[Institute]

[Institute address]

Internal Note/
ALICE reference number
ALICE-INT-2004-018 version 1.0
Institute reference number
[-]
Date of last change
2004-07-06

C.E.R.N., PH Department, AID Group

RMKI, RFFO Division, Detector Building Group

ALICE DETECTOR DATA LINK

ALICE-DDL

Interface Control Document

Appendix of the User Requirement Document

Issue: Final
Revision: 11

Reference: ALICE Detector Data Link IC
Created: 12/05/96
Last modified: 6/7/04

Prepared by: György Rubin, Pierre Vande Vyvre and Csaba Soós

1. Document Information

1.1 Abstract

This document is an appendix to the ***User Requirement Document*** (ALICE/96-42, Internal Note/DAQ, 12 December 1996). It describes the external interfaces for the ALICE-DDL.

1.2 Document Status Sheet

1. Document Title: ALICE-DDL Interface Control Document			
2. Document Reference Number: ALICE Detector Data Link IC			
3. Issue	**4. Revision**	**5. Data**	**6. Reason for change**
Draft	1	5 December, 1996	Original document
Draft	2	15 December, 1996	Changes are based on the comments of the ALD group review meeting
Draft	3	29 May, 1997	Priority rules have been defined for the protocol transactions
Draft	4	3 June, 1997	TXLOOP IFCMD has been introduced
Draft	5	6 June, 1997	New definitions for commands & status words
Draft	6	2 September, 1997	roCLKx2 signal is added to the RORC - DIU connector
Draft	7	8 September, 1997	Definition of the DDL configurations is added
Draft	8	10 September, 1997	New definition for status words
Draft	9	27 January, 1998	Redefinition of the interface status words
Final	10	11 March, 2004	Adding information about the new DDL and the usage of the DDL for the transfer of data to HLT
Final	11	6 July, 2004	Introducing the DTSTW continuation bit

Table 1 Document Status Sheet

1.3 Table Of Contents

1.4 List of Figures

1.5 List of Tables

2. Introduction

2.1 Purpose of the Document

This document describes the interfaces of the ALICE-DDL with the front-end electronics and the read-out receiver card.

It is an appendix of the ALICE-DDL User Requirement Document [1].

2.2 Scope

This Interface Control Document (ICD) contains only the DDL interface specification which defines the interface between the FEE and the SIU and the interface between the RORC[1] and the DIU (see Figure 1).

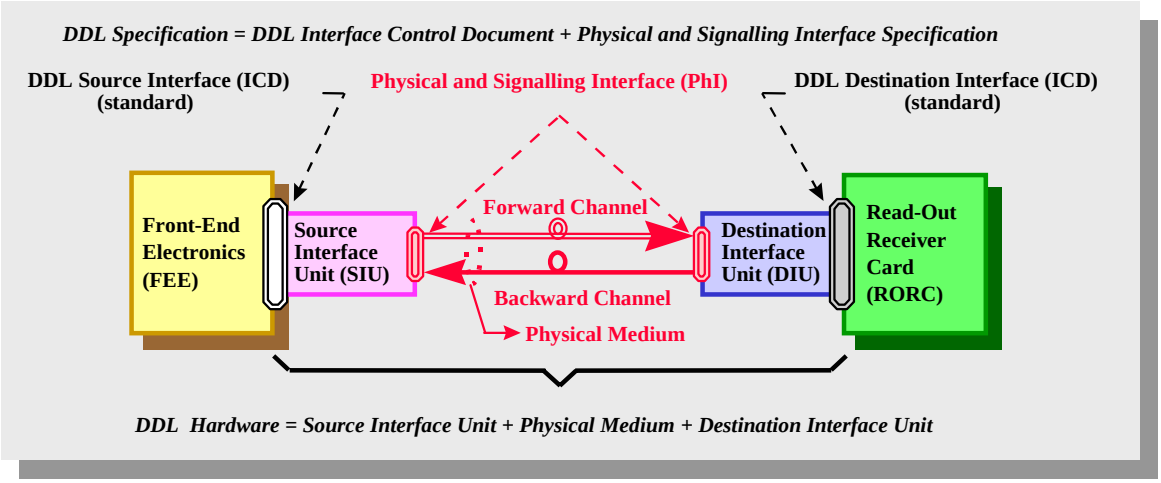


Figure 1 The DDL specifications and hardware components

Both the RORC-DIU and the FEE-SIU interfaces have identical architecture independently of the sub-detector they are connected to. Thus these interfaces are completely standardised. This document describes these interfaces.

The DDL is composed of three hardware components (see Figure 1):

- the SIU which is connected to the FEE,
- the DIU which is connected to the RORC,
- the physical medium which is a duplex optical fibre, connecting the SIU and the DIU over a maximum distance of 200 m.

The DDL is able to transmit information in both directions (see Figure 2):

- data blocks from FEE to RORC and from RORC to FEE
- data blocks from FEE to HLT RORC;
- commands from the RORC to the DIU, the SIU and the FEE;
- status information from the FEE, the SIU and the DIU to the RORC.

[1] **RORC means either D-RORC or pRORC**

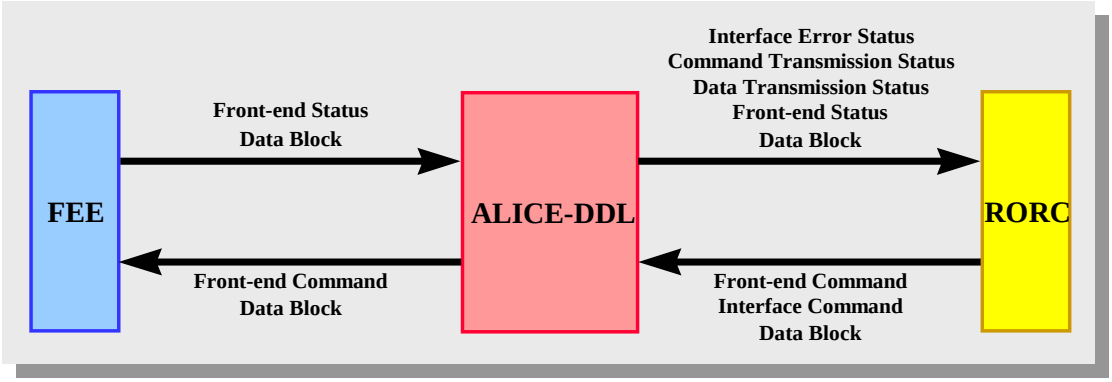


Figure 2 The information transfer of the DDL

2.3 Acronyms

CTSTW	Command Transmission Status Word
DDL	Detector Data Link
DIU	Destination Interface Unit
DRORC	DAQ Readout Receiver Card
DTSTW	Data Transmission Status Word
EOBTR	Front-end command: End of Block Transmission
EODB	Front-end status word bit: End of Data Block
EOTR	Front-end status word bit: End of Transaction
FECMD	Front-end Command Word
FECTRL#address	Front-end command: Front-end Control
FESTW	Front-end Status Word
FESTRD#address	Front-end command: Front-end Status Read-out
FWSTW	Firmware Status Word
HLT	High-Level Trigger
HWSTW	Hardware Status Word
ICD	Interface Control Document
IFCMD	Interface Command Word
IFSTW	Interface Status Word
PhI	Physical and Signalling Interface Specification
STBRD#address	Front-end command: Open a Data Block Read Transaction
STBWR#address	Interface command: Open a Data Block WriteTransaction
SUSPND	Interface command: DDL Suspend
PMSTW	Power Monitor Status Word
PRORC	PCI-based Readout Receiver Card
RDFWID	Interface command: Read Firmware Identification Word
RDHWID	Interface command: Read Hardware Identification Word
RDYRX	Front-end command: Ready to Receive

RORC	Read-out Receiver Card
RPMVAL	Interface command: Read Power Monitor Value
R&CIFST	Interface command: Read&Clear Interface Status Word
SIU	Source Interface Unit
SRST	Interface command: SIU Reset
TSTART	Interface command: Test Mode Start
TSTOP	Interface command: Test Mode Stop
TXLOOP	Interface command: Transmitter Loop-back
WAKEUP	Interface command: DDL Wakeup

2.4 References

1. ALICE DAQ, ALICE DDL User Requirement Document, Internal Note ALICE-INT-1996-42 V3.0.
2. IEEE 1386-2001 and IEEE 1386.1-2001, IEEE Standard for a Common Mezzanine Card Family, http://www.ieee.org, 2001.
3. ALICE DAQ, ALICE DDL Hardware Guide for the Front-end Designers, Internal Note ALICE-INT-1998-21 V1.3.

