Lpgbt communication explanation

Context

The lpGBT (Low Power Giga Bit Transceiver) is an ASIC for communication that uses high speed bidirectional optical links. Further information can be seen in [1].

This documents tries to be a guide for the firmware implelmentation that must be done inside EMP, but it can also be helpful to the E-link communication in the Charge Monitoring board side.

It is necessary to understand how lpgbt works, and how the IP provided by EMP/EMCI team interacts with the lpgbt communication standard, so the own firmware can correctly fit.

Communication with the lpgbt

In order to achieve high speed communication, the lpgbt ASIC expect an uncommon dataframe[2], which is different for up-link communication (goes from the front-end up to the EMP) versus the down-link communication (goes from the EMP down to the front-ends). In the down-link and up-link dataframes, there are groups of bits that maps to different channels of communication, which goes to different front-ends.

In general, the complete dataframe is built by the IP core given by EMP/EMCI group. However, to correctly use that IP, the user must give and receive the data that is imbeded inside that dataframe. From the EMP/EMCI firmware, the user will give or receive just the portion of the dataframe that is data, so the firmware that the user must build is the one that generates of decodes the data in the way expected by the dataframe.

1 https://lpgbt.web.cern.ch/lpgbt/v1/introduction.html
2 In this document, “dataframe” means the full group of bits that are the input or output in the lpGBT, when it is communicating itself with the EMP.

It mus be noticed that, to support the harsh radiation environment in the LHC, the dataframe is not transmitted uchanged, but instead, 24 bits inside the dataframe are for bit error correction. The original dataframe is rearranged according to an error correction algorithm that scaters the 24 error correction bits in the transmited frame. Because of that, if you try to read the transmitted dataframe, you won't see any meaning information, as the original dataframe has been "obfuscated"[3].

Downlink dataframe

The downlink communication has 4 E-links, and each E-link is able to handle up to 4 front-ends, so the downlink communication is able to communicate with up to 16 front-ends. The ammount of front-ends that are communicating with the lpgbt depends on the Mb/s set for the communication, the number of outputs of e-links for bandwidth is:[4]

Output eLinks (down-link)			
Bandwidth [Mb/s]	80	160	320
Maximum number	16	8	4

Figure 1: Number of output eLinks versus bandwidth

The logic behind this table is that always the ammount of bits transmitted is the same, but when more front-ends are connected, that ammount of bits must communicate more front-ends. In one extreme, with the lowest bandwidth, you will be communicating yourself with all front-ends, in the oder extreme, only one front-end per E-link will be communicated.

If the bandwidth is 80 Mb/s, the EMP will be communicating itself with all 16 front-ends. However, if bandwidth is 320, how do we know what front-ends will be communicated? The answer is the web-page[5], where you can select the bandwidth at which the lpgbt is set, and the web page will show what are the E-links activated. The activated E-links are hardcoded in the lpgbt, so it is not possible to set what E-links are activated at each bandwidth.

With this information in mind, now it is easier to understand the downlink dataframe, which has the following shape[6]:

3 To read about obfuscation, see: https://en.wikipedia.org/wiki/Obfuscation
4 https://lpgbt.web.cern.ch/lpgbt/v1/introduction.html#electrical-links-elinks
5 https://pigbt.web.cern.ch/#/Core
6 https://lpgbt.web.cern.ch/lpgbt/v1/highSpeedLinks.html#frame-format

