Controls Interface to Electronics Boards

Credit-Card PC and Local Control bus

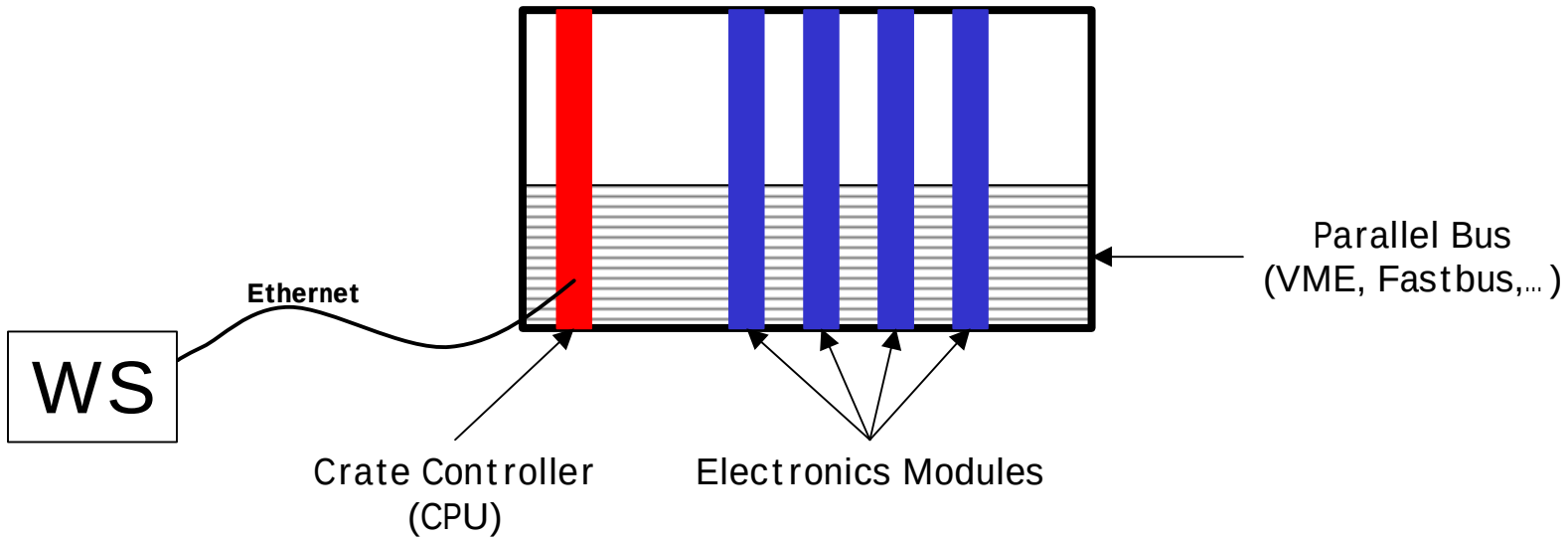
Beat Jost
Cern EP

Outline

- Introduction
- Goal
- Current Ideas for Solution
- Discussion and Conclusion

Problem Introduction

Classical Way of controlling electronics in HEP



Pros:

- Universally available
- simple? slave interface
- in the past bus could also be used for DAQ

Cons:

