MSU / AAPS Reconfigurable Computing Demonstration
Marshall Space Flight Center
1/7/10

Research Statement:

"Exploit Reconfigurable Hardware to Create Resilient Computing Systems for Military & Aerospace Applications"

Presenters:

Dr. Brock J. LaMeres
Assistant Professor

Clint Gauer
MSEE Candidate (5/10)

Department of Electrical and Computer Engineering
Montana State University
Bozeman, MT

Overview of Project Funding

- This work has been funded through a variety of NASA & Space Grant Programs:

NASA EPSCoR

Experimental Program to Stimulate Competitive Research

NASA Exploration Systems Mission Directorate *Higher Education Program*

$ $

Montana Space Grant Consortium

$

MSU Research Team

MONTANA STATE UNIVERSITY College of ENGINEERING

Faculty Research Initiation Grants	Graduate Fellowships	Senior Capstone Projects	Travel Grants
(Dr. LaMeres)	(Clint Gauer)	(3x at MSU)	(LaMeres & Gauer)

Acknowledgements

- Space Grant Funding Requires NASA Mentor/Collaboration:

- Special thanks to our project mentor from NASA's *Advanced Avionics & Processor Systems (*AAPS) Project

Dr. Andrew S. Keys
Marshall Space Flight Center
AAPS Project Manager

- And also to the APPS Individual Task Leaders

Dr. Robert E. Ray
Marshall Space Flight Center
Reconfigurable Computing Task

Michael A. Johnson
Goddard Space Flight Center
High Performance Processor Task

Overview of Work to Date

1) **Fall 2008 Capstone:** "TMR Soft Processor System on an FPGA"
   - Tony Thomason, Colin Tilleman
   - EE & CpE Undergraduate Students

2) **Spring 2009 Capstone:** "64 Processor Computing System with Spatial Fault Avoidance"
   - Patrick Kujawa, Dan Dunbar, Dave Racek
   - CpE Undergraduate Students

3) **Fall 2009 Capstone:** "Dynamic Recovery of IO Faults using Spare Lines"
   - Sam Harkness, Devin Mikes, Jeff Bahr
   - CpE Undergraduate Students

4) **Graduate Research Project:** "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
   - Clint Gauer
   - EE Graduate Student

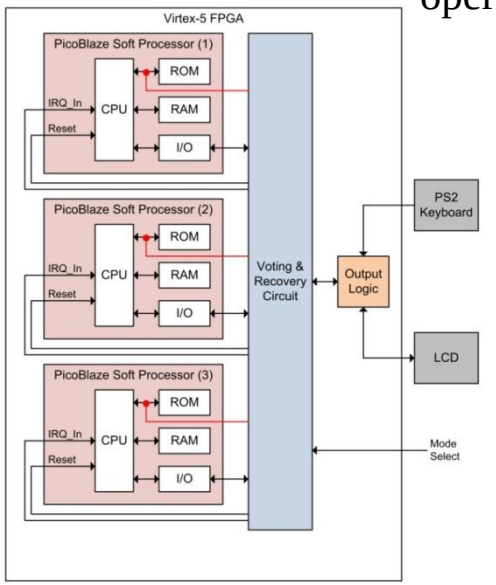
5) **Graduate Research Project:** "Spatial Radiation Sensor"
   - Brian Peterson, Eric Gowens
   - EE Graduate Student

Overview of Work to Date (Project #1)

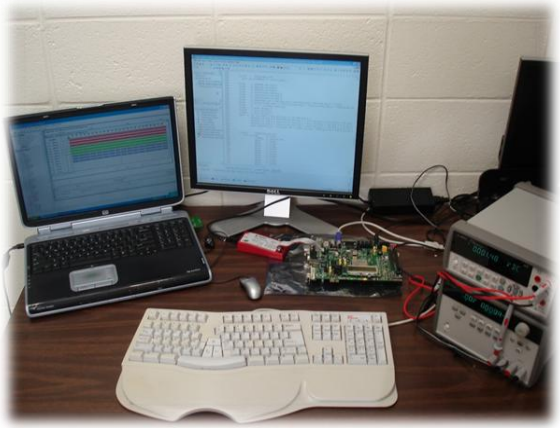
1) Fall 2008 Capstone: "TMR Soft Processor System on an FPGA"
Anthony Thomason & Colin Tilleman

Summary: Develop an FPGA-based computer system that can recover from emulated radiation-induced faults using triple modular redundancy of soft processors.

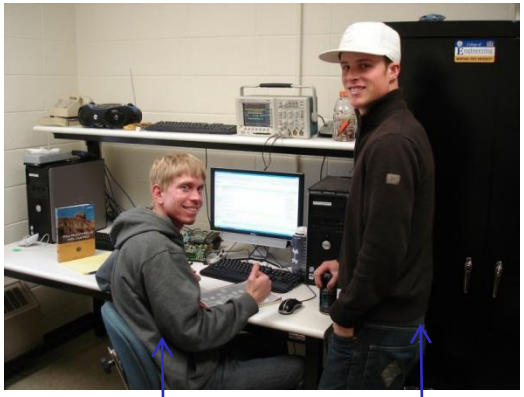
The system will continually service basic peripherals (keyboard & LCD) in the presence of faults. Upon a fault in a processor, the system will finish its current operation, reset & resynchronize the three processors, and continue operation.



Block Diagram



Lab Setup (Xilinx V5 FPGA)

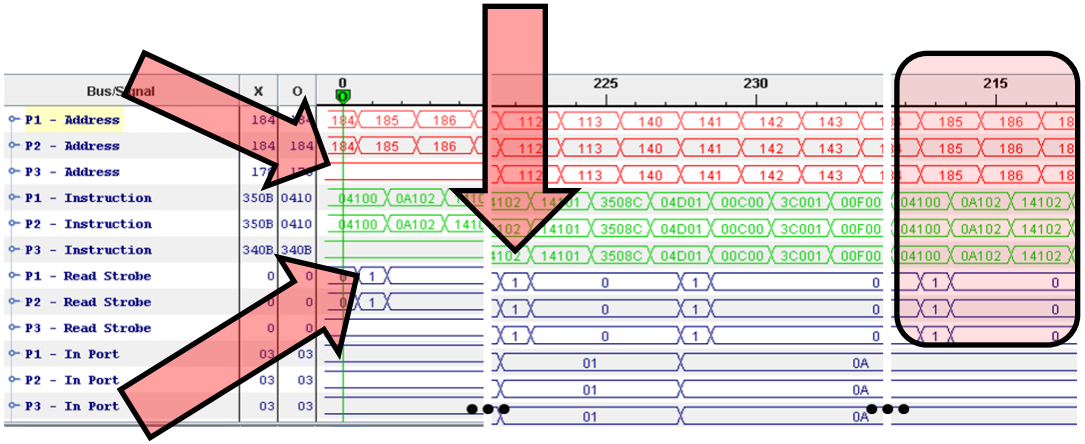


Colin **Tony**

Overview of Work to Date (Project #1)

1) Fall 2008 Capstone: "TMR Soft Processor System on an FPGA"
Anthony Thomason & Colin Tilleman

Highlights: - successfully demonstrated to Robert Ray and Clint Patrick at Fall-08 Design Fair
- won 2nd place ($500) in the IEEE NW Section Student Paper Contest (April 2009)



- Processor P3 has been faulted and processors P1 and P2 continue normal operation
- The three processors are reset and then load in the variable data
- The processors then continue with normal operation

Tony Colin

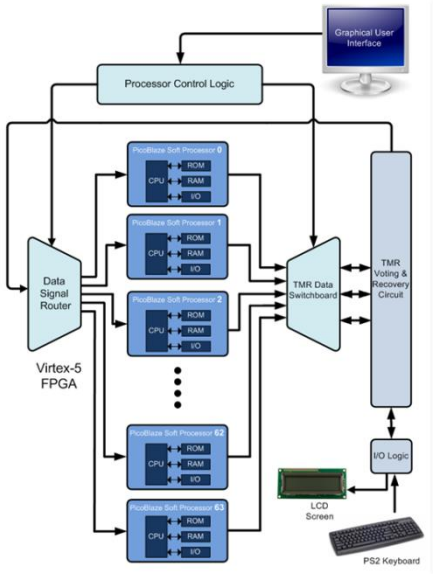
System Operation Measured by ChipScope Logic Analyzer

Fall 2008 Senior Design Fair

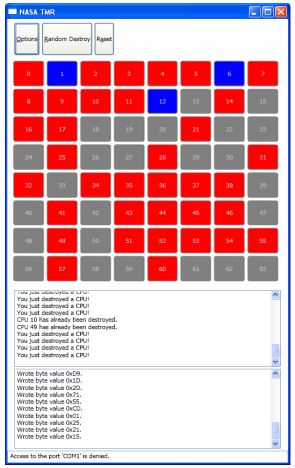
Overview of Work to Date (Project #2)

2) Spring 2009 Capstone: "64 Processor Computing System with Spatial Fault Avoidance"
Pat Kujawa, Dan Dunbar, & David Racek

Summary: Develop an FPGA-based computer system that can recover from emulated radiation-induced faults using spare processors. The system should contain 64 soft processors. 3 of the processors will be active at any given time and be running in TMR. Upon a fault, the system will bring a spare processor online to replace the faulted processor. A GUI should be developed to induce faults and display the active, faulted, and spare processors.



