable COUNTMAX; which corresponded to the total number of iterations to execute before terminating the program. This difference was again to keep the execution times equal under both conditions, thus COUNTMAX = 288 when L1 was enabled, and COUNTMAX = 40 when L1 was disabled. The flowchart for the ALUTEST program is given in Fig. 7, and the source code may be found in Appendix C.

The MCPDIAG Program

The floating-point unit (FPU) was tested with a program called MCPDIAG, which was acquired from the Intel Corporation. This program is a burn-in test program they use for testing the FPUs of new chips. When executed, it first verifies that the processor is installed correctly and has a working FPU. Then it continuously repeats a series of tests that check for proper FPU operation. After each iteration, the result of the test is displayed as either "*Passed*" or "*Failed.*" The user can stop the test at any time, at which point a summary of all tests is then displayed. In Fig. 8, the flowchart for the MCPDIAG test program is given. With this program, the data obtained will reflect the susceptibility of the floating-point unit hardware to single event upsets.

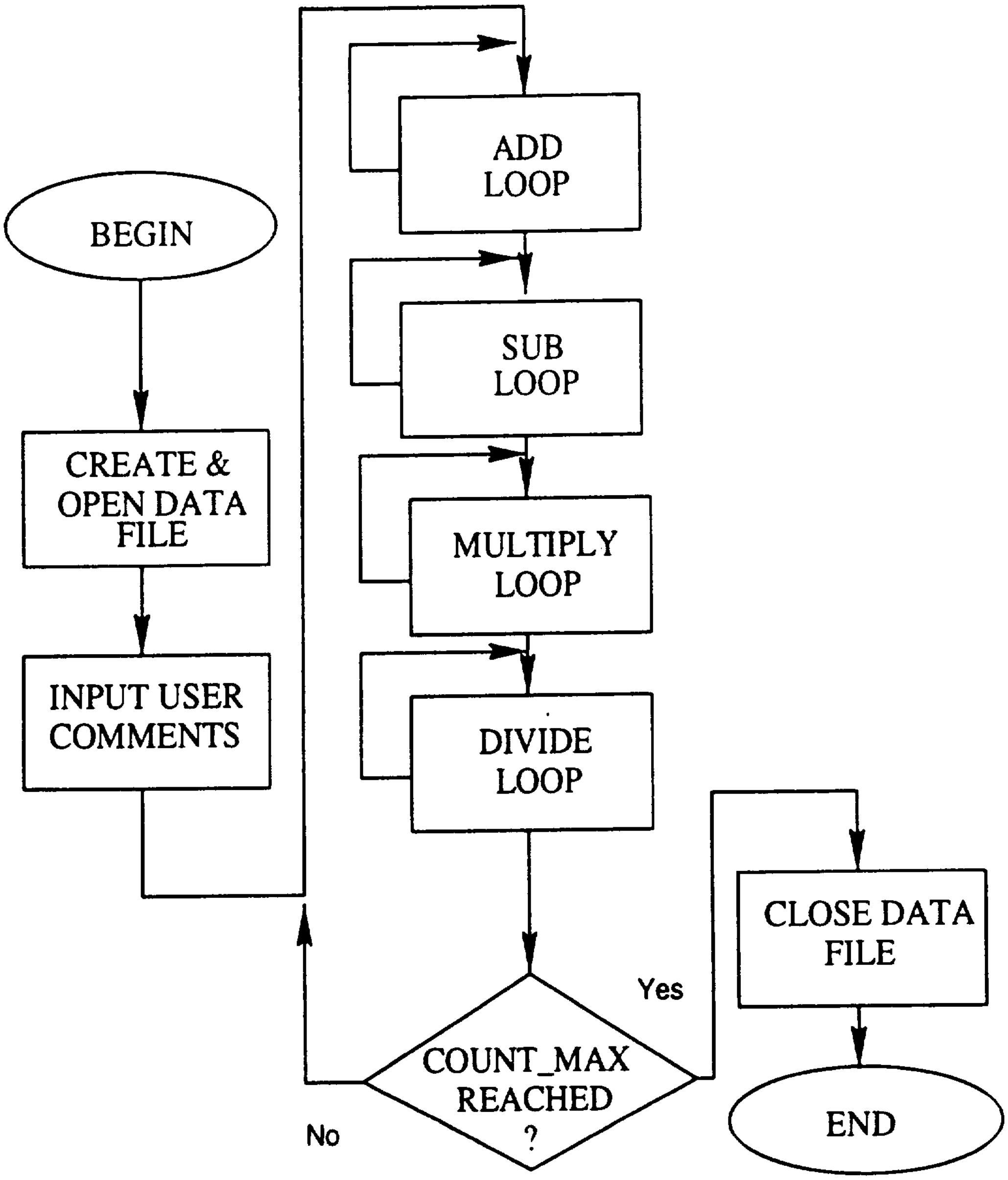


Figure 7: Flowchart for the *ALUTEST* program.

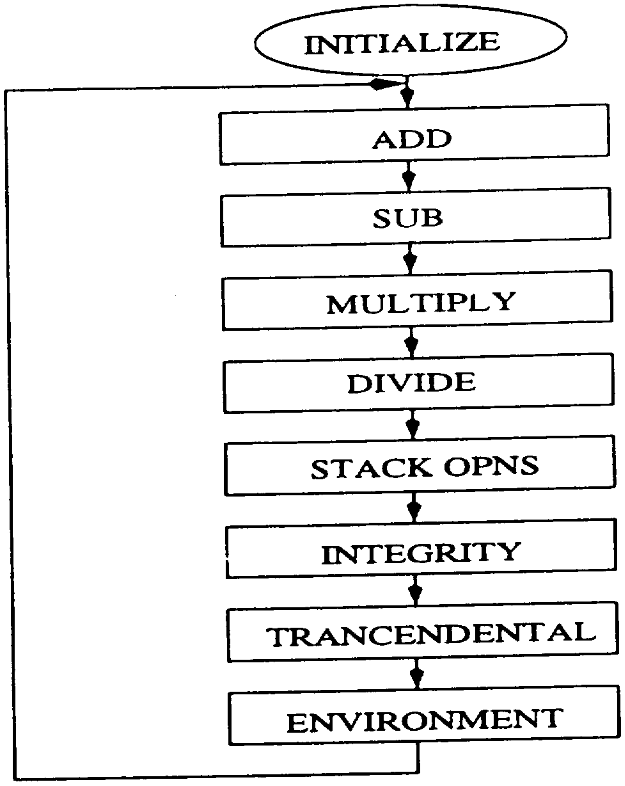


Figure 8: Flowchart for the MCPDIAG floating-point unit test program.

5.3 Test Plan

This section reviews the procedures used during the cyclotron test. The first action was to install the system board in the test chamber, set up the monitoring equipment at the Test Control Station, and install the interface cables. Next, the extender socket was attached to the system board. An operational checkout of the system was then made, by booting up and executing a few test programs. When the checkout was completed the test chamber was closed and de-pressurized. The vacuum pumps were turned on and in about fifteen minutes the operational pressure of 10^{-5} torr was achieved.

Before the first test could commence, the beam had to be calibrated and fine-tuned. This was achieved by the cyclotron personnel at the TCS. When it was determined the beam was within specifications, the experiment could begin.

The first test program to be used was loaded off the hard disk. The program execution was started before the ion beam was applied; since the software did not have to be synchronized with the beam it was easier to monitor the test if the program was already running in an operational mode before beam application.

The monitoring and detection took place by watching the video screen for proper program flow or erroneous program output, as well as monitoring the power supply current for signs of latchup. The test program stopped when it either normally terminated, was stopped by the experimenter, or when an error was detected.

Upon error detection, the first action was to halt the beam at the TCS. Depending on the program, error handling routines were activated, and the error information was entered into the test log. At the end of each test run or whenever an error was detected, the beam was stopped and a count of the particle fluence was recorded into the test log. If the processor was still functional after the error, and the test program was still capable of being restarted, the program continued and the ion beam was applied again until the next error.

After all test programs were run for both configurations of the L1-cache, the test chamber was vented and a new test device was installed. The same procedures were repeated for all test devices. Next, the ion beam was changed, and the same procedures were repeated until all devices hao been tested under all three beams. This sequence is summarized in a flowchart in Fig. 9.

In summary, the test variables were:

- □ 3 **Heavy-ion beams:** Xenon, Krypton, Argon
- □ 4 **Test devices:** two microprocessors from each vendor
- □ 2 **L1-cache configurations:** enabled versus disabled
- □ 3 **Test programs:** *REGTEST*, *ALUTEST*, and *MCPDIAG*
- □ **Indirect variables:** device temperature, vacuum pressure, and power supply voltages & currents

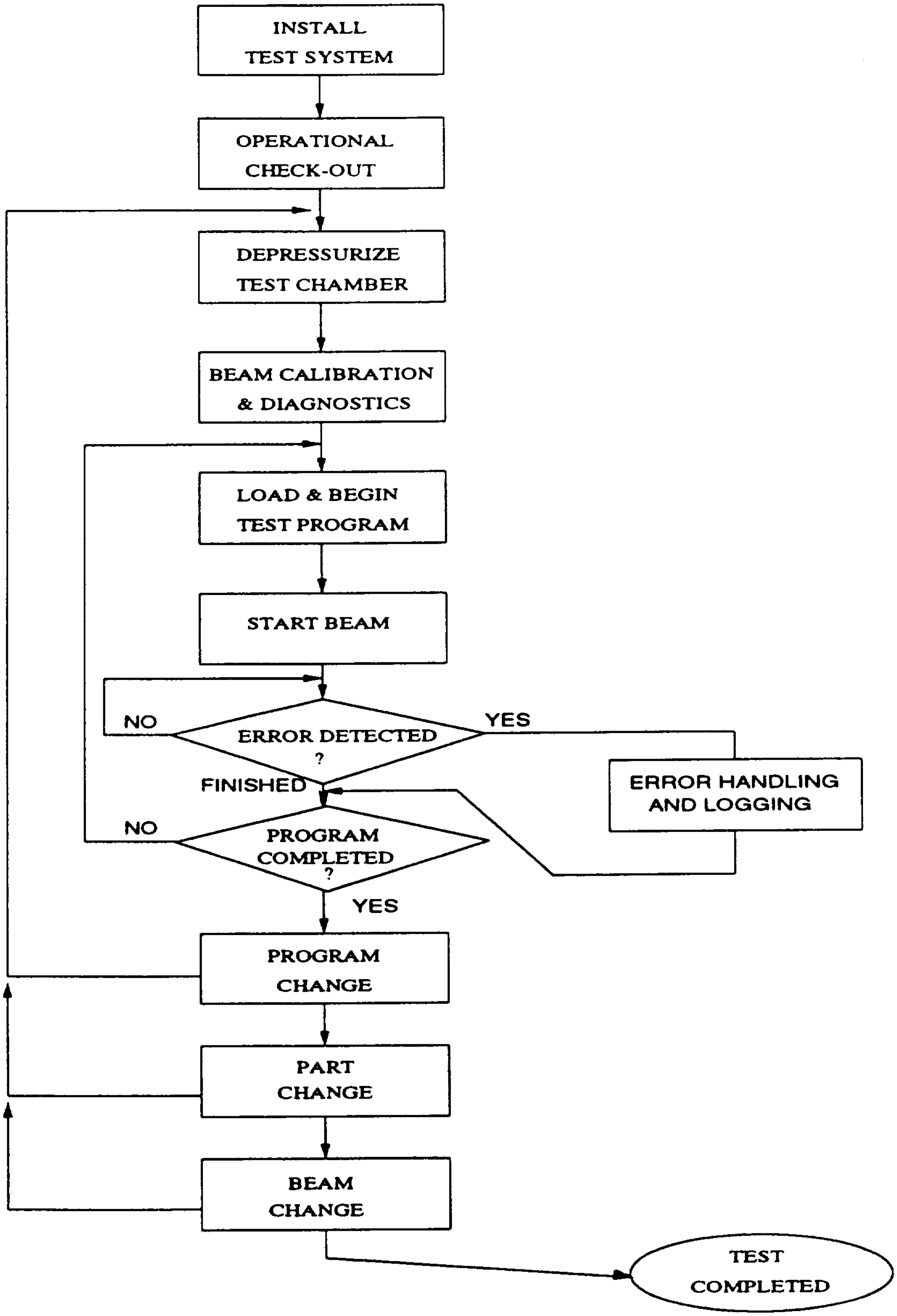


Figure 9: The Test Procedure flowchart for the 486-DX4 SEE test. The last three action blocks are not fully decomposed.

6.0 TEST RESULTS

6.1 Overview

The SEU testing took place on two separate dates, September 19 and November 26, 1996. Two beams, Xenon and Krypton, were run on the first date, and Argon was run on the latter date. A total of six microprocessor parts were tested producing a total of 234 data runs. The processors from AMD were tested a total of 88 times, and those from Intel a total of 146 times.

On average, each processor was tested about 23 times per beam. However, the first AMD part tested under Xenon experienced instantaneous latchup, and after only four data runs it was decided to forego further testing, and proceed to test the first Intel part, which were more successful. Table 3 breaks down the number of tests run on each part per beam.

Two parts from Intel were used in the experiment, and are denoted as Intel-1 and Intel-2. Four parts from AMD were tested, and are referred to as AMD-1 through AMD-4. Before testing with Argon, AMD-1 and AMD-2 were suspect of being permanently damaged by latchup, and were replaced with two identical parts, AMD-3 and AMD-4.

The complete test data log is given in Appendix B. It contains the test conditions, data parameters, failure signatures, and cross-sections obtained for each data run. Using this log, the raw test data was processed and analyzed to investigate the following issues:

(1) the observed error modes
(2) dependency on cache configuration
(3) dependency on the test program used
(4) comparison of performance between Intel and AMD
(5) upset cross-sections for each error mode
(6) SEU thresholds and saturated error rates
(7) SEL thresholds and behavior

Table 3: A breakdown of the number of tests run on each part per beam.

Test Processor	Xenon	Krypton	Argon
AMD-1	4	11	n/a
AMD-2	0	22	n/a
AMD-3	n/a	n/a	26
AMD-4	n/a	n/a	25
Intel-1	28	22	24
Intel-2	22	26	24

6.2 Observed Error Modes

The first step in analyzing the data was to identify a consistent set of error modes based on the failure signatures of each data run. Eight error modes were identified, and each data run was categorized into one of these modes. The eight error modes are presented in Table 4. In a few instances, two or more error modes could be identified for a particular data run, such as if a system error was detected right before a reboot occurred. In the following subsections, the details of each error mode are discussed and particular test cases are highlighted to help corroborate the 486-DX4's SEE behavior.

Table 4: The eight SEE error modes identified for the 486-DX4 microprocessor.

