An Experimental Study of Heterogeneous Fault-Tolerant Systems: Preamble

David A. Rennels

UCLA

1. Systems are partitioned and functionality is distributed in order to manage complexity

- Subsystems are fitted with controllers or local computers
- Central Computing is thus simplified and more effective
- Examples
  - Controller cards in personal computers
  - Major Subsystems in large DOD systems
  - Spacecraft Experiments

2. Microcontrollers are the device of choice for commercial applications:

Highly integrated systems on a **single** chip with/

- Multi-MIPS processor, memory, digital inputs and outputs, analog inputs with on-chip A-to-D converter, asynchronous (RS232) port, serial ports, counters/timers.
- Sometimes with a standard integrated bus interface (e.g., the automotive CAN bus)

Sold by the millions at a few dollars to a few tens of dollars each.

3. Fault-Tolerant Design is a Critical Enabling Technology for Space Use of Microcontrollers

- Without it the error rate is too high in the space radiation environment and the potential for latchup causes unacceptable reliability

- **It is also an Enabling Technology for using Microcontrollers in Critical Ground-Based Applications Needing very High Reliability**
  - COTS microcontrollers have little or no built-in fault-tolerance capability

4. This Project is aimed at developing fault-tolerant microcontroller-based systems that can be used in space and in critical ground applications

- **Resulting in:**
  - Improved performance using these powerful highly integrated devices
  - Simplified system development with a wide range of parts to choose from and COTS software development tools
  - Lower life-cycle costs based on high volume commercial parts.

An Experimental Study of Heterogeneous Fault-Tolerant Systems: Preamble (Cont'd)

D. Rennels

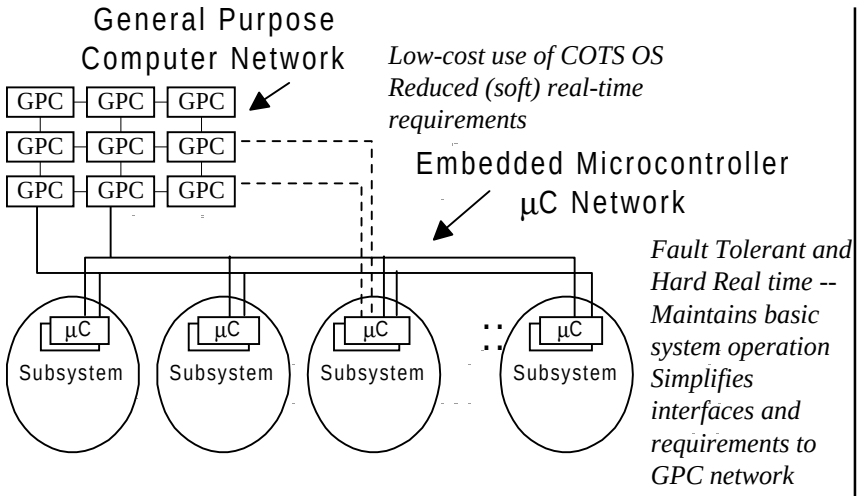
Typical Microcontroller Chip Properties (Intel 87C196CA)

- **16-bit computer with 20 mhz clock**
- **32KB ROM and 1280 KB RAM on-chip**
- **6 A/D Channels**
- **Parallel input and output ports**
- **3 serial I/O ports including an asynchronous serial port (RS-232)**
- **2 external interrupt pins**
- **timers**
- **a CAN (Controller Area Network) bus controller/interface**
- **typical power 0.25 Watt with lower for idle and power-down modes**

Objectives in the Fault-Tolerance Design

- to develop architectures for fault-tolerant embedded microcontroller nodes based on generic fault-tolerance techniques which are broadly applicable to many existing microcontroller chips.
- to develop a network of subsystem-embedded fault-tolerant microcontroller nodes that can operate like an autonomous nervious system for a complex system -- maintaining simple real-time operating modes for the system and guaranteeing system safety.
- to develop methods of interfacing high performance general purpose computing facilities to the underlying network of subsystems.
- to experimentally validate the techniques; and assess their effectiveness

An Experimental Study of Heterogeneous Fault-Tolerant Systems: Basic System Concepts



New Ideas

- New Fault-Tolerant Architectures using low-cost commodity microcontrollers to implement real-time dedicated subsystems.
- New Fault-Tolerance Hierarchy for complex dedicated systems -- Real-time and core-fault-tolerance functions off-loaded from General Purpose computers to network of embedded fault-tolerant microcontrollers.
- Simplifying GP functions, increasing performance, and allowing them to use COTS- derived hardware and software, conventional checkpointing, etc.
- Use of low-cost microcontrollers in space applications