- expensive CPUs (very small market)
- expensive crates
- expensive slave I/Fs

Assumption for the Following

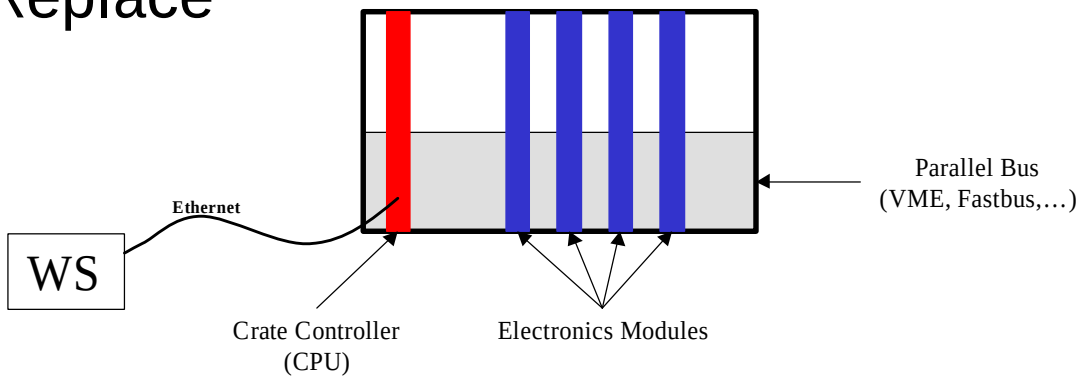
- The vast majority of the electronics boards are home-made
- Crate bus is not used for moving the physics data in LHC experiments, because
  - (at least) at higher levels of the readout systems, performance is insufficient
  - trigger rate too high for processor intervention on per event basis
- Crate bus is only used for control and monitoring
  - for this purpose a “high performance” bus is **not needed**
  - for this purpose a parallel bus is **not desired** for reliability reasons
    - One participant on the bus can prevent bus accesses even if it’s not involved
    - makes diagnostics more difficult
- Hence
  - A more reliable and perhaps even more cost effective alternative is desired.
- By the way: Crates are useless for dispersed individual boards

Goals

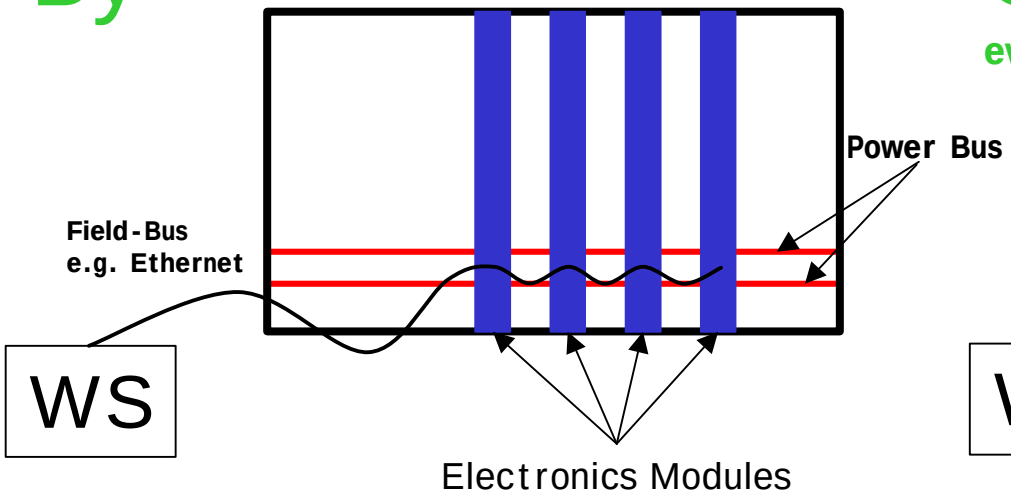
- Get rid of parallel busses for controls
- Find a cheaper solution for
  - per-board controls interface (slave)
  - per crate intelligence, by taking it out of the crate formfactor
    - Use commodity items
  - Crates (no parallel bus needed anymore)
    - Reduce crates to
      - Mechanical support (“Anti-Gravity device”)
      - Power Bus (could be arguable)
      - Cooling (Fantray)
- Take advantage of large market (low price)
- Provide a common controls interface for **ALL** electronics boards in LHCb

