CHALMERS
UNIVERSITY OF TECHNOLOGY

Fault Injection-based Assessment of Software Mechanisms for Hardware Fault Tolerance

Johan Karlsson

(with Ruben Alexandersson, Daniel Skarin, Raul Barbosa and Peter Öhman)

Department of Computer Science and Engineering
Chalmers University of Technology
Göteborg, Sweden

CHALMERS
UNIVERSITY OF TECHNOLOGY

Transistor variability and degradation

Shekhar Borkar, Intel Corp:

"*As technology scales, variability in transistor performance will continue to increase, making transistors less and less reliable. ….*

Finding solutions to these challenges will require a ***concerted effort on the part of all the players in a system design***."


DESIGNING RELIABLE SYSTEMS FROM UNRELIABLE COMPONENTS: THE CHALLENGES OF TRANSISTOR VARIABILITY AND DEGRADATION

AS TECHNOLOGY SCALES, VARIABILITY IN TRANSISTOR PERFORMANCE WILL CONTINUE TO INCREASE, MAKING TRANSISTORS LESS AND LESS RELIABLE. THIS CREATES SEVERAL CHALLENGES IN BUILDING RELIABLE SYSTEMS, FROM THE UNPREDICTIBILITY OF DELAY TO INCREASING LEAKAGE CURRENT. FINDING SOLUTIONS TO THESE CHALLENGES WILL REQUIRE A CONCERTED EFFORT ON THE PART OF ALL THE PLAYERS IN A SYSTEM DESIGN.

Shekhar Borkar
Intel Corp.

Borkar, S.; "Designing reliable systems from unreliable components: the challenges of transistor variability and degradation," IEEE Micro, December 2005.

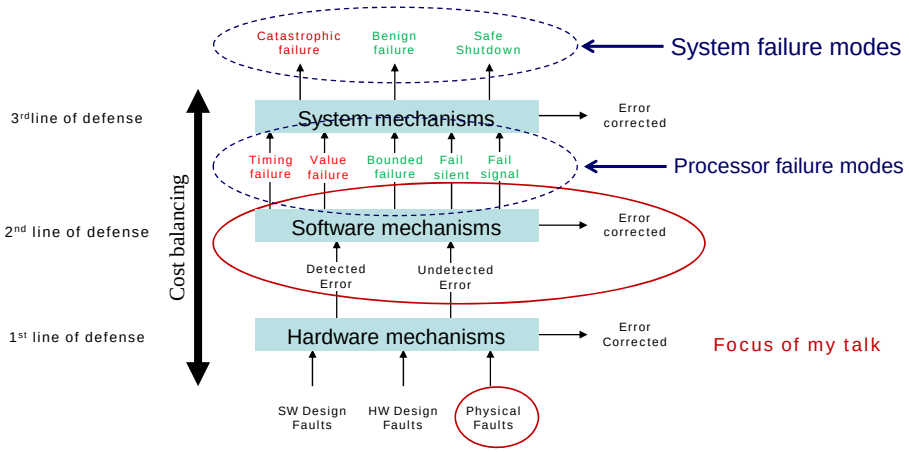
CHALMERS
UNIVERSITY OF TECHNOLOGY

Sources of transistor failures

- Process variations
  - Random variations related to lithography, etching, dopant count
  - Voltage and temperature variations
- Wear out effects
  - NBTI - negative bias temperature instability
  - HCI - hot carrier injection
  - Gate oxide breakdown
  - Electromigration
  - …
- Ionizing particle radiation
  - Cosmic neutrons, alpha particles, muons, pions, …