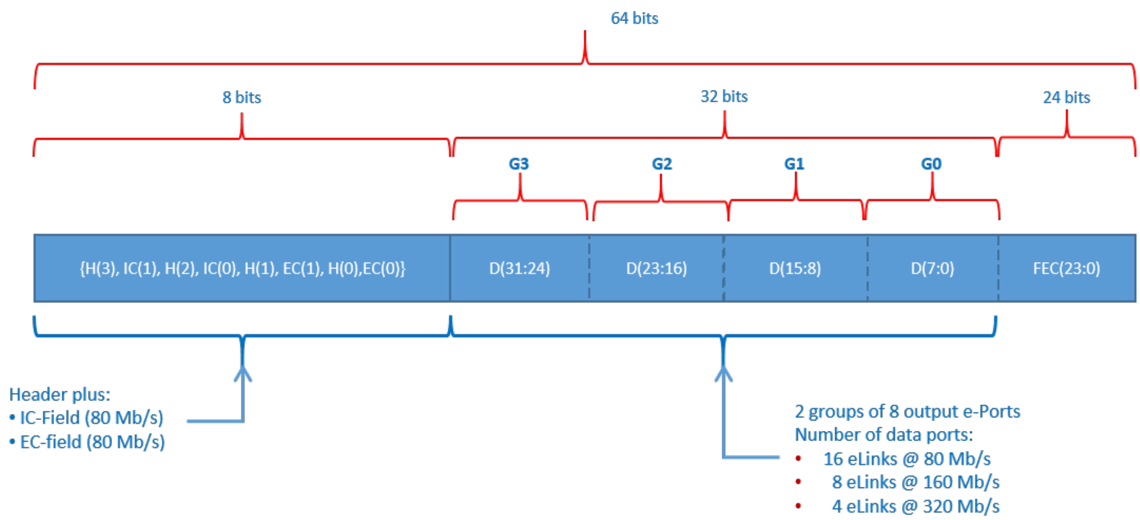


Figure 2: Downlink frame structure before interleaving.

In the EMP/EMCI firmware context, that complete dataframe is produced by the EMP/EMCI IP core. But what user must give as an input are the bits labeled with G3, G2, G1 and G0; in the following we will call each of those divisions a “Group”. Each Group will be the communication with one specific E-link, the ammount of bits that will end in each front-end is ordered from left to right in ascending order. An example of the bits used to communicate with each front-end in the Group X is shown in the following table:

Bandwidth	Bits assigned to each front-end
320 Mb/s	CH0; CH0; CH0; CH0; CH0; CH0; CH0; CH0
160 Mb/s	CH0; CH0; CH0; CH0; CH2; CH2; CH2; CH2
80 Mb/s	CH0; CH0; CH1; CH1; CH2; CH2; CH3; CH3

In the previos table, the color and “CHX” represents what bits in the Gropu X used to communicate with each of the 4 front-ends associated with that E-link. You can find firther explanation about this by consulting Fig. 4.2 in [7].

In summary, what the user must provide to the EMP/EMCI IP core called “emp_lpgbt_*” is each of the Groups’ bits, concatenated as shown in Figura 2.

7 https://lpgbt.web.cern.ch/lpgbt/v1/highSpeedLinks.html#downlink-frame

Uplink dataframe

The upper link logic is similar to the one used in the downlink. However, the uplink handles 7 E-links, each of whom can be connected to 4 front-ends; a total ammount of 28 possible front-ends sending information to the EMP. The summary of the ammount of front-ends connected, depending on lpgbt settings, is whons here[8]:

Input eLinks (up-link)												
Up-link bandwidth [Gb/s]	5.12						10.24					
FEC coding	FEC5			FEC12			FEC5			FEC12		
Bandwidth [Mb/s]	160	320	640	160	320	640	320	640	1280	320	640	1280
Maximum number	28	14	7	24	12	6	28	14	7	24	12	6

Figure 3: Number of input eLinks versus bandwidth

To know which front-end is connected depending on the settings, you can use the web page: [9] considering that the Uplink is set to work at rate 10.24 Gbps and with FEC5[10], and that CMB will operate at the highest possible speed (i.e. 7 communication channels). To know what bits correspond to which front-end connected, you must use the tables shown in images from Fig. 4.8 to Table in Fig. 4.11 from the link:[11].

How to use the EMP/EMCI IP Core to interact with the lpGBT

The main IP core in the Emp firmware is the one called “emp_lpgbt_X” (where “X” is a number that goes from 0 to 11). This IP core implements a portion of the E-link protocol, but we still must generate the remaining part.

At this moment (2024/07/22), the only documentation about the IP core “emp_lpgbt_X” is the “README.md” document inside the Gitlab containing the emp-firmware; its link is: https://gitlab.cern.ch/emci-emp/emp-firmware . That documentation explains some of the IP Core’s pins, describes pins that are nonexistent, and lacks the description for existing pins.

8 https://lpgbt.web.cern.ch/lpgbt/v1/introduction.html#electrical-links-elinks
9 https://pigbt.web.cern.ch/#/Core
10 https://indico.cern.ch/event/1344518/contributions/5694943/subcontributions/453757/attachments/2762498/4811117/FDR_Firmware.pdf
11 https://lpgbt.web.cern.ch/lpgbt/v1/highSpeedLinks.html#frame-formats