2.5 Overview of the document

The second chapter of this document describes:

1. the purpose of the document
2. the scope
3. the definitions
4. the acronyms
5. the abbreviations
6. an overview of the document

The third chapter describes:

1. the physical description
2. the electrical description
3. the interface signals
4. the data, the command and the status word
5. the configurations
6. the transactions
7. the flow control
8. the interface timing

The fourth chapter describes:

1. the memory bank organisation
2. the HLT interface

3. General Description

3.1 Physical description

The original requirements on the size of the SIU boards, as requested by the front-end electronics, were more stringent than the present ones. Therefore, two different boards were developed for the SIU and the DIU in the prototyping phase.
Meanwhile, some of the user requirements have changed and a new form factor has been proposed. In the final version of the DDL, the same board can play the role of either the SIU or the DIU. This board is compliant with the Common Mezzanine Card Family standard [2]. This specification describes the physical arrangement of the final version of the DDL cards (Figure 3).

3.1.1 DDL hardware

The size of the DDL card corresponds to the half size of a single CMC board (149 mm x 37 mm). According to the standard, the card has got two 64-pin interface connectors. The male type connectors[2] are mounted on the DDL cards, while the female type counterparts[3] are placed on the RORC, or the FEE.

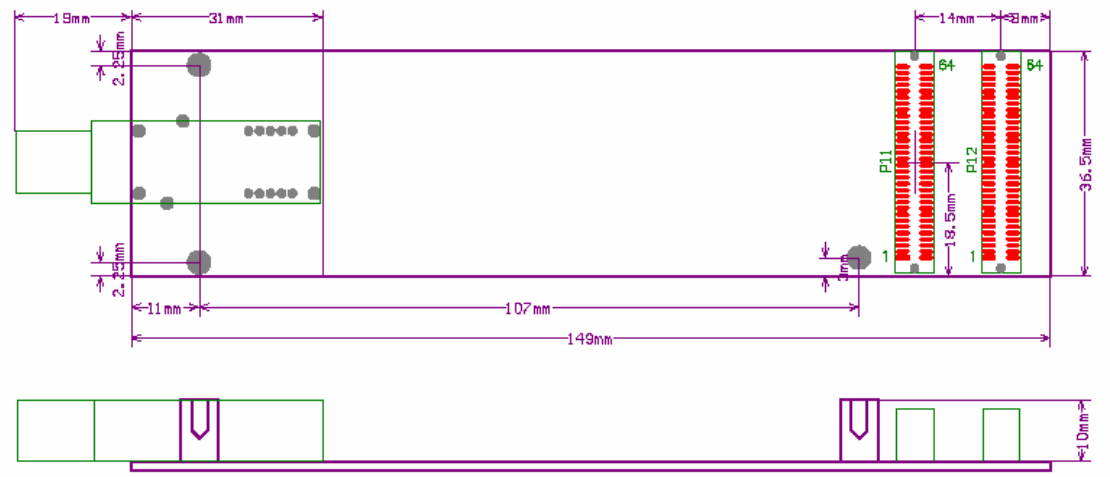


Figure 3 Physical arrangement of the DDL card

[2] **AMP – 120527-1**
[3] **AMP – 120521-1**

3.2 Electrical description

3.2.1 Power requirements

The SIU and the DIU operate from a single power supplies of +3.3V. The SIU is sourced from the FEE, while the DIU from the RORC. The maximum power consumption of the DDL card does not exceed 3 W.

3.2.2 Signal levels

The 2nd generation of the DDL card is +3.3V LVTTL compatible. Therefore, the designers of the FEE and the RORC **must not** use +5V TTL signals on the interfaces, because this can cause damage to the programmable device on the DDL card.

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC}	supply voltage		3.0	3.6	V
V_{IH}	high-level input voltage		1.7	4.1	V
V_{IL}	low-level input voltage		-0.5	0.8	V
V_{OH}	high-level output voltage	I_{OH} = -12 mA DC	2.4		V
V_{OL}	low-level output voltage	I_{OL} = 12 mA DC		0.4	V
I_I	input leakage current	V_I = 4.1V to -0.5V	-10	10	µA
I_{OZ}	tri-state output off-state current	V_O = 4.1V to -0.5V	-10	10	µA

Table 2 The LVTTL signal levels

3.3 Interface signals

3.3.1 Signal names

All the signal names are composed of four components: the interface identifier, the direction identifier, name identifier and the polarity identifier.

1. The DDL has two interfaces: the FEE-SIU interface and the RORC-DIU interface. The signals which belong to the FEE-SIU interface are prefixed with a letter "**f**", while the signals which belong to the RORC-DIU interface are prefixed with a letter "**r**".
2. The direction of the signals are defined from the point of view of the external systems (e.g. FEE, RORC). The input signals are indicated with an abbreviation "**i**", the output signals with "**o**", while the bi-directional signals with "**b**". In the signal name the direction identifier follows the interface identifier.
3. The name identifier is the abbreviation of the natural name of the signal. It is written by uppercase letters (e.g. DATA[31..0] -> **D[31..0]**, CONTROL -> **CTRL**). In the signal name the name identifier follows the direction identifier. Figure 4 shows all of the interface signals with their natural names and signal names.
4. The polarity identifier indicates the polarity of the active state of the signal. This identifier are "**_N**" characters for the active low signals and it is nothing for the active high signal. In the signal name the polarity identifier is located at the end.

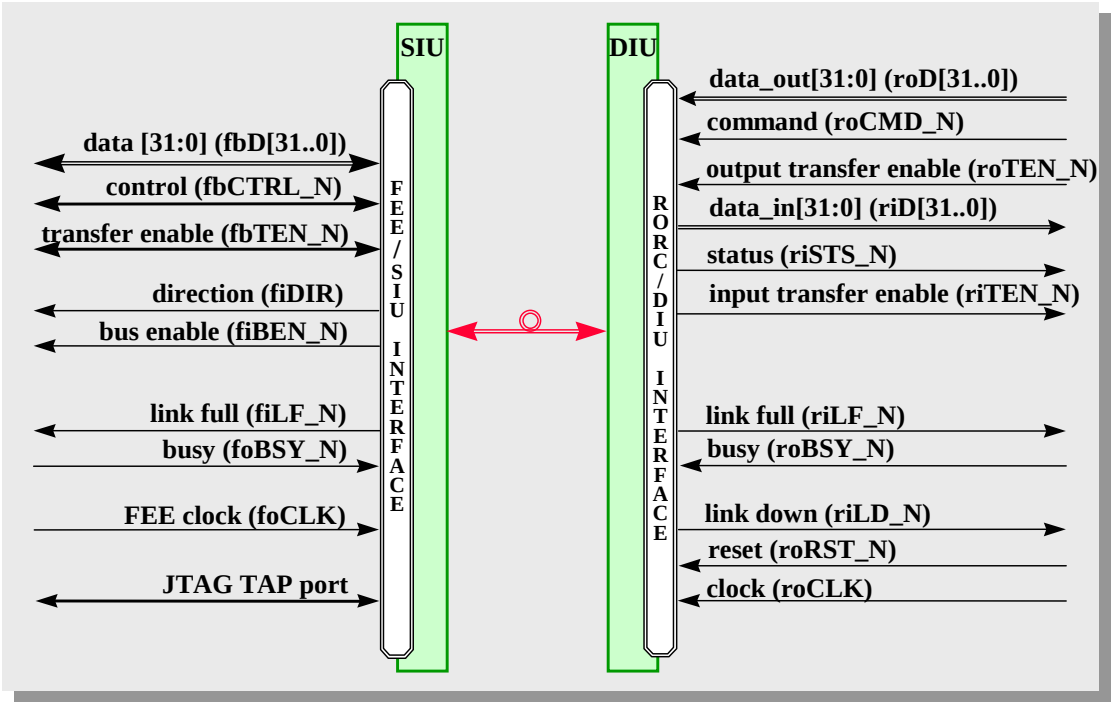


Figure 4 The names of the interface signals

3.3.2 Sample transactions

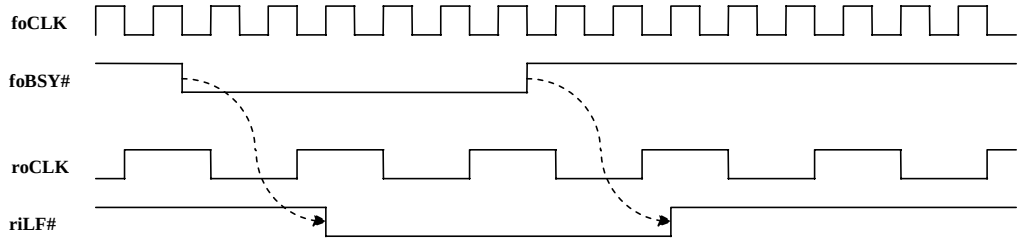


Figure 5 The timing diagram of the flow control signal transmission from the FEE to the RORC

