FEB2 Slow Control Requirements and Constraints

February 24, 2025

The FEB2 slow control, including board initialization and monitoring, is implemented through a series of IC and EC transactions that write or read lpGBT control and data lpGBT registers. The IC and EC channels are embedded in the downlink data stream to the two onboard control lpGBTs. Control lpGBT registers are written or read directly through IC transactions as defined in table 3.3 of the lpGBTv1 manual. EC transactions are passed on from the control lpGBTs to data lpGBTs using an onboard EC bus, and can write or read data lpGBT registers. This document summarizes the slow control implementation for the most recent v2.5 prototype version of the FEB2 board.

In terms of board initialization, there are several onboard components that need configuration including the COLUTAv4 ADCs, ALFE2 PA/S, VTRx+ and the lpGBTs. Hadronic Endcap (HEC) versions of the boards will have HPS2 chips in place of ALFE2s. The lpGBTs are configured via their slow control register interface. The remaining board components are configured via the lpGBT I2C buses, which are controlled through lpGBT slow control registers. Additionally there are two slow control modes that can be used, the default mode which relies on both control lpGBT optical links working and a redundant mode where only one optical link is available.

Each type of onboard component has a specific configuration scheme. However the implementation of the lpGBT I2C interface means that a similar set of IC and EC transactions are required in all cases. These transactions would ideally be implemented in firmware controlling the slow control interface to the FEBs.

Required IC and EC transactions for slow control

The following IC and EC transactions are most relevant for configuring the various chips integrated on the FEB2:
-IC or EC write transactions capable of writing from 1 to 4 sequential lpGBT registers
-IC or EC read transaction capable of reading either 1 or 8 sequential lpGBT registers
lpGBTs configuration would be faster with IC or EC transactions capable of writing and reading back 256 sequential lpGBT registers at a time, as this would allow an lpGBT to be configured with a single IC transaction. However this additional capability would not speed up the configuration of COLUTAv4s, ALFE2s or VTRx+s. The number of IC or EC write and read transactions required to configure each chip type, in either the default or redundant modes, as summarised in the tables below.

In principle the FEB2 slow control can be entirely implemented through IC or EC transactions that write or read only a single lpGBT register. However this generally more than doubles the total number of required write transactions, and consequently would increase the time required to initialise the FEB2.

	# of IC/EC Write Transactions to Configure Chip				
Chip, Configuration Mode	**1 register**	**2 register**	**3 register**	**4 register**	**256 register**
COLUTAv4, default (EC)	1264	292	0	680	0
COLUTAv4, redundant (IC)	12742	778	1264	3404	0
ALFE2, default (EC)	464	0	0	96	0
ALFE2, redundant (IC)	3040	112	464	688	0
VTRx+, default (EC)	126	0	9	0	0
VTRx+, redundant (IC)	918	36	126	216	0
Control lpGBT, default (IC)	0	0	0	0	1 (optimal)
Control lpGBT, redundant (IC)	1442	103	103	309	0
Data lpGBT, default (EC)	0	0	0	0	1 (optimal)
Data lpGBT, redundant (IC)	1442	103	103	309	0

	# of IC/EC Read Transactions to Configure Chip		
Chip, Configuration Mode	**1 register**	**8 register**	**256 Register**
COLUTAv4, default (EC)	584	98	0
COLUTAv4, redundant (IC)	4857	98	0
ALFE2, default (EC)	112	0	0
ALFE2, redundant (IC)	1182	0	0
VTRx+, default (EC)	63	0	0
VTRx+, redundant (IC)	368	0	0
Control lpGBT, default (IC)	0	0	1 (optimal)
Control lpGBT, redundant (IC)	1060	0	0
Data lpGBT, default (EC)	0	0	1 (optimal)
Data lpGBT, redundant (IC)	625	0	0

EC and I2C Bus Implementation on v2.5 Boards

The table below summarizes the implementation of the EC and I2C buses on the FEB2 board. EC channel 0 refers to the EC frames delivered to the board via the lpGBT12 downlink, and EC channel 1 frames are delivered via the lpGBT13 downlink. In the event that a control lpGBT is operating in redundant mode the corresponding EC bus will not be available and slow control for data lpGBTs on that bus must use the I2C interface. The naming of lpGBT I2C buses follows the convention in the lpGBT manual, where the three available I2C buses are identified as M0,M1 and M2.

