PERFORMANCE OF THE STANDARD FAIR EQUIPMENT CONTROLLER PROTOTYPE

Stefan Rauch, Ralph C. Bär, Wolfgang Panschow, Matthias Thieme

GSI, Darmstadt, Germany

Abstract

For the control system of the new FAIR accelerator facility a standard equipment controller, the Scalable Control Unit (SCU), is presently under development. First prototypes have already been tested in real applications. The controller combines an x86 COM Express™ Board and an Altera Arria™ II FPGA. Over a parallel bus interface called the SCU bus, up to 12 slave boards can be controlled. Communication between CPU and FPGA is done by a PCIe link. We discuss the real time behaviour between the Linux OS and the FPGA Hardware. For the test, a Front-End Software Architecture (FESA) class, running under Linux, communicates with the PCIe bridge in the FPGA. Although we are using PCIe only for single 32 bit wide accesses to the FPGA address space, the performance still is sufficient. The tests showed an average response time to IRQs of 50 μs with a 1.6 GHz Intel Atom CPU. This includes the context change to the FESA user space application and the reply back to the FPGA. Further topics are the bandwidth of the PCIe link for single/burst transfers and the performance of the SCU bus communication.

Description of the SCU

- SCUB connections with LVDS and PCIe
  - strict Master/Slave bus
  - 16Bit data bus, 16Bit address bus
  - in addition serial LVDS and PCIe channels
- COM Express™ module Type II with Intel Atom
- 32Mbyte Parallel Flash
- RJ45 Gigabit Ethernet from COM Express™ module
- two SFP connectors
  - Timing (White Rabbit over GbE)
  - AUX LAN (GbE)
- Arria™ II FPGA
- White Rabbit Timing Receiver (FPGA)
- 128Mbyte DDR3 memory
- IO Extension for existing timing system
  - DEV-BUS: GSI fieldbus
  - Event-In: Timing Events
  - both based on MIL-STD-1553
- USB 2.0
- 2x EIA-232

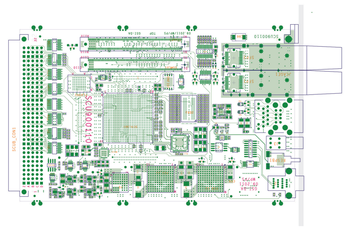

Figure 1: *Carrier Board Layout*

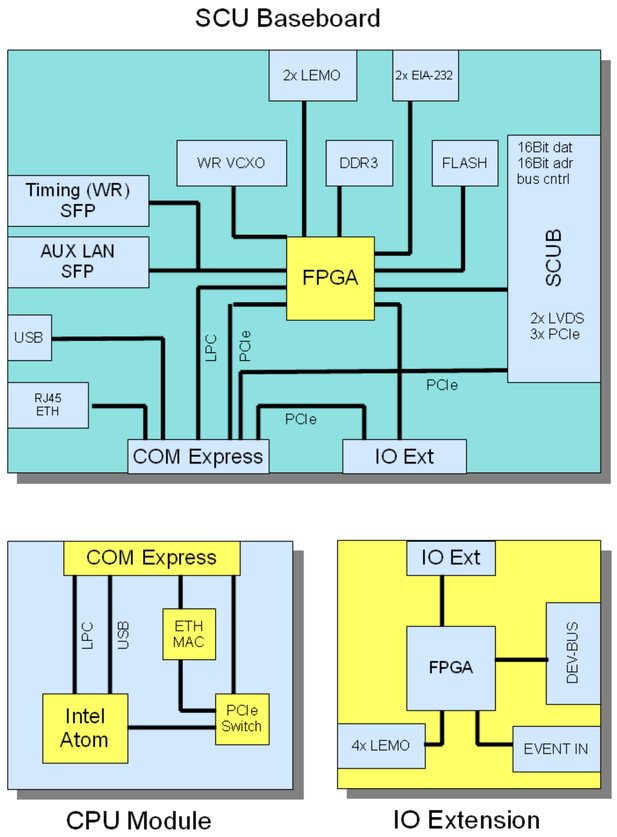

Figure 2: *SCU schematic*

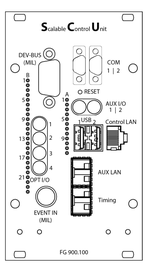

Figure 3: *Front Panel*

Problems during design phase

- fitting of high speed serial transceivers
- timing issues with DDR3 memory controller
- errors with DDR3 communication
- layout errors based on unclear documentation and design tool errors

The Arria™ FPGA comes with a hard IP cell for PCIe so no license fees are needed. This FPGA and the design tools were quite new, when the testing of base components of the system like PCIe or DDR3 communication began. In FPGA prototyping you typically implement the design first in the FPGA due to constraints in the pinning of the PCB. In this special case, the design tool allowed the hard IP cell to be connected to the transceiver block 1 of the Arria™. The problem is, there is no hardware connection in the FPGA and the PCBs did not run PCIe with the hard IP cell. One possibility to solve this problem was the use of a soft core implementation. This would result in high license fees and unnecessary loss of logic cells in the FPGA. Another temporary solution was rewiring the PCB. After patching the PCIe to transceiver block 0 the connection worked perfectly. Altera fixed the issue in the next version of their design tool, but for the final solution a redesign of the PCB was needed.

Lessons Learned

- plan a lot more time for testing new technologies
- new FPGAs with high speed transceivers and mem controllers have lots of constraints
- a lot of independent clock inputs are needed for optimal routing
- layout errors based on unclear documentation and design tool errors
- the HDL design has to be frozen early in the project
- most changes cannot be made after PCB is finished

In Detail: Implementing a LPC UART

- Problem: SuperIO Chip does not fit in Layout
- Solution: extending LPC bus to FPGA, implementing UART functionality in logic
- Low Pin Count Bus, four address/data lines, serial clk, frame signal
- running at 33MHz
- IO read/write, MEM read/write
- 16750 compatible UART core
- Whishbone LPC Peripheral Bridge
- emulating extended function mode of SuperIO chip
- up to 11 devices, e.g. UARTs, printer port, MIDI
- base address of UART can be configured
- 16750 UART is then mapped to

Figure 4: *LPC UART schematic*

Status Report of the Project

The prototype should have been ready in July 2011, but to complications during the design phase it came to a delay. The layout for the prototype baseboard is now finished and ready for production. The first boards are expected to be shipped in November 2011.