Current Ideas

Replace

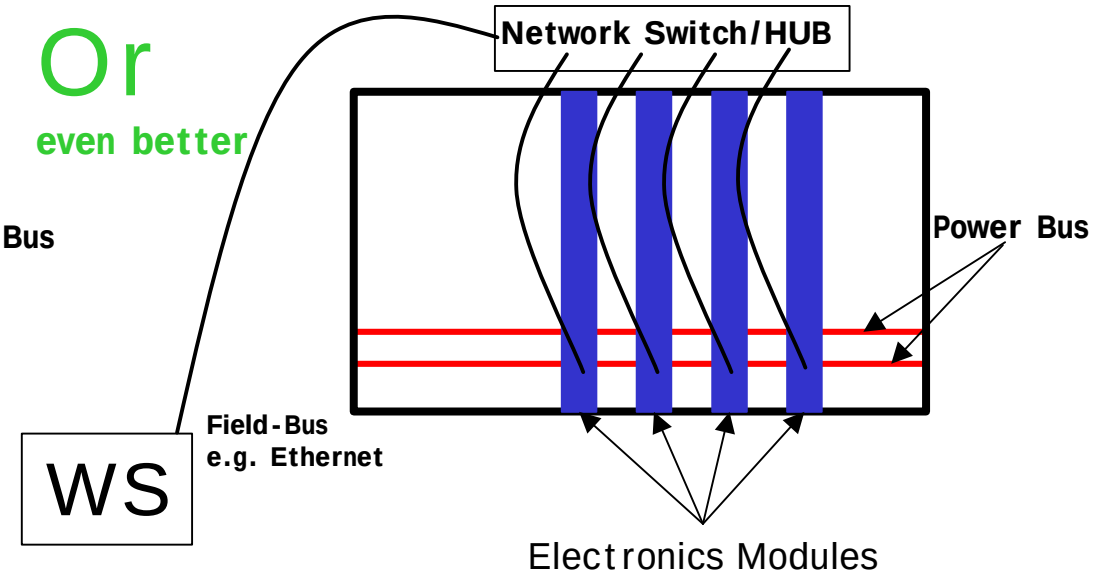


By



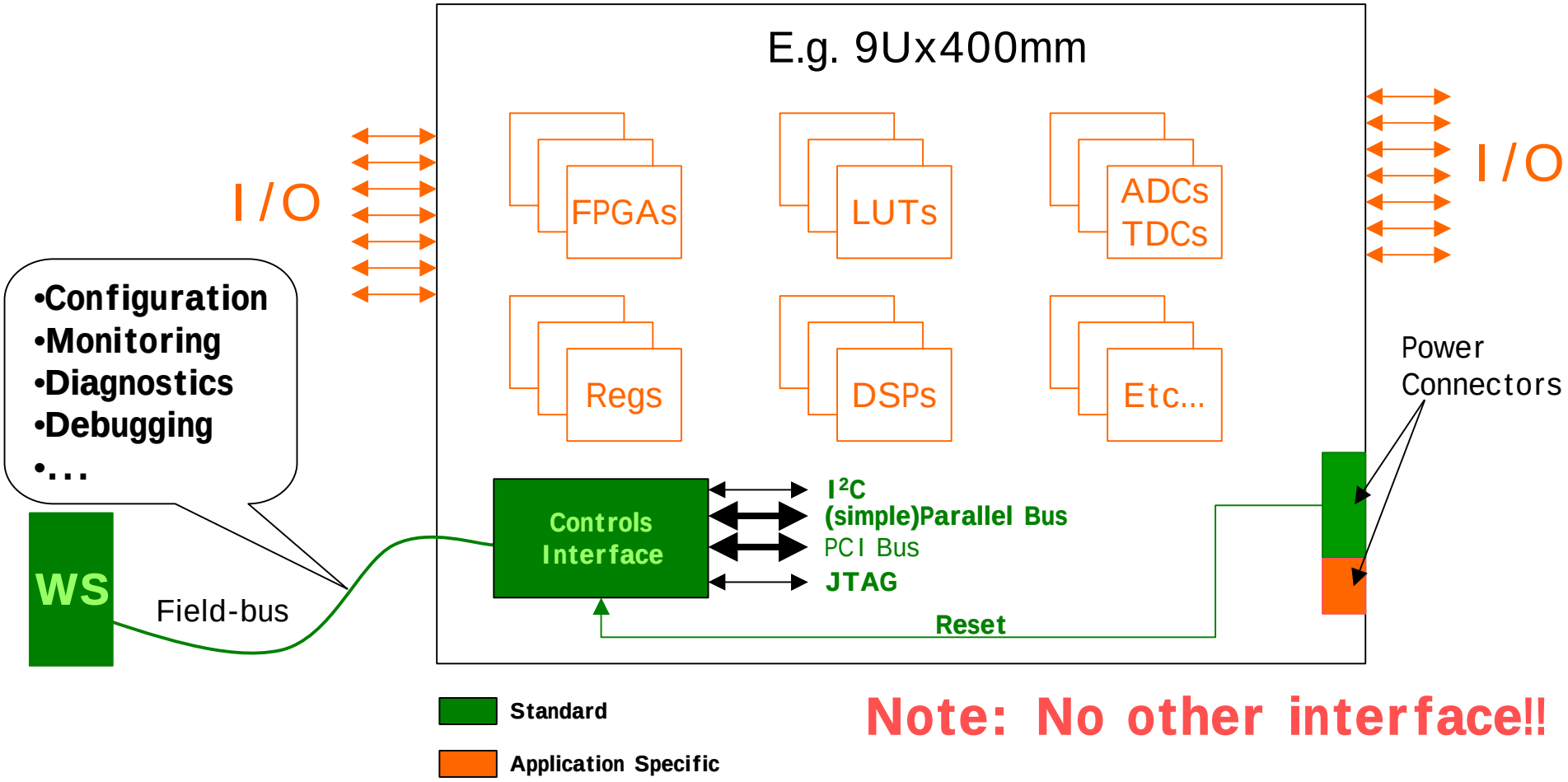
Or

even better



Electronics Board Architecture

The architecture of an electronics board could look like this:



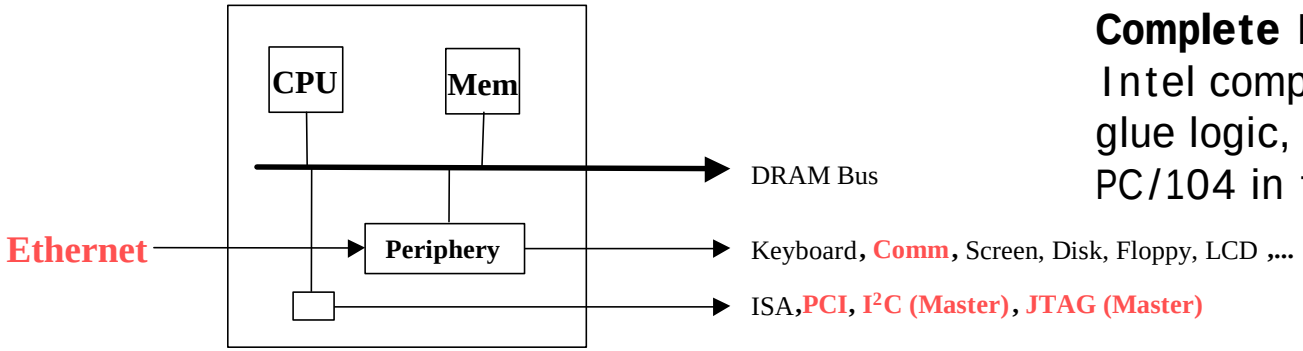
Note: No other interface!!

Requirements for On-Board ECS Interface

- Sufficient bandwidth into each board (10-100 Mb/s)
- Cost per board must be low
  - Uniform approach for "all" electronics (volume)
  - for Ethernet the cost of a switch or hub port has to be taken into account
    - 10 Mb/s hub port ~30 SFr. (today)