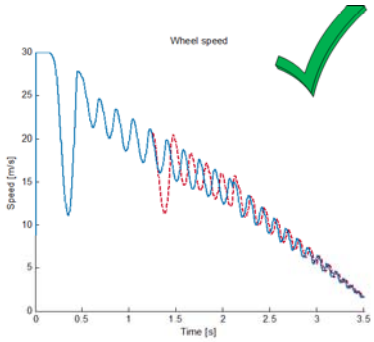
CHALMERS
UNIVERSITY OF TECHNOLOGY

Layered fault tolerance

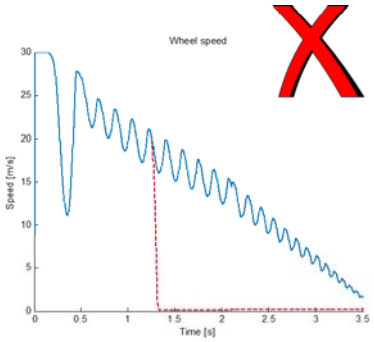


CHALMERS
UNIVERSITY OF TECHNOLOGY

Bounded failure vs unbounded failure for brake-by-wire controller



Benign failure



Critical failure

CHALMERS
UNIVERSITY OF TECHNOLOGY

Objectives of my talk

- To report on results from fault injection experiments with several software-based techniques aim at handling transient hardware faults
- Briefly discuss the use of aspect-oriented programming for implementing software-based fault tolerance
- Compare error coverage for two implementation techniques: Manual programming in C and Aspect-oriented programming
- Give some insights on how compiler optimizations influence the error coverage of software-based fault tolerance techniques

CHALMERS
UNIVERSITY OF TECHNOLOGY

Research Questions

RQ1 Can software-based mechanisms along with hardware exceptions enforce ***fail silent/fail reporting*** failure semantics for hardware faults that manifest as ***single-bit errors*** in instruction set architecture (ISA) registers and the data segment of main memory?

RQ2 How does compiler optimization influence error coverage?

RQ3 How does the implementation language influence error coverage?

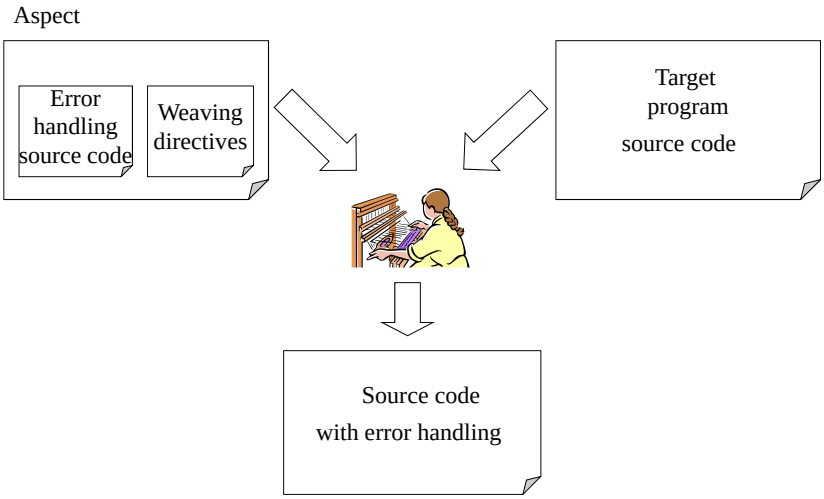
CHALMERS
UNIVERSITY OF TECHNOLOGY

Outline

- Aspect-oriented implementation of error handling mechanisms
- Experimental set-up
  - Target system, workload, fault injection tool
- Results for two software-based error handling mechanism
  - Execution time overhead
  - Error coverage
  - Impact of implementation language
    - Manual programming in C vs. Aspect-oriented programming
  - Impact of compiler optimization

CHALMERS
UNIVERSITY OF TECHNOLOGY

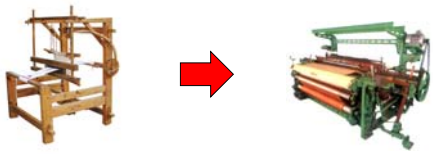
Aspect oriented programming



CHALMERS
UNIVERSITY OF TECHNOLOGY

Weaver versions

- Expressiveness study (LADC 2007)
  - Extension of the AspectC++ weaver
  - AspectC++$_{Ext}$
- Overhead study (EDCC 2010)
  - Optimized weaver
  - AspectC++$_{Opt}$