OBSERVED ERROR MODE	DESCRIPTION
SEU	SEU errors explicitly detected in the application register set by the REGTEST program. The error reporting subroutines were invoked and the erroneous word/byte in the affected register was reported
DATA ERROR	Errors detected in a data variable that did not cause a disruption in the program flow
PROGRAM FLOW ERROR	Errors that caused an abnormal flow in the program path, but did not cause the program to hang
PROGRAM HANG	Errors that caused the test program to crash and the processor to quit responding to further inputs; a manual reboot was required
FPU FAIL	An SEU error in the floating-point unit that was explicitly detected by the MCPDIAG program
SYSTEM ERROR	An upset that halted the processor due to a system-level error. System recovery was not possible; examples are internal stack overflow, divide by zero, memory allocation error, etc.
REBOOT	SEUs that caused the processor to initiate a system reboot
LATCHUP	Upsets that caused a latchup condition and was detected by a sudden increase in the power supply current

Program Hangs

Program hangs were observed on all test processors and test programs, and they were *the most frequent* error mode encountered. When a program hang was detected, the test program stopped executing and the entire system usually quit responding to further inputs. In nine times out of ten, the system had to be manually reset, but in a few cases a "CTRL-C" would allow the system to break to the C:\ prompt and the system still appeared to be functional. A manual reset was always performed to re-initialize the processor for the next test.

Whenever a program hang occurred, a "warm" reset was performed first as opposed to recycling the power with a "cold" reset. This warm reset was to help eliminate the possibility of an SEL condition. If SEL had occurred, the warm reset would not clear the latch and the subsequent reboot would fail. If this was the case, the power was recycled and the data run was classified as a latchup.

The error that caused the program hang could have manifested itself in a multitude of ways. If the program stack or instruction pointer (EIP register) was upset, the processor would lose track of the next instruction to execute and most likely attempt to process an invalid opcode. Another program hang scenario could occur if a memory address, either a segment value or an offset, was upset. The contents of the corrupted address may generate a number of exceptions or illegal accesses that could fail the processor. Another scenario could occur if an arithmetic or logic operand was upset, thus producing an erroneous result that halts the processor. Approximately 50% of the program hangs occurred with the internal L1-cache enabled and 50% with it disabled.

Latchups

The latchup error mode is defined as an upset caused by SEL, and was detected by the maximum current limit being exceeded on the power supply. The latchups were defined as either "*soft*," when the current slowly rose to the limit, or "*hard*," as when the current quickly jumped to the limit and the test program halted and the video monitor dimmed (due to excess current drain from the video card). Immediately after detecting a latchup, the power supplies were turned off and the beam stopped. Approximately two minutes were allowed until the power was turned back on.

As noted earlier, the first AMD part did not perform well under Xenon. Out of the four data runs taken, all were hard latchups that occurred within four seconds. Due to time constraints, the second AMD part was not tested under Xenon. Both AMD parts were tested under Krypton, and together they experienced latchups 47% of the time. With Argon, the AMD parts experienced latchups about 14% of the time.

The Intel parts faired much better against latchups, with *only one* latchup being detected on part Intel-2 under Xenon. *No other latchups were detected* on Intel's parts, and no permanent damage was observed.

SEU Errors

The SEU error mode could only be detected when the REGTEST program was being executed. A test was assigned this error mode when a miscompare was detected in one of the shaded registers of Fig. 2 (section 3.0). This detection would have occurred in the error-handling routine after the register failed the second compare against the test pattern. The register and byte location of the SEU was reported on the video monitor. Most of the time this upset was isolated to a single byte, but multiple upsets were detected in the same register in 40% of the SEU error cases. After one SEU, the program would resume checking all registers until the next error was detected.

A total of 38 SEU errors were detected out of all 234 data runs, and out of these only 9 occurred when the L1-cache was enabled, and 29 occurred with it disabled. No correlation between the upset rate and the test patterns could be determined from the data, that is no evidence exists to prove that writing all 1's or all 0's is more susceptible than the other. A breakdown of the percentage of SEUs detected for each register is given in Table 5. The data combines the errors across *all* beams for each vendor, and is a percentage of the total SEUs detected for each vendor.

In some executions of the REGTEST program, the instruction that contained the test pattern used to check the register contents was also upset. The test pattern value was a constant coded into the instruction, and it was sometimes upset during its residency on the chip. When the compare instruction was executed, a presumably good register was compared to a bad test pattern and a mismatch was flagged. The error-handling routine helped distinguish between true SEUs and bad test patterns by re-comparing the register in the error-handling routine. If this time the result was equal (hence compared by a different, good instruction) it was assumed that the register was not corrupted. This condition was then classified as a data error, described in the next subsection.

Table 5: The distribution of the 38 SEU errors detected in the 486-DX4's application register set for all beams. The errors were detected by the REGTEST program.

Percentage of SEUs detected in each register		
Register	**AMD**	**INTEL**
EAX	10 %	17.80 %
EBX	10 %	7.14 %
ECX	20 %	7.14 %
EDX	10 %	17.80 %
ESI	0 %	17.80 %
EDI	20 %	21.40 %
EBP	20 %	3.57 %
ES	10 %	0.00 %
FS	0 %	7.14 %
GS	0 %	0.00 %

Data Errors

A data error was defined as an upset in a data variable that was detected by erroneous output on the video monitor. The kinds of upsets were usually *single, isolated* errors, such as a wrong character being displayed, and did not cause a disruption in the program flow. One example of this type of error was the display of an incorrect loop iteration number, such as "loop iteration: 1&3" where the '&' was the upset location. Another example was the output message: "time: 00:00" being upset to: "time: 0d:00." In another case, only half of an output message was printed, probably due to an index pointer being upset. Some errors had an operational impact to the program, such as the case when an upset caused a loop delay variable to be changed to a higher value and effectively increased the delay time. In another test case, the maximum loop counter was upset from 256 to 400, thus extra loop iterations were performed.

Out of the 18 data errors detected, 16 occurred when the L1-cache was *enabled*. Most of these errors were due to an upset in an instruction or data variable that was stored in the cache, and the upset generally occurred before the instruction was processed. When the L1-cache was disabled all instructions and data (except in registers) resided in system DRAM, thus being somewhat protected from upsets.

Program-Flow Errors

A program-flow error was similar to a data error, except that the upset caused an abnormal flow in the program path such that proper execution could not continue. It differs from a program hang in that the processor did not crash. An example of a program-flow error is when the normal output on the video monitor was interrupted by stray ASCII characters. The processor continued to operate, but no meaningful program data was seen and a system reboot was required. In another test case, every other line of the program output contained erroneous spaces that were mixed in with the good program output.

Eighteen program-flow errors were detected and all but 2 occurred with the L1-cache enabled. As with the case of data errors, the cause of the error was presumably an instruction or data variable being upset in the cache or instruction pipeline, but the location and nature of these upsets were such that the program-flow was adversely affected.

System Errors and Processor Reboots

A system error was a type of upset that caused a system-level error to occur, which always caused the test program to abort and usually halted the processor. There were some cases where a "CTRL-C" input would break the system back to the C:\ prompt, but a warm boot was still always performed. The offending system error message was displayed on the video monitor, and a list of all the messages observed follows:

- DIVIDE OVERFLOW
- NO ROM BASIC: SYSTEM HALTED
- MEMORY ALLOCATION ERROR: SYSTEM HALTED
- INTERNAL STACK OVERFLOW: SYSTEM HALTED
- SECTOR NOT FOUND WRITING DRIVE C:\ A/R/I/F
- CANNOT ALLOCATE COMMAND.COM: SYSTEM HALTED

A reboot error was a special case of a system error in that it caused the microprocessor to initiate a self-reboot. Twelve system errors were detected, 9 with the cache enabled and 3 with it disabled. Five reboot errors were observed; only 1 with the cache on and 4 with it off.

Floating-Point Unit Errors

A FPU fail error is an upset that is explicitly detected in the floating-point unit by the MCPDIAG test program. If the program flags a certain FPU test as "failed," the data run was classified as an FPU fail. While the upset was actually detected in the FPU, it may have originated elsewhere and propagated into the FPU. Seven FPU errors were detected, with 2 occurring when the cache was enabled, and 5 when it was disabled.

Processing the ALUTEST output files

When an ALUTEST completed normally, the experimenter immediately halted the beam and the fluence was recorded. The term "normal termination" was entered in the test log. While the execution and termination of the program escaped any observable errors, it was not known if the output data files had been corrupted.

To determine if errors had occurred, an analysis program was written to compare each data file to an appropriate control file. These control files were generated by running ALUTEST in the absence of radiation and thus their results are error-free. The analysis program first opens both files and begins comparing the data, line by line, to the control file. If a mismatch was detected, the line numbers and mismatched data were written to another output file. After the entire file had been processed, the output file was then reviewed to see if and where any errors occurred, and what type of error mode was present.

To process the newly discovered upset, the byte location of the error in the data file was found, and using the total number of bytes in the file and the total fluence at program termination, interpolation was used to determine the fluence at the location of the error. This data was then amended to the test data log.

While the majority of ALUTEST executions resulted in program hangs, 20 executions produced "normally terminated" data files, and out of those, only 7 contained errors. From these 7 data runs, the only errors found were program flow errors and data errors.

6.3 The Error Mode Distribution

The distribution of error modes was almost identical for all devices of the same vendor, that is all AMD devices and all Intel devices exhibited nearly the same behavior for each beam. This indicates that the test devices were all good representative samples of each vendor's design.

For all test devices, the most common error mode observed was program hangs, however AMD exhibited a strong tendency for latchup, especially at the higher LET values. The combined error mode distribution for each vendor is presented in Table 6, and graphically in Figs. 10 and 11. It is important to note that the number of tests run on each processor was not the same, thus the percentages cannot be directly compared from one vendor to the other.

Table 6: The distribution of error mode percentages for each vendor over all three beams.

Distribution of Error Modes (in %)						
Error Mode	INTEL			AMD		
	Ar	*Kr*	*Xe*	*Ar*	*Kr*	*Xe*
SEU	14.8	18.2	17.3	15.2	2.9	0.0
data error	7.4	12.7	3.8	6.8	0.0	0.0
pgm flow err	3.7	7.3	11.5	6.8	2.9	0.0
pgm hang	68.5	50.9	55.8	45.8	41.2	0.0
FPU fail	3.7	1.8	1.9	3.4	2.9	0.0
system error	0.0	7.3	5.8	5.1	2.9	0.0
reboot	1.8	1.8	1.9	3.4	0.0	0.0
latchup	0.0	0.0	1.9	13.6	47.1	100.0

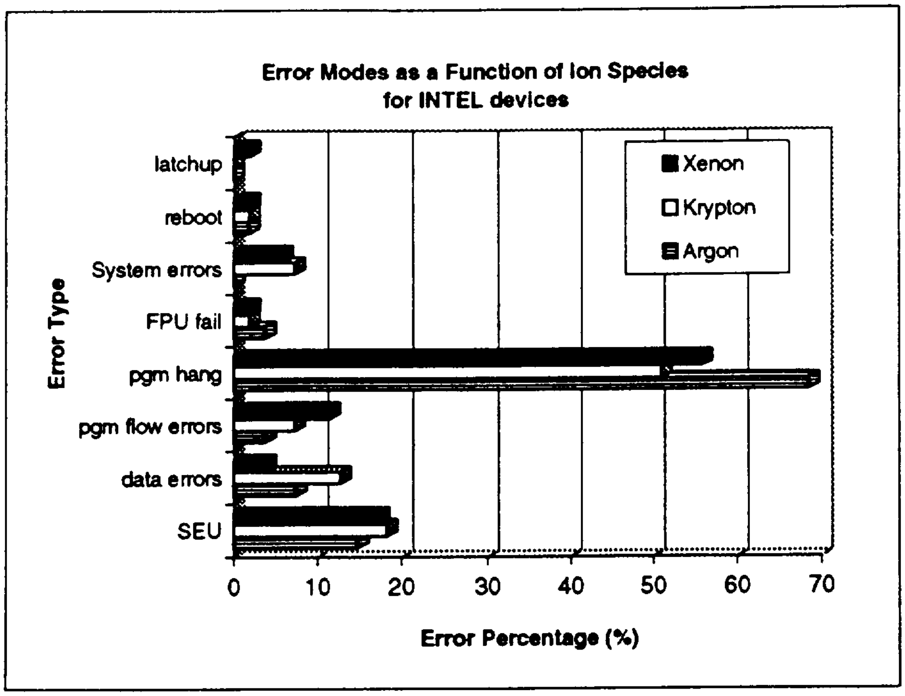


Figure 10: The error distribution percentages for all Intel data runs.

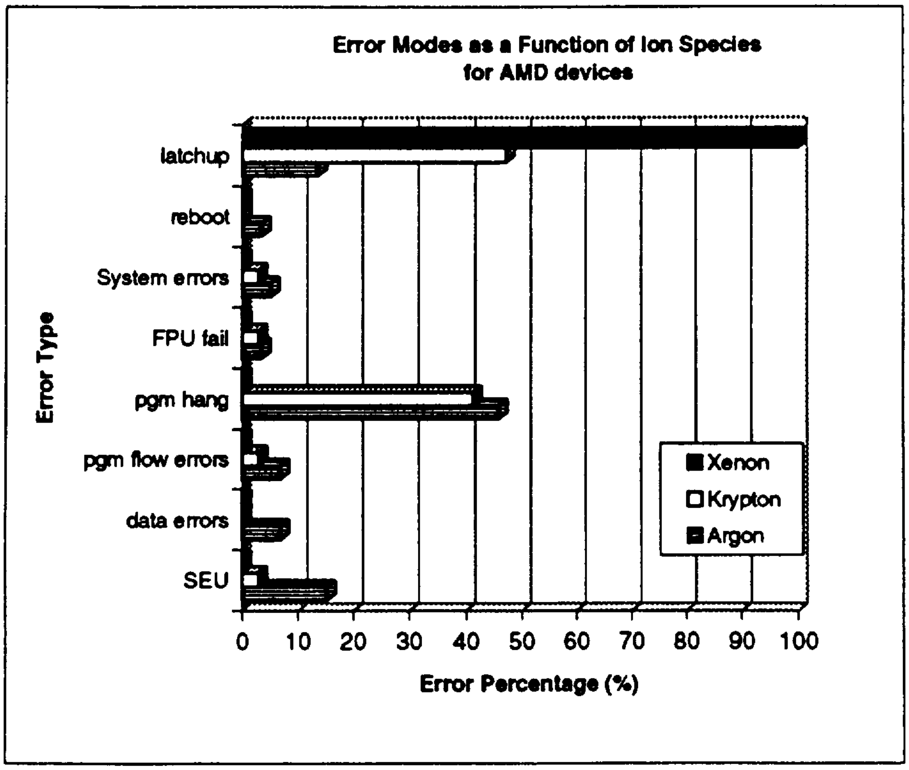


Figure 11: The error distribution percentages for all AMD data runs.

6.4 Calculating the Upset Cross-Section

The next major effort in analyzing the SEE test data was to compute the cross-sections for each error mode. These cross-sections represent the statistical susceptibility of each error mode to upsets, and can be related to upset rate. The cross-section is defined by: σ = (# of errors)/(total fluence), and is usually expressed in units of cm^2/device. The number of errors was usually one, but in some cases multiple upsets occurred and this number was higher. For the cross-section calculations, the total fluence is the total number of particles that a particular device was exposed to up to the *first occurrence* of *each* unique error. For example, if the first data error occurred on run #9, the total fluence for this particular error would be the sum of fluences for run numbers 1-9. After the first occurrence, the total fluence is then the number of particles between each subsequent occurrence of that error mode. By performing this analysis, the results give a representation of the device's susceptibility to each error mode. This could shed some light as to which error mode is more likely to occur in a particular operating configuration, or which aspect of the processor is more vulnerable to radiation.