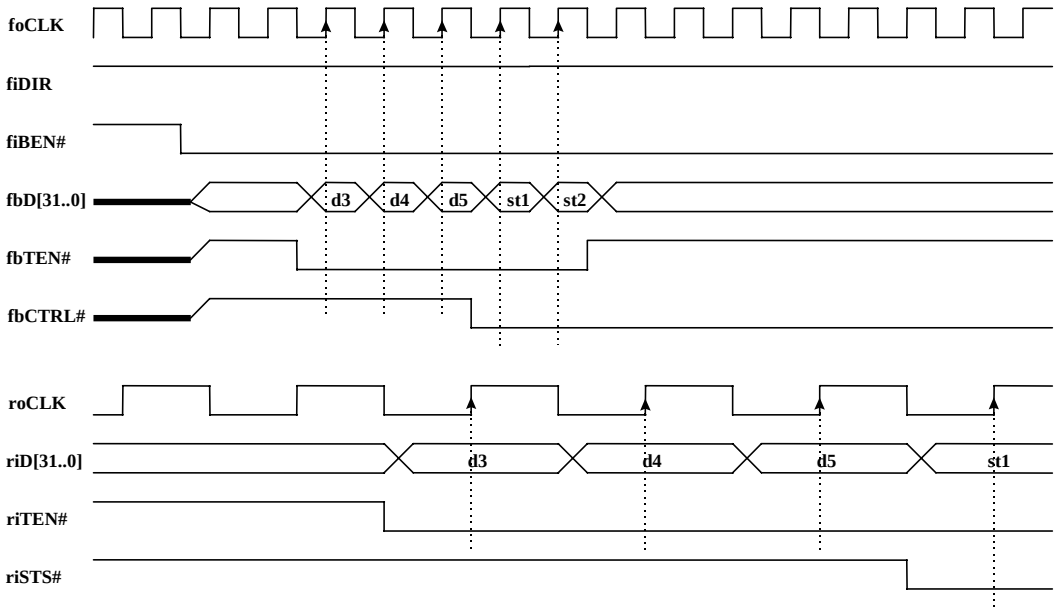


Figure 6 The timing diagram of the data transmission from the FEE to the RORC

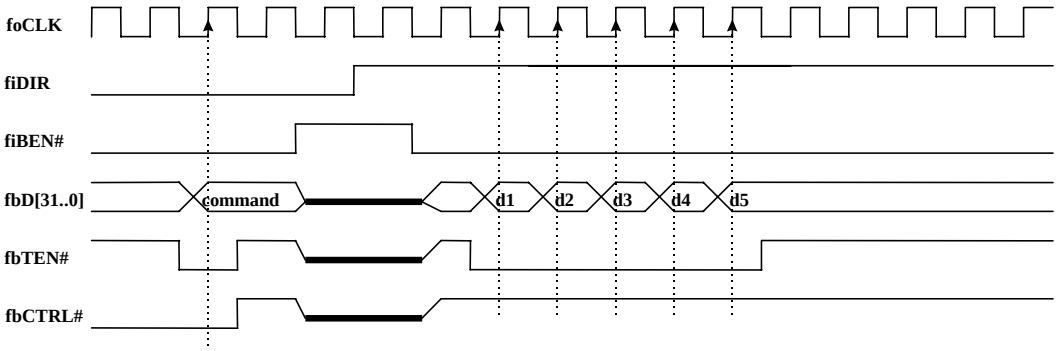


Figure 7 The timing diagram of the direction change of the data bus at the FEE-SIU interface

3.3.3 Signal line functions of the FEE-SIU interface

fbD[31..0]	**DATA[31..0]**	**bi-directional**	

The fbD[31..0] is a 32 bit wide, tri-state data bus. If the level of the fiDIR line is high, the data on these bus lines is transferred from the FEE to the SIU on a low-to-high transition of the foCLK when the fbTEN_N is active. The fbD[31..0] contains status words when the fbCTRL_N is active, otherwise it contains normal data words. If the level of the fiDIR line is low, the data on these bus lines is transferred from the SIU to the FEE on a low-to-high transition of the foCLK when the fbTEN_N is active. The fbD[31..0] contains commands when the fbCTRL_N is active, otherwise it contains normal data words.

fbCTRL_N	**CONTROL**	**bi-directional**	**active low**

The fbCTRL_N is a tri-state management line for the data bus. The active level of this line indicates that the data word to be transferred between the FEE and the SIU is a command or a status word, depending on the direction of the information transfer. It is a status word, when the information is transferred from the FEE to the SIU, otherwise it is a command.

fbTEN_N	**TRANSFER ENABLE**	**bi-directional**	**active low**

The fbTEN_N is a tri-state management line for the data bus. The active level of this line enables data to be transferred between the FEE and the SIU on the low-to-high transition of the foCLK. The direction of the information transfer depends on the level of the fiDIR line.

fiDIR	**DIRECTION**	**input**	**high: FEE to SIU transfer**

The fiDIR is a management line for the data bus. The high level of this line indicates that the information is transferred from the FEE to the SIU on the fbD[31..0] lines, while the low level indicates the opposite direction of the information transfer. When the level of this line is high, all of the bi-directional lines are driven by the FEE, otherwise they are driven by the SIU.

fiBEN_N	**BUS ENABLE**	**input**	**active low**

The fiBEN is a management line for the data bus. The inactive level of this line will put in high-impedance state the tri-state drivers of all the bi-directional lines of the FEE-SIU interface in the SIU and in the FEE.

fiLF_N	**LINK FULL**	**input**	**active low**

The fiLF_N is a flow control line. The active level of this line indicates that the FEE must stop information transfer to the SIU, because the SIU and/or the DIU and/or the RORC are busy. After this line becomes active, only one more data word or status word may be transferred from the FEE to the SIU.

foBSY_N	**BUSY**	**output**	**active low**

The foBSY_N is a flow control line. The active level of this line indicates that the FEE is not able to receive data from the DDL. After this line becomes active, only one more data word may be transferred from the SIU to the FEE.

foCLK	**FEE CLOCK**	**output**	

The foCLK is a clock line. This free running clock is generated by the FEE for the synchronisation of the information transfer between the FEE and the SIU. Every signal on the interface, except JTAG TAP lines, is synchronized to the rising edge of the foCLK.

3.3.4 Signal line functions of the RORC-DIU interface

roD[31..0]	**OUTPUT DATA[31..0]**	**output**	

The roD[31..0] is a 32 bit wide output data bus. Data on these lines is transferred from the RORC to the DIU on a low-to-high transition of the roCLK when the roTEN_N is active. The roD[31..0] contains commands when the roCMD_N is active, otherwise it contains normal data words.

roCMD_N	**COMMAND**	**output**	**active low**

The roCMD_N is a management line for the output data bus. The active level of this line indicates that the data word to be transferred from the RORC to the DIU is a command, while it is inactive the data word is a normal data.

roTEN_N	**OUTPUT TRANSFER ENABLE**	**output**	**active low**

The roTEN_N is a management line for the output data bus. The active level of this line enables data to be transferred from the RORC to the DIU on the low-to-high transition of the roCLK.

riD[31..0]	**INPUT DATA[31..0]**	**input**	

The riD[31..0] is a 32 bit wide input data bus. Data on these lines is transferred from the DIU to the RORC on a low-to-high transition of the roCLK when the riTEN_N is active. The riD[31..0] contains status words when the riSTS_N is active, otherwise it contains normal data words.

riSTS_N	**STATUS**	**input**	**active low**

The riSTS_N is a management line for the input data bus. The active level of this line indicates that the data word to be transferred from the DIU to the RORC is a status word, while it is inactive the data word is a normal data.

riTEN_N	**INPUT TRANSFER ENABLE**	**input**	**active low**

The riTEN_N is a management line for the input data bus. The active level of this line enables data to be transferred from the DIU to the RORC on the low-to-high transition of the roCLK.

riLF_N	**LINK FULL**	**input**	**active low**

The riLF_N is a flow control line. The active level of this line indicates that the RORC must stop information transfer to the DIU, because the DIU and/or the SIU and/or the FEE are busy. After this line becomes active, only one more data word or command may be transferred from the RORC to the DIU.

roBSY_N	**BUSY**	**output**	**active low**

The roBSY_N is a flow control line. The active level of this line indicates that the RORC is not able to receive data from the DDL. After this line becomes active, only one more data word may be transferred from the DIU to the RORC.

riLD_N	**LINK DOWN**	**input**	**active low**

The riLD_N is the link status line. The active level of this line indicates that the DIU is not in the on-line state, so no information can be transmitted through the DDL.

roRST_N	**RESET**	**output**	**active low**

The roRST_N is an asynchronous interface control line. The active level of this line initiates the reset cycle of the DIU. The reset cycle will be automatically initiated by the DIU after each power on.

roCLK	**RORC CLOCK**	**output**	

The roCLK is a clock line. This free running clock is generated by the RORC for the synchronisation of the information transfer between the RORC and the DIU. Every signal on the interface, excluding the asynchronous reset line, is synchronized to the rising edge of roCLK.

3.3.5 Pin-out diagrams

FEE-SIU interface connector

The pin-out (see Table 3) is derived from the CMC standard and it contains signals, which are not used on the SIU card. These signals – the BUSMODEx#, +12V, -12V and +5V – are internally not connected on the SIU card (marked with [1] in Table 3). Please note that the SIU uses only the +3.3V power supply (see 3.2.1).

