

CCCP Libera interface

Specification

Revision:	1.1
Status:	Released
Repository:	acc/projects/GSI/FAIR-BPM
Project:	ACC-GSI-SD-Libera
Folder:	doc
Document ID:	CSL-SPE-08-42862
File:	SPE-Libera_Interface.doc
Owner:	Igor Verstovšek
Last modification:	September 26, 2008
Created:	September 11, 2008

Document History

Revision	Date	Changed/ reviewed	Section(s)	Modification
1.0	2008-09-11	gjansa	All	Created.
1.1	2008-09-24	gjansa	1.1	Updated.

Confidentiality

This document is classified as a **confidential document**. As such, it or parts thereof must not be made accessible to anyone not listed in the Audience section, neither in electronic nor in any other form.

Scope

This document specifies Libera CCCP interface.

Audience

The audience of this document are:

- All members of the GSI-SD group.
- All members of the GSI-BEL group.
- All members of development team at Cosylab.

The document can be disclosed to all GSI staff and staff of all entities participating in the FAIR project.

Table of Contents

1. Libera Interface	4
1.1. Packet description	4
1.1.1. BPM Packet	4
1.1.2. AUX Packet.....	4
1.1.3. Raw Data Packet	5
1.2. Frame Description	5
1.2.1. BPM Frame.....	5
1.2.2. Control Frame	6
1.2.3. AUX Frame	6



1.2.4. Raw Data Frame.....	7
1.2.5. Packet ID.....	7
1.3. Libera Control Commands.....	8

Figures

Figure 1: BPM packet structure.....	4
Figure 2: AUX packet structure.....	5
Figure 3: Rae data packet structure.....	5
Figure 4: BPM Ethernet frame structure.....	6
Figure 5: Control Ethernet frame structure.....	6
Figure 6: AUX Ethernet frame structure.....	7
Figure 7: Raw data Ethernet frame structure.....	7

Tables

Table 1: FA commands and parameters.....	8
--	---

1. LIBERA INTERFACE

Communication between CCCP and each of the 13 LiberAs used in the system is done using two communication channels:

- 100Mbit network used for control
- 10Gbit network used for data retrieval

1.1. PACKET DESCRIPTION

1.1.1. BPM Packet

BPM packet is send by Libera during bunch to bunch and position offset calibration Libera modes. BPM packet is 96 bits long and consists of:

- Position X – 22 bit
- Position Y – 22 bit
- Overload status – 4 bit
- Underload status – 4 bit
- t1 – 12 bits
- t2 – 12 bits
- trf – 16 bits

Position X	Overload status	Position Y	Underload status	t1	t2	trf	Dummy
22 bit	4 bit	22 bit	4 bit	12 bit	12 bit	16 bit	4 bit

Figure 1: BPM packet structure

1.1.2. AUX Packet

AUX packed is send by AUX Libera during bunch to bunch data and raw data Libera mode. AUX packet is 80 bits long and consists of:

- Signal 1 – 16 bit
- Signal 2 – 16 bit
- Signal 3 – 16 bit



- Signal 4 – 16 bit
- trf – 16 bits

Signal 1	Signal 2	Signal 3	Signal 4	trf
16 bit	16 bit	16 bit	16 bit	16 bit

Figure 2: AUX packet structure

1.1.3. Raw Data Packet

Raw data packet is send by BPM Libera during raw data Libera mode. Raw data packet is 64 bits long and consists of:

- 4 * signal - 4 * 16 bits

Signal 1	Signal 2	Signal 3	Signal 4
16 bit	16 bit	16 bit	16 bit

Figure 3: Rae data packet structure

1.2. FRAME DESCRIPTION

Each of the Ethernet frame described in this section is sent via SFP 1Gbit port.

Each of the 12 BPM Libera will send the following Ethernet frames

- BPM frame. Normal frame send during acquisition in bunch to bunch data mode
- Raw data frame. Frame send during acquisition in raw data mode.
- Control frame. Frame to signal exception events (e.g. stop frame).

Libera 13 will send the following frames

- Signal frame. Normal frame send during acquisition.
- Control frame. Frame to signal exception events (e.g. stop frame).

1.2.1. BPM Frame

BPM frame is 9018 byte long Ethernet frame composed of the following

- Ethernet/IP/UDP header

- Payload/data which consist of the following:
 - Packet ID – 6 byte long. For details see 1.2.5
 - BPM packets – each 12 byte long
 - Dummy bytes – smallest size is 2 bytes but can be higher if stop trigger occurs before Libera fills up BPM frame.

