Hardware Implementation, Processors and EMC

Prof. Chris Johnson,

School of Computing Science, University of Glasgow.

johnson@dcs.gla.ac.uk

http://www.dcs.gla.ac.uk/~johnson

University of Glasgow

Introduction

- Hardware Implementation Issues
- COTS Microprocessors.
- Specialist Microprocessors.
- Programmable Logic Controllers
- Electromagnetic Compatability.

COTS Microprocessors

- As we have seen:
  - safety of software jeopardised
  - if flaws in underlying hardware.
- Catch-22 problem:
  - best tools for COTS processors;
  - most experience with COTS;
  - least assurance with COTS...
- Redundancy techniques help...
  - but danger of common failures;
  - vs cost of heterogeneity;

COTS Microprocessors

- Where do the faults arise?
  1. fabrication failures;
  2. microcode errors;
  3. documentation errors.
- Can guard against 1:
  - using same processing mask;
  - tests then apply to all of batch;
  - high cost (specialist approach).
- Cannot distinguish 2 from 3?
- Undocumented instructions...

COTS Microprocessors

“Modern microprocessor chips are getting very complex indeed. The current gate count can exceed 2.5 million. One must therefore expect that new versions of such chips will contain logical bugs. A common form of bug is in the microcode, but since the distinction between a microcode fault and another form of design bug is difficult to define, the distinction is not made here. We are *not* concerned with fabrication faults."

“Attempts to report bugs openly have not been successful.   A consequence of the above is that it is very difficult of users undertaking a critical application to protect themselves against a potential design bug. One approach that has been tried with one project is to use identical chips from the same mask so that rig and development testing will extrapolate to the final system. In some cases, the suppliers have provided information under a non-disclosure agreement, be this seems to be restricted to major projects. In contrast, quite a few software vendors have an open bug reporting scheme --- and almost all provide a version number to the user. Hence it appears in this area, software is in `advance' of hardware."

Brian Wichmann, Microprocessor design faults (comp.risks)

COTS Microprocessors

- Early chips are unreliable: There have been some dramatic errors in very early releases of chips.
- Rarely used instructions are unreliable: One report sent to me reported that some instructions not generated by the `C' compiler were completely wrong. Another report noted that special instructions for 64-bit integers did not work, and when this was reported, the supplier merely removed them from the documentation!
- Undocumented instructions are unreliable: Obviously, such instructions must be regarded with suspicion.
- Case handling is unreliable: A classic instance of this problem is an error which has been reported to me several times of the jump instructions on the 6502. When such an instruction straddled a page boundary, it did not work correctly. This issue potentially gives the user most cause for concern, since it may be very difficult to avoid the issue. For instance, with machine generated code form a compiler, the above problem with the 6502 would be impossible to avoid."

University of Glasgow

Specialist Microprocessors

- Commercial microprocessor flaws.
- What happens if illegal opcode?
  - or result may be undefined?
- Motorola 6801 test instruction
  - fetches infinite bytes from memory;
  - good to test for faults on bus;
  - but could be executed erroneously;
  - see Storey or comp.risks for more.
-

University of Glasgow

Specialist Microprocessors - AAMP2

- Collins Avionics/Rockwell group.
- AAMP2
  - 30+ in every Boeing 747-400.
- High criticality implies cost
  - can you sell enough to cover input?
- What is money spent on?
  - extra time spent on design?
  - bench testing (see later);
  - formal verification...

Specialist Microprocessors - AAMP5

``The AAMP5 verification was a project conducted to explore how formal techniques for specification and verification could be introduced into an industrial process. Sponsored by the Systems Validation Branch of NASA Langley and Collins Commercial Avionics, a division of Rockwell International, it was conducted by Collins and the Computer Science Research Lab at SRI International. The project consisted of specifying in the PVS language developed by SRI a portion of a Rockwell proprietary microprocessor, the AAMP5, at both the instruction set and register-transfer levels and using the PVS theorem prover to show the microcode correctly implemented the specified behavior for a representative subset of instructions. The formal verification was performed in parallel with the development of the AAMP5 and did not replace any production verification activities.''