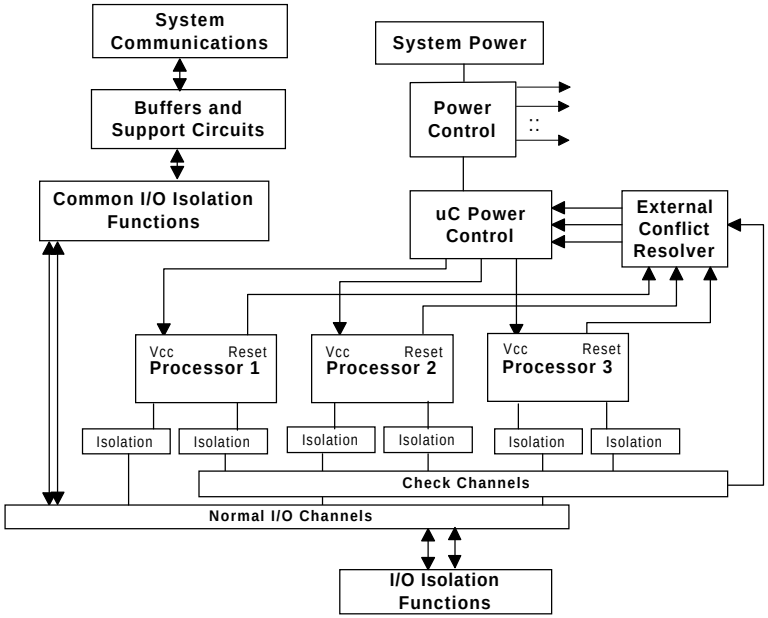
Impact

- Availability of highly fault-tolerant microcontrollers and small computers at costs in the range of $100.
- Systems where long-life dependable embedded computers serve as a trusted automated repair facility.
- Better understanding of how to partition complex dedicated systems derived from COTS technology to achieve high performance and fault-tolerance
- Space missions with
  - higher-performance electronics due to the use of more modern technology
  - more rapid turnaround at lower development costs

Schedule

Heterogeneous, Fault-Tolerant System Development	1996		1997				1998				1999	
	3Q	4Q	1Q	2Q	3Q	4Q	1Q	2Q	3Q	4Q	1Q	2Q
Fault-Tolerant Node	---	---	---	---	**1**	---	---	**2**				
Evaluate Node 1 - (fault-insertion testing)					---	---	**X**					
Evaluate Node 2							---	**---**	**X**			
Network of Nodes					---	---	---	**---**	---	**X**		
Test and Demonstrate Microcontroller Network								---	**---**	---	**X**	
Integrate with High Level Computers								---	---	**---**	**X**	
System Fault-testing and Demonstration									---	---	---	**X**

Node Design #1 A Spacecraft Node (PIC16C73A)



Objective

- To develop fault-tolerance techniques that will allow a low-cost (8-bit) COTS microcontroller node to be used in the space environment
  - recovery from a high rate of transient errors (single event upsets- SEU and “micro latchup)
- To implement a fault-tolerant node and verify the fault-tolerance features by experimental testing

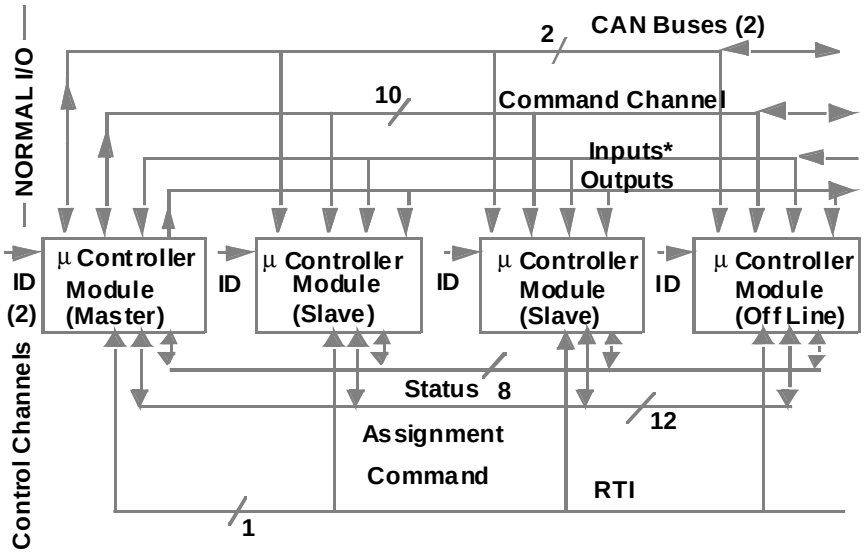
Technical Approach

- Triplicated processors exchange vote messages via a serial channel
- A special external conflict resolver (ECR) can reset or unpower processors based on two processor requests
- Each processor can request for itself or neighboring processsors to be reset based on the voting process
- Special I/O isolation is provided that protects against shorts and provides analog voting to protect outputs against transient errors
- Monitoring and power-cycling to clear latchup