6.4.1 Error Mode Cross-Sections

To calculate the error mode cross-sections, the first step was to group all the occurrences of each error mode in sequential order for each processor, and for each beam. Thus up to eight groups (the total number of error modes) could be generated for each test processor. This procedure was performed separately for each test device and for each beam.

After these groups are compiled, the total fluence is found for each error as described above. It is important to get the total number of particles between each unique error in a given error mode, because statistically, this is the number of particles the device "survived" before experiencing an upset of this type. When the total fluence is calculated for each error instance, the cross-section is obtained by dividing the number of errors (again usually one), by this total, summed fluence.

It should be noted that two of the error modes were program dependent, and could only be detected when certain test programs were executed. The SEU error mode could only be detected when the REGTEST program was running, and the FPU fail mode could only be detected when the MCPDIAG test program was running. When calculating the cross-sections for each of these errors, *only* the data runs that executed these programs can be counted in summing the total fluence.

An example of this cross-section analysis can be found below in Table 7. The data comes from device Intel-1 that was obtained under Xenon, and it shows the cross-section and total fluence values for the SEU and data error modes. Consult the test data log in Appendix B, and look for the first three SEU errors for this part, and compute the fluence between each SEU error.

Table 7: A partial list of the cross-section data points for Intel-1 under Xenon.

Partial SEU & Data Error Cross-Section Calculations for Intel-1 (under Xenon)							
Error Mode	#	Program Type	Run #	Total Run #s for cumulative fluence	Total cumulative fluence	Cross-section	L1 cache config.
SEU	1	FREG	9	6-9	10,170	9.83 E-5	on
	2	5REG	12	10-12	2,942	3.40 E-4	on
	3	FREG	25	13-16,24-25a	12,250	8.16 E-5	on
DATA ERROR	4	FREG	6	6	1,000	1.00 E-3	on
	5	AREG	14	7-14	12,552	7.97 E-5	on

Plotting Cross-Section versus LET Curves

After the upset cross-sections were calculated for all devices & beams, plots were made of the cross-section versus LET for each device. These plots represent the upset rates for the device at each LET value, and can be used for comparison to other devices. The combined plots for each vendor are shown below in Figs. 12 & 13. The LET values of the three beams again are: Xenon = 43.1, Krypton = 25.1, and Argon = 7.7 MeV cm^2/mg. Note that the x-coordinate values have been expanded around each LET value in order to allow an easier interpretation of the graph's legend. Several data labels in the legend are repeated due to the limitation with the graphing tool, but the top entry in each legend corresponds to the leftmost column of data for each beam, and proceeds down to the last graph legend corresponding to the rightmost data column.

From the cross-section plots, it can be seen that the variance within each data set is rather large; about one order of magnitude from the mean. The variance tends to become tighter as the LET increases. Also, the distribution within each error mode is fairly widespread. This may be attributed to the fact that the 486-DX4 is such a complex device, that there are a number of ways in which it could be upset for a particular error mode.

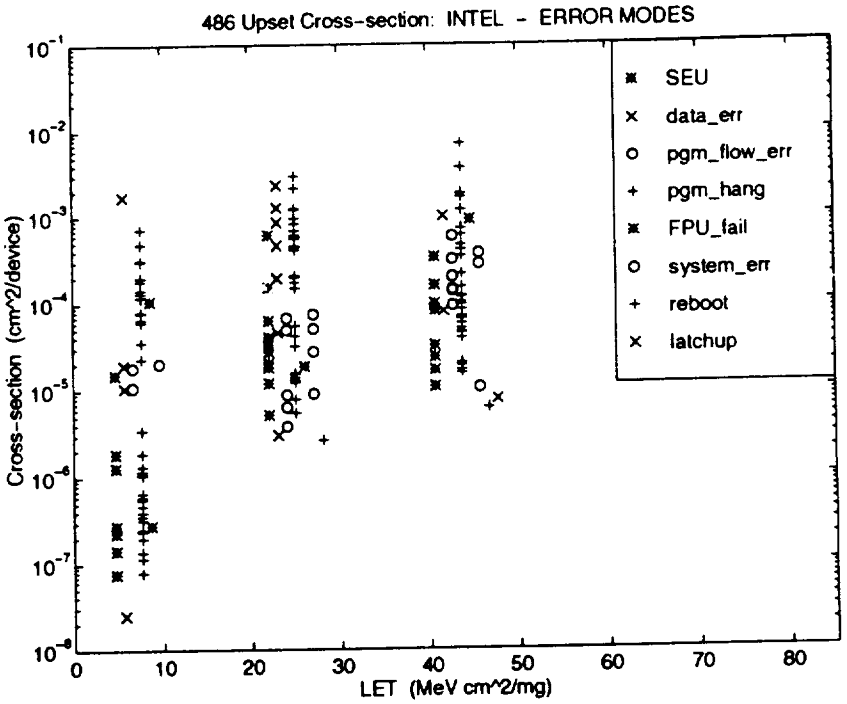


Figure 12: A combined plot of the upset cross-section versus LET for all Intel devices. Note that the LET values are: Xe=43.1, Kr=25.1, and Ar=7.7 MeV cm^2/mg, and that the x-axis has been expanded around each LET value to allow easier interpretation of the graph's legend.

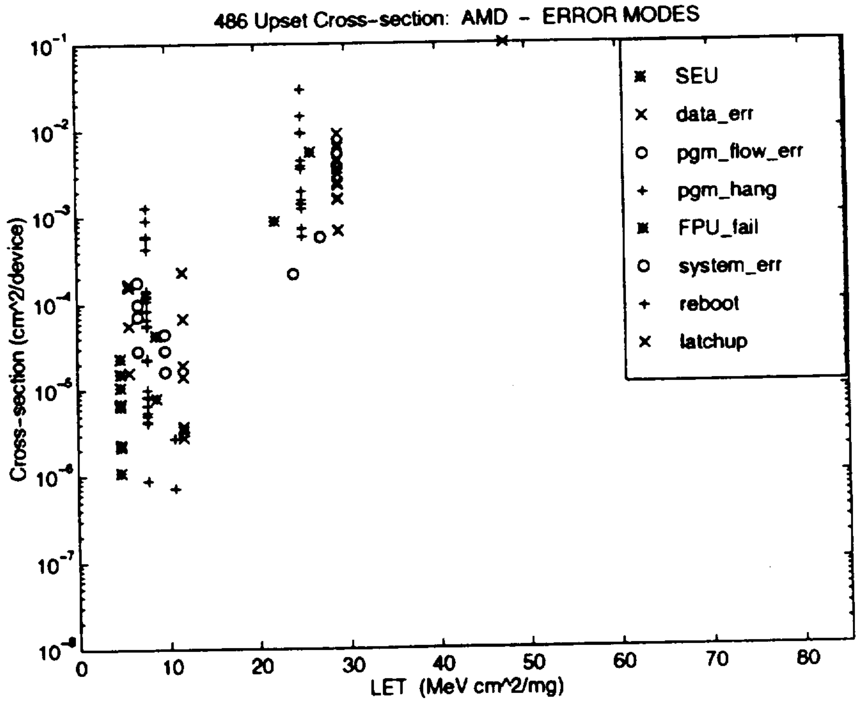


Figure 13: A combined plot of the upset cross-section versus LET for all AMD devices. Note that the LET values are: Xe=43.1, Kr=25.1, and Ar=7.7 MeV cm^2/mg, and that the x-axis has been expanded around each LET value to allow easier interpretation of the graph's legend.

6.4.2 Cache Dependency

It has already been shown that the 486-DX4 experiences a higher upset rate when the L1-cache is enabled. This is because program code and data reside onboard the processor, and are thus vulnerable to radiation upsets even before they are executed. The data used in the error mode upset cross-sections developed in the previous section has also been plotted to reflect the L1-cache configuration for each error. This information is given in Figs. 14 and 15, where the '+' represents an error while the cache was enabled, and an 'o' while the cache was disabled. From these plots, it is clear to see that higher SEU rates occur when the L1-cache is enabled versus disabled, by at least one order of magnitude.

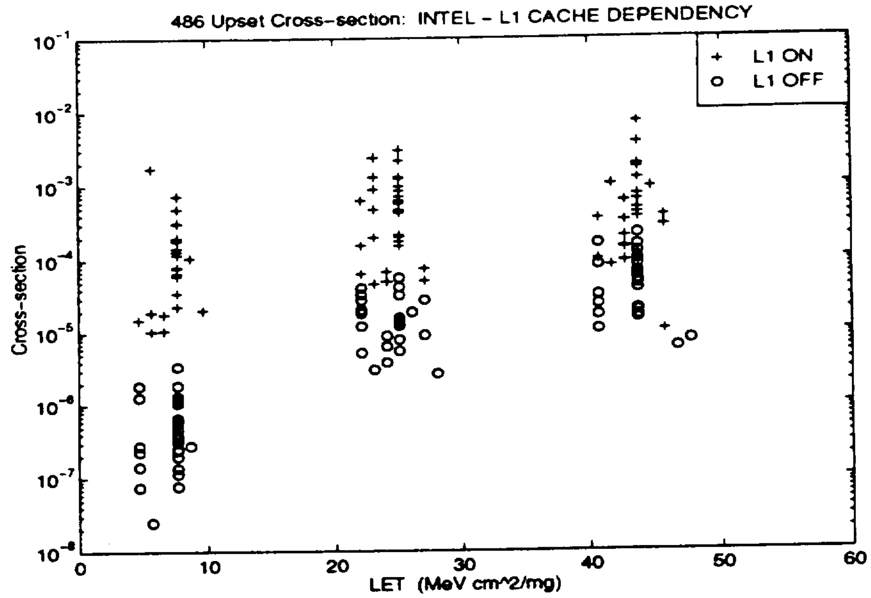


Figure 14: The dependency of the internal L1-cache on upset error rate for Intel devices.

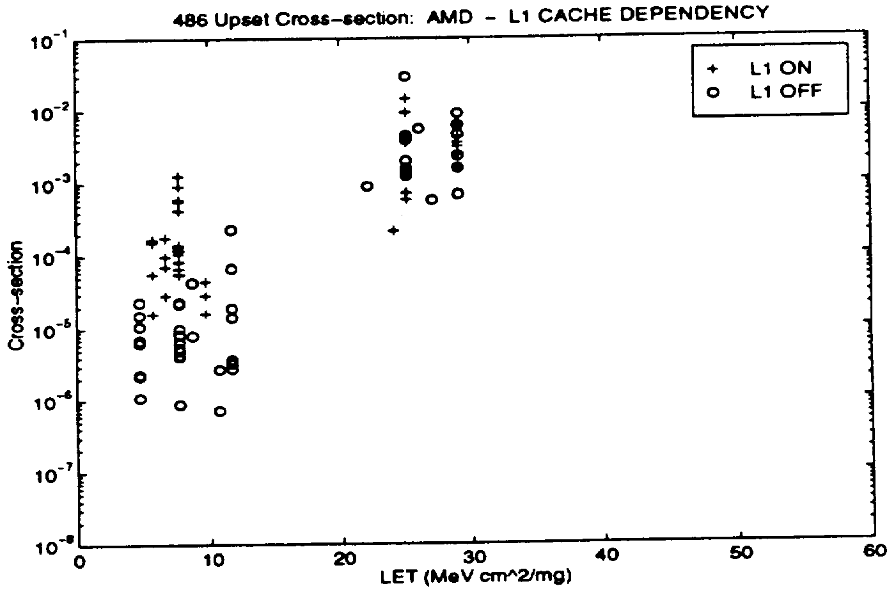


Figure 15: The dependency of the internal L1-cache on upset error rate for AMD devices.

6.4.3 Test Program Dependency

An investigation was made to determine if a clear dependency could be made between the upset rate and test programs that were used. By plotting the cross-section data in terms of which program was executed during each error, the results indicate that no consistent grouping could be made between the data points, thus no strong dependency on the test program versus the upset error rate could be made.

6.4.4 Log-Mean Average Cross-sections

In order to reduce the data to a more manageable form, the log-mean average was taken for each set of data to yield a statistical representation of the upset cross-section. Since there is such a wide difference between each cache configuration, the log-mean average was performed on each cache configuration for each beam. This was taken for all error modes for each beam with L1 enabled, and again for L1 disabled. This sequence was repeated for all devices and beams, and the log-mean average cross-sections are listed in Table 8. The data from this table was used to generate the log-mean average plots found in Figs. 16 and 17. A curve is fitted through the data points to get a better representation of the SEU behavior of the 486-DX4.

Table 8: The log-mean average upset cross-sections for each vendor. All the error modes have been combined for each cache configuration, beam, & device.

LOG-MEAN AVERAGE CROSS-SECTION DATA						
Vendor	Xenon		Krypton		Argon	
	L1 on	*L1 off*	*L1 on*	*L1 off*	*L1 on*	*L1 off*
AMD	n/a	n/a	1.4E-3	2.0E-3	5.7E-5	8.2E-6
INTEL	2.9E-4	3.2E-5	1.3E-4	1.0E-5	3.9E-5	2.5E-7

6.4.5 Estimating the SEU Threshold and Saturated Cross-Section

The log-mean average data in Figs. 16 and 17 was used to estimate the SEU threshold. From the fitted data points, **the SEU threshold for Intel appears to be 5.0 ±1.5 MeV-cm^2/mg**, and **for AMD the SEU threshold appears to be 3.0 ±1.5 MeV-cm^2/mg.** The plus and minus terms take into effect that the actual thresholds will be slightly lower when the L1-cache is enabled and slightly higher when it is disabled.

The *saturated cross-section* region can be determined by looking at where the slope of the curves go to zero. For Intel parts, these curves go to zero around 3.0E-4 cm^2/device when the cache is on, around 3.2E-5 cm^2/device when it is off. The saturated cross-sections for AMD tend to converge around 2.5E-3 cm^2/device, and this is probably due to the limiting factor being latchups at the higher LETs, regardless of the cache configuration.

From the experimental observations, AMD appears to have a SEL threshold of approximately 5.0 ±1.5 MeV-cm^2/mg, and for Intel it appears to be much higher, around 40.0 MeV-cm^2/mg.

In comparison to this research, the SEE test results on other related 80x86 devices obtained by different institutions are similar to those found here. JPL [27] tested the 80386 microprocessor and determined the SEU threshold LET (with heavy ions) to be 3.5 ± 1 MeV-cm^2/mg, and the SEL threshold to be 40 MeV-cm^2/mg. Goddard [29] tested the 486-DX2 and reported the SEU threshold LET to be around 5-6 MeV-cm^2/mg, and the SEL threshold to be around 59.6 MeV-cm^2/mg.