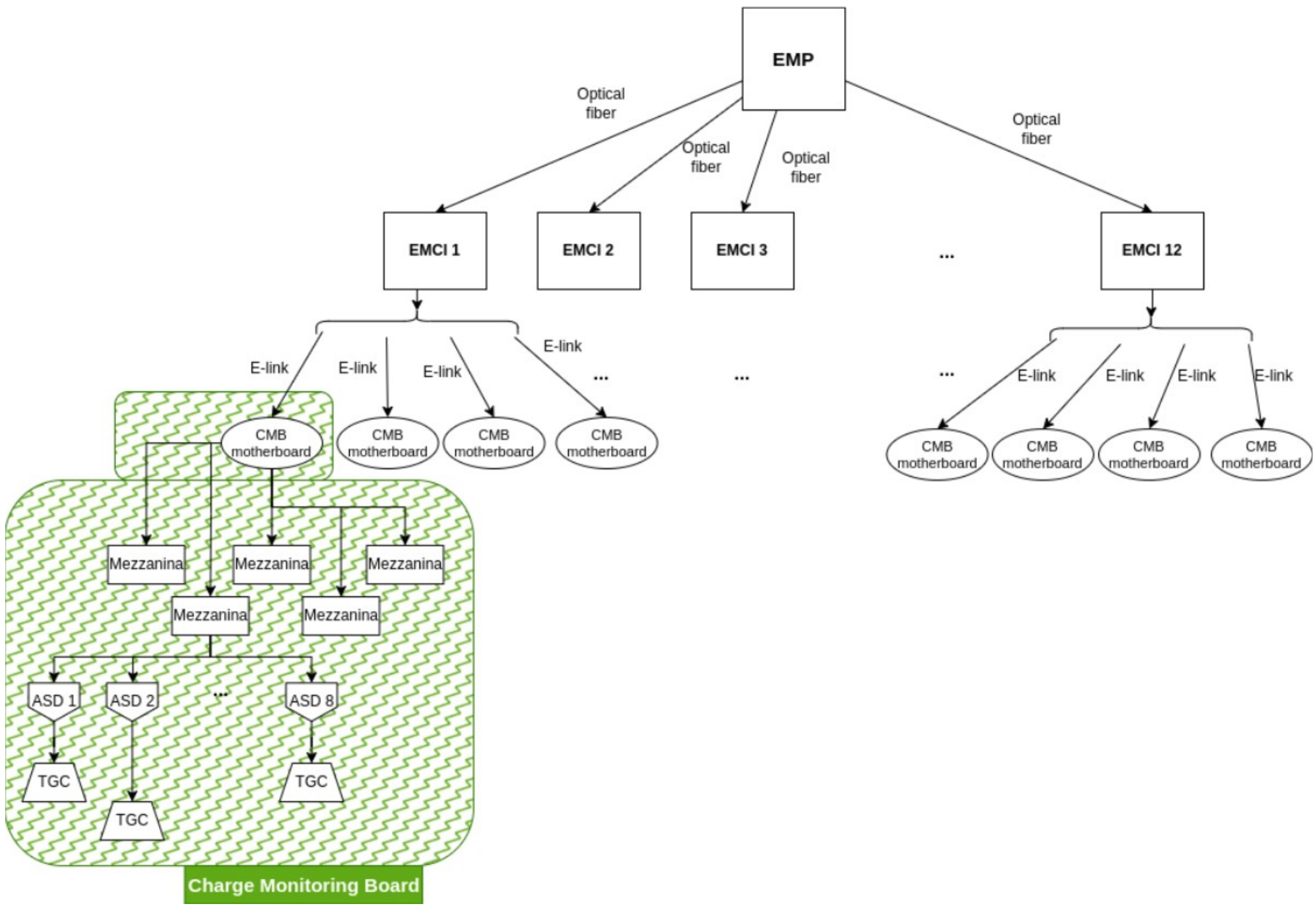
Downlink communication

According to my experiments with the IP core “emp_lpgbt_X”, the signals paths at the firmware diagram, when you send information from the EMP down to the front-end is this:

1. Message must have 32 bits and be written at pin “lpgbtfpga_downlinkUserData_i” from emp_lpgbt IP core.
    1. This message must be synchronous with the 40 Mhz clock that inputs at pin “logicCLK40_i” in emp_lpgbt IP core.
    2. This 32-bits word is the one that must contain the Groups from 3 to 0 in the Figure 2 of this document.
2. The already coded message (already obfuscaded/interleaved) comes out from the IP core “emp_lpgbt” through pin “MGT_USRWORD_i”
3. That coded message goes to module “transceivers_TE0807$_0$” through its pin “CH*_MGT_USRWORD_o” (* means a number corresponding to the emp_lpgbt_X number).
4. A diferential signal comes out of the EMP, to the EMCI.

Diagram of the communication between CMB and EMP

The diagram for connections can be found in the next page


EMP
Optical fiber
Optical fiber
Optical fiber
Optical fiber
EMCI 1
EMCI 2
EMCI 3
EMCI 12
E-link
CMB motherboard
Mezzanina
ASD 1
ASD 2
ASD 8
TGC
Charge Monitoring Board