Block Diagram



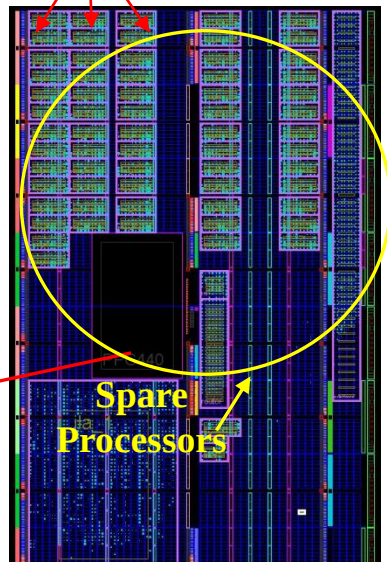
GUI (Blue=active, Red=faulted, gray=spare)

Pat

Dan

Dave

Xilinx Virtex-5 Platform



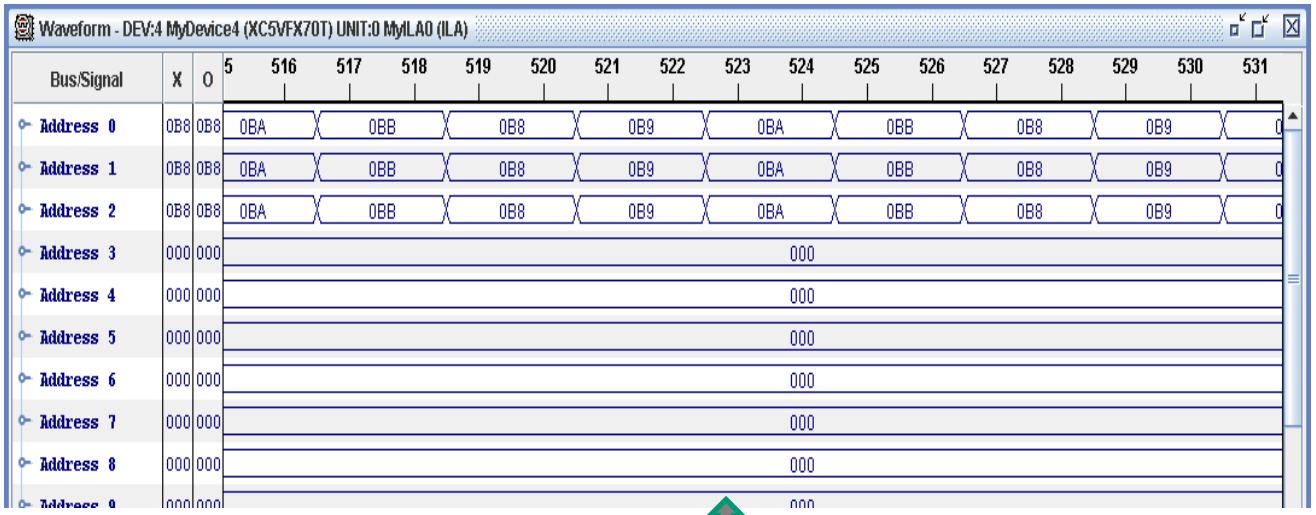
FPGA Floor plan

MONTANA STATE UNIVERSITY

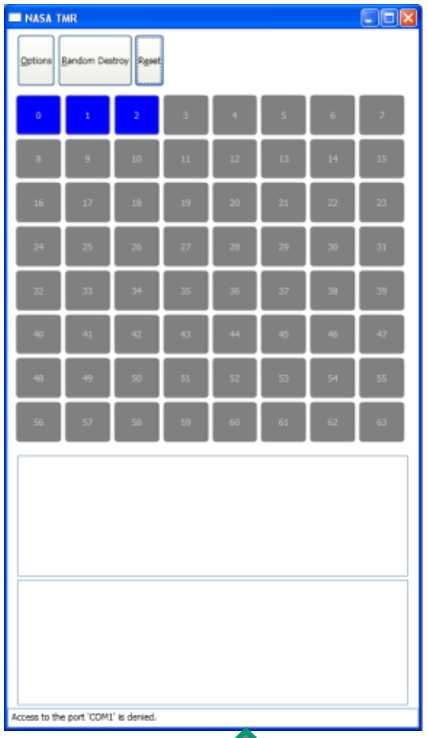
System Demonstration

- **Initial Operation**

  - Processors **0, 1, and 2** are active (blue) and operating in TMR
  - Processors **3-63** provide 61 spare *picoBlaze* processors (gray)



ChipScope shows uP 1,2,3 are running in synch with no faults



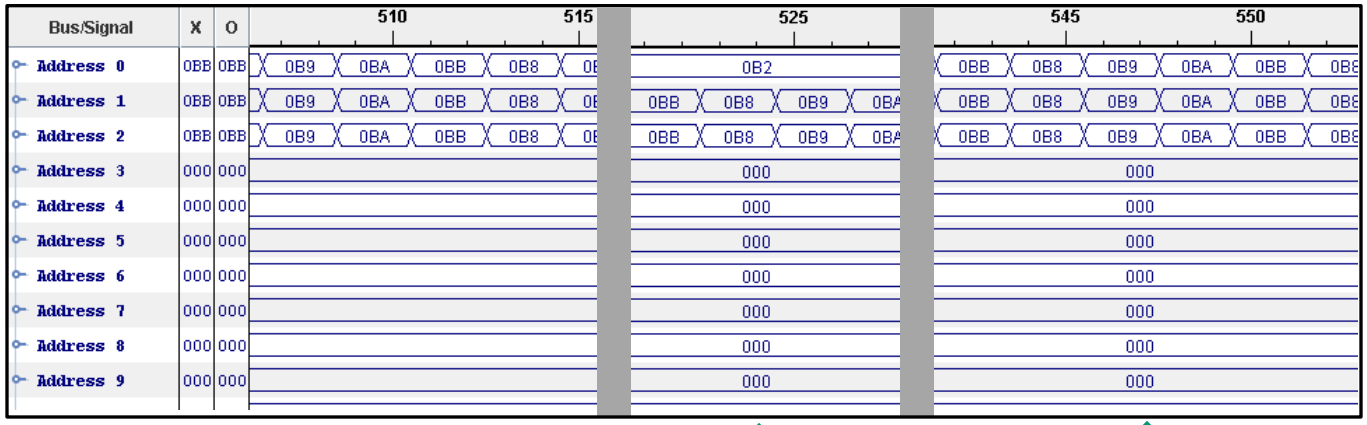
GUI indicates uP 0, 1, and 2 are active (blue)

(showing address lines between uP and memory for all 64 processors)

System Demonstration

- **Soft Fault Recovery**

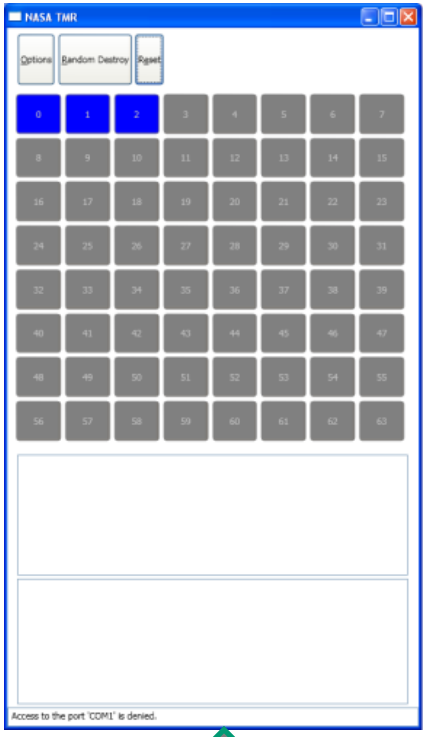
  - Processors **0, 1, and 2** are active (blue) operating in TMR
  - Processors **0** undergoes a soft fault and then recovers and resynchronizes



System initialized and running normally in TMR mode.

Processor 0 has been corrupted by an SEU. The TMR detects the failure.

Processor 0 brought back into synch with other two processors.

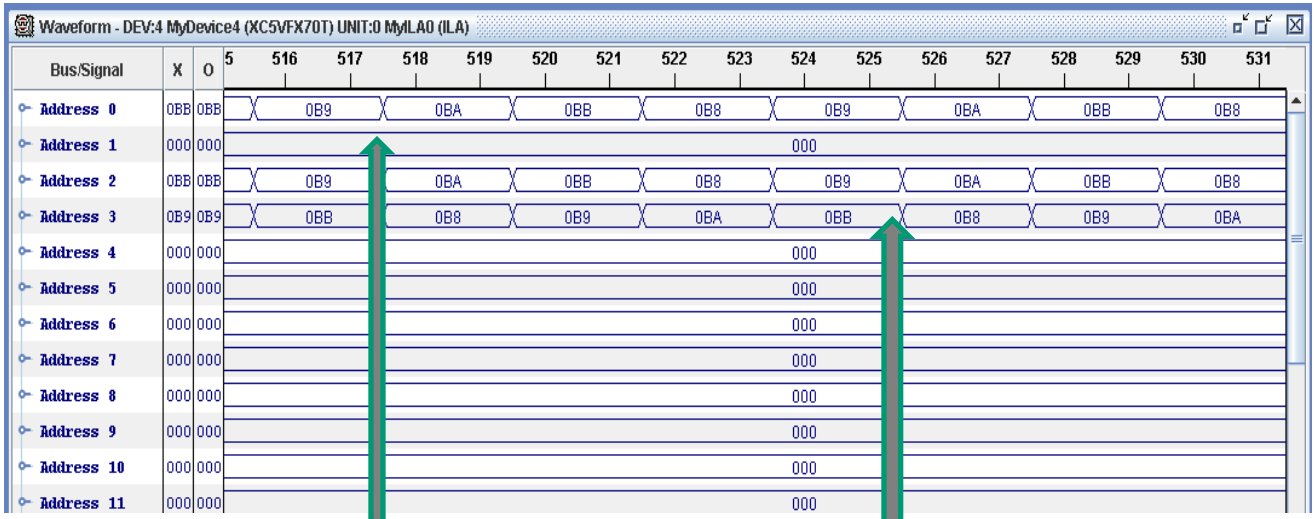


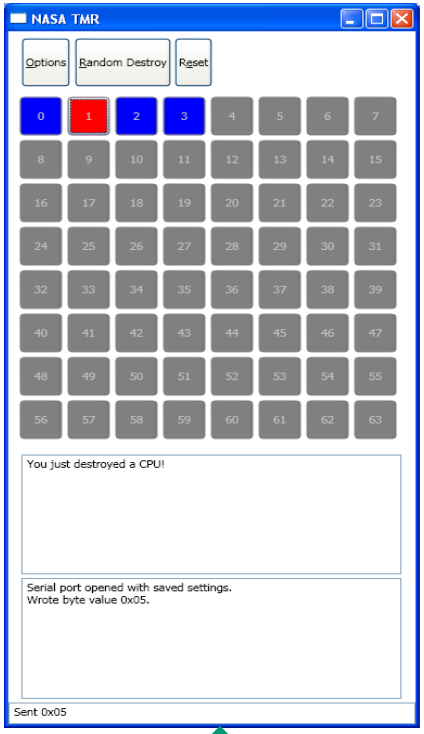
GUI indicates uP 0, 1, and 2 are active (blue)

