Proposal for a miniFEC

DRAFT 0

The purpose of the miniFEC (mFEC) is to read information from other sources than the TPC, e.g. the event number from the TLU (Trigger Logic Unit), via a distributor. The mFEC will have the same interface to the ALTRO bus as the normal FEC, i.e. the bus driver and Board Controller. At least one ALTRO chip will be replaced by an FPGA handling the readout of the non-TPC equipment, and is interfaced to the standard ALTRO bus driver on the mFEC. The PASA will not exist, instead there can be other types of front end receivers, either as input to an FPGA or to an ALTRO chip for readout. The FPGA must obey the ALTRO interface standard with timings and signals for the communication with the ALTRO bus driver on the FEC. Figure 1 show a schematic view of a miniFEC with a FPGA to get the event number and ALTROs.

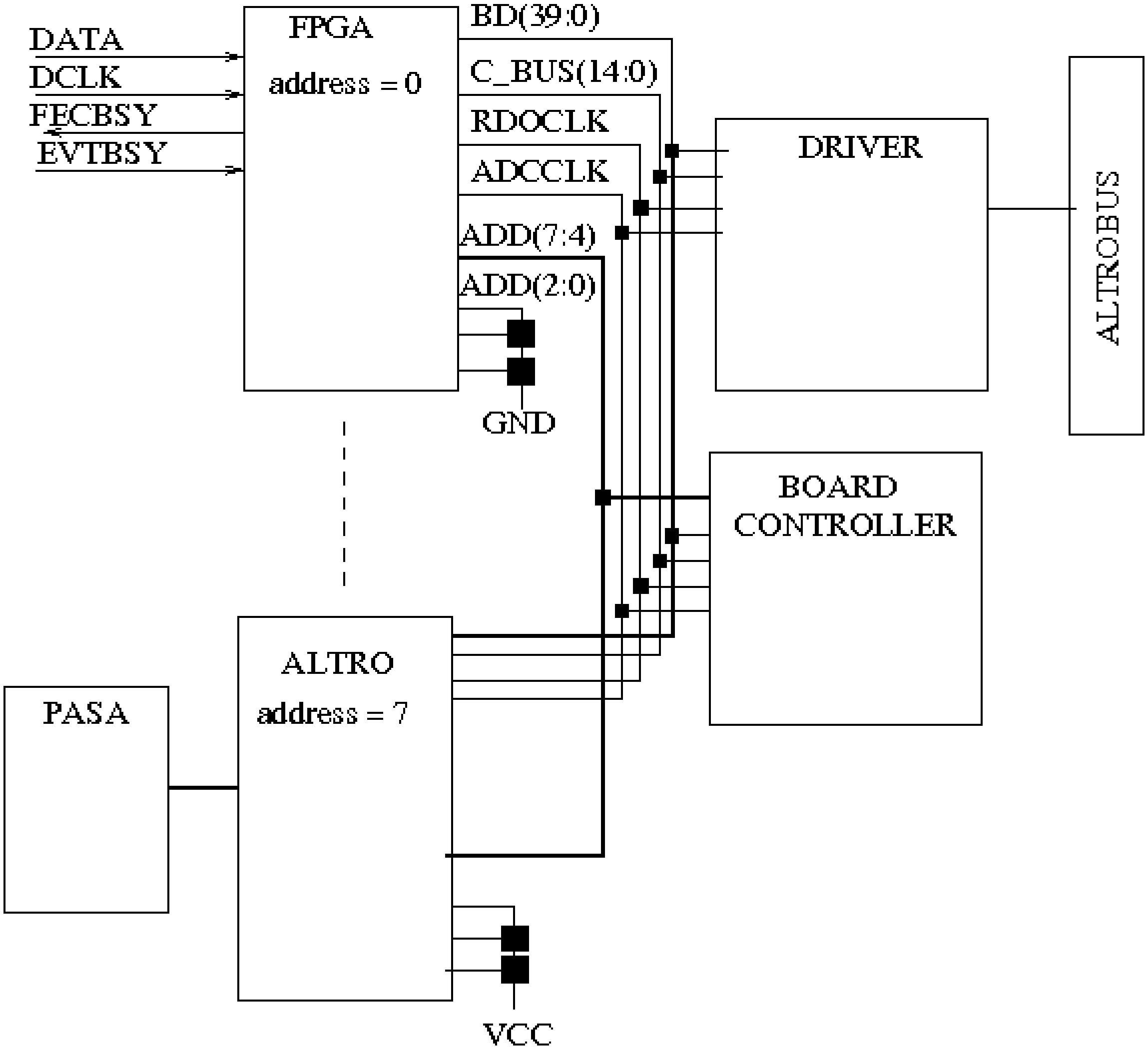

Figure 1: A schematic view of a mFEC card with a FPGA to read the event number

The FPGA should have the number of front end buffers, 4 or 8, as is the case in the ALTRO chip. Below are lists of signals/register that exist in the ALTRO chip and which should be used in the FPGA version. The FPGA corresponds to one ALTRO chip, with one to sixteen channels. The data format from the ALTRO consists of 10bits for each channel, a time stamp and length for each cluster, followed by hardware id, and is adjusted to 40 bit word boundary. The FPGA data should have the same structure.

ALTRO signals

Signals to/from ALTRO (= not needed on FPGA version)*

C_BUS (14:0) ControlBus

BD (39:0) DataBus

ADD (7:4) Card hardware address

ADD(2:0) ALTRO hardware address

RDOCLK ReadoutClock

ADCCLK SamplingClock

TEST_X TEST mode *

ADC_ADD (1:0) Channel Address in TEST mode *

PASAGND PASA input signals ground *

PASA (15:0) PASA inputs *

3V3_PASA: 3.3 V Power supply *

VRA voltages for the ALTROs *

VRT reference voltage for all the PASA *

VRM reference voltage for all the PASA *

VRB reference voltage for all the PASA *

The controlbus content (= not needed on FPGA version)*

TRSF Transfer, asserted to take control of the bus

DSTB Data strobe, asserted for each word transferred

ACKN acknowledge that instruction has been latched, or data put on the bus

ERROR FEC ERROR flag

RST_TBC Reset to BC from RCU *

L1 Level1 trigger