CMC SIU (P11 connector)

1	FIBEN_N	-12V [1]	**2**
3	GND	FILF_N	**4**
5	FOBSY_N	FIDIR	**6**
7	BUSMODE1# [1]	+5V [1]	**8**
9	FBCTRL_N	FBTEN_N	**10**
11	GND	FBD0	**12**
13	FOCLK	GND	**14**
15	GND	FBD1	**16**
17	FBD2	+5V [1]	**18**
19	+3.3V	FBD3	**20**
21	FBD4	FBD5	**22**
23	FBD6	GND	**24**
25	GND	FBD7	**26**
27	FBD8	FBD9	**28**
29	FBD10	+5V [1]	**30**
31	+3.3V	FBD11	**32**
33	FBD12	GND	**34**
35	GND	FBD13	**36**
37	FBD14	+5V [1]	**38**
39	GND	FBD15	**40**
41	FBD16	FBD17	**42**
43	FBD18	GND	**44**
45	+3.3V	FBD19	**46**
47	FBD20	FBD21	**48**
49	FBD22	+5V [1]	**50**
51	GND	FBD23	**52**
53	FBD24	FBD25	**54**
55	FBD26	GND	**56**
57	+3.3V	FBD27	**58**
59	FBD28	FBD29	**60**
61	FBD30	+5V [1]	**62**
63	GND	FBD31	**64**

CMC SIU (P12 connector)

1	+12V [1]	TAP_TMS	**2**
3	TAP_TDI	TAP_TDO	**4**
5	TAP_TCK	GND	**6**
7	GND	TAP_TRST	**8**
9	-	-	**10**
11	BUSMODE2# [1]	+3.3V	**12**
13	-	BUSMODE3# [1]	**14**
15	+3.3V	BUSMODE4# [1]	**16**
17	-	GND	**18**
19	-	-	**20**
21	GND	-	**22**
23	-	+3.3V	**24**
25	-	-	**26**
27	+3.3V	-	**28**
29	-	GND	**30**
31	-	-	**32**
33	GND	-	**34**
35	-	+3.3V	**36**
37	GND	-	**38**
39	-	GND	**40**
41	+3.3V	-	**42**
43	-	GND	**44**
45	-	-	**46**
47	GND	-	**48**
49	-	+3.3V	**50**
51	-	-	**52**
53	+3.3V	-	**54**
55	-	GND	**56**
57	-	-	**58**
59	GND	-	**60**
61	-	+3.3V	**62**
63	GND	-	**64**

Table 3 The pin-out of the FEE-SIU interface connector

RORC-DIU interface connector

The pin-out (see Table 4) is derived from the CMC standard and it contains signals, which are not used on the DIU card. These signals – the BUSMODEx#, +12V, -12V and +5V – are internally not connected on the DIU card (marked with [1] in Table 4). Please note that the DIU uses only the +3.3V power supply (see 3.2.1).

CMC DIU (P11 connector)

1	RORST_N	-12V [1]	2
3	GND	RILF_N	4
5	ROBSY_N	RILD_N	6
7	BUSMODE1# [1]	+5V [1]	8
9	RISTS_N	RITEN_N	10
11	GND	RID0	12
13	ROCLK	GND	14
15	GND	RID1	16
17	RID2	+5V [1]	18
19	+3.3V	RID3	20
21	RID4	RID5	22
23	RID6	GND	24
25	GND	RID7	26
27	RID8	RID9	28
29	RID10	+5V [1]	30
31	+3.3V	RID11	32
33	RID12	GND	34
35	GND	RID13	36
37	RID14	+5V [1]	38
39	GND	RID15	40
41	RID16	RID17	42
43	RID18	GND	44
45	+3.3V	RID19	46
47	RID20	RID21	48
49	RID22	+5V [1]	50
51	GND	RID23	52
53	RID24	RID25	54
55	RID26	GND	56
57	+3.3V	RID27	58
59	RID28	RID29	60
61	RID30	+5V [1]	62
63	GND	RID31	64

CMC DIU (P12 connector)

1	+12V [1]	TAP_TMS	2
3	TAP_TDI	TAP_TDO	4
5	TAP_TCK	GND	6
7	GND	ROTEN_N	8
9	ROCMD_N	ROD0	10
11	BUSMODE2# [1]	+3.3V	12
13	ROD1	BUSMODE3# [1]	14
15	+3.3V	BUSMODE4# [1]	16
17	ROD2	GND	18
19	ROD3	ROD4	20
21	GND	ROD5	22
23	ROD6	+3.3V	24
25	ROD7	ROD8	26
27	+3.3V	ROD9	28
29	ROD10	GND	30
31	ROD11	ROD12	32
33	GND	ROD13	34
35	ROD14	+3.3V	36
37	GND	ROD15	38
39	ROD16	GND	40
41	+3.3V	ROD17	42
43	ROD18	GND	44
45	ROD19	ROD20	46
47	GND	ROD21	48
49	ROD22	+3.3V	50
51	ROD23	ROD24	52
53	+3.3V	ROD25	54
55	ROD26	GND	56
57	ROD27	ROD28	58
59	GND	ROD29	60
61	ROD30	+3.3V	62
63	GND	ROD31	64

Table 4 The pin-out of the RORC-DIU interface connectors

3.4 Data, command and status word

3.4.1 Data word

At the RORC-DIU interface the roD[31..0] output data bus contains data words, when the roCMD_N interface line is held in inactive state, while the roTEN_N interface line is active. The riD[31..0] input data bus contains data words, when the riSTS_N interface line is held in inactive state, while the riTEN_N interface line is active.

At the FEE-SIU interface the fbD[31..0] bi-directional data bus contains data words, when the fbCTRL_N interface line is held in inactive state, while the fbTEN_N interface line is active.

3.4.2 Command

The roD[31..0] output data bus contains commands at the RORC-DIU interface, when both the roCMD_N and roTEN_N interface lines are held in active state.

The fbD[31..0] bi-directional data bus contains commands at the FEE-SIU interface, when the fbCTRL_N and fbTEN_N interface lines are held in active state and the level of the fiDIR line is low. Figure 8 shows the general structure of the commands.

D31	D30 D12	D11 D8	D7	D6	D5	D4	D3	D2	D1	D0
don't use	PARAMETER FIELD	IDENTIFIER FIELD	CODE FIELD				DESTINATION FIELD			
X	FEE address	transaction ID	write/read	command code			JTAG	FEE	SIU	DIU

Figure 8 The general structure of the commands

All the commands consist of a destination field, a code field and an identifier field. Some of the commands may also use the parameter field (see Figure 9 and Figure 10):

- The destination field is composed from the D[3..0] bits. It defines the target system (e.g. JTAG, FEE, SIU, DIU) where the command is sent to.
- The code field is composed from the D[7..4] bits. It defines function of the commands. When the SIU receives a FECMD in which the value of the D[7] bit is set to logical '0' (*read*), the SIU first transfers the command to the FEE and then it automatically sets the front-end bus in the opposite direction, preparing the FEE-SIU interface for read operation (see Figure 7).
- The identifier field is composed from the D[11..8] bits. It defines transaction ID of the commands. All the commands and status words, belonging to the same transaction, should have identical transaction ID. The transaction IDs of the consecutive transactions should be different.
- The parameter field is composed from the D[30..12] bits. When the destination is the FEE, this field may contain the address inside the front-end.
- The D[31] bit is not used in the commands.

Front-end commands

The FECMDs are send from the RORC to the FEE through the DDL. There are only two standard FECMDs which must be implemented and used for the FEE of each sub-detector:

- *Ready to Receive* (RDYRX);
- *End of Block Transfer* (EOBRT).

An event data transmission transaction can be opened by using a RDYRX command (see 3.6.6). The RORC may only send this command to the FEE through the DDL, when it is ready to receive event data. When the FEE receives this command, it can transmit an event data block from the event memory after each read-out command from the trigger system. The DDL will be able to transmit event data from the FEE to the RORC, after transmitting the RDYRX command.

Any open block transfer transactions (e.g. event data transmission, user defined block write/read) must be closed by the FEE, when an EOBTR command is received from the RORC (see 3.6.6, 3.6.7, 3.6.8).

The implementation and application of the following 4 user defined FECMDs is optional:

- *Start of Block Write* (STBWR#address);
- *Start of Block Read* (STBRD#address);
- *Front-end Control* (FECTRL#address);
- *Front-end Status Read-out* (FESTRD#address).

A block write transaction can be opened by using a STBWR command. In this transaction a data block will be downloaded from the RORC to the FEE at the *address* defined in the parameter field of the STBWR command (see 3.6.7)

A block read transaction can be opened by using a STBRD command. In this transaction a data block will be read by the RORC from the FEE at the *address* defined in the parameter field of the STBRD command (see 3.6.8)

The STBWR and the STBRD commands can be used only, if the memories of the FEE are organised as memory banks. The memory bank organisation allows to read and write data blocks sequentially word-by-word from/to a given starting address of the FEE memories, without generating address cycle on the front-end bus (see 4.1).

