MSU/NSCL May 2009

CoBo Module

Specifications

Version 1.0

Nathan USHER

1. Introduction

This document specifies the design of the CoBo module of GET. The primary task of the CoBo is to readout the ASICs on the AsAd, compress the data, and forward it to the data acquisition system. This must be done with minimum dead time to allow the high event rates required by GET. Therefore, the CoBo firmware must be designed such that the data readout time is limited by the speed of the SCA/ADC sample clock, not the CoBo.

Other tasks that must be performed by the CoBo include:

- Clock distribution from MUTANT to AsAds
- Multiplicity from AsAd to MUTANT
- AsAd SCA read / write control
- Add time stamp to data
- Slow control pass-through to AsAds
- Fast control to setup sparse readout mode of the SCA

Particular attention must be paid to the clock distribution paths and as a poor design could cause synchronization errors between data packets from different CoBos.

The CoBo is proposed to be a single-wide NIM (or µTCA) module based around a Xilinx Virtex-5 FXT FPGA (XC5VFX70T). It will be controlled by VxWorks real-time operating system running on the embedded PowerPC 440 core in the FPGA. The operating system will handle the control and calibration tasks and networking as well as debugging modes.

2: Specifications

2.1 Standards

The CoBo module will be designed to NIM (or µTCA) standards. The hardware will be identical, regardless of where each CoBo is placed within the GET system. The CoBo is designed as a single-wide NIM (or µTCA) module. A maximum of 10 CoBos and 1 MUTANT will be placed in each crate.

2.2 Clock Tree

The CoBo receives a 100 MHz master clock from the MUTANT module. From this master clock, it must synthesize the SCA write and read clocks and the AsAd ADC sample clock. The clock frequency synthesis will be performed by a National Semiconductor LMK02002 clock jitter cleaner.

The LMK02002 has four output channels, one will output the SCA write clock, the second will output the SCA read clock, the third will output the ADC sample clock, and the fourth will provide the clock signal to the FPGA. The LMK02002 is specified as having <200 fs RMS jitter on its outputs. To keep the divided clocks synchronized on all CoBos, the MUTANT must send a synchronization pulse to the CoBo after any SCA write clock frequency change.

Since the clock signals must be sent to four AsAd boards, the three AsAd clocks are sent to LMK01020 clock distribution ICs, which have <30 fs additive jitter. These ICs will provide clean LVPECL clock signals for transmission to the AsAds. The LMK02002 and LMK01020 ICs all have programmable delays for each of their outputs, so some adjustment can be made between the phase of the SCA read clock and ADC sample clock (±48°) without extra logic.

2.3 Multiplicity

The multiplicity from AsAd is transmitted via the same high-speed ADC used for data transmission. Therefore, the multiplicity is 12 bits per ASIC and received every 40 ns (25 MHz sampling rate). The multiplicity of the 16 ASICs connected to the CoBo is summed then scaled to 11 bits for transmission to MUTANT. The CoBo will add a 12th bit for parity (based on MUTANT Specifications draft). Since the connection to MUTANT only contains four bits for multiplicity, the multiplicity will be split and transmitted over three clock cycles.

2.4 SCA Readout

In full read-out mode, when the MUTANT triggers, it will send the "STOP_WRITE_SCA" signal. When this signal is received by the CoBo, it sends the "STOP_WRITE_SCA" signal to AsAds, followed by the "START_READ_SCA" signal to begin data readout. The CoBo also sends a "DEAD_TIME" signal back to the MUTANT until readout is completed and the CoBo is ready to begin sending multiplicity information again.

The CoBo will also support a sparse readout of the switched capacitor array. In this mode, following the "STOP_WRITE_SCA" signal, the CoBo reads the hit channel register of each AGET using the fast control interface. If necessary, it can modify the hit register and write the new value back to the AGET. Then, it sends the "START_READ_SCA" signal. In this mode, the AGETs only send data from the hit channels, which should shorten readback time considerably. As in the full read-out mode, the CoBo sends a "DEAD_TIME" signal to the MUTANT while it is reading data from the SCA.