System Demonstration

- **Hard Fault Recovery**

  - Processors **1** undergoes hard fault (induced by GUI, **red**)
  - The system shuts down uP #1 and brings on spare processor uP #3 into TMR





Processor 1 has hard fault so is shut down

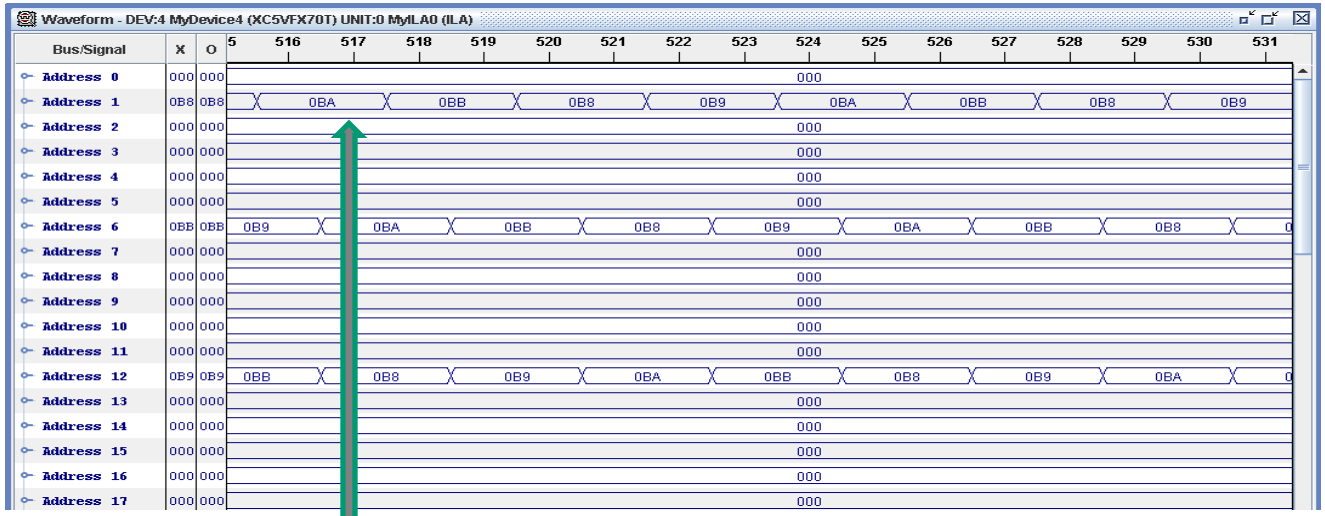
Spare processor 3 is brought online, resynchronized, and reinitialized to form TMR

GUI indicates uP 1 is in hard fault (red). uP 0,2,3 form TMR (blue).

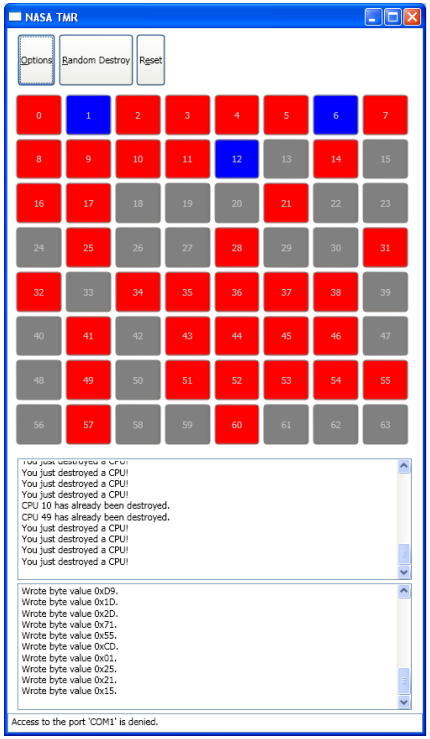
System Demonstration

- **Multiple Hard Faults**

  - Multiple hard faults are present
  - uPs 1, 6, and 12 form TMR



Processor 1, 6, & 12 are active



GUI indicates uP 1 , 6, & 12 are active. Multiple hard faults are present

Timing/Area Impact

- **Soft Fault Recovery** (reset, reload variable information)

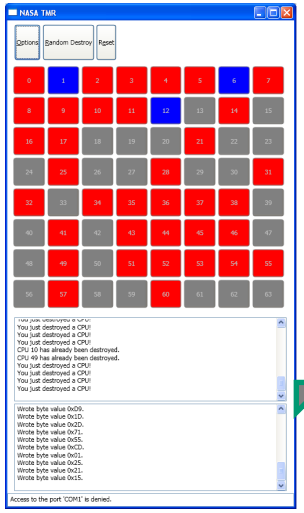
Timing Overhead

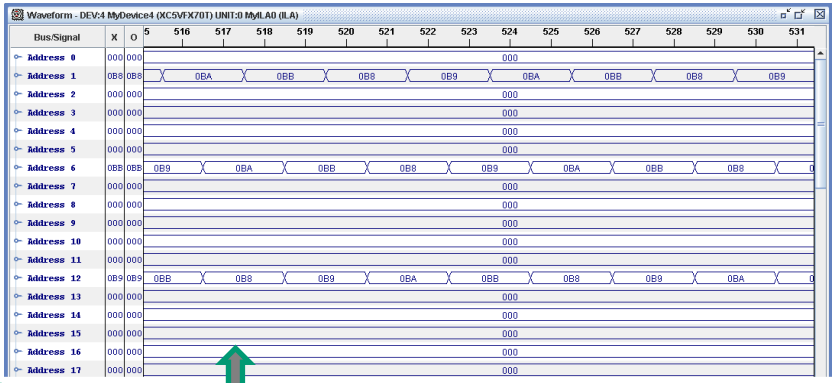
- TMR interrupt	2 clocks	
- Reset	2 clocks	
- Read variable data from good processors:	128 clocks	(2 clks/inst, 64 bytes of RAM)
- Write variable data to reset processor:	128 clocks	(2 clks/inst, 64 bytes of RAM)
Total	**260 clocks = 2.6 us**	(100 MHz V5 Clock)

Overview of Work to Date (Project #2)

2) Spring 2009 Capstone: "64 Processor Computing System with Spatial Fault Avoidance"
Pat Kujawa, Dan Dunbar, & David Racek

Highlights:
- successfully demonstrated to Robert Ray at Spring-09 Design Fair
- demonstrated at 09 Europa Jupiter Systems Mission (EJSM) Instrument Workshop
- published at 2009 *MAPLD Conference*





Processors 1, 6, and 12 are active (blue). May processors have been faulted (red). There are still spares remaining.

Logic analyzer shows processors 1, 6, and 12 continue to run in TMR despite failures.

GUI & System Operation Measured by ChipScope Logic Analyzer

System Demonstration at 2009 EJSM Instrument Workshop.
(Robert Ray shown here in front of demo)

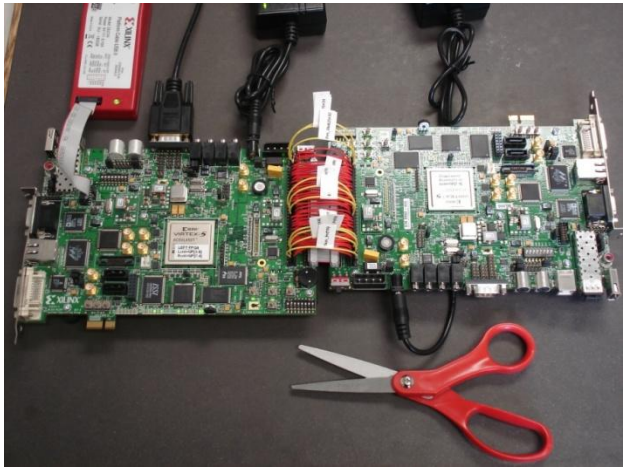
MONTANA STATE UNIVERSITY

Overview of Work to Date (Project #3)

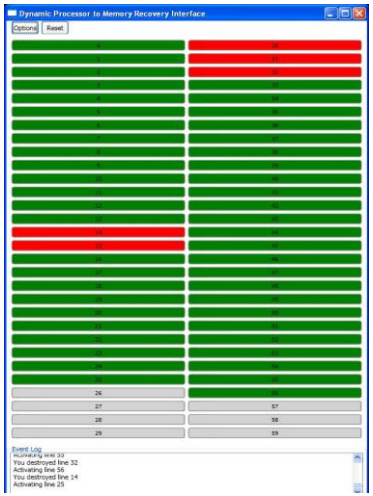
3) Spring 2009 Capstone: "Dynamic Recovery of IO Faults using Spare Lines"
Sam Harkness, Devin Mikes, & Jeff Bahr

Summary: Develop an IO system that can continue to operation when a fault occurs on the physical lines of the bus (due to radiation strikes or broken conductors). The system should be able to detect faults and switch the active signals to spare lines on the bus. A GUI should be developed to monitor which lines of the IO system have been faulted.