`This methodology was used to formally verify a core set of eleven AAMP5instructions representative of several instruction classes. The core set did not include floating point instructions. Although the number of instructions verified is small, the methodology and the formal machinery developed are adequate to cover most of the remaining AAMP5 microcode. The success of this project has lead to a sequel in which the same methodology is being reused to verify another member of the AAMP family of processors''

Steve Miller and Manadayam Srivas, Formal Verification of AAMP5, Comp.risks

University of Glasgow

Specialist Microprocessors - AAMP5

“Another key result was the discovery of both actual and seeded errors. Two actual microcode errors were discovered during development of the formal specification, illustrating the value of simply creating a precise specification. Both errors were specific to the AAMP5 and corrected prior to first fabrication. Two additional errors seeded by Collins in the microcode were systematically uncovered by SRI while doing correctness proofs. One of these was an actual error that had been discovered by Collins during testing of an early prototype but left in the microcode provided to SRI. The other or simulation."

Steve Miller and Manadayam Srivas, Formal Verification of AAMP5, Comp.risks.

- Formal verification can be done,
  - but still very expensive;
  - need techniques and tools;
  - reduce costs and increase subsets

Specialist Microprocessors

- Verifiable Integrated Processor for Enhanced Reliability (VIPER)
- This is an old story
  - but still very controversial.
- Royal Signals & Radar Establishment.
- LCF-LSM and Ella.
  - Big claims about confidence levels:
  - did MOD claim ``fully proven''?
  - proof from spec to production?

Specialist Microprocessors: VIPER

- Charter technologies market the chip.
- Sue MOD over ``ungrounded'' claims.
- Charter into liquidation as costs rise.
- Key lesson:
  - general ignorance about proof;
  - argument not absolute guarantee.

Specialist Microprocessors - 1750A, 1750B

- Introduced in 1979, Revised in 1982.

- Deactivated in 1996.
- Well documented/understood.
- But dont forget safety of language.
- Not just processor reliability.
- Started but never completed?

- 1750A remains a de facto standard.

Specialist Microprocessors - ERC32

- Reliable not safety-critical?
- Space and radiation tolerant.
- Ada development tools.
  - integer unit (IU);
  - floating-point unit (FPU);
  - memory controller (MEC).
- Single-chip (TSC695E) June 2009.

Microprocessors

- MIL-STD1750 - expensive.
- Select processor for application.
  - Storey cites widespread use;
  - range of less critical areas.
  - specifically for airbag applications;
  - “general purpose in-system programmable microcontrollers".

Programmable Logic Controllers (PLCs)

- Self contained:
  - power supply; interface circuitry; 1+ processors.
- Different from GPCs (eg Shuttle):
  - replace electromechanical relays;
  - perform simple logic functions.
- Designed for high MTBFs:
  - kernels provide trusted functions;
  - proprietary source for firmware.

Programmable Logic Controllers (PLCs)

- Widely used, well tested
- But hard/software proprietary.
- Beware: STUXNET
- Certification by trusted bodies.
- However...

Programmable Logic Controllers (PLCs)

``Lin Zucconi'' lin_zucconi@lccmail.ocf.llnl.gov

People using Modicon 984 Series programmable controllers with Graysoft Programmable Logic Controller (PLC) software Version 3.21 are advised to contact Graysoft (414) 357-7500 to receive the latest version (3.50) of the software. A bug in Version 3.21 can corrupt a controller's logic and cause equipment to operate erratically. PLCs are frequently used in safety-related applications. Users often assume that if their ``logic'' is correct then they are ok and forget that the underlying logic is implemented with software whichmay not be co rrect.

Lin Zucconi zucconi@llnl.gov

Programmable Logic Controllers (PLCs)

- PC emulators to develop software:
  - download to target PLC;
  - volatile store is dangerous;
  - wide use of EEPROMS.
- Fail safe PLC's:
  - two or more independent CPU's;
  - voting forms of redundancy;
  - if conflict close down in safe state
- Several graphical design techniques:
  - ladder & function block diagrams...
  - See Storey for more detail.

Electromagnetic Compatability (EMC)

- Work in presence of interference.
- AND not create interference.
- Interference from external noise.
- Interference from external source.
- 2 radio signals on same frequency.

University of Glasgow

Electromagnetic Compatability (EMC)

- Difficult to predict.
- Intensity changes over time;
  - eg with work patterns;
- Sources may also be mobile;
  - or you may be mobile!
- Mobile telephones, car ignitions...