The CoBo firmware will be pipelined so that zero suppression and additional compression takes place simultaneously with SCA readout. As a result, the dead time during readout is limited by the maximum rate that the SCAs can be readout. The prototype firmware will implement zero suppression; additional compression algorithms will be added in future versions.

At the end of the compression pipeline, the data will be stored in a DDR2 SDRAM double buffer. Each half of the double buffer will contain 512 Mb. In the worst case of full readout with no data being suppressed, this is enough to buffer 16 events. (This assumes 32-bit data: 9 bits for time bucket, 2 bits for AsAd, 2 bits for ASIC, 7 bits for pad, 12-bit SCA value.) Assuming zero suppression of 10% on average, the buffer could hold more than 150 events.

An embedded processor running VxWorks performs data transmission the data acquisition system. The data will be sent via two gigabit Ethernet links. Since a double buffer is used, there is no particular timing requirement between reading of the current event and transmission of previous events to from the CoBo. When the embedded processor has completed sending all buffered events in its memory and the compression pipeline finishes its current event, the buffers are swapped and the process repeats.

2.5 Slow Control

The embedded PowerPC 440 core in the FPGA will run VxWorks real-time operating system. VxWorks will handle the slow control. Slow control commands will be sent to the CoBo via its Ethernet link. Slow control commands for each AsAd will be sent to the CoBo, which will forward the command to the AsAd.

2.6 AsAd Connection

The connection to each AsAd will use a Samtec QTE-DP connector and a 3-meter shielded twisted pair cable. Each cable contains 28 controlled impedance differential pairs. The signals are detailed in section 3.

2.7 MUTANT Connection

If the µTCA standard is chosen, the connection will be made via the µTCA backplane. If the NIM standard is chosen, the connection to MUTANT will use the stacked 2x15 points MDSM connectors also used on the MUTANT end of the connection (ITT-Cannon MDSM-30PE-Z10-VR22). This will connect via two shielded twisted pair cables. Each cable carries seven twisted pairs, and one single-ended signal. The first cable is used for clock reception and transmission of multiplicity. The other cable is used primarily for serial information reception. These connections also use differential pairs.

2.8 Ethernet

The CoBo will use two gigabit Ethernet links for data transmission. The FPGA contains hardware gigabit Ethernet drivers and will use a firmware TCP/IP stack to build data packets. In order to maximize the throughput of the data, the firmware will assemble data packets, and the VxWorks operating system will only need to control their transmission.

3: I/O Signals

The connections to other GET components will be detailed here as they are finalized.

3.1 AsAd Inputs

LVPECL:

CLK_W – SCA write clock
CLK_R – SCA read clock (25 MHz)
CLK_ADC – ADC sample clock (locked to CLK_R)

LVDS:

SCA_W – SCA write command
SCA_R – SCA read command
AGET_IN – fast control data from CoBo to AsAd
AGET_CLK – fast control clock (100 MHz)

CMOS 3.3V:

SPI_MOSI – slow control data from CoBo to AsAd
SPI_CLK – slow control clock (20 MHz)
PWR_DN – command to turn off AsAd power

3.2 AsAd Outputs

LVDS:

CLK_F – data frame clock
CLK_B – data bit clock
OA0 – AGET 0 data LSB
OA1 – AGET 0 data MSB
OB0 – AGET 1 data LSB
OB1 – AGET 1 data MSB
OC0 – AGET 2 data LSB
OC1 – AGET 2 data MSB
OD0 – AGET 3 data LSB
OD1 – AGET 3 data MSB
AGET_OUT – fast control data from AsAd to CoBo

CMOS 3.3V:

SPI_MISO – slow control data from AsAd to CoBo
ALARM<3:0> - bus used to signal fault conditions

3.3 MUTANT Inputs

LVDS:

MULT_DATA<3:0> – multiplicity data
MULT_FRAME – multiplicity frame clock
DEAD_TIME – CoBo dead time signal

3.4 MUTANT Outputs

LVDS:

MASTER_CLK – system master clock
STOP_WRITE_SCA – event trigger signal
SERIAL_CLOCK – serial clock
SERIAL_FRAME – serial frame
SERIAL_DATA – serial data

CMOS 3.3V:

CLK_SYNC – clock synchronization pulse

3.5 DAQ

The connection for the data acquisition system will be two gigabit Ethernet links. Data will be transmitted using standard TCP/IP protocols.