Jeff Devin Sam



Prototype System
IO Bus Implemented with Wires between two Virtex-5 FPGAs



GUI (Green=active, Red=faulted, gray=spare)

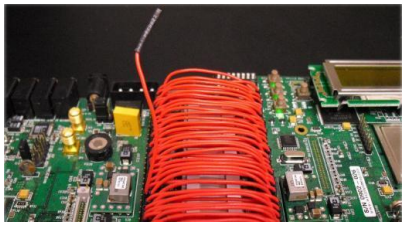
Overview of Work to Date (Project #3)

3) Spring 2009 Capstone: "Dynamic Recovery of IO Faults using Spare Lines"
Sam Harkness, Devin Mikes, & Jeff Bahr

Theory of Operation: -

1) Spare Lines are included on the bus to be used in case of a line failure
2) A Hamming code is used to check for errors on the bus and are transmitted on the bus
3) When an error is detected, the system begins a detect/ & recovery process

- Agent A sends all 1's
- Agent B looks for all 1's, logs failures
- Agent A sends all 0's
- Agent B looks for all 0's, logs failures
- Agent B sends all 1's
- Agent A looks for all 1's, logs failures
- Agent B sends all 0's
- Agent A looks for all 0's, logs failures
- The bus lines are remapped into good lines



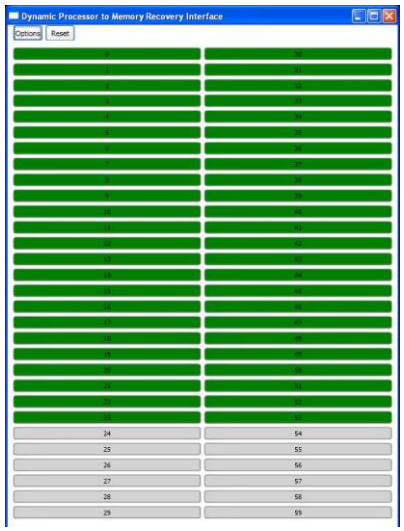
Total Time = (10 + n + log(n) + spare lines) where n = # of lines on bus

Our system = 10 + 18 + 6 + 6 = 40 clocks

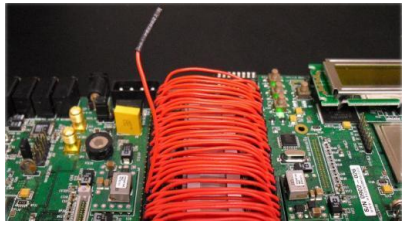
Overview of Work to Date (Project #3)

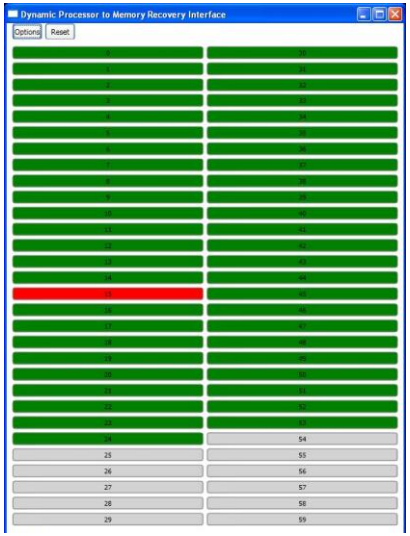
3) Spring 2009 Capstone: "Dynamic Recovery of IO Faults using Spare Lines"
Sam Harkness, Devin Mikes, & Jeff Bahr

Highlights: - successfully demonstrated to Robert Ray & Leigh Smith at Fall-09 Design Fair
- currently filing an invention disclosure with MSU (first time for the students)



IO bus in Tact, GUI indicates all lines good





Wire pull on line 15, GUI indicates fault and that a spare has been brought online

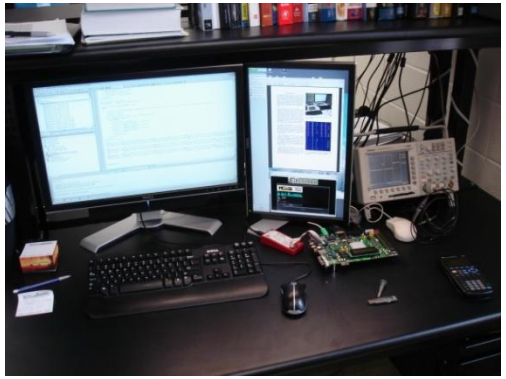
System Demonstration at MSU Fall-2009 Design Fair
(Sam Harkness giving Leigh Smith Demo)

Overview of Work to Date (Project #4)

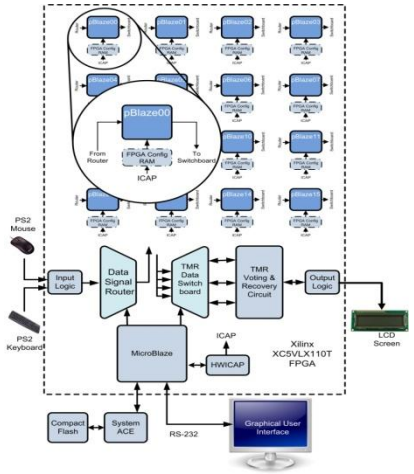
4) Graduate Research Project: "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
(2007-present) *Clint Gauer*

Summary: Develop an FPGA-based computer system which can recover from, avoid, and repair radiation induced faults in both the circuit fabric and configuration SRAM. The system uses a many-core architecture where three soft processors run in TMR with *n* spares. Each processor resides in a partially reconfigurable *tile* on the FPGA. Upon a fault, the system brings on a spare processor to replace the faulted processor (SEU/TID recovery & avoidance). The faulted tile is then partially reconfigured to repair and re-introduce it as a spare (SEFI recovery).

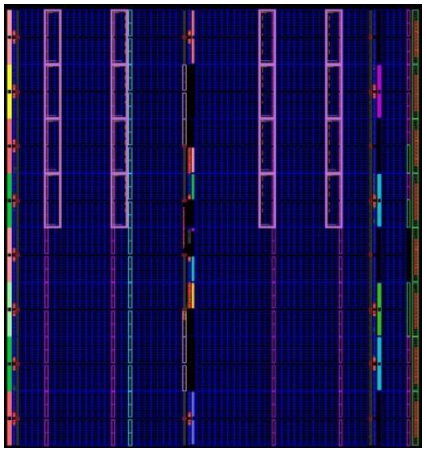
Clint



Lab Setup
(Virtex-5 FPGA)



Block Diagram
(3+13 soft processors)



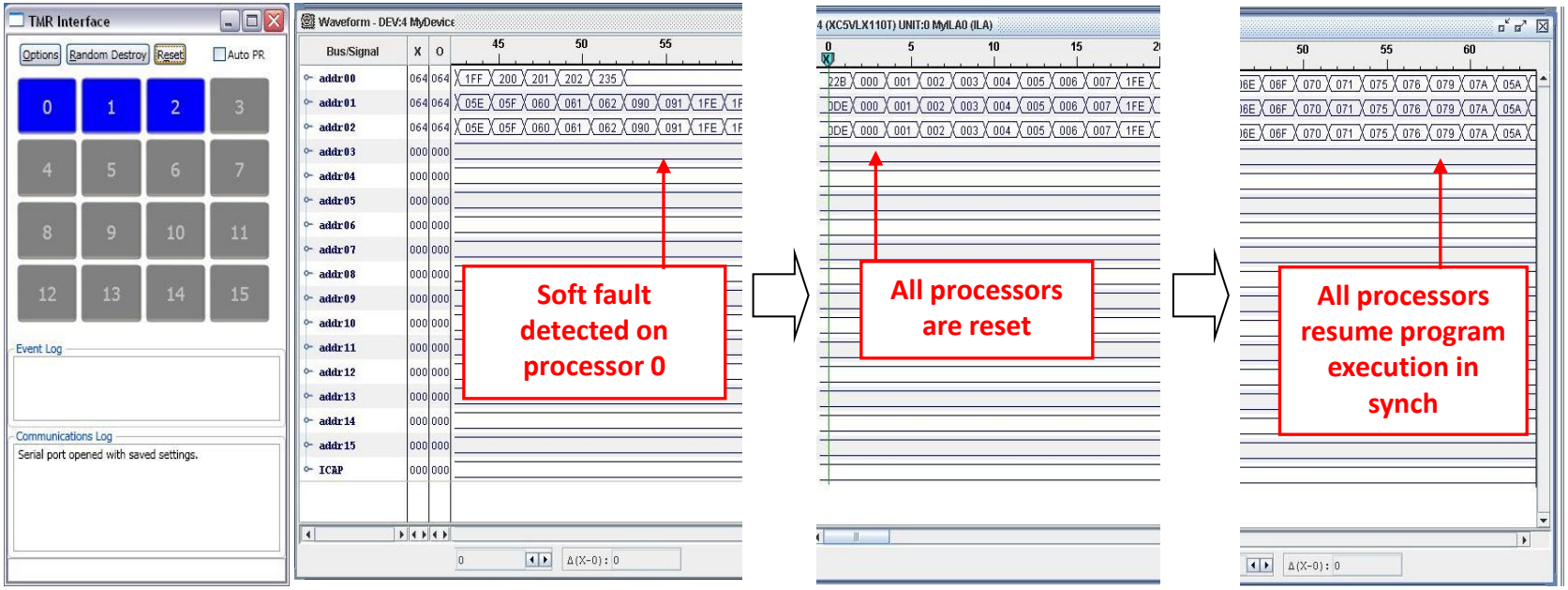
FPGA Floor plan
(16 picoBlaze processors)

MONTANA STATE UNIVERSITY

Overview of Work to Date (Project #4a)

4) Graduate Research Project: (2007-present) "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
Clint Gauer

System Operation of 3+13 picoBlaze Architecture: Recovery from SEU in Circuit Fabric (Processor 0)

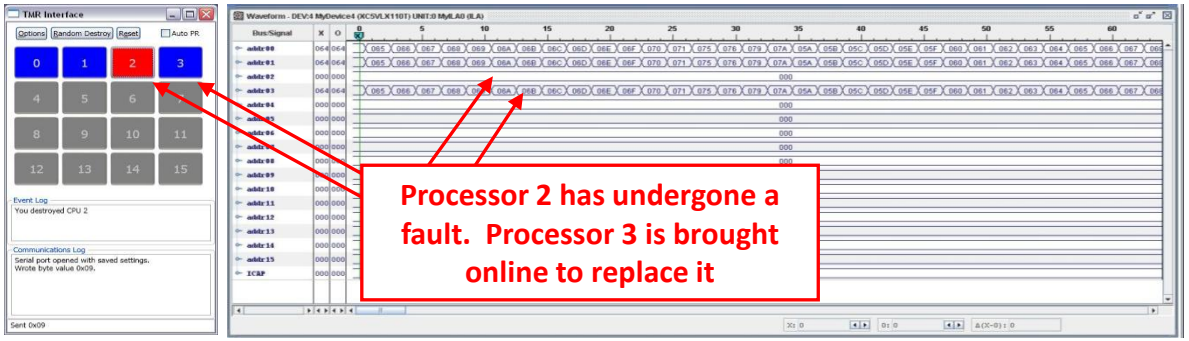


Overview of Work to Date (Project #4a)