    - 100 Mb/s switch port ~80 SFr. (today)
    - Can be mixed!!
- Mechanical
  - minimal height (thickness)
  - minimal surface
- Software support
  - Low-level access libraries from WS to board components
  - Tools supporting the programming of a CPU (if present) on the interface
- Reset of ECS interface without disturbing the operation of the rest of the electronics on board

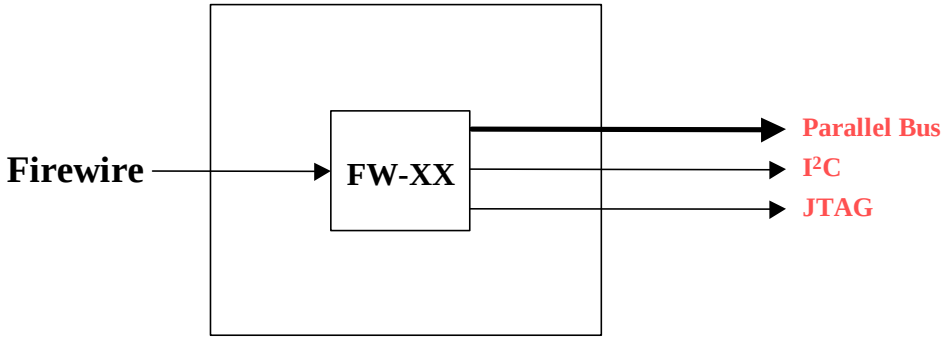
Possible Solutions under Study

- Credit-card PCs (e.g. smartModules)



Complete PCs!! Typically based on Intel compatible microcontroller plus glue logic, could replace partially PC/104 in the future.