4: CoBo Registers

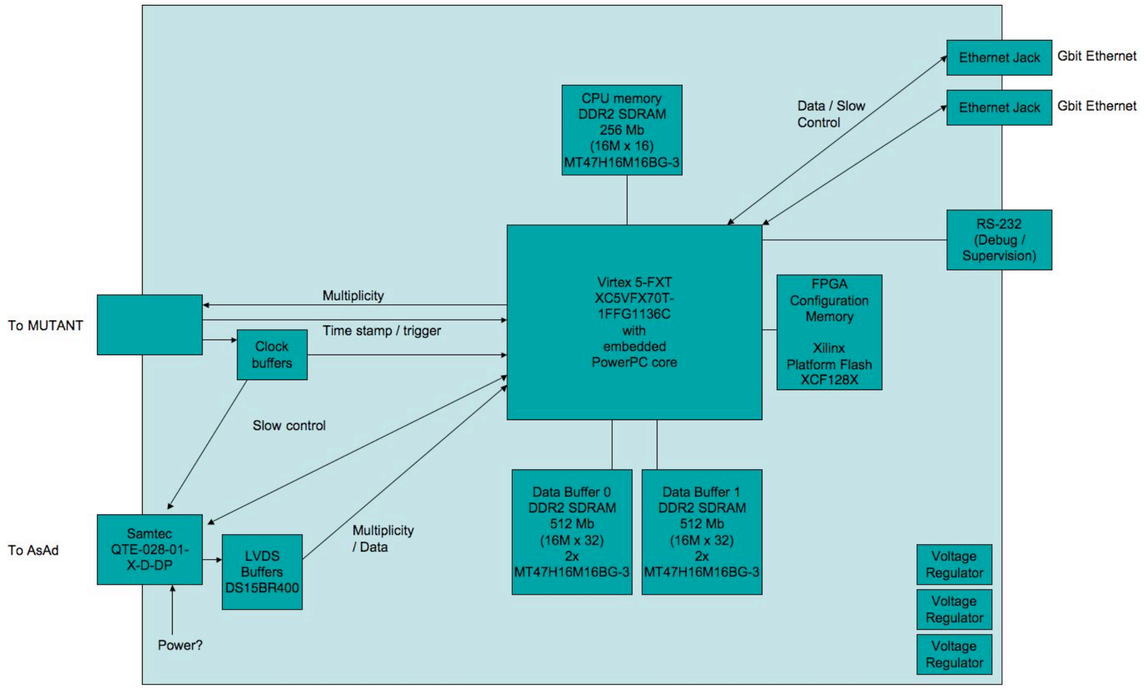
The connections to other GET components will be detailed here as they are finalized.

4.1 General Configuration Registers (Slow Control)

4.2 Inspection / Diagnostic Registers

Appendix 1: CoBo Diagram

CoBo Block Diagram


Ethernet Jack
Gbit Ethernet
Ethernet Jack
Gbit Ethernet
Data / Slow Control
CPU memory DDR2 SDRAM 256 Mb (16M x 16) MT47H16M16BG-3
RS-232 (Debug / Supervision)
Virtex 5-FXT XC5VFX70T-1FFG1136C with embedded PowerPC core
FPGA Configuration Memory
Xilinx Platform Flash XCF128X
To MUTANT
Multiplicity
Time stamp / trigger
Clock buffers
Slow control
To AsAd
Samtec QTE-028-01-X-D-DP
LVDS Buffers DS15BR400
Multiplicity / Data
Power?
Data Buffer 0 DDR2 SDRAM 512 Mb (16M x 32) 2x MT47H16M16BG-3
Data Buffer 1 DDR2 SDRAM 512 Mb (16M x 32) 2x MT47H16M16BG-3
Voltage Regulator
Voltage Regulator
Voltage Regulator