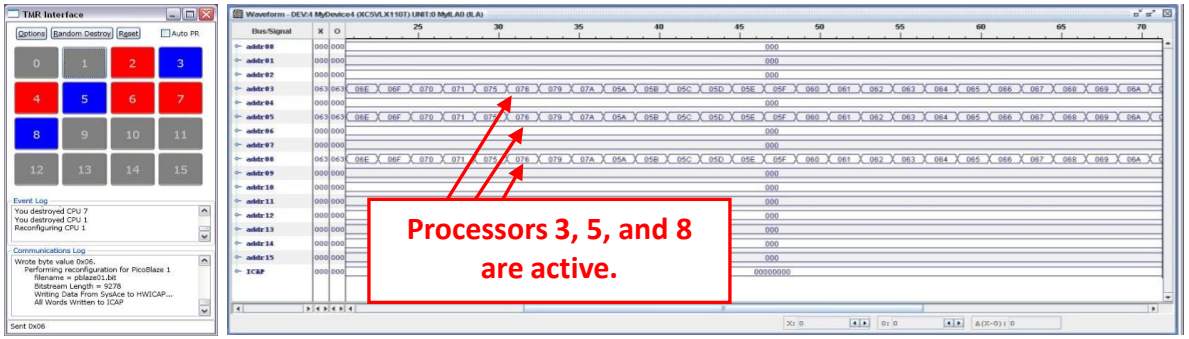
4) Graduate Research Project: (2007-present) "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
Clint Gauer

System Operation of 3+13 picoBlaze Architecture: Spatial Avoidance of Faulted Tile/uP

SEFI or TID on Processor 2



SEFI or TID on Processors 2, 4, 6, and 7



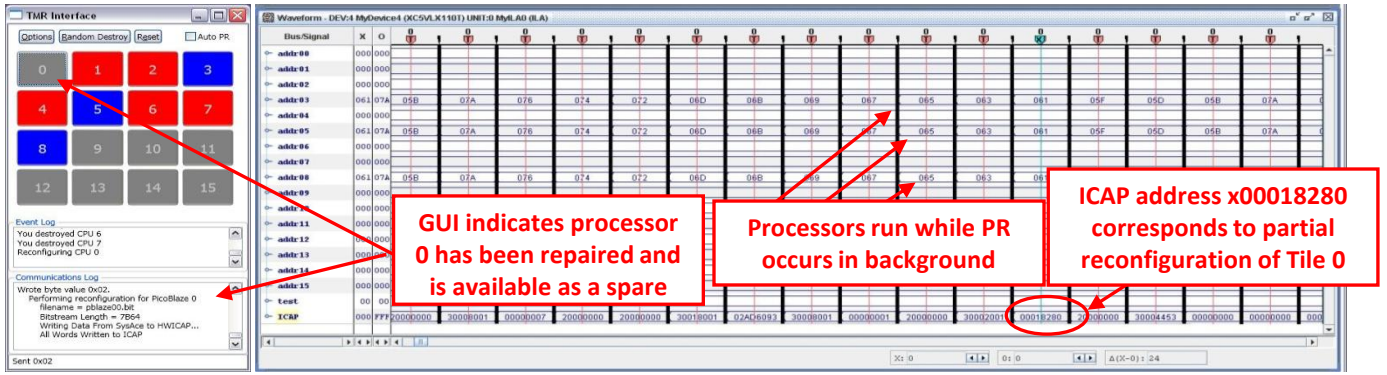
Overview of Work to Date (Project #4a)

4) Graduate Research Project: "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery" (2007-present)

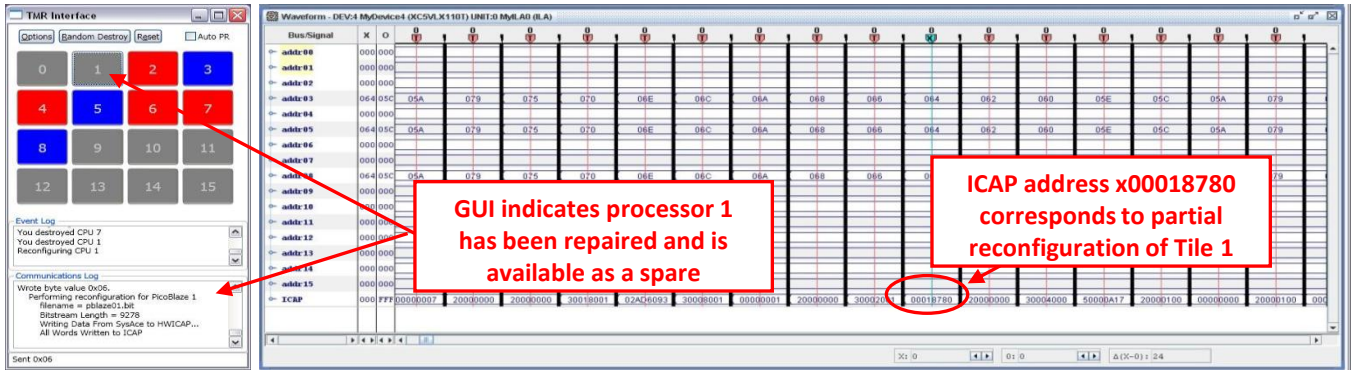
Clint Gauer

System Operation of 3+13 picoBlaze Architecture: SEFI Repair using Partial reconfiguration of faulted tile

Repair of Processor 0 (66ms)



Repair of Processor 1 (66ms)

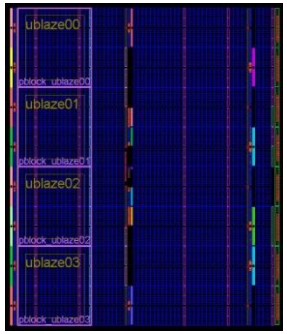


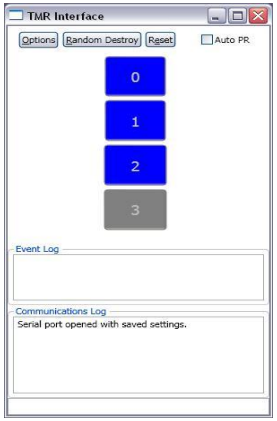
Overview of Work to Date (Project #4b)

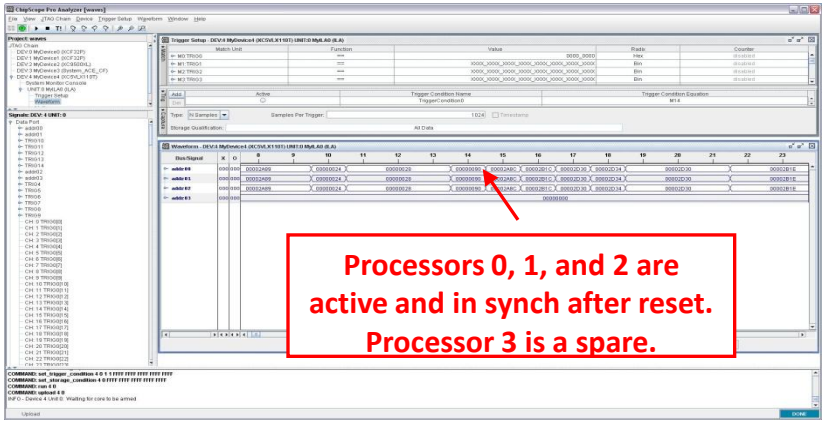
4) Graduate Research Project: "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
(2007-present) *Clint Gauer*

System Operation of 3+1 microBlaze Architecture: Spatial Avoidance of Faulted Tile/uP

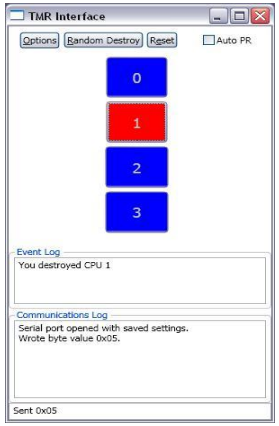
Normal Operation
uP 0,1,2 are active
uP 3 is a spare

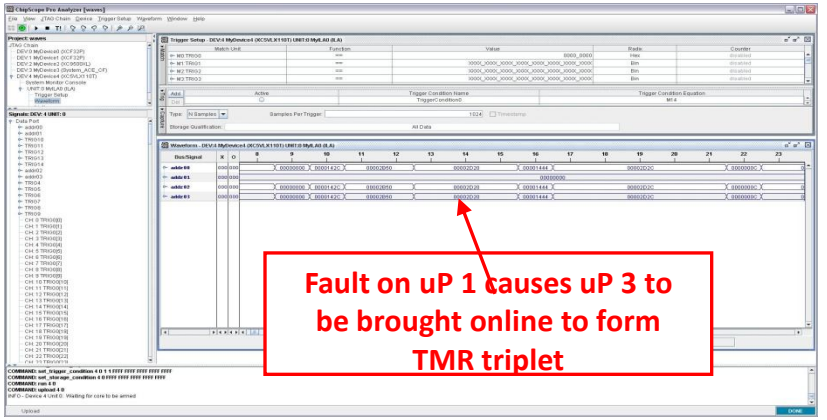






SEFI or TID on uP 1
uP 3 is brought online

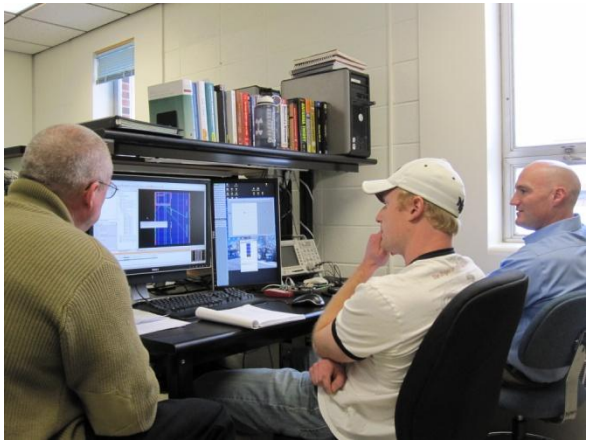




Overview of Work to Date (Project #4)

4) Graduate Research Project: "Many-Core Computing System using Partial Reconfiguration for fault detection, avoidance, and recovery"
(2007-present) *Clint Gauer*

Highlights:
- published work twice at *Military & Aerospace Programmable Logic Devices (MAPLD)* Conference (08 & 09)
- published work twice at *IEEE Aerospace* Conference (09 & 10 accepted)
- this work will be submitted as Clint's Masters thesis in May 2010.