- FireWire

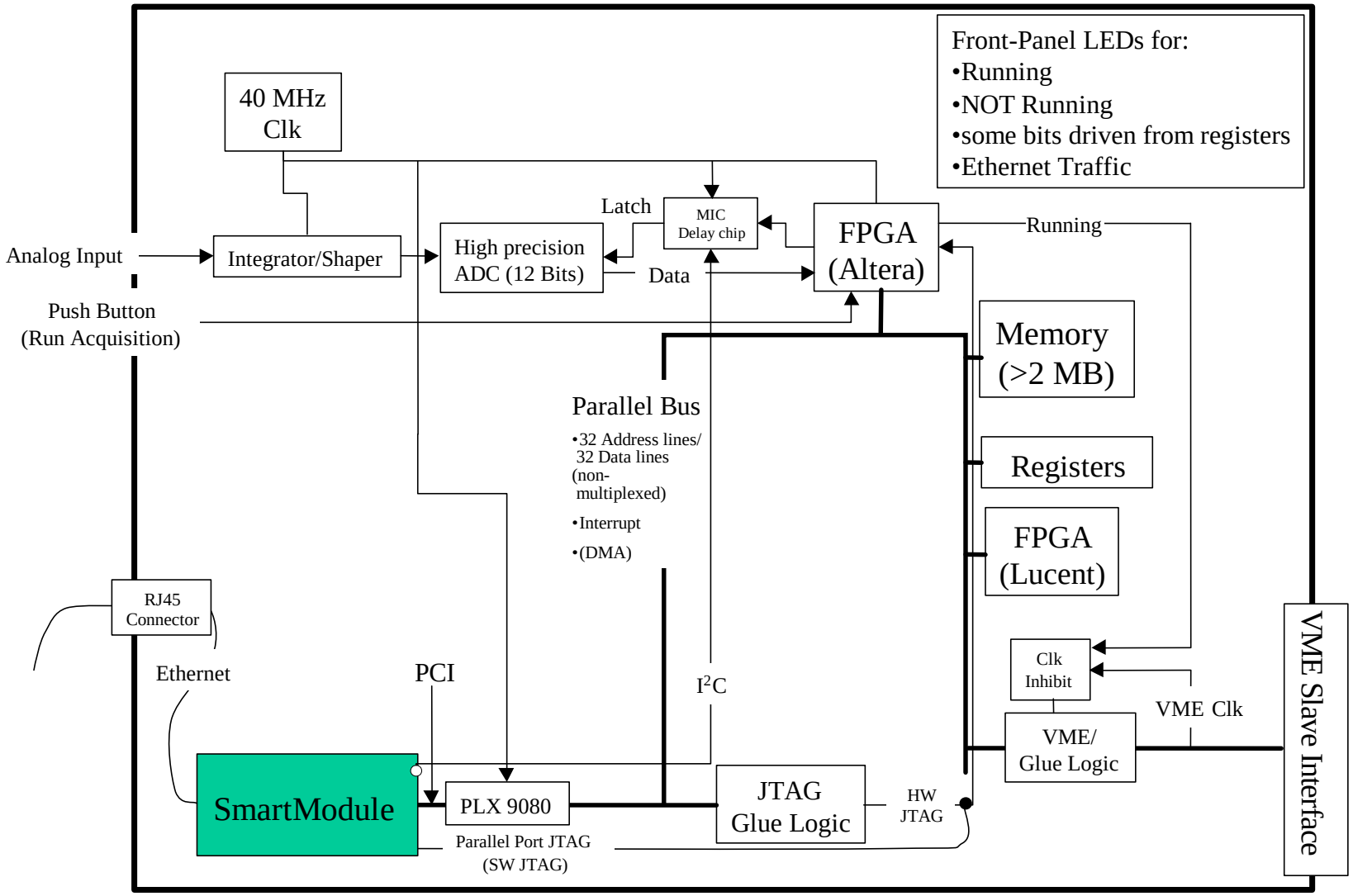


- We also looked at PC/104, but it doesn't seem suitable mechanically

Immediate Plans

- For SmartModules
  - acquire one evaluation kit and one “bare” module
    - Evaluation Kit with 486 acquired, porting currently Linux to it
    - Final (586-based) module not yet available
  - learn how to program the processor using the evaluation kit
  - build an evaluation board with FPGA(s), memory, registers, LEDs, to study how to use the modules and whether they are suited for our application
    - being designed now. Ready in ~3 months?
  - Should be able to give guidelines for designers by November
- FireWire
  - try to understand the availability of interfaces/bridges
    - for example: we know that there exists a Firewire to IDE chip. Can this be used?
    - Similar (or same) evaluation board as above to prove usability

SmartModule Evaluation Board



Local Bus Characteristics

Based on PLX PCI 9080 chip
(see also http://lhcb.cern.ch/computing/controls/pdf/9080db-106.pdf)

Goal: It should be easy to interface to components to local bus

- 32 bit address
- 32 bit data + Byte Parity

} Non-multiplexed

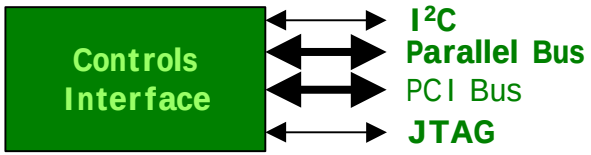
- Control lines (address strobe, R/W, Byte enable, etc.)
- Local bus clock frequency 0-40 MHz
  -> allows to run bus synchronous with LHC Clock
- local bus can be Little or Big Endian
- One interrupt line from local bus to PCI -> might need external interrupt register/generator
- not terribly fast:
  - 8.3 MB/sec PCI reading from local bus
  - 8.8 MB/sec PCI writing to local bus

} Single Cycle from timing diagrams

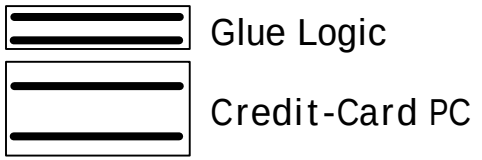
- PCI DMAs translated into local bus bursts

Other Issues

- Are the proposed interfaces (I^2C, JTAG, parallel bus, (PCI)) acceptable and sufficient?