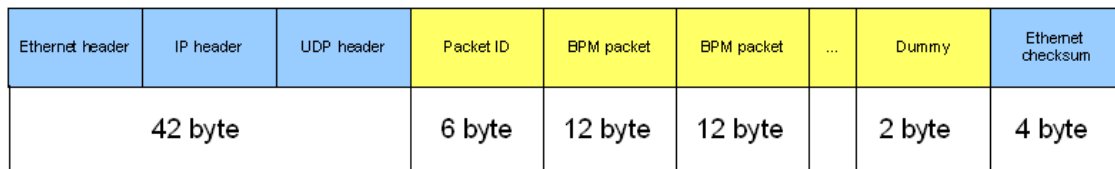


Figure 4: BPM Ethernet frame structure

1.2.2. Control Frame

Control frame is 64 bytes long Ethernet frame composed of the following

- Ethernet/IP/UDP header
- Payload/ data which consist of the following:
 - Packet ID – 6 bytes long. For details see 1.2.5
 - Control command - 2 bytes long. Control command can be one of the following:
 - Stop command - signaling CCCP end of data stream
 - Dummy bytes – 10 bytes long (this is used so that frame is 64 bytes long, which is standard minimum Ethernet frame length)

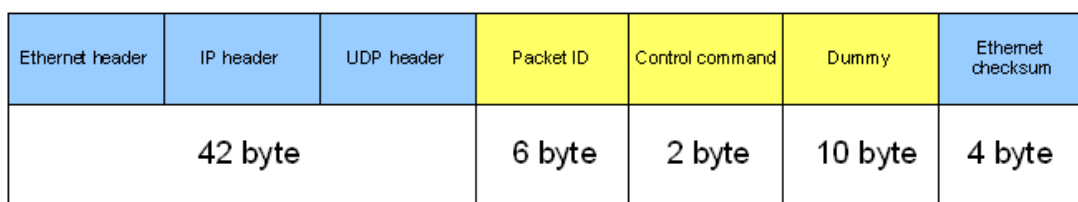


Figure 5: Control Ethernet frame structure

1.2.3. AUX Frame

AUX frame is 9018 byte long Ethernet frame composed of the following

- Ethernet/IP/UDP header
- Payload/data which consist of the following:
 - Packet ID – 6 byte long. For details see 1.2.5



- AUX packets – each 8 bytes long
- Dummy – smallest size is 6 bytes but can be higher if stop trigger occurred before Libera fills up raw data frame.

Ethernet header	IP header	UDP header	Packet ID	Signal packet	Signal packet	...	Dummy	Ethernet checksum
46 byte			6 byte	10 byte	10 byte		6 byte	4 byte

Figure 6: AUX Ethernet frame structure

1.2.4. Raw Data Frame

Signal frame is 9018 byte long Ethernet frame composed of the following

- Ethernet/IP/UDP header
- Payload/data which consist of the following:
 - Packet ID – 6 byte long. For details see 1.2.5
 - Raw data packets – each 8 bytes long
 - Dummy – smallest size is 6 bytes but can be higher if stop trigger occurred before Libera fills up raw data frame.

Ethernet header	IP header	UDP header	Packet ID	Raw data packet	Raw data packet	...	Dummy	Ethernet checksum
42 byte			6 byte	8 byte	8 byte		6 byte	4 byte

Figure 7: Raw data Ethernet frame structure

1.2.5. Packet ID

Each Ethernet frame contains packet ID used to detect missing (lost) UDP packets. Moreover, packet ID is generated of 2 parts:

- Packet counter (4 byte) which is reset to zero on start trigger and it increments by each packet sent from Libera.
- Cycle counter (2 byte) which is incremented by start trigger.

Both are unsigned numbers that wrap around (set to 0) when maximum value is reached. This allows very precise detection of lost UDP packets within each cycle.

1.3. LIBERA CONTROL COMMANDS

To control each of the Libera a fast application (FA) will be used. This is done using Libera CSPI library using generic server. The following configuration/commands will be performed on each of the Libera.

- Mode change. User can select one of the following modes on Libera:
 - Bunch to bunch mode
 - Zero line calibration mode
 - Position offset mode
 - Raw data mode
- Arm. Used to arm Libera (this will put Libera in a state where it will wait for start/stop triggers).
- Unarm. Used to unarm Libera.
- Set Libera MAC and IP address.
- Set CCCP MAC and IP address.
- Zero line position bin size setup. Used in zero line calibration mode to setup the number of acquisitions Libera acquires during calibration procedure.
- Get Zero line position offset. Used in zero line calibration mode to retrieve 4 offset values (one for each analog input) to be saved on CCCP and displayed in GUI.
- Set Zero line position offset. Used in zero line calibration mode to restore 4 offset values.

Commands:

ID	Description	FA address	Parameters
1	Mode change	0x14028000	
2	Arm/Unarm	0x14028004	
3	Set Libera MAC		
4	Set Libera IP		
5	Set CCCP MAC		
6	Set CCCP IP		
7	Zero line position bin size set		
8	Zero line position offset get		
9	Zero line position offset set		

Table 1: FA commands and parameters