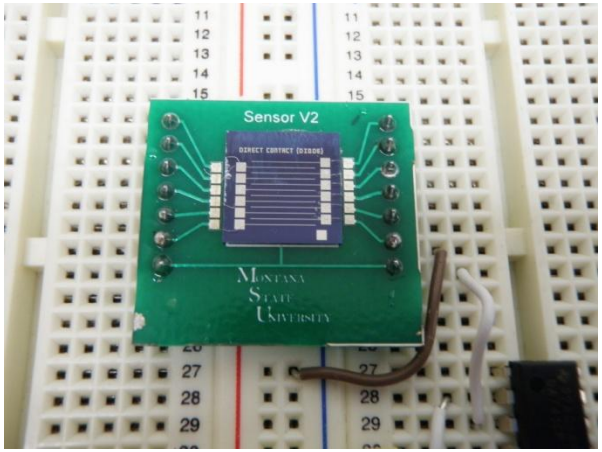
System Demonstration in MSU Research Lab 12/14/09
(Clint Gauer giving Robert Ray & Brock LaMeres Demo)

Overview of Work to Date (Project #5)

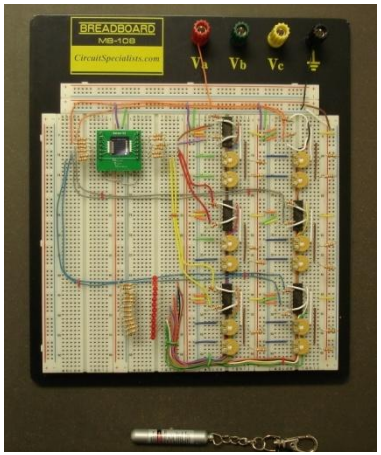
4) Graduate Research Project: "Spatial Radiation Sensor"
Brian Peterson, Eric Gowens

Summary: Develop a sensor which can give the location and trajectory of incoming radiation strikes. This sensor is designed to be used in conjunction with a many-core computing system. The computer system can use the spatial radiation information to more effectively avoid faults in the circuit fabric and repair faults in the configuration SRAM.

LaMeres, Smith, Gowens, and Kaiser At MSU 12/14/09



Sensor & Packaging Prototype

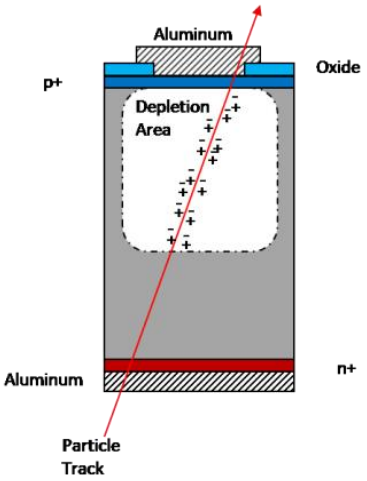


Prototype System

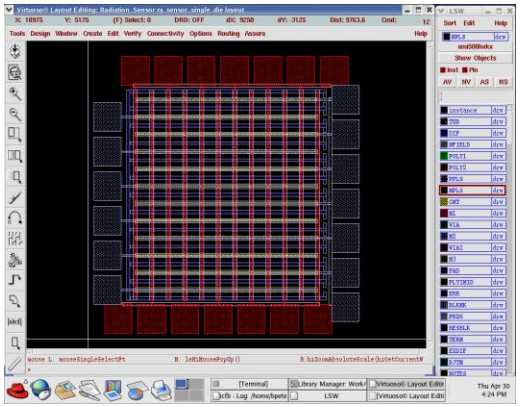
Overview of Work to Date (Project #5)

5) Graduate Research Project: "Spatial Radiation Sensor"
Brian Peterson, Eric Gowens

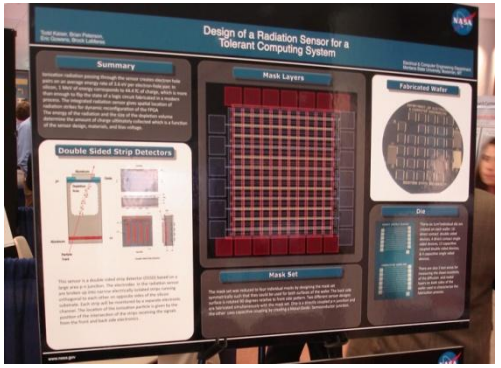
Highlights: - idea presented at 09 Europa Jupiter Systems Mission (EJSM) Instrument Workshop
- prototype demonstrated to Robert Ray & Leigh Smith at MSU on 12/14/09



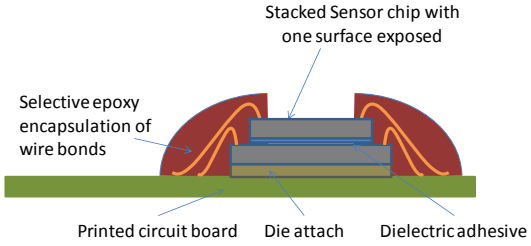
Theory of Operation



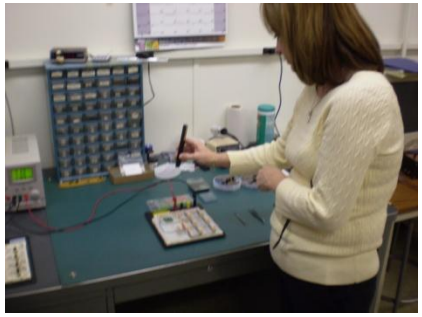
Sensor Mask Design



EJSM Poster Presentation on Radiation Sensor (2009)



Trajectory Detection



Leigh Smith getting demo at MSU 12/14/09

MONTANA STATE UNIVERSITY

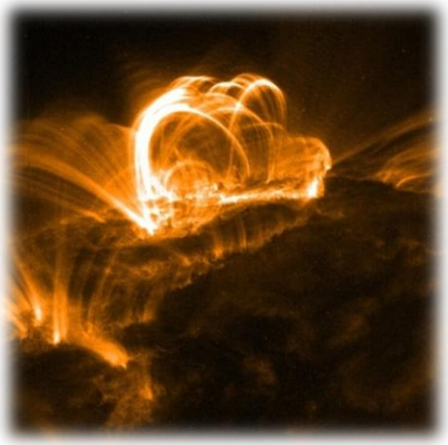
Background

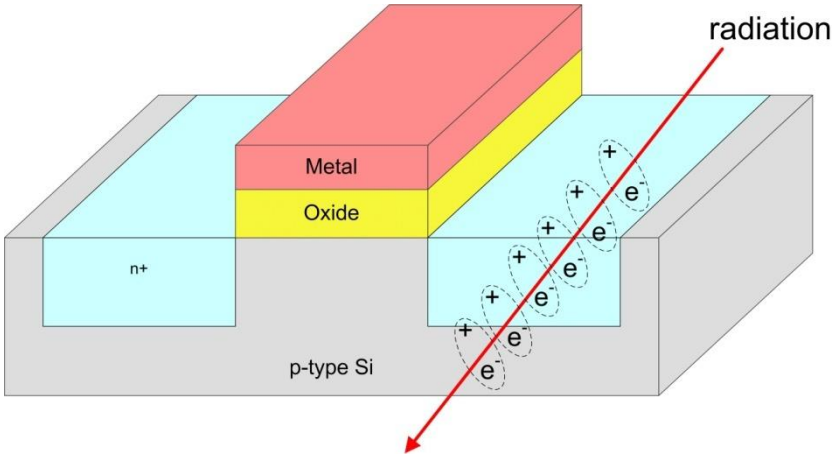
- Background

Motivation

- Radiation has a detrimental effect on electronics in space environments.

- The root cause is from electron/hole pairs creation as the radiation strikes the semiconductor portion of the device and ionizes the material.