Progress Highlights

- Construction of the fault-tolerant node was completed
- A fault-insertion mechanism has been developed that allows registers and memory locations in any processor to be faulted as normal programs run
- Preliminary fault-insertion testing has been carried out with encouraging results
  - for over 99.9% of transient faults inserted, computations are restored
  - More extensive tests are being prepared

Node Design #2: (87C196CA)



Objective

- To develop fault-tolerance techniques for general embedded real-time control applications using more capable 16-bit microcontrollers with expanded memory and integrated intercommunication buses
  - covering a wider range of faults
  - configurable for different levels of redundancy (duplex, duplex-duplex, TMR, or 4-processor hybrid)
  - to investigate a fully distributed recovery approach (complementing the node design approach above)
- To be integrated into a network of similar nodes and integrated with high-performance general-purpose computers

Technical Approach

- Master and Slave microcontroller modules:
  - Execute Identical Applications code, with OS software hiding voting and other redundancy functions
  - Master sends outputs and inputs via Master Channel for comparison/voting by slaves
  - Slaves signal agreement, disagreement, time-out by dedicated Status lines to all modules
  - Assignment Channel allows hardware vote of modules to replace faulty Master, and designate Slave, and Off-Line modules
- Power-switching for low-power and latchup recovery.

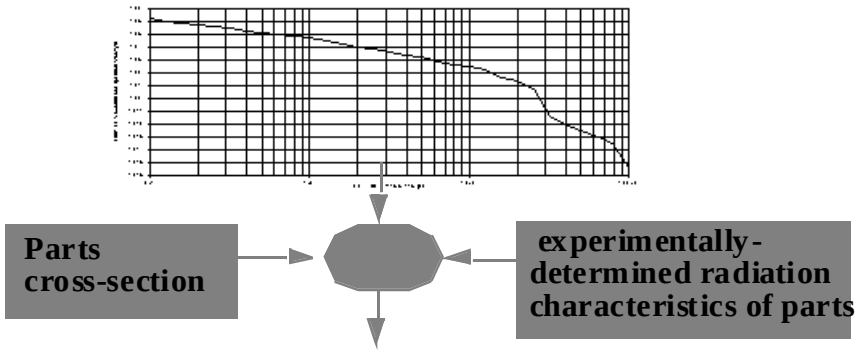
Progress Highlights

- Node Hardware Design is complete
  - Improvements made on preliminary design after studing failure modes and recovery algorithms
- An existing small real-time operating system for embedded applications is being ported to the microcontroller node, and a software development environment is currently under construction. (Consisting of a wire-wrapped processor interfaced to a PC fitted with software development tools and serial and parallel interfaces to the 87C196CA)
- Board layout of the fault-tolerant node is beginning and should be completed before summer.
- A paper was presented at the 1997 Pacific Rim International Symposium on Fault-Tolerant Systems, Dec. 1997, Taiwan.

Commodity Microcontrollers in Space -- Radiation Effects and Economics Studies

UCLA

Radiation Effects - Fault Environment



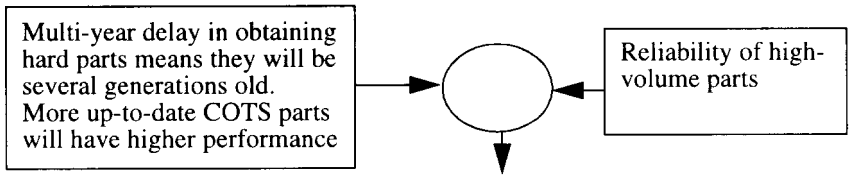
SEU - Rate, SEL - Rate, SEL Recovery Time Required
(SEU - Single-event upsets, SEL Single-event latchup)

Objective

- To determine how microcontrollers can be used cost-effectively in a space environment
  - Estimate rate of expected SEUs and latchup to more precisely determine the fault-tolerance requirements and to predict the reliability of the microcontroller nodes
- To quantify the potential advantages of using fault-tolerance with COTS parts over specially produced hard parts
  - Compare a non-fault-tolerant implementation of distributed microcontrollers using rad-hard parts, vs. our fault-tolerant node designs with respect to development costs, and launch costs in terms of weight and power for various orbits/trajectories, development time, and relative performance

Trade-offs of Hardened Parts vs. COTS Parts

	Rad Hard Characteristics	RAD Hard Cost	Automotive Characteristics	Automotive Costs
Microprocessor core	80C31RH	$ 3000	87C196CB(16 bit)	$ 30
ROM	4KB ROM	$1000	56KB in uC	$ 0
RAM	4KB RAM	$500	2 KB in uC	$ 0
Serial Bus (CAN)	FPGA	$ 2000	In uC	$0
A/D, D/A Converter	12-bit	$2000	In uC	$0
Other Comparisons: power, launch cost/Watt, weight, launch cost/Kg, latchup mitigation circuitry, .SEU detection circuitry, Power switching, etc.				