- Hardware configuration, e.g. amount of memory required
- Cost
- Interference with on-board analog electronics
- Reset of controls interface while taking data (SUE recovery)
- Final Implementation: Glue logic on separate board or integrated on motherboard?



Pros and Cons

- Pros
  - only point-to-point connections
  - flexible concerning association between controls WS and electronics board (scalability)
  - programmable element on board allows extensive self-test and diagnostics
  - allows to use (design) cheap crates
  - test-benches and repair stands are very simple and cheap: just need a PC or terminal and an Ethernet cable and a power supply
- Cons
  - potentially cost (even though it seems that e.g. smartModules are competitive with VME)
  - difficult to mix in one crate commercial (VME-based?) modules
  - currently only few vendors (issue is more the pin-out), however market is growing, Effort going on to standardize pin-out.

Conclusions

- We are studying alternatives to crate-based solutions (VME) to be used for the control, configuration and monitoring of board-level electronics
- Our Goal is to find a solution that is, compared to VME,
  - more flexible
  - more reliable
  - not more expensive (actually much cheaper if cheap crates are used)
- Credit-Card PCs could be a viable solution
- The results of the evaluation board will make us know more

smartModule vs. VME Cost Analysis

SmartModule Cost (Elan based)		
	Component cost [CHF]	
smartModule	250	
Glue Logic	80	e.g. PLX 9080
Ethernet HUB	30	per port
Total per Board	**360**	

VME Based System Cost			
	Unit Cost	Average Per Slot Cost [CHF]	
VME Slave Interface Chip (TUNDRA Trooper II)	60$	100	Quantity of 500
VME Processor	5000	333	assuming 15 slots occupied (in average)
Subtotal		433	
Glue Logic		17	
		450	