Types

- *alpha particles* (Terrestrial, from packaging/doping)
- *Neutrons* (Terrestrial, secondary effect from Galactic Cosmic Rays entering atmosphere)
- *Heavy ions* (Aerospace, direct ionization)
- *Proton* (Aerospace, secondary effect)

Motivation

- Two types of failures mechanics are induced by radiation

1) Total Ionizing Dose (TID)

  - The cumulative, long term ionizing damage to the device materials
  - Caused by low energy protons & electrons

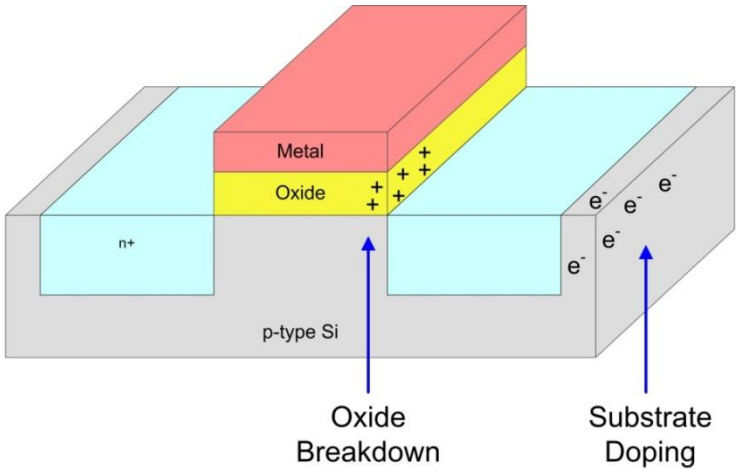
2) Single Event Effects (SEE)

  - Transient spikes caused by Heavy Ions and protons
  - Can be both destructive & non-destructive

Motivation (TID)

1) Total Ionizing Dose (TID)

- As the electron/holes try to recombine, they experience different mobility rates ($\mu_n > \mu_p$)
- Over time, the ionized particles can get trapped in the oxide or substrate of the device prior to recombination
- This can lead to:

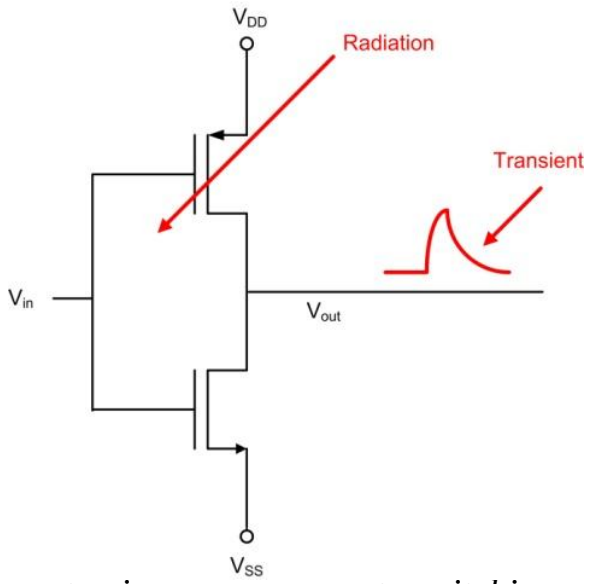


- Threshold Shifting
- Leakage Current
- Timing Skew

Motivation (SEEs)

2) Single Event Effects (SEEs)

- Transient voltage/current induced in devices
- This can lead to both Non-Destructive and Destructive effects



Non-Destructive	Behavior
Single Event Transient (SET)	*A transient spike of voltage/current noise, can cause gate switching*
Single Event Upset (SEU)	*A transient captured in a storage device (FF/RAM) as a state change*
Single Event Func. Interrupt (SEFI)	*A fault that cannot be recovered from using a reset.*
Multi-Bit Upsets (MBU)	*Multiple, simultaneous SEUs*

Destructive	Behavior
Single Event Latchup(SEL)	*Transient biases the parasitic bipolar SCR in CMOS causing latchup*
Single Event Burnout (SEB)	*Transient causes the device to draw high current which damages part*
Single Event Gate Rupture (SEGR)	*The energy is enough to damage the gate oxide*

Mitigation of TIDs

1) Current Mitigation Techniques (TID)

- Parts can be "hardened" to TID through:

  - layout techniques (sizing of Q_{crit}, enclosed layout)
  - guard rings
  - substrate doping
  - redundant circuitry

- Parts are specified in terms of:

  - "the amount of energy that can be tolerated by ionizing particles before the part performance is out of spec"

  - units are given in krad (Si), typically 300krad+

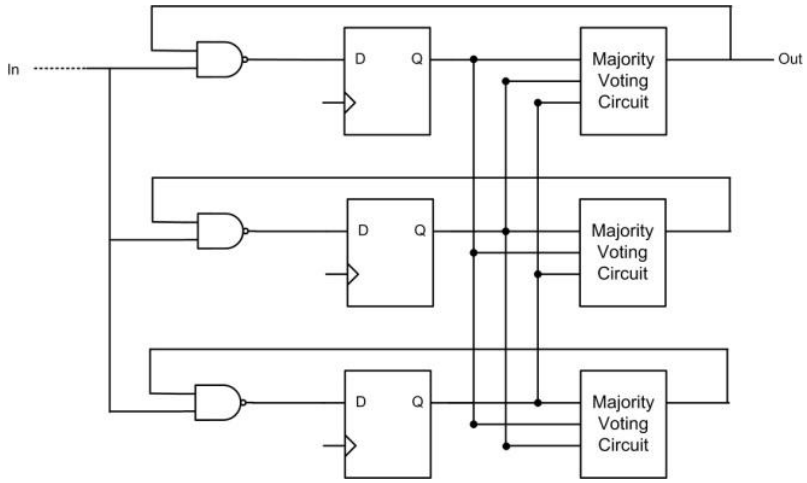
- Shielding Does Help

  - low energy protons/electrons can be stopped at the expense of weight

Mitigation of SEEs

2) Current Mitigation Techniques (SEEs)

- Triple Modular Redundancy (TMR)



- Reboot/Recovery Sequences

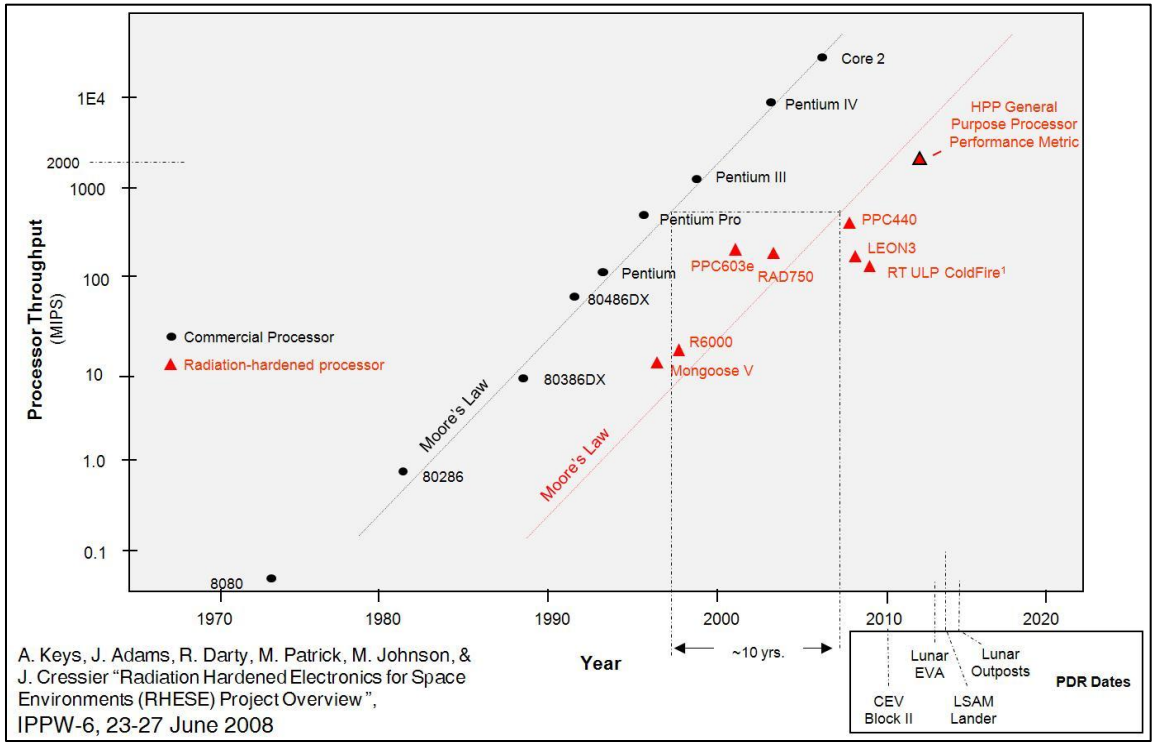
- Shielding Does NOT eliminate all SEEs

  - impractical to shield against high energy particles and Heavy Ions due to necessary mass

Drawback of Mitigation

- **Radiation Hardening = Slower Performance**

  - All TID mitigation techniques lead to slower performance



  - TID mitigation **DOES NOT** prevent SEEs

FPGAs & Radiation

- **Radiation Mitigation in FPGAs**

- RAM based FPGAs are traditionally *soft* to radiation

- Fuse-based FPGAs provide some hardness, but give up the flexibility of real-time programmability

- **Exploiting Reconfiguration**

- The flexibility of FPGAs enables novel techniques to radiation tolerant computing

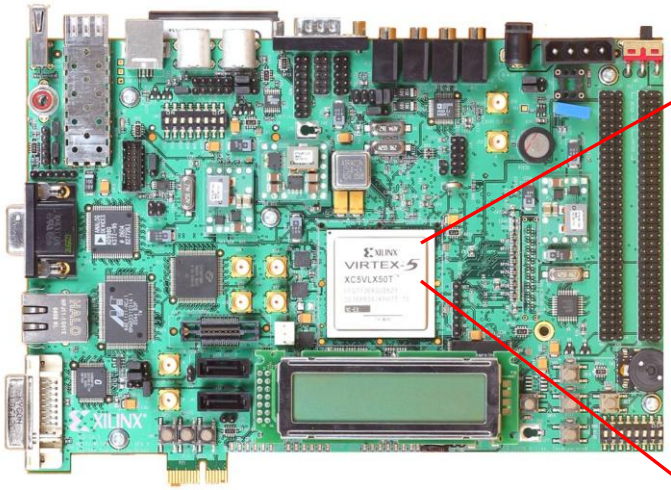
ex) Dynamic TMR, Spatial Avoidance of TID failures,