The FEE can be controlled by using FECTRL commands (see 3.6.1). Each *address* can represent an independent control function, so the FEE designers can define up to 2^{19} different front-end control commands.

Status words can be read-out from the FEE by using FESTRD commands (see 3.6.2). Each *address* can represent an independent status word, so the FEE designers can define up to 2^{19} different front-end status words.

D31	D30 D12	D11 D8	D7	D6	D5	D4	D3	D2	D1	D0	data bits
X	PARAMETER FIELD	IDENTIFIER FIELD	WRITE	UD	CLOSE	BTR	0	1	0	0	FECMD
X	is not used	transaction ID	0	0	0	1	0	1	0	0	RDYRX
X	is not used	transaction ID	1	0	1	1	0	1	0	0	EOBTR
X	address in FEE	transaction ID	1	1	0	1	0	1	0	0	STBWR#address
X	address in FEE	transaction ID	0	1	0	1	0	1	0	0	STBRD#address
X	address in FEE	transaction ID	1	1	0	0	0	1	0	0	FECTRL#address
X	address in FEE	transaction ID	0	1	0	0	0	1	0	0	FESTRD#address

block transfer
close block transfer
user defined command
write/read operation

Figure 9 The front-end commands

Figure 9 shows all the valid front-end commands. Any other combinations in the code field (D[7..4] bits) are illegal. ***The transfer of the illegal front-end commands to the DIU is forbidden!*** It must be prevented by the DDL control program, because the command validity check is not implemented in the DDL hardware.

Interface commands

The IFCMDs are sent from the RORC to either the SIU or the DIU. The following IFCMDs are standard commands independently of the sub-detectors which the DDL is connected to:

- *DDL Suspend* (SUSPND);
- *DDL Wakeup* (WAKEUP);
- *SIU Reset* (SRST);
- *DIU Transmitter Loop-back* (TXLOOP);
- *SIU Start Test Mode* (TSTART);
- *SIU Stop Test Mode* (TSTOP);
- *Read Firmware Identification Word* (RDFWID);
- *Read Hardware Identification String* (RDHWID):
- *Read Power Monitor Value* (RPMVAL):
- *Read & Clear Interface Status Word* (R&CIFST).

The DIU can be switched to low-power state using the SUSPND command. The DIU puts the SIU in low-power state first, and then turns off its laser. The DIU can be waken up from the low-power state using the WAKEUP command. The DIU also wakes up the SIU followed by the power-on reset state before going to on-line state.

The reset cycle of the SIU can be initiated remotely by sending a SRST command from the RORC. This reset cycle will be automatically started after each power on.

The DIU can be tested without external system, if its transmitter is looped-back to its receiver. This working mode can be set by sending a TXLOOP command from the RORC to the DIU.

The SIU enters in or exits from the test mode upon receiving the TSTART or TSTOP commands, respectively. When the SIU is in test mode, the data or command words that are coming from the DDL are sent back to the DIU without presenting them on the FEE interface.

The firmware identification status word or the hardware identification string can be obtained from the DIU and SIU cards using the RDHWID and RDFWID commands, respectively (see 3.4.3).

The supply current of the laser of the DIU and SIU cards can be read out using the RPMVAL command.

The SIU and the DIU send back to the RORC an IFSTW and than delete the content of its error register, when it receives and R&CIFST command.

D31	D30 D12	D11 D8	D7	D6	D5	D4	D3	D2	D1	D0	data bits
X	parameter field	identifier field	Command code				0	0	1 or 0	1 or 0	IFCMD
X	is not used	transaction ID	1	0	1	0	0	0	0	1	SUSPND
X	is not used	transaction ID	1	0	1	1	0	0	0	1	WAKEUP
X	is not used	transaction ID	1	0	0	1	0	0	0	1	TXLOOP
X	is not used	transaction ID	1	1	0	1	0	0	1	0	TSTART
X	is not used	transaction ID	1	1	0	0	0	0	1	0	TSTOP
X	is not used	transaction ID	1	1	1	1	0	0	0	1	SRST
X	is not used	transaction ID	0	1	0	0	0	0	1 or 0	1 or 0	RDFWID
X	EEPROM address	transaction ID	0	1	1	0	0	0	1 or 0	1 or 0	RDHWID
X	is not used	transaction ID	0	1	1	1	0	0	1 or 0	1 or 0	RPMVAL
X	is not used	transaction ID	0	0	0	0	0	0	1 or 0	1 or 0	R&CIFST

Figure 10 The interface commands

Figure 10 shows the valid interface commands Any other combinations in the code field (D[7..4] bits) are illegal. ***The transfer of the illegal interface commands to the DIU is forbidden!***

3.4.3 Status word

At the RORC-DIU interface the riD[31..0] output data bus contains status words, when the riSTS_N interface line is held in active state. At the FEE-SIU interface the fbD[31..0] bi-directional data bus contains status words, when the fbCTRL_N interface line is held in active state, while the fbTEN_N interface line is active and the level of the fiDIR line is high. Figure 11 shows the general structure of the status words.

D31	D30 D12	D11 D8	D7 D6 D5 D4	D3 D2 D1 D0
ERROR BIT	PARAMETER FIELD	IDENTIFIER FIELD	CODE FIELD	SOURCE FIELD
error/ok	status parameter	transaction ID	status code	JTAG / FEE / SIU / DIU

Figure 11 The general structure of the status words

All the status words consist of a source field, a code field, an identifier field, a parameter field and an error bit:

- The source field is composed from the D[3..0] bits. It defines the source system (e.g. JTAG, FEE, SIU, DIU) where the status word was generated in.
- The code field is composed from the D[7..4] bits. The D[7..6] bits define the type of the status words, while the D[4] bit indicates the start of a data block transaction and the D[5] bit the end of a data block transaction or the end of a data block.
- The identifier field is composed from the D[11..8] bits. It defines transaction ID of the status words except the DTSTW, where D[11..9] bits are set to '0' and D[8] is used to indicate the data block continuation (see later).
- The parameter field is composed from the D[30..12] bits. It defines the parameters of the status words.
- The error bit is composed from the single D[31] bit. The status word reports about the error(s) in the system (FEE and DDL), when this bit is set to logical '1'.

There are eight different status words (see Figure 12):

- *Command Transmission Status Word* (CTSTW);
- *Front-end Status Word* (FESTW);
- *Data Transmission Status Word* (DTSTW);
- *Interface Status Word* (IFSTW);
- *Firmware Status Word* (FWSTW);
- *Hardware Status Word* (HWSTW);
- *Power Monitor Status Word* (PMSTW).

D31	D30 D12	D11 D8	D7	D6	D5	D4	D3	D2	D1	D0	data bits
ERROR	PARAMETER FIELD	IDENTIFIER FIELD	STATUS CODE				JTAG	FEE	SIU	DIU	STATUS WORD
error	Command parameter	transaction ID	0	0	IL(1	TO(2	0	1 or 0	1 or 0	1 or 0	CTSTW
error	front-end status	transaction ID	0	1	EODB(3	0	0	1	0	0	FESTW
error	DDL block length	continuation	1	0	0	0	0	0	1	0	DTSTW
error	interface status	transaction ID	1	1	0	0	0	0	1 or 0	1 or 0	IFSTW
error	firmware ID	transaction ID	0	1	0	0	0	0	1 or 0	1 or 0	FWSTW
error	EEPROM data	transaction ID	0	1	1	0	0	0	1 or 0	1 or 0	HWSTW
error	Current value	transaction ID	0	1	1	1	0	0	1 or 0	1 or 0	PMSTW

1. IL – I*llegal Command*
The IL bit will be set to '1' , if the command is illegal.
2. TO – Time-out
The TO bit will be set to '1', if the front-end exceeds the time-out interval during a transaction.
2. EODB - *End of Data Block*
The EODB bit will be set to '1' by the FEE at the end of each data block, transferred to the SIU.

Figure 12 The status words

Command Transmission Status Word

The DIU generates a CTSTW and transfers it to the RORC, when it receives any IFCMDs (see 3.6.3). The SIU generates a CTSTW and sends it to the RORC, when it receives any IFCMDs or FECMDs (see 3.6.1 - 3.6.9, excepting 3.6.3). In the parameter field of the CTSTW the original command parameter should be returned, if any. All the transactions are terminated by a CTSTW. It indicates by setting the error bit to logical '1', if any errors are occurred during the transaction. The CTSTW is also used for the acknowledgement of that commands which open block transfer transactions. In this case, it indicates by setting the error bit to logical '1', if any errors are occurred during the transmission and execution of the command (see 3.6.6 - 3.6.9). ***In any cases, when the error bit in the CTSTW is set to '1', the DDL control software must read-out the interface status words from both interface units by sending a R&CIFST command, in order to identify all the errors in the DDL!***

Front-end Status Word

The RORC can read-out a FESTW by sending a FESTRD#address command (see 3.6.2). All the FESTWs are sub-detector specific and they must be defined by the FEE designers. The parameter field is reserved for FEE specific status information, while the error bit can be used for the indication of any defined error states inside the FEE. A FESTW must automatically be generated by the FEE (with the EODB bit is set to '1'), when the transfer of a data block to the SIU is terminated. Upon receiving the block termination FESTW, the SIU generates and sends the DTSTW to the RORC (see 3.6.6, 3.6.8).