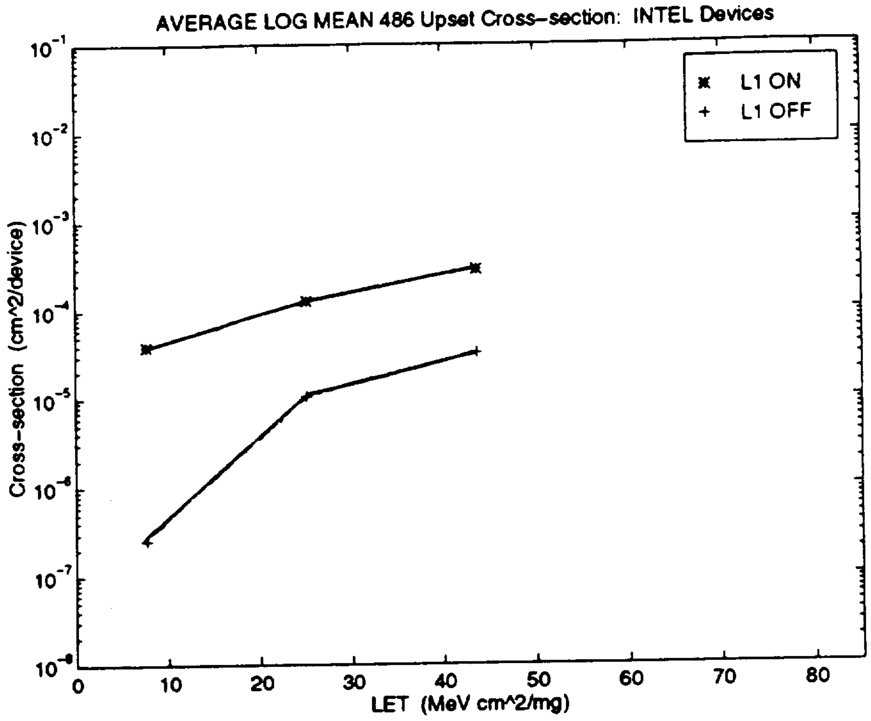


Figure 16: The log-mean average cross-section data for all **Intel** parts. Each point in the graph is the log-mean average of all error modes combined in each beam set, and for each cache configuration.

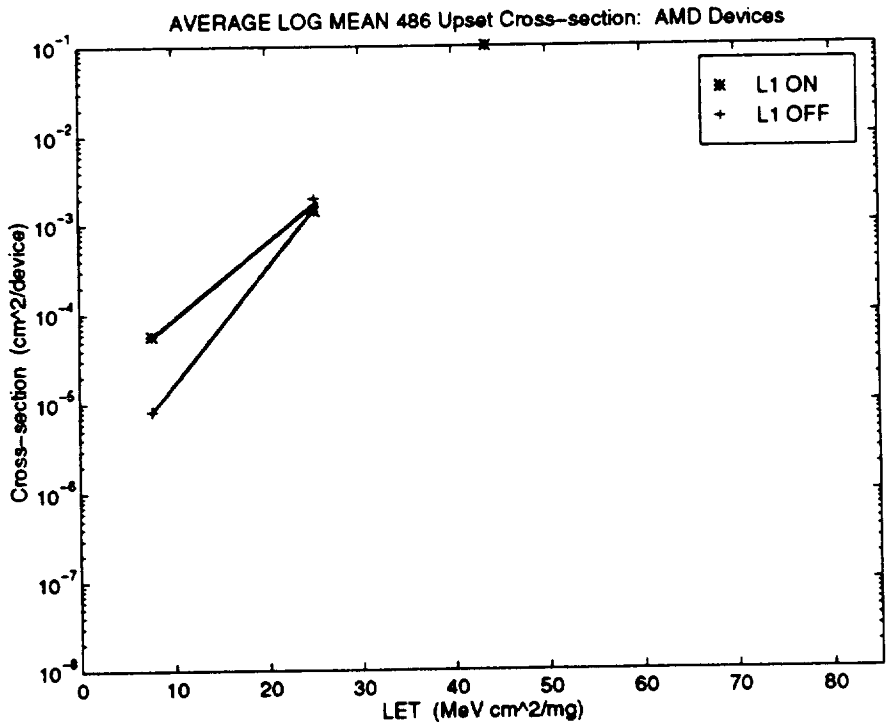


Figure 17: The log-mean average cross-section data for all **AMD** parts. Each point in the graph is the log-mean average of all error modes combined in each beam set, and for each cache configuration.

7.0 SPACEFLIGHT PREDICTIONS FOR THE 486-DX4

In order to provide an upset rate estimate and mean-time-between-failure (MTBF) prediction for the 486-DX4 operating in a space environment, a software program called "HIZ" was used. The algorithm combines experimental test results, device geometry, and environmental parameters of interest to compute an integrated, numerical estimation of the device's SEU rate, and is based on the methods similar to Chenette [2]. The orbital parameters used to compute these predictions were characteristic of the *International Space Station* environment; 51.6° inclination and 270 nm altitude. The HIZ program was used under the direction of Dr. Pat O'Neill at NASA's Johnson Space Center in Houston, Texas [15].

The key environmental inputs to HIZ were to compute the predictions at a solar minimum, to include galactic cosmic rays & heavy-ions, and to include solar flares. Next, the upset cross-section data obtained from the SEU testing was entered, as well as some additional 486-DX4 device information. HIZ assumes that each measured upset cross-section point is a step function and is equal to zero below the threshold. The algorithm takes each data point and integrates the upset cross-section over the given LET range, to produce error rates and MTBF predictions due to the contribution of that LET value. As each data point is processed, the new error rates and MTBF are integrated with the previous values.

The output of HIZ gives the error rate contribution at each LET value, the accumulated error rate, the total dose rate, and the MTBF for the device in terms of number of days until failure. The overall results from HIZ are summarized in Table 9, which presents the expected MTBF for each vendor, in each L1-cache configuration, and for two different shielding arrangements. The first shielding assumed 100 mils of aluminum, and the second used the shielding distribution found inside the Space Shuttle Orbiter. The shielding inside the Orbiter is much greater than 100 mils of aluminum, and it increases the MTBF by almost an order of magnitude. There is a vast difference between the results, especially between vendors, and this may be attributed to the high latchup susceptibility of AMD, as well as the smaller number of overall AMD data samples taken.

Table 9: The HIZ program results that predict the estimated SEU rate of the 486-DX4 in the International Space Station environment.

MTBF PREDICTIONS FOR THE 486-DX4 IN A SPACE STATION ORBIT
inclination: 51.6 degrees / altitude: 270 nmi

Shielding	INTEL		AMD	
	L1 on	*L1 off*	*L1 on*	*L1 off*
100 mil (AL)	4,150 days	333,039 days	377 days	282 days
Inside Orbiter	32,404 days	971,147 days	4,181 days	3,197 days

8.0 CONCLUSIONS

In this research, the importance of the 486-DX4 was addressed, and motivation was given to pursue SEE testing. A cyclotron-based experiment was devised and three heavy-ion sources were used to determine the SEE thresholds and saturation levels. All test goals were achieved and the major objectives were met. The results from this 486-DX4 experiment also agree with trends from previous 80x86 microprocessor SEU testing. From the experimental observations and of the analysis performed, the following conclusions can be drawn:

(1) The 486-DX4 is a very complex device that has many sensitive areas and many possible mechanisms in which a particle can cause an upset. The large variance in the data gives upper and lower bounds on the *operational performance* for different workloads, L1-cache configurations, and temporal spatial activity. The eight error modes identified represent some of the mechanisms in which an error is detected at the user level.

(2) The L1 internal cache plays a large role in determining the susceptibility of the 486-DX4, for there is over a magnitude of difference in the upset cross sections between cache configurations. When the cache is enabled, program hangs, data errors, and program flow errors can be frequently expected as code and data are vulnerable and easily upset in the cache. One possible recommendation for operation in the space environment would be to disable the L1-cache during intense periods of solar activity or when traversing the South Atlantic Anomaly (if the reduction in performance can be accepted).

(3) No distinct program-dependent relationship could be found on SEU rate.

(4) A significant difference in vendor performance was observed, with AMD exhibiting a much higher susceptibility to SEE as compared to Intel. In terms of latchup, AMD experienced SEL at all three LETs, and its SEU and SEL thresholds appear to be very close together. The reason for this could lie in AMD's fabrication process and smaller feature sizes.

(5) From the test data obtained in this research, the estimated SEU and SEL thresholds are:

Estimated Threshold (MeV-cm^2/mg)	**INTEL**	**AMD**
SEU	5.0 ±1.5	3.0 ±1.5
SEL	40.0	5.0 ±1.5

(6) The HIZ program was used to predict the MTBF for the 486-DX4 in the *International Space Station* environment. The results for both vendors indicated that there was over a ten-fold increase in MTBF when operating with the internal L1-cache disabled.

In the radiation testing community, it is common to classify newly tested parts for SEE into one of four categories. Category 1 recommends the part for all spaceflight operations. Category 2 recommends the part for spaceflight, but some SEE mitigation techniques may be required. Category 3 states the part is recommended for some operations, but extensive SEE preventative and recovery techniques are required, and Category 4 does not recommend the part for any spaceflight operations. For the 486-DX4, it appears that Intel may be placed in Category 3, and AMD placed in Category 4.

In conclusion, ground-based SEE testing is a valuable and necessary requirement for the complex digital devices used in highly-reliable systems. If current industry trends prevail, SEUs will continue to remain a viable threat to future digital systems as long as the device's size, weight, volume, cost and power are reduced, and the device's performance, density and complexity are increased. As highly-reliable systems become more complex, traditional design evaluation & validation techniques that rely on experience and prior knowledge become impractical. It is therefore imperative to obtain accurate upset rates and behavior data at both the component level and the system level. Using a cyclotron as a fault-injection source provides a realistic means of simulating a critical part of the space environment, and at a fraction of the cost of an actual spaceflight.

9.0 REFERENCES

[1] Penn Brumm and Don Brumm, 80486 Programming, Windcrest Books, Blue Ridge Summit, PA, First Edition, 1991.

[2] D. Chenette, T. Schumaker, A. Williamson, ``An accurate and efficient method to calculate the rate of single event upsets from the LET spectrum and SEU cross section test results," Lockheed Technical Report, PL-TR-92, Palo Alto, CA, March 20, 1992.

[3] G. Choi and R. Iyer, ``FOCUS: An experimental environment for fault sensitivity analysis," *IEEE Trans. on Computers*, Vol. 41, No. 12, pp. 1515-1526, December 1992.

[4] Gary Cox and Pat O'Neill, ``MC68000 Radiation Test Plan, Procedure, and Results," Flight Data Systems Division, NASA-Johnson Space Center, Houston, TX, March, 1994.

[5] J. Elder, J. Osborn, W. Kolasinski, and R. Koga, ``A method for characterizing a microprocessor's vulnerability to SEU," *IEEE Transactions on Nuclear Science*, Vol. 35, No. 6, December, 1988, pp. 1678-1681.

[6] V. Zajic, T. Fogarty, E. Stassinopulos, ``Single event upset and total dose radiation effects on rad-hard SRAMs," *Journal of Electronic Materials*, Vol. 19, No. 7, pp. 689-697, 1990.

[7] Saverio Gaudiano, Branch Chief/EV2, Avionic Systems Division, personal communication, NASA-JSC, Houston, TX, 1996.

[8] Ulf Gunneflo and Johan Karlsson, ``Latchup and SEU test of Motorola MC68020 32-bit microprocessor," Technical Report No. 60-Revision 2, Department of Computer Science, Chalmers University of Technology, Goteborg, Sweden, 1990.

[9] A. Holmes-Siedle and L. Adams, Handbook of Radiation Effects, Oxford Science Publications, New York, 1993.

[10] IntelDX4 Processor Data Book, INTEL Corporation, Mt. Prospect, IL, February, 1994.

[11] Johan Karlsson, Ulf Gunneflo, Jan Torin, ``Use of heavy-ion radiation from Californium-252 for fault-injection experiments," Department of Computer Science, Chalmers University of Technology, Goteborg, Sweden, 1990.

[12] Johan Karlsson, ``Reliability evaluation of a fault-tolerant computer for multi-phased mission and use of heavy-ion radiation for fault injection experiments," Technical Report No. 202, Department of Computer Science, Chalmers University of Technology, Goteborg, Sweden, 1990.

[13] D. Nichols, W. Price, C. Malone, ``A guideline for heavy ion radiation testing for single event upset (SEU)," JPL Publication 84-22, Jet Propulsion Laboratory, Pasadena, CA, January 1, 1984.

[14] Pat O'Neill, ``Space shuttle single event upsets experience," Orbiter & GFE Project Office, NASA-Johnson Space Center, Houston, TX, March 10, 1994.

[15] Pat O'Neill, Space Shuttle/Space Station Engineering Integration Office/EA44, personal communication, NASA-JSC, Houston, TX, 1996.

[16] ``Orbiter Avionics Radiation Handbook," JSC-23160, NASA-Johnson Space Center, Houston, TX, October 18, 1988.

[17] Burr Passenheim, How to Do Radiation Tests, Ingenuity Ink, San Diego, CA, 1988.

[18] R. Pease, A. Johnston, and J. Azarewicz, ``Radiation testing of semiconductor devices for space electronics," *Proc. of the IEEE*, Vol. 76, No. 11, pp. 1510-1526, November, 1988.

[19] Barry Johnson, Design and Analysis of Fault-Tolerant Digital Systems, Addison-Wesley Publishing Company, Reading, MA, pp. 2, 1989.

[20] R. Rasmussen, ``Spacecraft electronics design for radiation tolerance," *Proceedings of the IEEE*, Vol. 76, No 11, pp. 1527-1537, November 1988.

[21] R. C. Rogers, The Texas A&M University Cyclotron Institute Single Event Effect Facility, Texas A&M University, College Station, TX, 1996.

[22] Paul Schmisseur, ``Total dose and dose rate test of the Intel MQ80486DX microprocessor," Report No. 3-52400/4TR-087, LORAL Vought Systems, Dallas, TX, June, 1994.

[23] T. Scott, ``Single event test method and results for Intel's 80386," IBM # 89-PN6-005, Federal Systems Division, Manassas, VA, March 9, 1989.

[24] F.W. Sexton, ``Single event upset in space environments," a White Paper, Div. 2147, Radiation Technology and Assurance Division, Sandia National Labs, Albuquerque, NM, 1994.

[25] E. G. Stassinopoulos and J. Raymond, ``The space radiation environment for electronics," *Proceedings of the IEEE*, Vol 76, No 11, pp. 1423-1442, November 1988.

[26] Michael Thorne, Computer Organization and Assembly Language Programming: for IBM PCs and Compatibles, The Benjamin/Cummings Publishing Company, Inc., Menlo Park, CA, Second Edition, 1990.

[27] R. Watson, H. Schwartz, and D. Nichols, ``Test report for single event effects - 80386DX microprocessor," Technical Report, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, February, 1993.

[28] R. Woodruff and P. Rudeck, ``Three-dimensional numerical simulation of single event upset of an SRAM cell," *IEEE Transactions on Nuclear Science*, Vol. 40, No. 6, pp. 1795-1803, December, 1993.