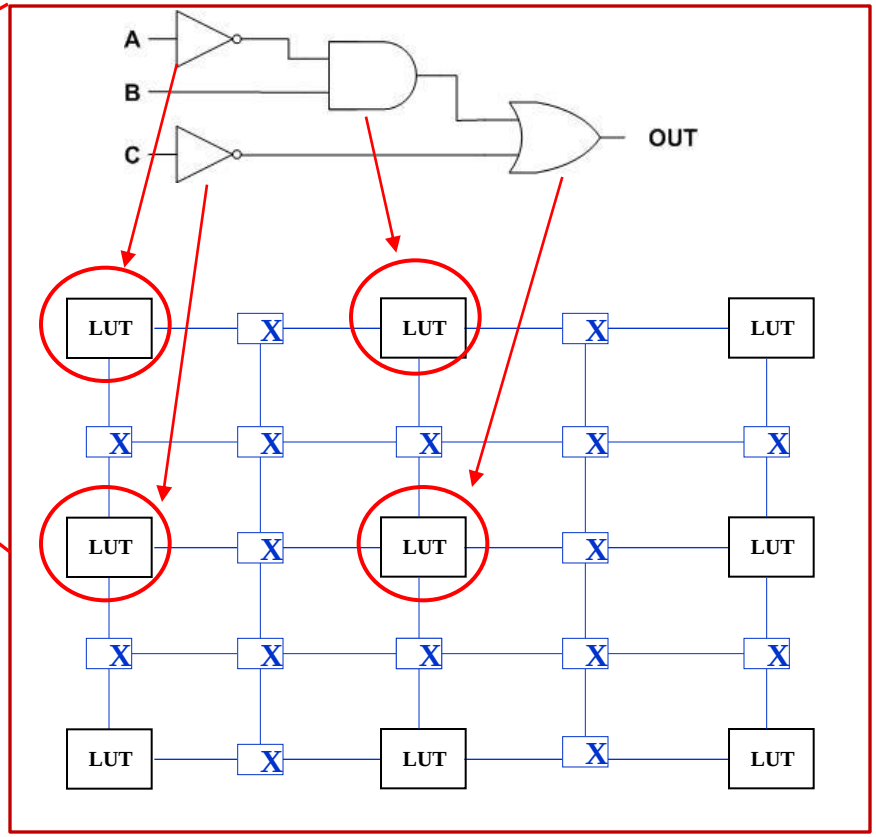
- The flexibility of FPGAs is attractive to weight constrained Aerospace applications

ex) Reduction of flight spares, internal spare circuitry

FPGAs as a Solution?

- **Field Programmable Gate Arrays**





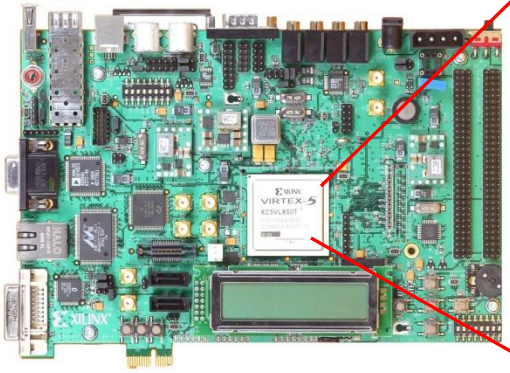
- FPGAs have followed Moore's Law and now yield comparable processing power to ASICs

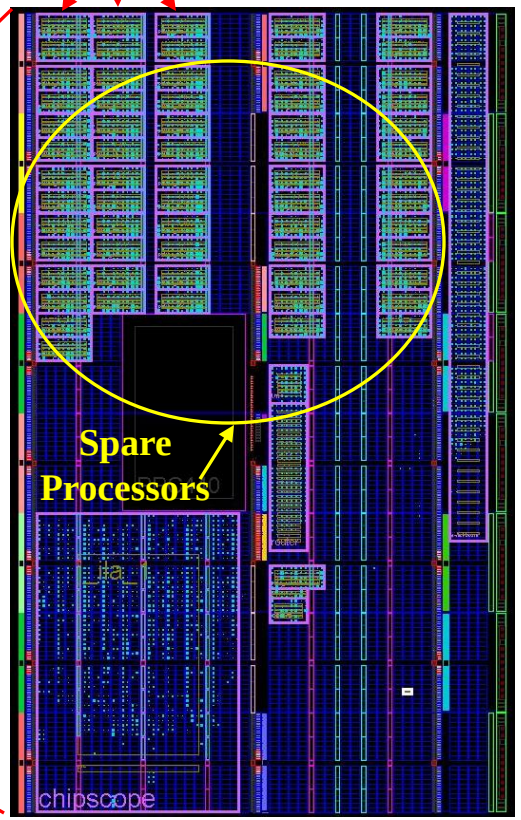
Many-Core Architecture

- **Radiation Tolerance Through Architecture**

  - Redundant, Homogenous, Soft Processors

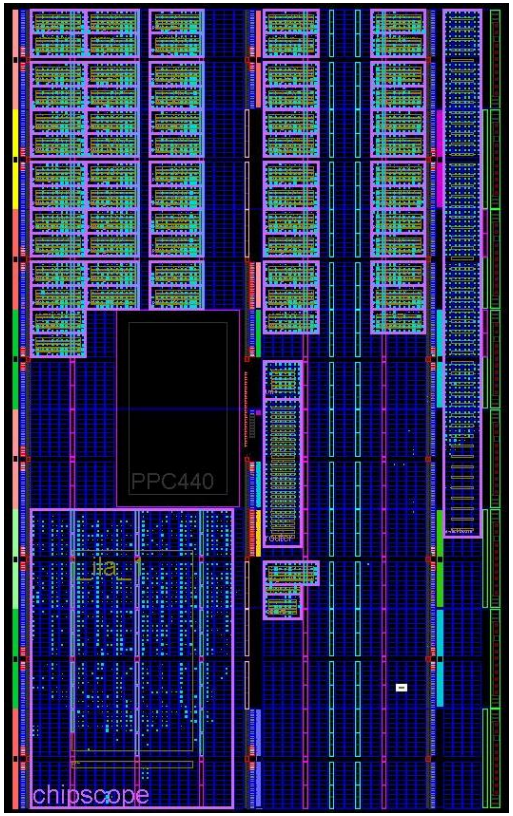
  - At Any Given Time, 3 are configured in *Triple Modular Redundancy* (TMR)





Many-Core Architecture

- **Types of Radiation Faults Seen in FPGAs**



1) Soft (SEU, SET)

- SEUs that can be recovered from using a reset

2) Medium (SEFI)

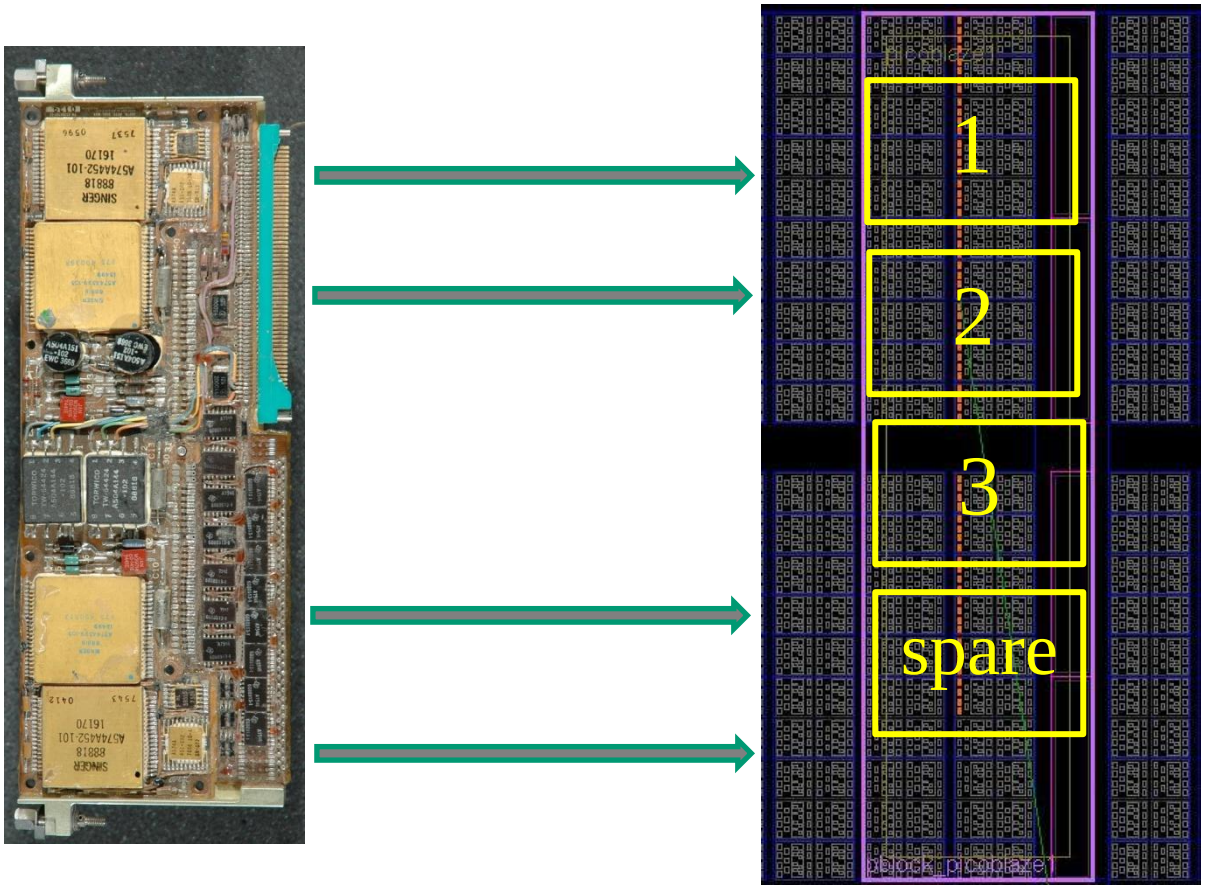
- SEUs in reconfiguration memory, can only be recovered using reconfiguration

3) Hard (TID / Displacement Damage)

- Damage to part of the chip due to TID or Displacement Damage

Potential Flight Computer

- ***microBlaze* Soft Processor**



Shuttle Processor Board

Virtex-5/6/7

Final Acknowledgements

- This work was supported by:

Montana Space Grant Consortium
(NASA EPSCoR)
http://spacegrant.montana.edu

NASA Exploration Systems Mission Directorate
"*Higher Education Program*"
http://education.ksc.nasa.gov/esmdspacegrant/

- Special thanks to our project mentors from NASA's *Advanced Avionics & Processor Systems (*AAPS) Project

Dr. Robert E. Ray
Marshall Space Flight Center
Reconfigurable Computing Task

Dr. Andrew S. Keys
Marshall Space Flight Center
AAPS Project Manager

Dr. Michael A. Johnson
Goddard Space Flight Center
High Performance Processor Task

Questions?