Data Transmission Status Word

The SIU generates a DTSTW and sends it to the RORC at the end of each DDL data block. The maximum size of one DDL data block is 524287 word (32-bit). If the size of the data block produced by the front-end or the HLT computers is larger than the maximum size of one DDL data block, the SIU will split the data block by inserting special DTSTWs after each DDL data block transmitted to the RORC. This special DTSTW is marked by the SIU by setting the so-called *continuation bit* (D[8]).

The RORC should also generate a DTSTW and send it to the SIU at the end of each download data block (see 3.6.7). The parameter field of the DTSTW contains the length of the transmitted data block in words (4 bytes). The transmission errors are detected by the following interface units in the different block transactions:

- DIU in the *Event Data Transmission* and *Data Block Read* transactions;
- SIU in the *Block Downloading* transaction.

In the first two cases, the error is reported in the DTSTW by setting the error bit (bit 31) to logical '1'. In case of the data block download, the error is reported in the CTSTW, which is sent as a reply for the EOBTR command issued by the RORC.

Interface Status Word

The RORC can read-out an IFSTW from the SIU and the DIU by sending a R&CIFST command to them (see 3.6.5). The bits of parameter field are connected to the internal status and error registers of the interface units. Table 6 and Table 5 show the meaning of these bits. The error bit (bit 31) is set to logical '1', if any error occurred in the interface units, it means this bit constitutes the *logical OR* function of the error bits. The error bits in the interface status registers are automatically reset, when this command is executed.

Firmware Status Word

The RORC can read-out an FWSTW from the SIU and the DIU by sending a RDFWID command to them. The parameter field of the reply will contain the version code of the firmware, as well as the date of the compilation (Figure 13).

D30 D25	D24 D21	D20 D17	D16 D12
Version code	Year after 2000	Month	Day

Figure 13 The parameter field of the firmware identification status word

Hardware Status Word

The hardware identification string can be obtained from the SIU and DIU cards by reading out the on-board serial EEPROM. Reading one single character from the EEPROM consists of sending a RDHWID command to the interface card and interpreting the answer encoded in the parameter field of the HWSTW (Figure 14).

D30 D28	D27 D20	D19 D12
reserved	EEPROM data (ASCII code)	EEPROM address

Figure 14 The parameter field of the hardware status word

Power Monitor Status Word

The RORC can read-out the value of the laser power monitor circuitry on the interface cards by sending the RPMVAL interface command. The parameter field of the PMSTW will contain the binary encoded value (Figure 15).

The transmitter current can be calculated as: 0.034*PMV [mA].

D30 D24	D23 D12
reserved	Power Monitor Value (PMV)

Figure 15 The parameter field of the power monitor status word

data bit	acronym	description	remark
30 – 17	-	ERRORS	
30	LEVNT	Too long event fragment	***error***
29	ILLFDS	Illegal front-end data/status	***error***
28	TXOF	Transmitter overflow (FEE to DDL)	***error***
27	ILLWRD	Illegal data from the DDL	***error***
26	OSINFR	Ordered set inside data frame	***error***
25	INVCH	Invalid character inside data frame	***error***
24	CRCERR	CRC error	***error***
23	BLERR	Download block length error (DTCC)	***error***
22	DOUT	Data word outside data frame	***error***
21	INVSOF	Invalid start of frame delimiter	***error***
20	FLERR	Frame length error	***error***
19	RXOF	Receiver overflow (DDL to FEE)	***error***
18	FRERR	Command or data frame error	***error***
17	PRERR	Protocol error	***error***
16 – 15	-	STATUS BITS	
16	FBLOOP	Front-end bus loopback	normal state
15	FETRAN	Front-end transaction is active	normal state
14– 12	-	LINK MANAGER STATUS	
000	PWRON	Power-on reset state (SIU reset)	normal state
001	SIUOF1	SIU is in off-line state, no error	normal state
010	SIUONL	SIU is in on-line state	normal state
011	PWROF	SIU is going to low power state	normal state
100	SIUOF2	SIU is in off-line state, no optical signal	***error*** state
101	SIUOF3	SIU is in off-line state, weak signal	***error*** state

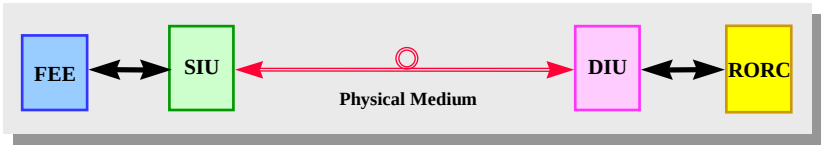
Table 5 The SIU error and status bits

data bit	acronym	description	remark
30	TXLOOP	Transmitter loop-back	normal state
29	LOSY	Loss of synchronisation	***error***
28	TXOF	Transmitter overflow (RORC to DDL)	***error***
27	RSVD	Reserved (read as zero)	reserved bit
26	OSINFR	Ordered set inside data frame	***error***
25	INVCH	Invalid character inside data frame	***error***
24	CRCERR	CRC error	***error***
23	RSVD	Reserved (read as zero)	reserved bit
22	DOUT	Data word outside data frame	***error***
21	INVSOF	Invalid start of frame delimiter	***error***
20	FLERR	Frame length error	***error***
19	RXOF	Receiver overflow (DDL to RORC)	***error***
18	FRERR	Command or data frame error	***error***
17 - 15	-	SIU PORT STATES	
000	TXIDLE	SIU is sending IDLE characters	normal state
001	SIUOF1	SIU is in off-line state, no error	normal state
010	SIUONL	SIU is in on-line state	normal state
011	SIUTXS	SIU is sending SUSPEND signal	normal state
100	SIUOF2	SIU is in off-line state, no signal	***error*** state
101	SIUOF3	SIU is in off-line state, unknown error	***error*** state
110	NOSIG	No optical signal	***error*** state
14 - 12	-	DIU PORT STATES	
000	PWRON	Power-on reset state	normal state
001	DIUOF1	DIU is in off-line state, no error	normal state
010	DIUONL	DIU is in on-line state	normal state
011	DIUTXS	DIU is sending SUSPEND signal	normal state
100	DIUOF2	DIU is in off-line state, no optical signal	***error*** state
101	DIUOF3	DIU is in off-line state, weak signal	***error*** state
110	PWROF	DIU is in low power state	normal state
111	DIURXS	DIU receives SUSPEND signal	normal state

Table 6 The DIU status and error bits

3.5 Configurations

The DDL can be used in the following two configurations: SIU - DIU configuration (see Figure 16/a) and DIU - DIU configuration (see Figure 16/b).



a.) SIU - DIU configuration

b.) DIU - DIU configuration

Figure 16 The DDL configurations

In the SIU - DIU configuration a SIU, a DIU and a RORC are used. All the subsystems of this configuration are controlled by the DDL control software. Not a single local signal is needed for the control of the SIU, located on the other side of the physical medium. Figure 17 shows the information transfer between the different subsystems in the SIU - DIU configuration.

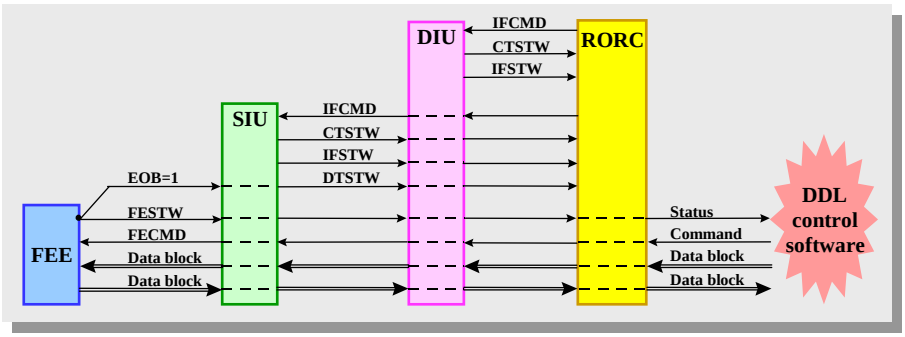


Figure 17 The information transfer in the SIU - DIU configuration

In the DIU - DIU configuration two DIUs and two RORCs are used. The Master RORC and its DIU are controlled by the DDL control software, while the Slave RORC and its DIU by the FEE/SIU emulator software. In this configuration the Slave RORC plays the role of the FEE, while its DIU the role of the SIU. Figure 18 shows the information transfer between the different subsystems in the DIU - DIU configuration.