L2 Level2 trigger

WRITE access is read or write

CSTB Command strobe (released when ACKN seen)

RCLK Redaout clock (upstreams of fanout)

free

DOLO_EN drives the BD bus when reading a register ?

TRSF_EN drives the BD bus when transferring an event ?

ACKN_EN frames ACKN ?

RST_FBC reset from BC

The address lines are hardwired on the FEC, and must be compared with the address as received on the data bus. The control bus contains all the control signals for the communication.

ALTRO registers

per channel (= not needed on FPGA version):*

K1- K3 Filter coefficients *

L1-L3 Filter coefficients *

VFPED Variable/fixed pedestal data *

PMDTA Pedestal memory dara for a given address *

ADEVL Chip address + event length

Global registers (= not needed on FPGA version):*

ZSTHR Offset + Threshold for zero skipping *

BCTHR Threshold HI + Threshold LO *

TRCFG Trigger delay + number of samples per event *

DPCFG configuration post/presamples *

BFNPT pretrigger samples, nb of buffers (4/8), digital filter, power save * (need nunber of buffers?)

PMADD pedestal memory address *

ERSTR Error and status register

TRCNT Trigger counter

FPGA specific signals

DATA event number, and possible other data from distributor

EVTBSY event busy from distributor

DCLK clock from distributor when next bit of DATA is available

FECBSY asserted by mFEC when EVTBSY is seen and deasserted when data readout has finished.

FPGA specific registers

EVTNB last read event number from distributor

FECBSY number of clock cycles to wait before deasserting FECBSY to the distributor (0=disabled).

A possible Event number readout

The event number is pushed from the distributor to the mFEC as:

1. The distributor reads the event number from the TLU and asserts the BUSY to the TLU
2. The distributor assert EVTBSY to the mFEC
3. All active mFEC asserts FECBSY
4. The distributor clocks the event number to the mFEC
5. The distributor wait until all active mFEC has deasserted FECBSY (the length of deasserting FECBSY in the mFEC is taken from the FECBSY register)
6. The distributor deasserts EVTBSY
7. The distributor deasserts the BUSY to the TLU

Note: in the case when the DAQ PC is also reading the event number, then the distributor waits for its RSTBSY as well before deasserting the BUSY to the TLU.