World Wide Web sites:
[29] http://flick.gsfc.nasa.gov/radhome.htm
[30] http://radnet.jpl.nasa.gov

APPENDIX A - GLOSSARY

ALU	-	arithmetic logic unit
AMD	-	Advanced Micro Devices
Ar	-	Argon
ASICS	-	application-specific integrated circuits
CPU	-	central processing unit
cross-section	-	a statistical measure of a device's upset rate determined by the total fluence and number of upsets.
data run	-	one execution of a test program under radiation to generate a data point
DRAM	-	dynamic random access memory
eV	-	electron-Volt
flux	-	number of particles/cm^2 per second per device
fluence	-	total number of particles/cm^2 per device; the integral of flux over time
FPU	-	floating-point unit
GPC	-	general purpose computers (aboard the space shuttle)
HDD	-	hard disk drive
IC	-	integrated circuit
I/O	-	input/output
JSC	-	Johnson Space Center
latchup	-	a form of permanent circuit damage associated with a single event effect
LET	-	linear energy transfer; the rate that energy is deposited into a material per unit length
KB	-	Kilobyte (thousand bytes)
Kr	-	Krypton
LEO	-	low earth orbit
MB	-	Megabyte (million bytes)
NASA	-	National Aeronautics & Space Administration
nm	-	nautical mile
ns	-	nano-second
PC	-	personal computer
PGA	-	pin grid array
SAA	-	South Atlantic Anomaly
SEE	-	single event effects
SEL	-	single event latchup
SEU	-	single event upset
TAMU	-	Texas A&M University
TCS	-	Test Control Station
Xe	-	Xenon

APPENDIX B - TEST DATA SHEETS

(see attached)

486-DX4 MICROPROCESSOR SEU RADIATION TEST DATA

Coy Kouba
Dept. of Electrical Engr
Texas A&M Cyclotron Institute

September 19, 1996

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Mode	Fluence (particles/cm^2)	Cross-section
1	Xe	43.7	AMD-1	on	FREG	latchup	n/a	n/a
2	Xe	43.7	AMD-1	on	FREG	latchup	n/a	n/a
3	Xe	43.7	AMD-1	on	FREG	latchup	n/a	n/a
4	Xe	43.7	AMD-1	on	FREG	latchup	n/a	n/a
5	Xe	43.7	Intel-1	on	FREG	n/a	n/a	n/a
6	Xe	43.7	Intel-1	on	FREG	data error	1.00 E3	1.00 E-3
7	Xe	43.7	Intel-1	on	0REG	pgm flow error	4.00 E3	2.50 E-4
8	Xe	43.7	Intel-1	on	0REG	pgm hang	4.80 E3	2.08 E-4
9	Xe	43.7	Intel-1	on	FREG	SEU	0.37 E3	2.70 E-3
10	Xe	43.7	Intel-1	on	FREG	pgm hang	1.01 E3	9.90 E-4
11	Xe	43.7	Intel-1	on	5REG	pgm flow error	1.09 E3	9.17 E-4
12	Xe	43.7	Intel-1	on	5REG	SEU	0.842 E3	1.19 E-3
13	Xe	43.7	Intel-1	on	AREG	pgm hang	0.21 E3	4.76 E-3
14	Xe	43.7	Intel-1	on	AREG	data error	0.23 E3	4.35 E-3
15	Xe	43.7	Intel-1	on	AREG	pgm hang	1.39 E3	7.19 E-4
16	Xe	43.7	Intel-1	on	AREG	pgm flow error	0.46 E3	2.17 E-3
17	Xe	43.7	Intel-1	on	ALU	normal termination	0.27 E3	n/a
18	Xe	43.7	Intel-1	on	ALU	pgm hang	0.12 E3	8.33 E-3
19	Xe	43.7	Intel-1	on	ALU	pgm hang	0.14 E3	7.14 E-3
20	Xe	43.7	Intel-1	on	ALU	pgm hang	0.55 E3	1.82 E-3
21	Xe	43.7	Intel-1	on	MCPDIAG	pgm hang	0.60 E3	1.67 E-3
22	Xe	43.7	Intel-1	on	MCPDIAG	pgm hang	0.27 E3	3.70 E-3
23	Xe	43.7	Intel-1	on	MCPDIAG	FPU fail	0.22 E3	4.55 E-3
24	Xe	43.7	Intel-1	off	FREG	pgm hang	7.96 E3	1.26 E-4
25	Xe	43.7	Intel-1	off	FREG	SEU (a) pgm hang (b)	2.00 E3 22.2 E3	9.01 E-5 (used 22.2e3 w/ 2 errs)
26	Xe	43.7	Intel-1	off	0REG	pgm hang	49.6 E3	2.02 E-5
27	Xe	43.7	Intel-1	off	5REG	pgm hang	21.5 E3	4.65 E-5

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Signature	Fluence (particles/cm^2)	Cross-section
28	Xe	43.7	Intel-1	off	ALU	normal termination	18.0 E3	n/a
29	Xe	43.7	Intel-1	off	MCPDIAG	pgm hang	1.90 E3	5.26 E-4
30	Xe	43.7	Intel-1	off	MCPDIAG	pgm hang	15.5 E3	6.45 E-5
31	Xe	43.7	Intel-1	off	MCPDIAG	normal termination	9.11 E3	n/a
32	Xe	43.7	Intel-1	off	MCPDIAG	pgm hang	8.29 E3	1.21 E-4
33	Xe	43.7	Intel-2	on	FREG	system error	2.67 E3	3.74 E-4
34	Xe	43.7	Intel-2	on	FREG	pgm hang	0.16 E3	6.25 E-3
35	Xe	43.7	Intel-2	on	0REG	SEU (a) pgm hang (b)	0.10 E3 2.42 E3	8.26 E-4 (used 2.42e3 w/ 2 errs)
36	Xe	43.7	Intel-2	on	0REG	system error	0.99 E3	1.01 E-3
37	Xe	43.7	Intel-2	on	0REG	pgm flow error	4.60 E3	2.17 E-4
38	Xe	43.7	Intel-2	on	AREG	pgm flow error	1.70 E3	5.88 E-4
39	Xe	43.7	Intel-2	on	AREG	pgm flow error	6.98 E3	1.43 E-4
40	Xe	43.7	Intel-2	on	5REG	pgm hang	0.114 E3	8.77 E-3
41	Xe	43.7	Intel-2	on	5REG	pgm hang	0.534 E3	1.87 E-3
42	Xe	43.7	Intel-2	off	FREGC	pgm hang	6.63 E3	1.51 E-4
43	Xe	43.7	Intel-2	off	FREGC	pgm hang	4.62 E3	2.16 E-4
44	Xe	43.7	Intel-2	off	0REGC	SEU (a) pgm hang (b)	31.4 E3 53.1 E3	3.77 E-5 (used 5.31e4 w/ 2 errs)
45	Xe	43.7	Intel-2	off	0REGC	SEU (a) SEU (b) pgm hang (C)	9.91 E3 16.2 E3 25.8 E3	1.16 E-4 (used 2.58e4 w/ 3 errs)
46	Xe	43.7	Intel-2	off	AREGC	latchup	20.2 E3	4.95 E-5
47	Xe	43.7	Intel-2	off	AREGC	pgm hang / reboot	40.5 E3	2.47 E-5
48	Xe	43.7	Intel-2	off	5REGC	pgm hang	12.0 E3	8.33 E-5
49	Xe	43.7	Intel-2	off	5REGC	SEU (a) SEU (b) pgm hang (C)	11.7 E3 54.3 E3 65.2 E3	4.60 E-5 (used 65.2e3 w/ 3 errs)
50	Xe	43.7	Intel-2	off	ALUC	normal termination	7.65 E3	n/a
51	Xe	43.7	Intel-2	off	MCPDIAG	pgm hang	2.71 E3	3.69 E-4
52	Xe	43.7	Intel-2	off	ALUC	normal termination	6.31 E3	n/a
53	Xe	43.7	Intel-2	on	ALU	system error	1.81 E3	5.52 E-4
54	Xe	43.7	Intel-2	on	ALU	pgm hang	0.534 E3	1.87 E-3

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Signature	Fluence (particles/cm^2)	Cross-section
55	Kr	25.1	Intel-2	on	FREG	SEU / pgm hang	1.64 E3	6.10 E-4
56	Kr	25.1	Intel-2	on	FREG	data error	0.510 E3	1.96 E-3
57	Kr	25.1	Intel-2	on	FREG	pgm hung	1.29 E3	7.75 E-4
58	Kr	25.1	Intel-2	on	0REG	pgm hung	2.27 E3	4.40 E-4
59	Kr	25.1	Intel-2	on	0REG	data error	1.53 E3	6.53 E-4
60	Kr	25.1	Intel-2	on	0REG	SEU (a) pgm hang (b)	1.02 E3 1.20 E3	1.67 E-3 (used 1.20e3 w/ 2 errs)
61	Kr	25.1	Intel-2	on	AREG	data error	3.96 E3	2.52 E-4
62	Kr	25.1	Intel-2	on	AREG	data error system error	1.16 E3	8.62 E-4
63	Kr	25.1	Intel-2	on	5REG	pgm hang	0.661 E3	1.51 E-3
64	Kr	25.1	Intel-2	on	5REG	data error	0.124 E3	8.06 E-3
65	Kr	25.1	Intel-2	on	ALU	pgm hang	0.338 E3	2.96 E-3
66	Kr	25.1	Intel-2	on	ALU	data error (a) pgm flow error (b)	0.091 E3 0.234 E3	8.55 E-3 (used 234 w/ 2 errs)
67	Kr	25.1	Intel-2	on	ALU	pgm hang	1.22 E3	8.20 E-4
68	Kr	25.1	Intel-2	on	MCPDIAG	pgm hang	0.335 E3	2.98 E-3
69	Kr	25.1	Intel-2	on	MCPDIAG	pgm hang	2.16 E3	4.63 E-4
70	Kr	25.1	Intel-2	off	FREGC	pgm hang	0.24 E3	4.17 E-3
71	Kr	25.1	Intel-2	off	FREGC	system error	7.64 E3	1.31 E-4
72	Kr	25.1	Intel-2	off	0REGC	SEU (a) SEU (b) pgm hang (c)	12.0 E3 42.5 E3 75.6 E3	3.97 E-5 (used 75.6e3 w/ 3 errs)
73	Kr	25.1	Intel-2	off	AREGC	pgm flow error	3.22 E3	3.10 E-4
74	Kr	25.1	Intel-2	off	AREGC	system error	33.0 E3	3.03 E-5
75	Kr	25.1	Intel-2	off	5REGC	SEU (a) pgm hang (b)	14.4 E3 30.5 E3	6.56 E-5 (used 30.5e3 w/ 2 errs)
76	Kr	25.1	Intel-2	off	5REGC	SEU (a) pgm hang (b)	20.7 E3 76.6 E3	2.61 E-5 (used 76.6e3 w/ 2 errs)
77	Kr	25.1	Intel-2	off	ALUC	normal termination	20.2 E3	n/a
78	Kr	25.1	Intel-2	off	ALUC	normal termination	21.7 E3	n/a
79	Kr	25.1	Intel-2	off	ALUC	pgm hang	30.4 E3	3.29 E-5
80	Kr	25.1	Intel-2	off	MCPDIAG	pgm hang	31.2 E3	3.20 E-5

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Signature	Fluence (particles/cm^2)	Cross-section
81	Kr	25.1	Intel-1	on	FREG	pgm hang	0.795 E3	1.26 E-3
82	Kr	25.1	Intel-1	on	FREG	pgm hang	1.71 E3	5.85 E-4
83	Kr	25.1	Intel-1	on	0REG	pgm hang	0.829 E3	1.21 E-3
84	Kr	25.1	Intel-1	on	AREG	pgm hang	1.05 E3	9.52 E-4
85	Kr	25.1	Intel-1	on	AREG	pgm hang	2.40 E3	4.17 E-4
86	Kr	25.1	Intel-1	on	5REG	pgm hang	6.67 E3	1.50 E-4
87	Kr	25.1	Intel-1	on	5REG	SEU	2.42 E3	4.13 E-4
88	Kr	25.1	Intel-1	on	0REG	pgm hang	2.57 E3	3.89 E-4
89	Kr	25.1	Intel-1	on	ALU	system error	1.51 E3	6.62 E-4
90	Kr	25.1	Intel-1	on	ALU	pgm flow error	0.515 E3	1.94 E-3
91	Kr	25.1	Intel-1	on	MCPDIAG	data error	1.53 E3	6.53 E-4
92	Kr	25.1	Intel-1	on	ALU	pgm hang	1.19 E3	8.40 E-4
93	Kr	25.1	Intel-1	off	FREGC	pgm hang	64.3 E3	1.55 E-5
94	Kr	25.1	Intel-1	off	0REGC	SEU (a) SEU (b) pgm hang (c)	1.28 E5 1.53 E5 1.85 E5	1.62 E-5 (used 1.85e5 w/ 3 errs)
95	Kr	25.1	Intel-1	off	AREGC	pgm flow error	12.5 E3	8.00 E-5
96	Kr	25.1	Intel-1	off	5REGC	SEU (a) data error (b)	11.0 E3 72.0 E3	2.78 E-5 (used 72e3 w/ 2 errs)
97	Kr	25.1	Intel-1	off	5REGC	pgm hang / reboot	44.9 E3	2.23 E-5
98	Kr	25.1	Intel-1	off	ALUC	pgm flow error	42.5 E3	2.35 E-5
99	Kr	25.1	Intel-1	off	ALUC	normal termination	31.9 E3	n/a
100	Kr	25.1	Intel-1	off	ALUC	pgm hang	5.27 E3	1.90 E-4
101	Kr	25.1	Intel-1	off	MCPDIAG	pgm hang	18.1 E3	5.52 E-5
102	Kr	25.1	Intel-1	off	MCPDIAG	FPU fail	34.5 E3	2.90 E-5
103	Kr	25.1	AMD-1	on	FREG	latchup	6.36 E2	1.57 E-3
104	Kr	25.1	AMD-1	on	FREG	latchup	2.31 E2	4.33 E-3
105	Kr	25.1	AMD-1	on	0REG	latchup	1.46 E2	6.85 E-3
106	Kr	25.1	AMD-1	on	ALU	latchup	6.15 E2	1.63 E-3
107	Kr	25.1	AMD-1	on	ALU	pgm hang	0.76 E2	1.31 E-2
108	Kr	25.1	AMD-1	off	ALUC	system error / latchup	0.34 E2	2.94 E-2
109	Kr	25.1	AMD-1	off	FREGC	pgm hang	4.66 E2	2.14 E-3