CHALMERS
UNIVERSITY OF TECHNOLOGY

Experimental set-up

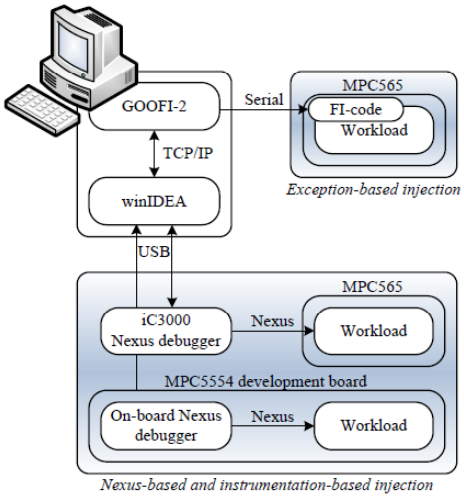
- Programs executed on a Freescale MPC565 PowerPC microcontroller

- Nexus-based fault injection
  - Injection via debug port
  - No need to change target program
- Experiments conducted with the GOOFI-2 tool

CHALMERS
UNIVERSITY OF TECHNOLOGY

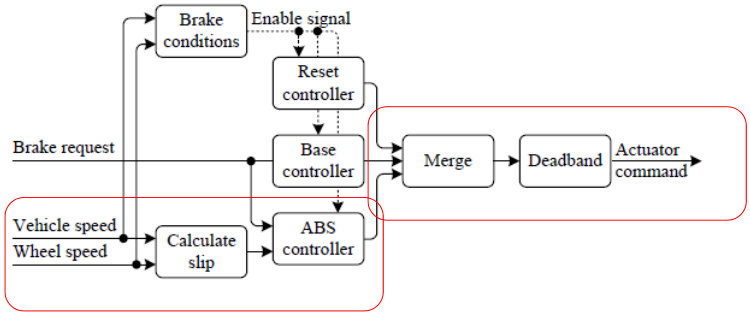
Overview of GOOFI-2

Generic Object-Oriented Fault Injection tool



CHALMERS
UNIVERSITY OF TECHNOLOGY

Workload: Brake-by-wire control loop



Parts of the program executed during fault injection experiments encircled in red

CHALMERS
UNIVERSITY OF TECHNOLOGY

Error model

- *Single bit-flip errors* in CPU registers and the data segment of main memory
- Bit-flips are injected in register/memory word just before the register/memory word is read. This avoids injection of errors in *dead data*
- We use *pre-injection analysis* of the execution trace to identifying live data items

Note: No guarantee for not injecting errors in data items that are *transitively dead*

CHALMERS
UNIVERSITY OF TECHNOLOGY

Triple time redundant execution with forward recovery (TTR-FR)

Purpose: Error masking and error detection

- Executes each control loop three times
- Errors masked by majority voting
- Three copies of program state
- Forward recovery: erroneous program state replaced with program state of correct version
- Error signaled if no majority result found

CHALMERS
UNIVERSITY OF TECHNOLOGY

Compiler optimization levels

- Low compiler optimization
  - GCC … -finline
- High compiler optimization
  - GCC … -O3 -fno-strict-aliasing

CHALMERS
UNIVERSITY OF TECHNOLOGY

Comparison of overhead for TTR-FR

	Low compiler optimization		High compiler optimization	
	No. of instructions	% overhead	No. of instructions	% overhead
Without TTR-FR	635	0%	245	0%
Manual C	2647	317%	943	285%
AspectC++$_{Opt}$	3428	440%	973	297%

No. of instructions = Number of machine instructions executed in one control loop
Target program: Brake-by-wire controller
Fault tolerance technique: Triple time redundant execution and voting with forward recovery (TTR-FR)
Implementation techniques: Manual programming in C and Aspect-oriented programming using the optimized weaver

CHALMERS
UNIVERSITY OF TECHNOLOGY

Error coverage – TTR-FR
(Triple Time Redundant execution with Forward Recovery)

Coverage	Over-head	No Effect	Corrected by Software	Detected by Software	Detected by HW Exception	Program Hang	Total Coverage
Low compiler optimization							
Manual C	317%	34.5%	15.2%	0.9%	45.6%	0.2%	96.4%
AspectC++$_{Opt}$	440%	33.2%	17.1%	0.5%	45.3%	0.3%	96.5%
High compiler optimization							
Manual C	285%	34.2%	18.4%	1.3%	41.9%	0.1%	95.9%
AspectC++$_{Opt}$	297%	32.6%	20.7%	1.7%	40.4%	0.2%	95.6%