Data lpGBT Name	EC Channel	I2C Controller	I2C Bus
lpGBT1	0	lpGBT13	M0
lpGBT2	0	lpGBT13	M0
lpGBT3	0	lpGBT13	M0
lpGBT4	0	lpGBT13	M0
lpGBT5	0	lpGBT13	M0
lpGBT6	0	lpGBT13	M1
lpGBT7	0	lpGBT13	M1
lpGBT8	0	lpGBT13	M1
lpGBT9	0	lpGBT13	M1
lpGBT10	0	lpGBT13	M1
lpGBT11	0	lpGBT13	M1
lpGBT14	1	lpGBT12	M1
lpGBT15	1	lpGBT12	M1
lpGBT16	1	lpGBT12	M1
lpGBT17	1	lpGBT12	M1
lpGBT18	1	lpGBT12	M1
lpGBT19	1	lpGBT12	M1
lpGBT20	1	lpGBT12	M0
lpGBT21	1	lpGBT12	M0
lpGBT22	1	lpGBT12	M0

lpGBT23	1	lpGBT12	M0
lpGBT24	1	lpGBT12	M0

Chips Configured via I2C

COLUTAv4s, ALFE2s and VTRx+s are configured through data lpGBT I2C interfaces. Additionally data lpGBTs can be configured via a corresponding control lpGBTs I2C interface as a backup to the EC interface. Control lpGBTs operating in redundant mode will also be configure via the other control lpGBT’s I2C interface. All chips on the FEB2 uses 7-bit I2C addresses, except for the COLUTAv4s which used 10-bit. Additional details on the I2C configuration process for different chip types are described below.

I2C write and read transaction are initiated by a series of lpGBT slow control register writes and reads. This series of lpGBT register writes and reads can be built by combining single or multiple register commands, as has been done in the various development test stands. However some additional reduction in configuration time might be obtained by implementing in firmware generic I2C read and write processes that will co-ordinate the set of IC or EC transactions required to initiate an I2C transaction.

COLUTAv4 I2C Configuration

The COLUTA configuration data is written through a data lpGBT I2C bus using 10-bit addressing write commands each containing 8 data bytes. COLUTA configuration bits are divided between global and channel-specific bits as described in the COLUTAv4 datasheet. For channel-specific configuration bits the I2C write command data bytes include two bytes containing subaddressing information and 6 bytes of configuration data. For global configuration bits each I2C write command includes 8 bytes of configuration data. The lpGBT “I2C_WRITE_MULTI_EXT” command is required to send these multiple bytes of configuration data using 10-bit addressing instead of the “I2C_WRITE_MULTI” command used for other ASICs.

ALFE2 I2C Configuration

ALFE2 configuration data is written through a data lpGBT I2C bus using 10-bit addressing write commands. The ALFE2 has 16 8-bit registers used to store configuration data and which are defined in the ALFE2 datasheet. The registers are configured using I2C write commands in a three step process, where the 2 LSBs of the ALFE2 7-bit I2C address act as a control register and are modified for each step:

1. Set control register to 0 and send register number via I2C single-byte write command
2. Set control register to 1 and send 0 via I2C single-byte write command
3. Set control register to 2 and register value via I2C single-byte write command

lpGBTv1 I2C Configuration

If the EC interface is not available then data lpGBTs are configured via I2C by a control lpGBT following the procedure described in section 3.5.1 of the lpGBT datasheet. The control lpGBT I2C write command process is defined in section 12.3 of the lpGBT datasheet.

Board Monitoring

FEB2 voltage and temperature monitor measurements are performed using the lpGBT v1 built-in ADCs. The operation and read out of these lpGBT ADC measurements is controlled through lpGBT slow control registers writes and reads as described in chapter 13 of the lpGBTv1 manual. These monitoring measurements generally require write and reading only 1 or 2 lpGBT registers at a time, and so do not introduce any additional IC and EC interface requirements beyond the initialization and configuration of FEB2 chips via I2C. However implementing the control of the ADC measurement process in firmware might increase the measurement rate and allow more frequent monitoring of the various FEB2 board voltages and temperature.