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Signature	Fluence (particles/cm^2)	Cross-section
110	Kr	25.1	AMD-1	off	0REGC	pgm hang	6.37 E2	1.57 E-3
111	Kr	25.1	AMD-1	off	AREGC	pgm hang	0.34 E2	2.94 E-2
112	Kr	25.1	AMD-1	off	5REGC	pgm hang	2.23 E2	4.48 E-3
113	Kr	25.1	AMD-1	off	MCPDIAG	latchup	1.06 E2	9.43 E-3
114	Kr	25.1	AMD-2	off	FREGC	pgm hang	2.56 E2	3.90 E-3
115	Kr	25.1	AMD-2	off	FREGC	pgm hang	7.19 E2	1.39 E-3
116	Kr	25.1	AMD-2	off	0REGC	SEU (a) latchup (b)	1.47 E2 4.83 E2	4.14 E-3 (used 4.83e2 w/ 2 errs)
117	Kr	25.1	AMD-2	off	0REGC	pgm hang	3.22 E2	3.10 E-3
118	Kr	25.1	AMD-2	off	AREGC	latchup	1.06 E2	9.43 E-3
119	Kr	25.1	AMD-2	off	5REGC	latchup	1.60 E2	6.25 E-3
120	Kr	25.1	AMD-2	off	ALUC	latchup	2.15 E2	4.65 E-3
121	Kr	25.1	AMD-2	off	ALUC	pgm hang	2.27 E2	4.40 E-3
122	Kr	25.1	AMD-2	off	MCPDIAG	FPU fail	1.79 E2	5.59 E-3
123	Kr	25.1	AMD-2	off	MCPDIAG	latchup	2.21 E2	4.52 E-3
124	Kr	25.1	AMD-2	on	MCPDIAG	latchup	3.16 E2	3.16 E-3
125	Kr	25.1	AMD-2	on	ALU	latchup	4.14 E2	2.42 E-3
126	Kr	25.1	AMD-2	on	ALU	normal termination	1.05 E2	n/a
127	Kr	25.1	AMD-2	on	ALU	pgm hang	1.50 E2	6.67 E-3
128	Kr	25.1	AMD-2	on	FREG	latchup	0.22 E2	4.54 E-2
129	Kr	25.1	AMD-2	on	FREG	pgm hang	2.56 E2	3.90 E-3
130	Kr	25.1	AMD-2	on	0REG	pgm hang	0.69 E2	1.45 E-2
131	Kr	25.1	AMD-2	on	0REG	latchup	0.72 E2	1.39 E-2
132	Kr	25.1	AMD-2	on	AREG	pgm hang	0.34 E2	2.94 E-2
133	Kr	25.1	AMD-2	on	AREG	pgm hang	1.08 E2	9.26 E-3
134	Kr	25.1	AMD-2	on	5REG	latchup	0.17 E2	5.88 E-2
135	Kr	25.1	AMD-2	on	5REG	pgm flow error	1.47 E2	6.80 E-3

486-DX4 MICROPROCESSOR SEU RADIATION TEST DATA

Coy Kouba
Dept. of Electrical Engr
Texas A&M Cyclotron Institute

November 27, 1996

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Mode	Fluence (particles/cm^2)	Cross-section
136	Ar	7.7	AMD-3	on	FREG	pgm hang	1.20 E4	8.33 E-5
137	Ar	7.7	AMD-3	on	FREG	data error	5.82 E3	1.72 E-4
138	Ar	7.7	AMD-3	on	0REG	pgm hang	3.47 E3	2.88 E-4
139	Ar	7.7	AMD-3	on	0REG	data error	3.01 E3	3.32 E-4
140	Ar	7.7	AMD-3	on	AREG	pgm hang	3.97 E3	2.52 E-4
141	Ar	7.7	AMD-3	on	AREG	data error	1.99 E3	5.02 E-4
142	Ar	7.7	AMD-3	on	5REG	pgm flow error	5.02 E3	1.99 E-4
143	Ar	7.7	AMD-3	on	5REG	pgm hang	8.03 E3	1.24 E-4
144	Ar	7.7	AMD-3	on	ALU	pgm flow error	6.02 E3	1.66 E-4
145	Ar	7.7	AMD-3	on	ALU	pgm hang	2.46 E3	4.06 E-4
146	Ar	7.7	AMD-3	on	MCPDIAG	pgm hang	1.66 E3	6.02 E-4
147	Ar	7.7	AMD-3	on	MCPDIAG	pgm flow error	6.01 E3	1.66 E-4
148	Ar	7.7	AMD-3	on	FREG	system error	2.14 E3	4.67 E-4
149	Ar	7.7	AMD-3	on	0REG	pgm flow error	3.52 E3	2.84 E-4
150	Ar	7.7	AMD-3	off	FREGC	pgm hang	1.72 E5	5.81 E-6
151	Ar	7.7	AMD-3	off	FREGC	latchup	3.33 E4	3.00 E-5
152	Ar	7.7	AMD-3	off	0REGC	latchup	7.06 E4	1.42 E-5
153	Ar	7.7	AMD-3	off	0REGC	latchup	1.49 E4	6.71 E-5
154	Ar	7.7	AMD-3	off	AREGC	SEU (a) pgm hang (b)	9.55 E4 1.14 E5	1.75 E-5 (used 1.14e5 w/ 2 errs)
155	Ar	7.7	AMD-3	off	AREGC	SEU (a) pgm hang (b)	2.47 E4 4.51 E4	4.43 E-5 (used 4.51e4 w/ 2 errs)
156	Ar	7.7	AMD-3	off	5REGC	SEU (a) latchup (b)	7.29 E4 1.16 E5	1.72 E-5 (used 1.16e5 w/ 2 errs)
157	Ar	7.7	AMD-3	off	5REGC	pgm hang	3.63 E3	2.75 E-4
158	Ar	7.7	AMD-3	off	ALUC	normal termination	3.54 E4	n/a
159	Ar	7.7	AMD-3	off	ALUC	latchup	1.31 E4	7.63 E-5
160	Ar	7.7	AMD-3	off	MCPDIAG	pgm hang	5.17 E4	1.93 E-5

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Mode	Fluence (particles/cm^2)	Cross-section
161	Ar	7.7	AMD-3	off	MCPDIAG	FPU fail	6.63 E4	1.51 E-5
162	Ar	7.7	AMD-4	on	FREG	pgm hang	2.34 E3	4.27 E-4
163	Ar	7.7	AMD-4	on	FREG	pgm hang	8.09 E3	1.24 E-4
164	Ar	7.7	AMD-4	on	0REG	system error	1.22 E4	8.20 E-5
165	Ar	7.7	AMD-4	on	0REG	pgm hang	5.43 E3	1.84 E-4
166	Ar	7.7	AMD-4	on	AREG	pgm hang	1.10 E3	9.09 E-4
167	Ar	7.7	AMD-4	on	AREG	pgm hang	7.85 E2	1.27 E-3
168	Ar	7.7	AMD-4	on	5REG	pgm hang	1.21 E4	8.26 E-5
169	Ar	7.7	AMD-4	on	5REG	pgm hang	7.40 E3	1.35 E-4
170	Ar	7.7	AMD-4	on	ALU	pgm hang	1.75 E3	5.71 E-4
171	Ar	7.7	AMD-4	on	ALU	system error	6.60 E3	1.51 E-4
172	Ar	7.7	AMD-4	on	ALU	normal termination	2.37 E3	n/a
173	Ar	7.7	AMD-4	on	MCPDIAG	data error	3.20 E3	3.12 E-4
174	Ar	7.7	AMD-4	on	MCPDIAG	pgm hang	5.91 E3	1.69 E-4
175	Ar	7.7	AMD-4	off	FREGC	pgm hang	2.44 E5	4.10 E-6
176	Ar	7.7	AMD-4	off	FREGC	latchup	4.91 E4	2.04 E-5
177	Ar	7.7	AMD-4	off	0REGC	latchup	4.33 E3	2.31 E-4
178	Ar	7.7	AMD-4	off	0REGC	latchup	3.04 E5	3.29 E-6
179	Ar	7.7	AMD-4	off	AREGC	SEU (a) SEU (b) SEU (c) pgm hang / reboot (d)	2.67 E5 4.12 E5 4.78 E5 7.82 E5	5.11 E-6 (used 7.82e5 w/ 4 errs)
180	Ar	7.7	AMD-4	off	AREGC	SEU (a) SEU (b) pgm hang (c)	1.57 E5 " 2.02 E5	1.48 E-5 (used 2.02e5 w/ 3 errs)
181	Ar	7.7	AMD-4	off	5REGC	pgm hang	4.38 E4	2.28 E-5
182	Ar	7.7	AMD-4	off	5REGC	SEU (a) pgm hang/reboot (b)	6.81 E4 1.23 E5	1.63 E-5 (used 1.23e5 w/ 2 errs)
183	Ar	7.7	AMD-4	off	ALUC	pgm hang	4.34 E4	2.30 E-5
184	Ar	7.7	AMD-4	off	ALUC	normal termination	8.02 E4	n/a
185	Ar	7.7	AMD-4	off	MCPDIAG	FPU fail	1.41 E4	7.09 E-5
186	Ar	7.7	AMD-4	off	MCPDIAG	pgm hang	6.00 E4	1.67 E-5
187	Ar	7.7	Intel-1	on	FREG	pgm hang	3.26 E3	3.07 E-4

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Mode	Fluence (particles/cm^2)	Cross-section
188	Ar	7.7	Intel-1	on	FREG	pgm hang	5.52 E3	1.81 E-4
189	Ar	7.7	Intel-1	on	0REG	pgm hang	5.59 E3	1.79 E-4
190	Ar	7.7	Intel-1	on	0REG	pgm hang	1.54 E4	6.49 E-5
191	Ar	7.7	Intel-1	on	AREG	pgm hang	3.18 E3	3.14 E-4
192	Ar	7.7	Intel-1	on	AREG	pgm hang	5.04 E3	1.98 E-4
193	Ar	7.7	Intel-1	on	5REG	pgm hang	8.51 E3	1.17 E-4
194	Ar	7.7	Intel-1	on	5REG	pgm hang	2.07 E3	4.83 E-4
195	Ar	7.7	Intel-1	on	ALU	system error / reboot	1.26 E3	7.94 E-4
196	Ar	7.7	Intel-1	on	ALU	data error (a) pgm flow error (b) normal termination(c)	2.13 E3 5.80 E3 7.81 E3	3.45 E-4 (used 5.80e3 w/ 2 errs)
197	Ar	7.7	Intel-1	on	MCPDIAG	pgm hang	3.95 E3	2.53 E-4
198	Ar	7.7	Intel-1	on	MCPDIAG	pgm hang	1.39 E3	7.19 E-4
199	Ar	7.7	Intel-1	off	FREGC	SEU	4.95 E5	2.02 E-6
200	Ar	7.7	Intel-1	off	FREGC	pgm hang	1.65 E6	6.06 E-7
201	Ar	7.7	Intel-1	off	0REGC	SEU (a) pgm hang (b)	1.98 E6 5.13 E6	3.90 E-7
202	Ar	7.7	Intel-1	off	0REGC	pgm hang	2.95 E5	3.39 E-6
203	Ar	7.7	Intel-1	off	AREGC	pgm hang	4.00 E6	2.50 E-7
204	Ar	7.7	Intel-1	off	AREGC	pgm hang	1.79 E6	5.59 E-7
205	Ar	7.7	Intel-1	off	5REGC	pgm hang	5.37 E5	1.86 E-6
206	Ar	7.7	Intel-1	off	5REGC	pgm hang	9.20 E5	1.09 E-6
207	Ar	7.7	Intel-1	off	ALUC	pgm hang	8.51 E5	1.18 E-6
208	Ar	7.7	Intel-1	off	ALUC	normal termination	2.11 E6	n/a
209	Ar	7.7	Intel-1	off	MCPDIAG	pgm hang	2.10 E6	4.76 E-7
210	Ar	7.7	Intel-1	off	MCPDIAG	FPU fail	1.52 E6	6.58 E-7
211	Ar	7.7	Intel-2	on	FREG	pgm hang	5.10 E3	1.96 E-4
212	Ar	7.7	Intel-2	on	FREG	pgm hang	1.26 E4	7.94 E-5
213	Ar	7.7	Intel-2	on	0REG	pgm hang	1.65 E4	6.06 E-5
214	Ar	7.7	Intel-2	on	0REG	pgm hang	2.85 E4	3.51 E-5
215	Ar	7.7	Intel-2	on	AREG	SEU (a) pgm flow error (b)	4.37 E3 3.10 E4	6.45 E-5

Run#	Ion	LET (MeV-cm^2/mg)	Test Device	L1 cache on/off	Test Program	Failure Mode	Fluence (particles/cm^2)	Cross-section
216	Ar	7.7	Intel-2	on	AREG	data error	1.42 E3	7.04 E-4
217	Ar	7.7	Intel-2	on	5REG	data error	5.82 E2	1.72 E-3
218	Ar	7.7	Intel-2	on	5REG	pgm hang	1.09 E4	9.17 E-5
219	Ar	7.7	Intel-2	on	ALU	normal termination	6.18 E3	n/a
220	Ar	7.7	Intel-2	on	ALU	pgm hang	8.16 E2	1.22 E-3
221	Ar	7.7	Intel-2	on	MCPDIAG	pgm hang	7.65 E3	1.31 E-4
222	Ar	7.7	Intel-2	on	MCPDIAG	FPU fail	1.82 E3	5.49 E-4
223	Ar	7.7	Intel-2	off	FREGC	pgm hang	7.33 E6	1.36 E-7
224	Ar	7.7	Intel-2	off	FREGC	SEU (a) SEU (b) pgm hang (c)	5.98 E6 6.76 E6 8.75 E6	3.43 E-7
225	Ar	7.7	Intel-2	off	0REGC	SEU (a) pgm hang (b)	2.42 E6 2.79 E6	7.17 E-7
226	Ar	7.7	Intel-2	off	0REGC	pgm hang	1.29 E7	7.75 E-8
227	Ar	7.7	Intel-2	off	AREGC	SEU (a) SEU (b) pgm hang (c)	7.62 E5	3.94 E-6
228	Ar	7.7	Intel-2	off	AREGC	pgm hang	1.70 E6	5.88 E-7
229	Ar	7.7	Intel-2	off	5REGC	pgm hang	1.53 E6	6.54 E-7
230	Ar	7.7	Intel-2	off	5REGC	pgm hang	3.13 E6	3.19 E-7
231	Ar	7.7	Intel-2	off	ALUC	pgm hang	9.44 E5	1.06 E-6
232	Ar	7.7	Intel-2	off	ALUC	data error (a) normal termination(b)	5.60 E5 1.74 E6	1.78 E-6
233	Ar	7.7	Intel-2	off	MCPDIAG	pgm hang	8.16 E5	1.22 E-6
234	Ar	7.7	Intel-2	off	MCPDIAG	pgm hang	4.10 E6	2.44 E-7
235	Ar	7.7	Intel-2	on	Windows3.1/WARP	application error	3.22 E3	3.11 E-4
236	Ar	7.7	Intel-2	on	BURNIT	pgm hang	immediate	n/a
237	Ar	7.7	Intel-2	on	Windows3.1/Write	application error	immediate	n/a

APPENDIX C - TEST SOFTWARE SOURCE CODE

The complete source code may be obtained by contacting the author:

Coy Kouba
Avionic Systems Division \ EV211
NASA - Johnson Space Center
Houston, TX 77058

(281) 483-8069
CKouba@ems.jsc.nasa.gov

APPENDIX D - THE SPACE RADIATION ENVIRONMENT

The space radiation environment is a very hostile and dangerous place, especially for spacecraft electronics. The environment consists of a diverse suite of radiation particles with energies ranging from the kilo electron-volts[1] (KeV) to GeV and greater. These particles are either trapped in the Earth's magnetic field or are transiting the solar system through Earth's domain [25]. The transiting radiation is comprised of a solar contribution and a galactic contribution. As reported in Holmes-Sielde [9], the three main components of the space radiation environment are:

1. Trapped radiation: a very broad spectrum of energetic charged particles trapped in the Earth's magnetosphere (which includes the magnetic fields and radiation belts).