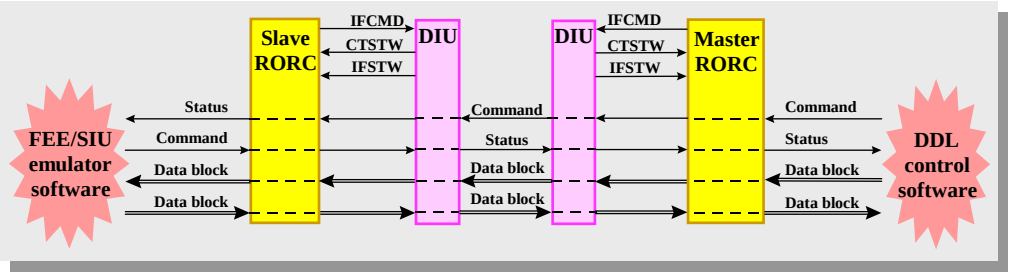


Figure 18 The information transfer in the DIU - DIU configuration

3.6 Transactions

3.6.1 User defined front-end control transaction

The FEE can remotely be controlled by the RORC, sending user defined FECTRL#address commands. The FECTRL commands are transmitted from the RORC to the FEE through the DDL. All the FECTRL commands are automatically acknowledged by the SIU, sending back a CTSTW to the DIU. The CTSTW indicates, if any error occurred during the transmission of a FECTRL command. The FECTRL command is not transferred from SIU to FEE, if a transmission error has occurred. The CTSTW is transferred from DIU to RORC for the termination of the transaction and error reporting. Figure 19 shows the front-end control transaction.

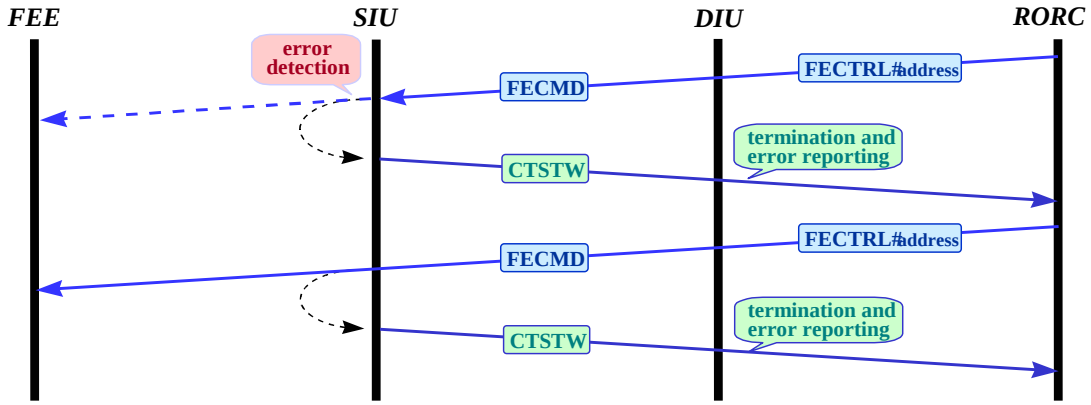


Figure 19 The front-end control transaction

3.6.2 User defined front-end status word read-out transaction

RORC can read-out FESTW from the FEE by sending a user defined FESTRD#address command to the FEE through the DDL. Receiving a FESTRD command, the FEE transfers a FESTW to the SIU and then it is transmitted to the RORC through the DDL. At the end of the transaction the FESTRD command is acknowledged by SIU, using a CTSTW. The CTSTW indicates, if any error occurred during the transmission of a FESTRD command. The FESTRD command is not transferred from SIU to FEE, if a transmission error has occurred. The CTSTW is transferred from DIU to RORC for termination of the transaction and error reporting. Figure 20 shows front-end status word read-out transaction.

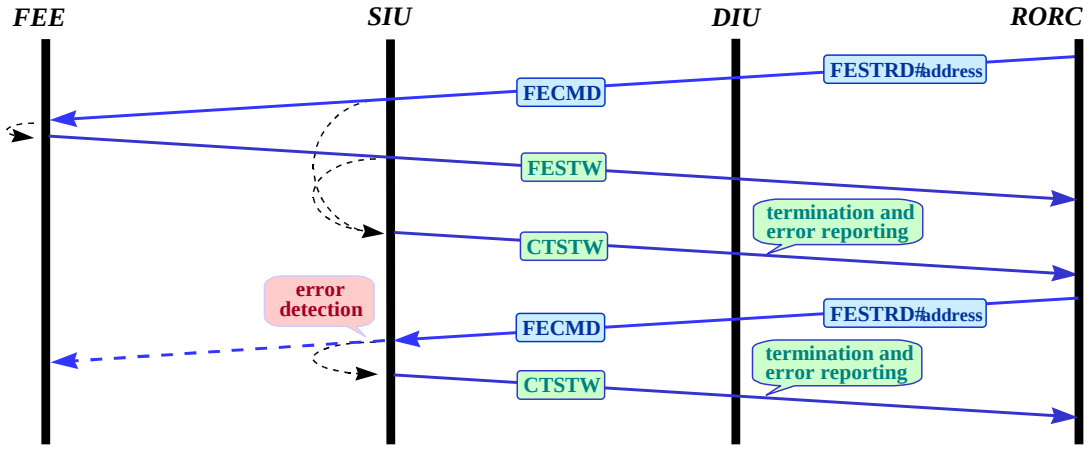


Figure 20 The front-end status word read-out transaction

3.6.3 DIU control transaction

The DIU is controlled directly by the RORC, using IFCMDs. The IFCMDs are transferred from RORC to DIU and automatically are acknowledged by using CTSTWs. The CTSTW indicates, if any error occurred during the transfer and execution of the IFCMD. The CTSTW transferred from the DIU to the RORC for the termination of the transaction and error reporting. Figure 21 shows the DIU control transaction.

Figure 21 The DIU control transaction

3.6.4 SIU control transaction

The SIU is remotely controlled by the RORC, using IFCMDs. All the IFCMDs are automatically acknowledged by the SIU, using CTSTWs. The CTSTW indicates, if any error occurred during the transmission and execution of the IFCMD. The CTSTWs are transmitted from the SIU to the DIU and then transferred from the DIU to the RORC for the termination of the transaction and error reporting. Figure 22 shows the SIU control transaction.

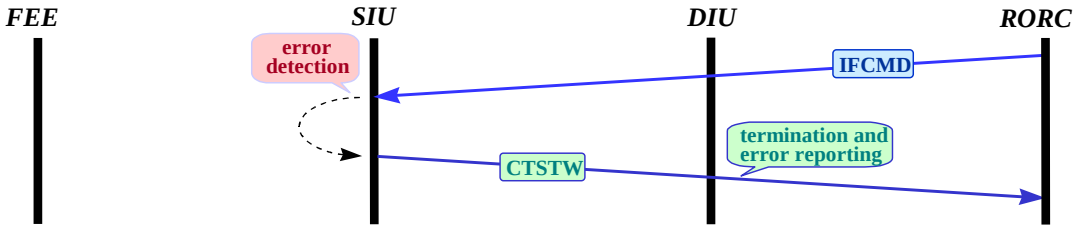


Figure 22 The SIU control transaction

3.6.5 Interface status read-out transaction

RORC can read-out IFSTWs from the SIU and the DIU by sending a R&CIFST command to the interface units. The IFSTW contains detailed information on the internal status and errors of the interface units. The transaction errors are reported and the transaction is terminated by sending a CTSTW from the interface unit to the RORC. Figure 23 shows the interface status read-out transaction.

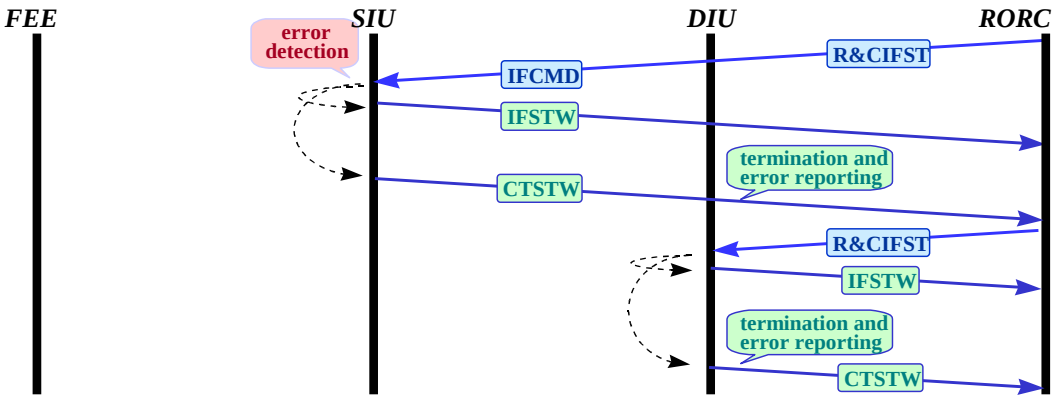


Figure 23 The interface status read-out transaction

3.6.6 Event data transmission transaction

Event data can be transmitted from the FEE to the RORC via the forward channel. A RDYRX command is sent first to the FEE by the RORC, in order to open the event data transmission transaction. The transmission of this command is acknowledged by the SIU. Event data blocks are transmitted, possibly with flow control, from the FEE to the RORC after each read-out command from the trigger system. The end of a data block is indicated by the FEE, transferring a FESTW to the SIU (see 3.4.3). When a FESTW is received, a DTSTW is generated by the SIU and it is transmitted to the RORC. Data blocks can be continuously transmitted form the FEE to the RORC until the data transmission is closed by the RORC, sending an EOBTR command to the FEE. The transmission of this command is acknowledged by the SIU sending back to the DIU a CTSTW. This status word is transferred from DIU to RORC for termination of the transaction and error reporting. Figure 24 shows the event data transmission transaction.