Error model: Single bit -flips in CPU registers and volatile main memory
No. of injected errors for each program: 10.000

CHALMERS
UNIVERSITY OF TECHNOLOGY

Time redundancy and more (TRAM)

Purpose: error detection

- Six checking mechanisms
  - Double time redundant execution and result comparison
  - Stack pointer and stack frame pointer integrity checks
  - Check that writes are made to correct data set
  - Counter-based control flow checking
  - Check for fake resets

CHALMERS
UNIVERSITY OF TECHNOLOGY

Comparison of overhead for TRAM

	Low compiler optimization		High compiler optimization	
	No. of instructions	% overhead	No. of instructions	% overhead
Without TRAM	635	0%	245	0%
Manual C	1824	187%	689	181%
AspectC++$_{Opt}$	2358	271%	746	204%

No. of instructions = Number of machine instructions executed in one control loop
Target program: Brake-by-wire controller
Fault tolerance technique: Double time redundant execution + 5 other error detection mechanisms
Implementation techniques: Manual programming in C and Aspect-oriented programming using the optimized weaver

CHALMERS
UNIVERSITY OF TECHNOLOGY

Error coverage – TRAM

(Double time Redundant execution + 5 other mechanisms)

Coverage	Over-head	No Effect	Corrected by Software	Detected by Software	Detected by HW Exception	Program Hang	Total Coverage
Low compiler optimization							
Manual C	187%	33.3%	0%	21.5%	44.8%	0.3%	100%
AspectC++$_{Opt}$	271%	29.6%	0%	22.9%	47.4%	0.1%	100%
High compiler optimization							
Manual C	181%	34,2%	0%	24.2%	40.4%	0.1%	100%
AspectC++$_{Opt}$	204%	30.6%	0%	30.9%	38.4%	0.2%	100%

Error model: Single bit -flips in CPU registers and volatile main memory

No. of injected errors for each program: 10.000

CHALMERS
UNIVERSITY OF TECHNOLOGY

Research Questions

(No definite answers)

RQ1 Can software-based mechanisms along with hardware exceptions enforce ***fail silent/fail reporting*** failure semantics for hardware faults that manifest as ***single-bit errors*** in instruction set architecture (ISA) registers and the data segment of main memory?

Answer: Yes, it seems so, at least for simple control programs.

RQ2 How does compiler optimization influence error coverage?

Answer: We observed small mall or no differences in error coverage between different optimization levels. High compiler optimization led to a slightly reduced error coverage for the TTR-FR mechanisms.

RQ3 How does the implementation language influence error coverage?

Answer: We observed small variations in error coverage among different implementation techniques

CHALMERS
UNIVERSITY OF TECHNOLOGY

Future research

- Assess the validity of the single-bit flip approximation
- Experiments with other target programs
- Investigate impact of multiple-bit errors
- Develop pre-injection analysis techniques for identifying transitively dead registers/memory words
- Investigate impact of “out-of-spec” behavior of microprocessors
- And many more …

CHALMERS
UNIVERSITY OF TECHNOLOGY

Acknowledgements

- ***Ruben Alexandersson*** (now with Volvo Cars)
  - Designed and implemented the aspect weavers
  - Inventor/implementor of the TTR-FR, DS-CFC and TRAM mechanisms
- ***Daniel Skarin*** (now with the Swedish Technical Research Institute)
  - Designed the GOOFI-2 plug-in based software architecture
  - Adapted GOOFI-2 to the brake-by-wire application
  - Conducted a large number of fault injection campaigns with the brake-by-wire application
- ***Raul Barbosa*** (now ass. professor, University of Coimbra, Portugal)
  - Implemented the pre-injection analysis module for GOOFI-2
  - Develop support for instrumentation-based and exception-based fault injection
- ***Martin Sanfridson*** (Volvo Technology)
  - Developed the brake-by-wire application and the associated environment simulator
- ***Peter Öhman*** (now Head of Test Site Sweden AB)
  - Co-advisor to Ruben Alexandersson

CHALMERS
UNIVERSITY OF TECHNOLOGY

Questions?