2. Cosmic rays: energetic heavy-ions of low concentration (called flux) that includes all ions in the periodic table with energies exceeding the TeV range.

3. Solar flares: intense solar eruptions that emit energetic protons with energies up to several hundred MeV, and includes small amounts of alpha particles, heavy-ions, and electrons.

Also found throughout space is a continuous plasma of electrons and protons with energies up to 100 KeV at high fluxes (up to $1x10^{12}$ $cm^{-2}s^{-1}$). Within the Earth's radiation belts, these particles are seen as the low-energy population of trapped particles, and at the fringe of the magnetosphere they are associated with the solar wind and are found in high flux concentrations. The magnetosphere constitutes that region surrounding the Earth which is influenced by the magnetic field.

The Cosmic Radiation Environment
A major contribution of the transiting radiation comes from cosmic rays, and they originate from three sources: galactic, solar, and terrestrial. Galactic cosmic rays originate outside the solar system but are associated with the galaxy. These rays are detected in a fairly continuous low flux concentration and are referred to as the space "background radiation." Their composition is about 85% protons, 14% alpha particles, and 1% heavier nuclei [25]. As reported in Sexton [24], it is the heavy-ion contribution of the galactic cosmic rays that are most harmful to spacecraft electronics. In Fig. D1, a breakdown is given of the heavy-ion portion of the galactic cosmic ray spectrum. As can be seen from the graph, the flux is minimal for those particles with an atomic number greater than 30. At a distance of one AU from the sun[2] and outside the Earth's magnetosphere, the cosmic ray flux is approximately four particles per cm^2 per second [25].

Terrestrial cosmic rays are those galactic and solar cosmic rays which penetrate and interact with the Earth's atmosphere and are transformed into secondary radiation. These rays constitute the majority of cosmic radiation experienced at the Earth's surface (i.e., UV radiation), and can still be a threat to integrated circuits; there have been rare cases observed of single event upsets occurring to Space Shuttle computers while the vehicle was still on the launch pad!

[1] One eV is the energy gained by one electron in accelerating through a potential difference of one volt.
1 eV = $1.6x10^{-19}$ Joules.

[2] One Astronomical Unit (AU) is the distance from the Sun to the Earth; approximately 93,000,000 miles.

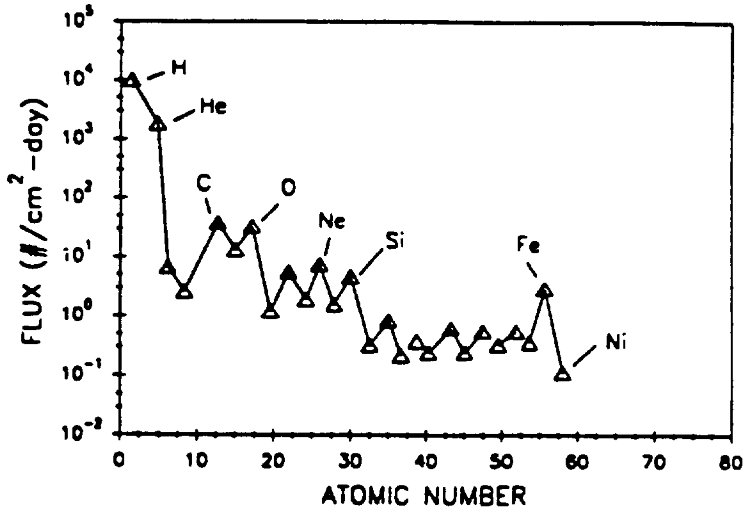


Figure D1: The heavy-ion components of the Galactic Cosmic Ray particle spectrum [24].

The Solar Contribution

Solar cosmic rays are direct products of the sun, and are regulated by changes in solar activity. The sun normally emits protons, neutrons, X-rays, alpha particles, ultra-violet rays and gamma rays. Much of this material is carried away from the sun and into Earth orbit by the solar wind. While the solar wind is generally comprised of low energy particles, they are major contributors to the total overall cosmic ray flux, and it acts as the "driver" for the transiting radiation experienced near Earth. During periods of solar maximum, the increased solar wind tends to dilute the more energetic galactic cosmic rays from the Earth, while at solar minimum, the galactic contribution is more prevalent.

Solar activity varies in approximately eleven year cycles. During periods of increased activity, violent eruptions associated with sun spots occur and are called "solar flares." These solar flares emit a heavy concentration of energetic protons, as well as smaller amounts of alpha particles, electrons, and heavy-ions [9]. Intense solar flares can last several days and can increase the normal cosmic radiation flux by several orders of magnitude. Fig. D2 illustrates the change in proton fluences observed over three solar cycles. The solar flare protons are usually soft protons and do not directly cause damage to integrated circuits, but they can interact with other materials (i.e., spacecraft or IC packaging) to produce secondary radiation, including the more penetrating bremsstrahlung radiation and neutrons [9].

The solar flare heavy-ions normally consist of the helium ion (in 5-10% total concentrations), with heavier ions representing an even smaller population (below the level found in the background radiation).

Earth's Radiation Environment

The Earth's natural radiation environment is closely correlated to its magnetic field, which is a dipole consisting of north and south poles with closed field lines. The magnetic dipole is offset from the Earth's axis of rotation by eleven degrees and is displaced some 500 km toward the Western Pacific [25]. The magnetic field is not symmetrical and is distorted by geological features on Earth and also by the sun. The magnetosphere is shaped and molded in large part by the solar wind, producing a hemispherical shape on the sun side, and a cylindrical shape on the night side [9]. Refer to Fig. D3 for a depiction of the magnetosphere and the Earth's radiation belts.

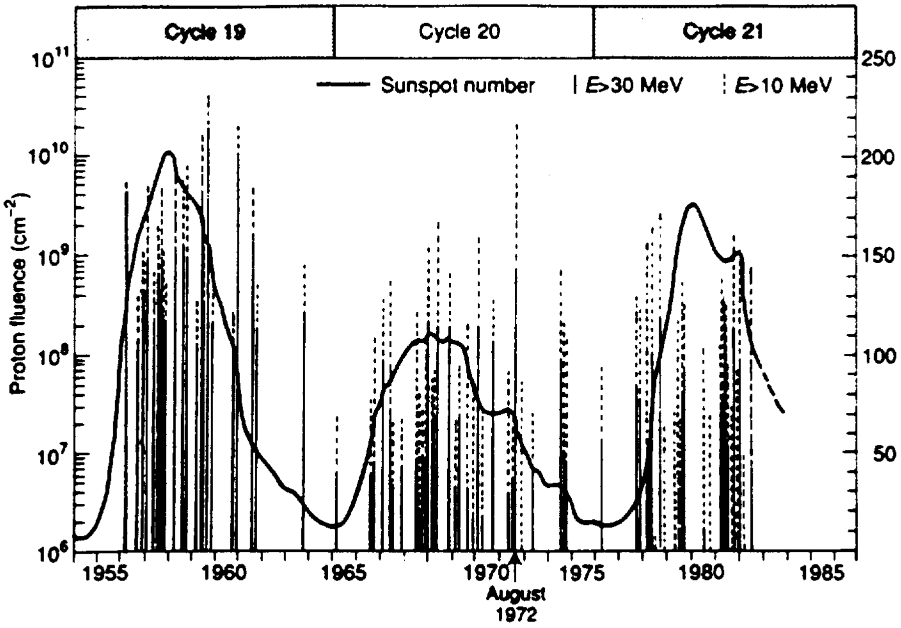


Figure D2: The solar flare proton activity over three cycles [9].

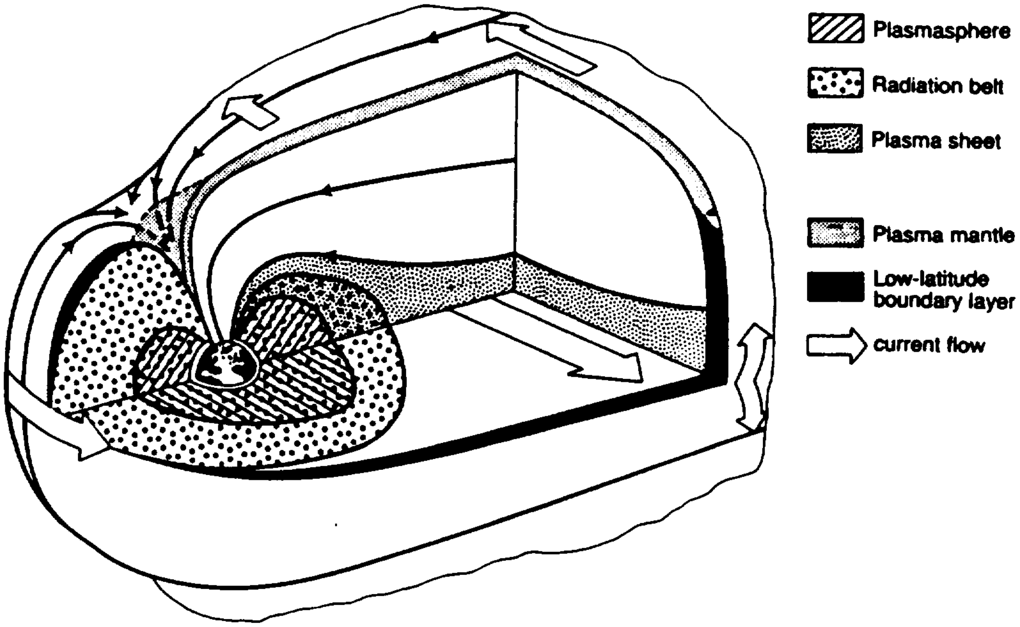


Figure D3: The composition of Earth's magnetosphere and radiation belts [9].

Within the Earth's magnetic field and above the dense atmosphere, trapped electrons, protons, and sparse amounts of low-energy heavy-ions are found. The main particle trapping region of interest is called the plasmasphere, as depicted in Fig. D3. As reported by Stassinopoulos [25], "these particles gyrate around and bounce along magnetic field lines, and are reflected back and forth between pairs of conjugate mirror points in opposite hemispheres." Due to their charge, electrons drift eastward around the Earth, while protons and heavy-ions drift westward. Fig. D4 illustrates this behavior. Besides trapped particles, transiting cosmic rays of solar or galactic origin are also encountered in the magnetosphere.

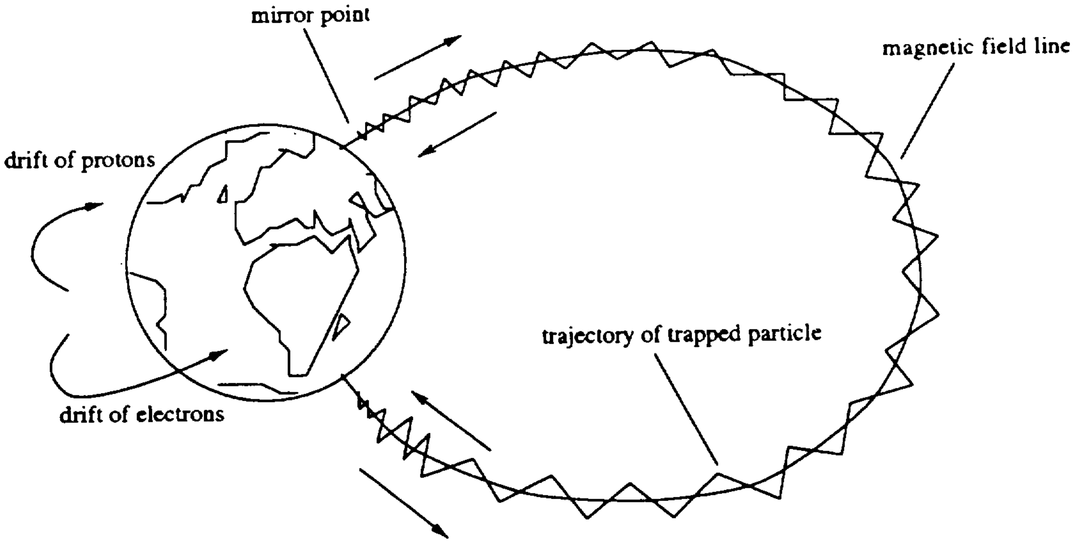


Figure D4: The movement of trapped particles in the Earth's magnetosphere [25].

The Earth's radiation environment is a complex function of both place and time, as certain regions of the magnetosphere possess different trapping abilities. The trapped electron profile consists of two distinct zones, with the inner zone extending to about 2.4 Earth radii (R_e), and the outer zone from 2.8 to 12.0 R_e [9]. The area in between these two zones is commonly referred to as the slot. The trapped electrons have energies up to 7 MeV with the most energetic particles found in the outer zone.

In comparison, the profile of the trapped protons exhibits a maximum flux at about 1.75 (R_e). At this point forward the flux decreases proportionally with increasing distance, and subsides around 3.8 R_e [9]. The trapped protons have energies up to several hundred MeV, with the more energetic ones occurring at lower altitudes.

The South Atlantic Anomaly (SAA) is a particular region of the magnetosphere that is depressed inward towards the Earth due to the tilt of its magnetic axis relative to its rotational axis. This depression lowers the shielding protection and allows more trapped radiation to be encountered over the South Atlantic. The SAA is responsible for most of the intense and penetrating trapped radiation in low-earth orbit (LEO), thus a higher number of single event upsets (SEUs) can be expected in this region. Flight data from the Space Shuttle's general purpose computers (GPCs) supports this claim as Fig. D5 shows. Each dot in the figure represents a single SEU hit, and each triangle represents a multiple SEU hit. Note the increased number of upsets inside the SAA as well as at the higher inclinations. Flight data also supports the dependence of altitude on the SEU rate. As Fig. D6 suggests, more upsets are observed at higher altitudes as opposed to the lower ones shown in the previous figure.

The magnetosphere offers some protection to transiting radiation in the form of "geomagnetic shielding." This occurs when the moving charged particles are deflected by the Earth's magnetic field. These deflections occur perpendicular to the field lines, thus at low altitudes and latitudes (up to 45 degrees) cosmic rays are easily repelled. At polar inclinations however, the field lines converge and geomagnetic shielding is greatly reduced. A particle's penetrating ability is determined by its momentum and charge, thus heavier and faster particles can penetrate further into the magnetosphere.

Spacecraft computer systems may be afforded more protection by shielding the sensitive components against radiation. Low-energy particles can easily be stopped with thin shielding, but high-energy ones are more threatening. As high-energy particles such as solar flare protons impact with a material (such as the spacecraft or IC packaging) the particles can undergo a transformation into secondary radiation, such as highly penetrating bremsstrahlung and neutrons. As these particles collide with the spacecraft they slow down, and a continuous spectrum of x-rays are emitted in the direction of penetration [25]. It is this secondary radiation that can sometimes be more threatening to electronics than the original particle.