Trade-offs on Using Fault-Tolerant COTS Microcontrollers vs. Hardened Parts

Technical Approach

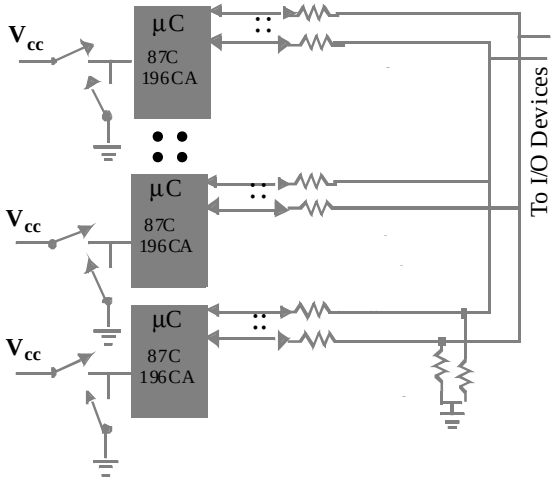
- Gather representative models of the space radiation environment for different mission types
  - Evaluate existing database on radiation resistance of commercial parts and experimentally evaluate selected microcontroller parts for radiation resistance
  - Develop generic design techniques for dealing with SEU and SEL and develop Reliability Models to predict the reliability of our Node designs.
  - Build comparative cost models of spacecraft computers using rad-hard parts vs.

Progress Highlights

- This work has been largely completed and will be included in D. Caldwell's Ph.D. Thesis

Electronic Design: Power Switching and Circuit Isolation (Node Design #1)

Isolation for Short Protection, Voting, and to Support Power Switching for Latchup Mitigation



Objective

- To prevent shorts as single failure points in a design where many signals are bussed due to limited interconnects.
- To make it possible to remove power from a module for latchup circumvention
- To provide protection against SEU's creating output errors.

Technical Approach

- A new approach is used to protect against transient errors changing outputs.
  - Inputs are resistor isolated and passive pull-down resistors are set so to create a logical OR.
  - Outputs are only set after a successful comparison
  - Two flip flops must be set in the microcontroller chips to generate a one. They are both set to zero to output zero and both to one to output a one. A single bit-flip in these flip flops will not change the composite output.
- Resistors prevent back-powering of turned-off devices
- Vcc is switched to ground to clear latchup in a device

Progress Highlight

- This innovative approach was developed this year and is implemented in the 8-bit microcontroller module.
- It combines circuit isolation for short protection and power swiching and provides a considerable degree of protection against SEUs upsetting outputs.

  (Of course a module can on rare occasion still fail in such a way as to generate an erroneous "one" output. This is detected by the voting process and periodic output comparisons, which will result in the module being reset.)

UCLA

An Experimental Study of Heterogeneous Fault-Tolerant Systems
Concluding Comments

- This project is aimed at taking very highly integrated, low-cost microcontrollers and augmenting their reliability by adding fault-tolerance so that they can be used in military and space applications, while still maintaining the low-cost features that make them attractive.
- Building systems of this type represents a significant challenge because the redundancy necessary to achieve fault-tolerance must also be inexpensive -- so not to destroy their cost advantage. Among these challenges are:
- developing fault-tolerant software that will fit in the memory available on these chips (e.g. 56KB ROM, 2KB RAM for the 16-bit chip and much smaller for the 8-bit chip)
- developing interfaces not too different from the non-redundant microcontrollers so that they can be put into existing subsystems, and so that differing amounts of redundancy can be easily employed in different applications using the same low-cost MCM-L (chip on board) building blocks.
- dealing with the hard-core issues of: i) exposing latent errors in highly integrated single-chip systems, ii) synchronization of redundant units, iii) algorithms for error detection, isolation and recovery, iv) achieving interactive consistency, and v) achieving a stable restart under multiple errors.

- Significant progress has been made toward the goals of understanding how to design and evaluate these systems:
    - A fault-tolerant 8-bit microcontroller node tailored for space use has been constructed, experimental testing has been started, and preliminary results are very encouraging
    - The design for the 16-bit microcontroller node is complete and hardware and software implementation are being started.
- We expect to complete testing of the 8-bit node and debug hardware and software of the 16-bit node in the summer followed by implementation of a distributed network of nodes in the fall.