University of Glasgow

Electromagnetic Compatability (EMC)

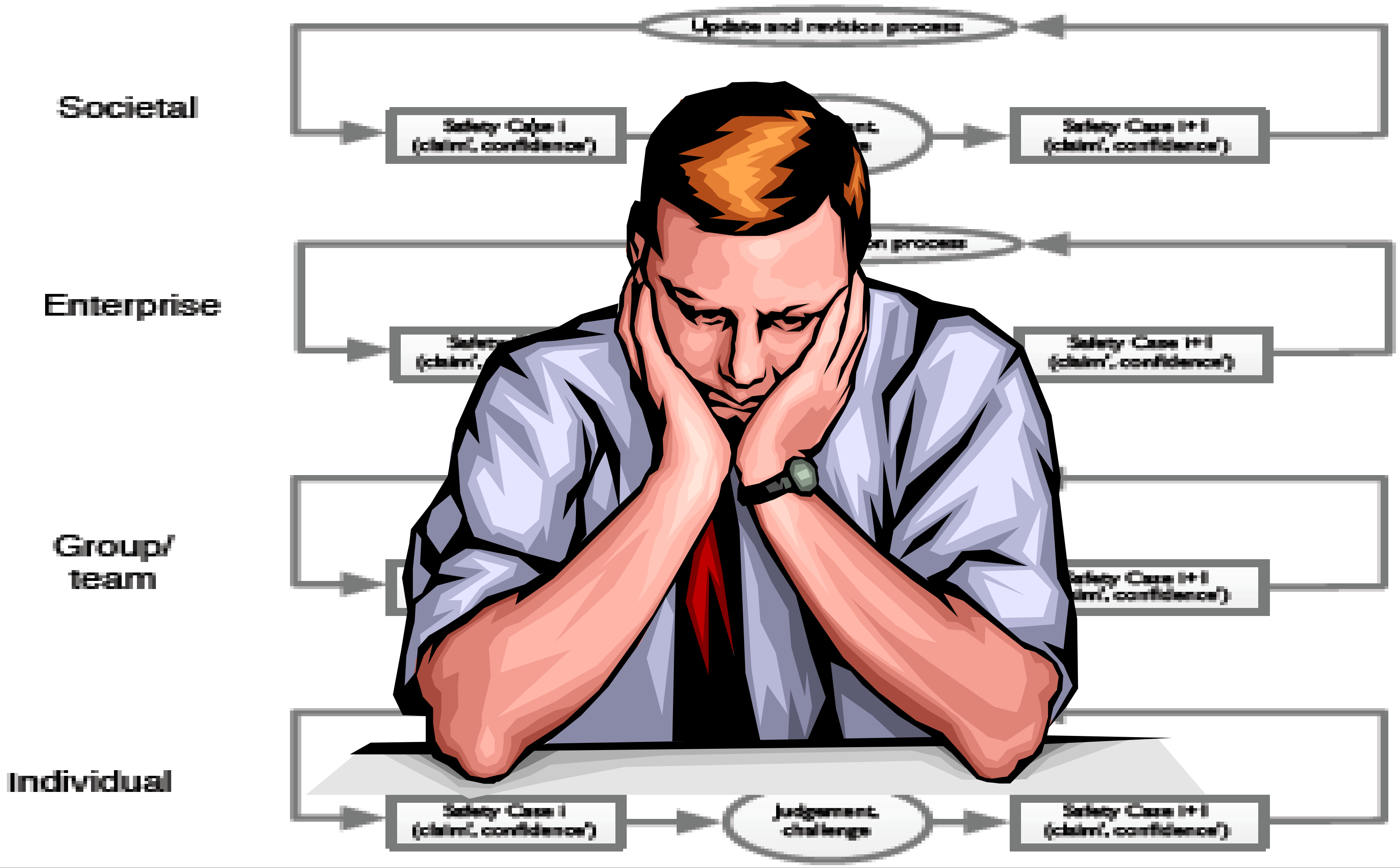
- Protection.
- Screening: use conductive cage/enclosure.
- Check design of PCBs if possible:
  - (power) loops form antennas;
  - check use of ground planes.
- Check CMOS output capacitance;
  - can buy chips (Philips 8051);
  - help discriminate signal edges.
- Seek help from a specialist...

Conclusions

- Hardware Implementation Issues
- COTS Microprocessors.
- Specialist Microprocessors.
- Programmable Logic Controllers
- Electromagnetic Compatability.

University of Glasgow

Any Questions…



Update and revision process
Societal
Safety Case i (claim', confidence')
Safety Case i+1 (claim', confidence')
Enterprise
Safety Case i+1 (claim', confidence')
Group/ team
Safety Case i+1 (claim', confidence')
Individual
Safety Case i (claim', confidence')
Judgement, challenge
Safety Case i+1 (claim', confidence')