In summary, the radiation particles that are most harmful to spacecraft electronics are: protons, electrons, heavy-ions, alpha particles, and x-rays. The three main sources of space radiation are cosmic rays, solar flares, and trapped particles in the Earth's magnetosphere. While the magnetosphere offers some degree of protection to space radiation, higher flux concentrations can be expected in high altitude orbits, high inclination orbits, the polar regions, and the South Atlantic Anomaly. In low earth orbits, the most intense and penetrating radiation is encountered in the South Atlantic Anomaly in the form of protons. The vulnerability of spacecraft electronics to radiation depends on several factors including altitude, orbital inclination, solar activity, and particle flux. The target parameters depend on its shielding, the size of the integrated circuit features, and the fabrication technology used in the process.

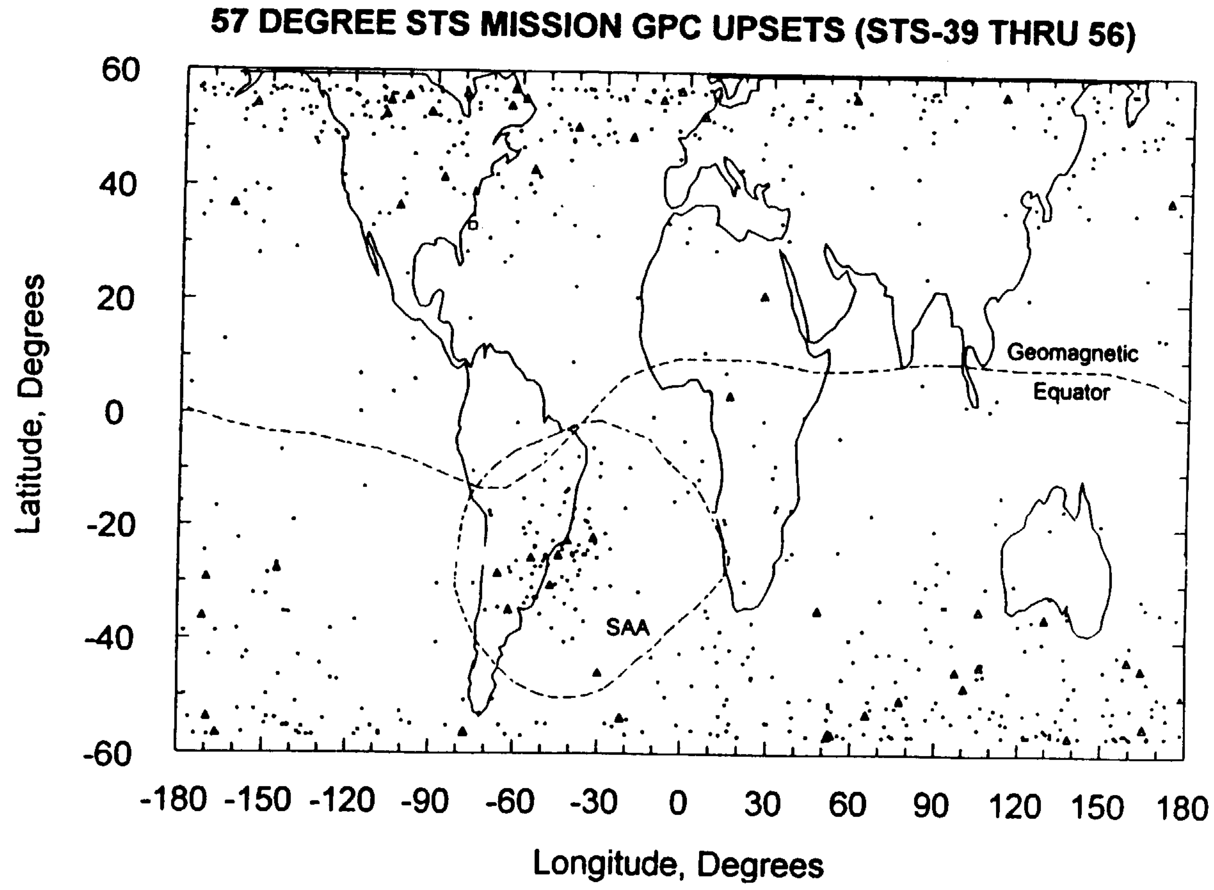


Figure D5: Space Shuttle GPC data depicting the number of SEUs detected over a series of flights. Each dot represents a single-bit upset and each triangle represents a multiple-bit upset [14].

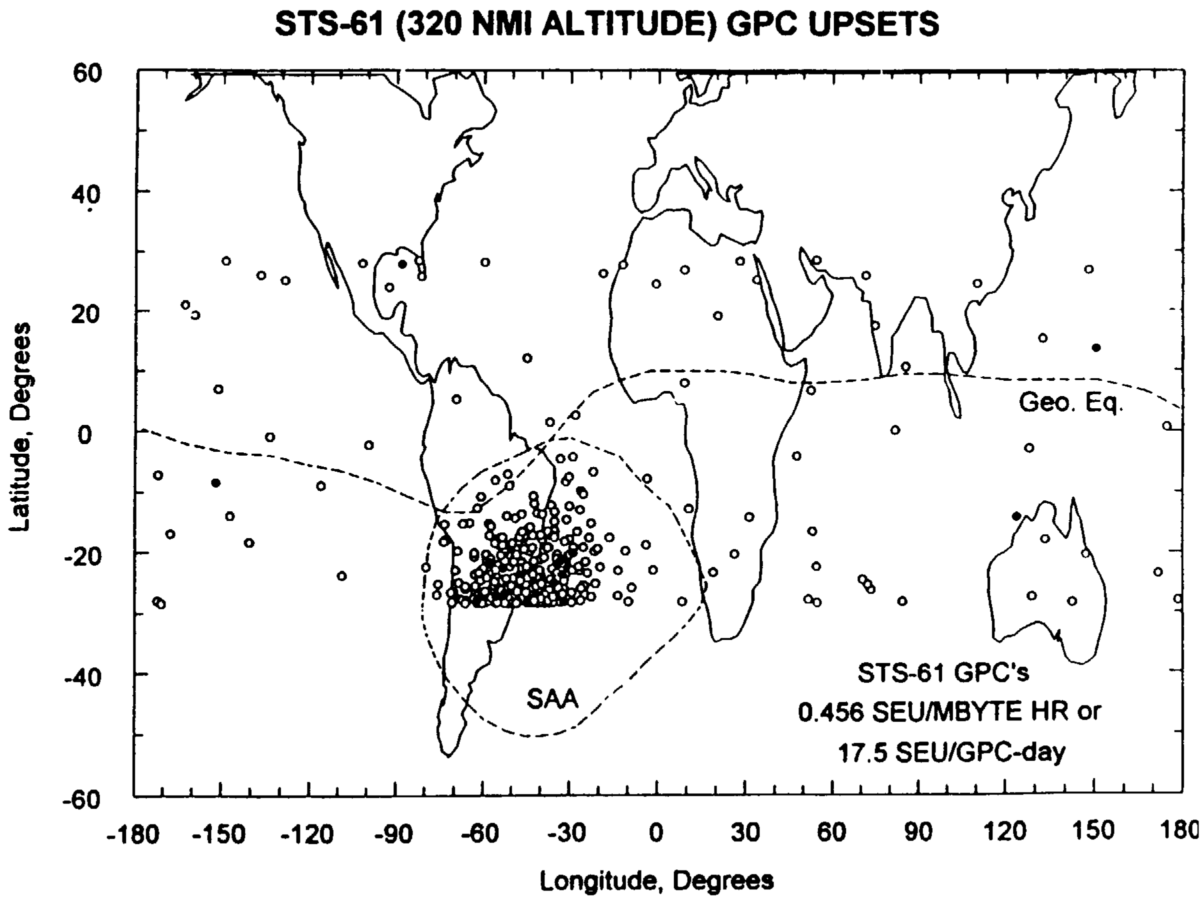


Figure D6: Space Shuttle GPC flight data depicting the number of SEUs detected for a 320-nm orbit [14].

APPENDIX E - RADIATION EFFECTS ON SEMICONDUCTORS

Radiation can effect the normal operating behavior of semiconductors in many different ways. For spaceflight computers and electronics, there are two major types of effects: *total dose* and *single event effects*.

Total dose refers to the long-term accumulation of charge, which breaks down the operating characteristics of the device. When this happens permanent alterations to the device occur, such as a breakdown in its voltage versus current relationships, a parametric shift in the device thresholds, and a decrease in transistor switching speeds to name a few. The total dose accumulation occurs over a long period of time, much longer than for single event effects, but it is directly related to the flux of the environment.

Single event effects (SEE) occur when a single radiation particle strikes a sensitive junction in an operating semiconductor and causes a disruption in its logic state. An occurrence of a single event effect on a semiconductor is often referred to as a "hit." SEE effects may be further classified into three subtypes: *transients*, *single event upsets* (SEUs), and *single event latchup* (SEL). A transient occurs when a hit is not latched by the circuitry, and an SEU occurs when the hit is latched and the logic state of the device is changed. A single event latchup is a form of permanent damage induced by the hit. SEU and SEL will be discussed in further detail in the following subsection.

It is the SEUs are of most interest to this research, because their impacts to digital devices are generally more threatening than total dose. Additionally, many of the potential spaceflight applications for the 486-DX4 microprocessor would not be in space long enough for total dose to be much of an issue. Thus, the remainder of this work primarily focuses only on single event effects in semiconductors.

SEU Circuit Effects

When a high-energy particle strikes a junction in an operating semiconductor, it loses energy as it collides with the electrons and nuclei in the target material. [25][20][24]. Some of this energy is transformed into a very dense plasma of electron-hole pairs along the track of the particle and ionization occurs. Since the junction is biased with an electric field, the electron-hole pairs are separated and a current spike is observed at the circuit node. This phenomena is illustrated in Fig. E1 which shows the electric field as the hashed area of the figure. The ion path through the node distorts the electric field into the substrate to create a highly conductive "funnel" of electron-hole pairs. The funnel eventually collapses as the free carriers are collected by the PN junction and equilibrium is reestablished. The aforementioned current spike has two components, a prompt and a delayed response. The prompt component occurs on the order of 0.1 ns after the hit and is due to the original depletion and funnel region. The delayed component may last hundreds of nanoseconds and is due to the carrier diffusion. This current response is depicted in Fig. E2.

In order for an SEU to change a node's logic state, the charge that is deposited in the sensitive region must be greater than the critical charge required to store information on that node. In other words, the particle must deposit enough energy to exceed the node's logic threshold.

One commonly used measure for the rate of energy deposited in a material is called the *linear energy transfer* (LET). It is also called the mass stopping power, $1/\rho\ dE/dx$, or the rate of energy loss per unit length in a material with density ρ (in our case silicon). LET is expressed in units of MeV-cm^2/mg, and is a useful quantity since it can correlate the total amount of energy deposited in a material for particles with different characteristics.

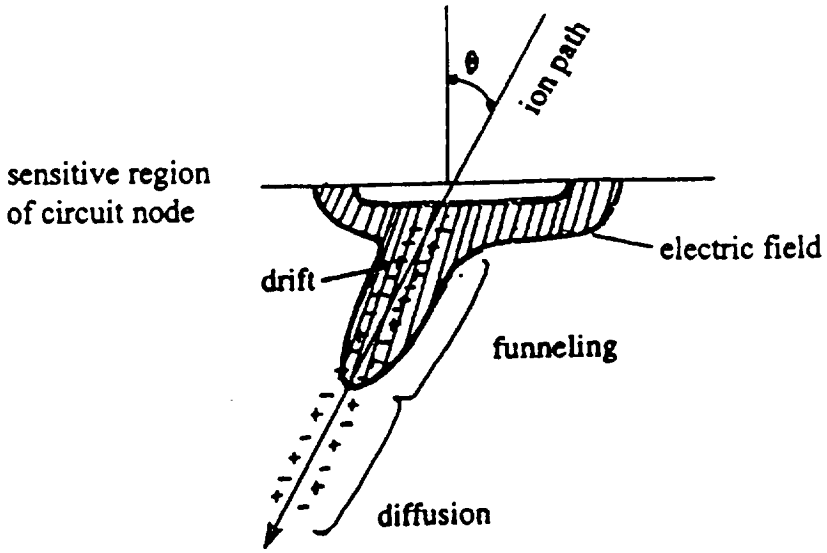


Figure E1: The interaction of a heavy-ion penetrating an active region of an integrated circuit node [24].

Single event latchup (SEL) may sometimes occur when an energetic particle strikes a junction with a high electric field and turns on a parasitic bipolar PNPN structure. The range of the particle travels deep into the node and the funnel region extends well into the substrate. Since this funnel region is highly conductive, a "virtual short" is created which may allow a nearby capacitor to discharge. If sufficient energy is stored on this capacitor, thus high electric fields are present, the discharge may be fast enough that excessive local heating occurs at the node and a thermal runaway situation develops. Temperatures may get so hot that the dielectric melts or the conductive layers evaporate, thus permanently damaging the node [24].

System Impacts of SEUs

When an SEU causes a change in the logic state, a "bit flip" occurs at the upset circuit node [20]. SEUs are usually non-destructive soft errors and can be corrected by reprogramming the affected location [24]. The impact to the system is either corrupted data or program-flow anomalies, depending upon the location and nature of the upset. Normally SEUs only affect a single bit, but multiple bit hits from the same particle can also be expected.

As reported in [14], O'Neill states that SEUs can be more of a threat to system integrity than some permanent failures. One major reason for this is that they are sometimes very difficult to detect and analyze due to their temporal and spatial characteristics. For instance, an upset may occur in one location and propagate to another before it is observed at the system level [14][18]. The location of an upset is therefore one of the biggest factors in determining the impact to the system; for instance, if an SEU occurs in an unused memory location of a microprocessor's cache, it will probably never be detected. But if an SEU occurs in the instruction pointer or in a critical register, the system may undergo a catastrophic failure

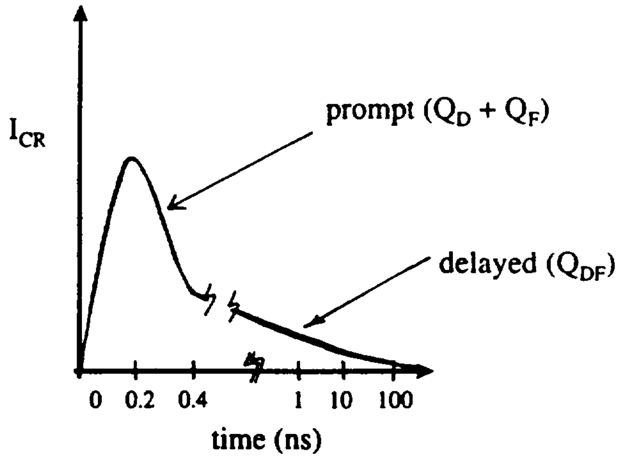


Figure E2: The induced current response observed at the node of an integrated circuit after an SEU hit [24].

in the worst case (such as the case of a spacecraft's trajectory parameters being upset during re-entry). The latency time from upset to detection is also another contributing factor in the difficulty of SEU analysis. An upset may occur in a dormant register that is not accessed until quite some time later. Thus, the impact of an SEU to the system depends on the location of the upset, how the corrupted data is handled, and the latency from upset to detection.