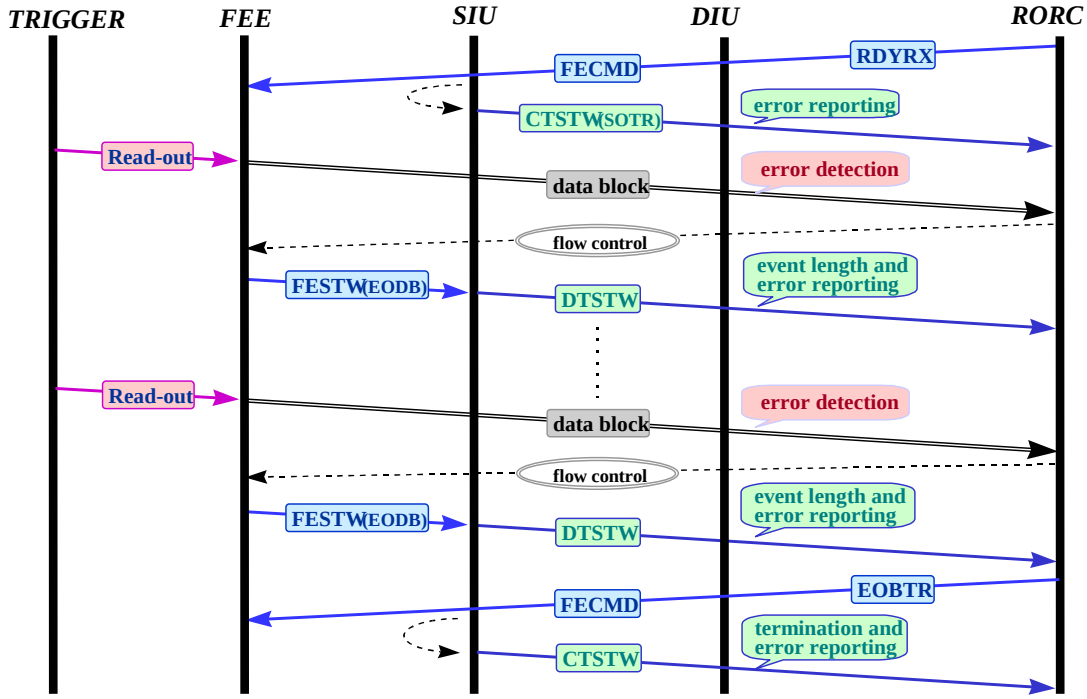


Figure 24 The event data transmission transaction

The data transfer from the FEE to the SIU must be suspended for at least 16 foCLK period between the data blocks, in order to allow the SIU to invert the bus direction and deliver a command to the FEE.

3.6.7 User defined data block downloading transaction

Detector parameter data and event data can be downloaded from the RORC to the memories of the FEE via backward channel. First a STBRW#address (see 4.1) command is sent by the RORC to the FEE in order to select the target memory bank for the write transaction. The transmission of this command is acknowledged by the SIU. A data block is transmitted from RORC to the selected memory bank of the FEE, possibly with flow control, followed by the DTSTW generated by the RORC. The end of a data block is indicated by the RORC, sending an EOBTR command to the FEE. Receiving this command, first the target memory bank is deselected in the FEE, and the command transmission is acknowledged by the SIU sending back a CTSTW to the DIU. This status word is transferred from DIU to RORC for termination of the transaction and error reporting. Figure 25 shows the data block downloading transaction.

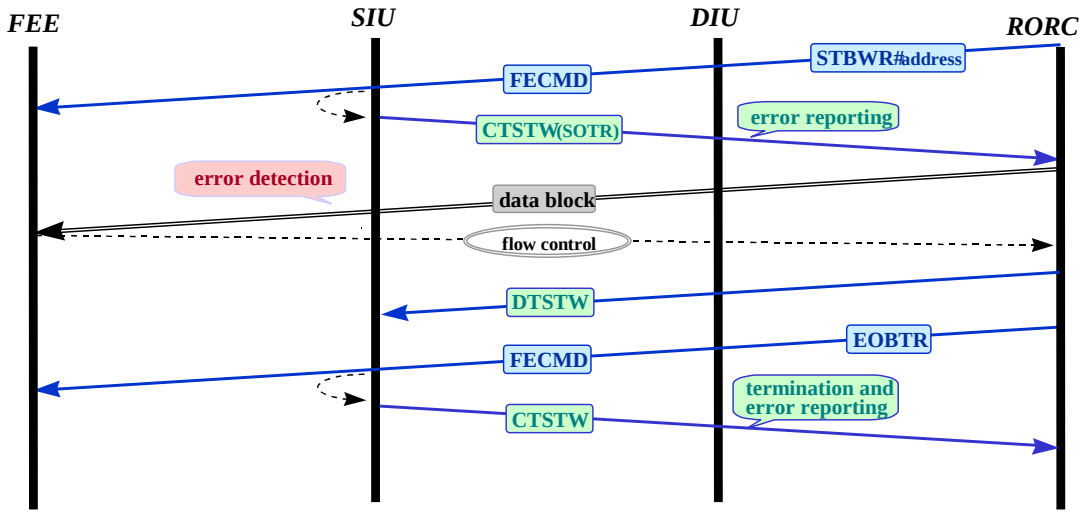


Figure 25 The data block downloading transaction

3.6.8 User defined data block read transaction

Data blocks can be read-back by the RORC from the memories of the FEE via the backward channel. First a STBRD#address command is sent by the RORC to the FEE, in order to select the target memory bank for the read transaction. The transmission of this command is acknowledged by the SIU. A data block is transmitted from the selected memory bank of the FEE to the RORC, possibly with flow control. The end of a data block is indicated by the FEE, transferring a FESTW (with EODB=1) to the SIU (see 3.4.3). When this FESTW is received, a DTSTW is generated by the SIU and it is transmitted to the RORC. The read-back process is closed by the RORC, sending an EOBTR command to the FEE. Receiving this command, first the target memory bank is deselected in the FEE and than the command transmission is acknowledged by the SIU, sending back a CTSTW to the DIU. This status word is transferred from DIU to RORC for termination of the transaction and error reporting. Figure 26 shows the data block read transaction.

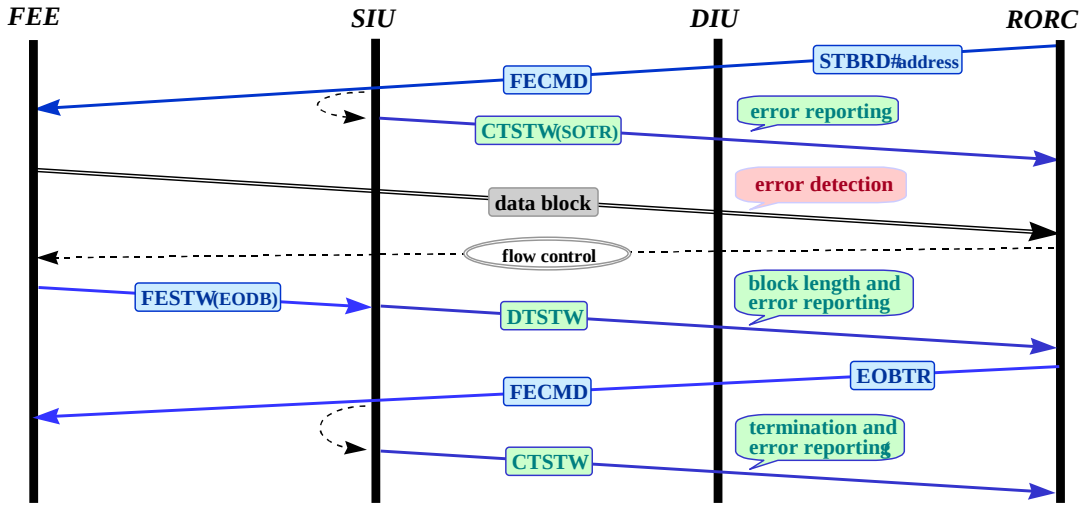


Figure 26 The data block read transaction

3.6.9 Self-test transaction

The host processor initiates the self-test process by sending a TSTART command to the DIU and the SIU. The transmission of this command is acknowledged by the SIU. A test data block is transmitted from the memory of the host computer by the RORC through the DIU to the SIU. The data words of the test data block are looped-back from the SIU to the input buffer of the RORC. The data is transferred back into the memory of the host computer, where the software can perform the check. This procedure can be repeated many times. The test between the RORC and the SIU continues until the host processor closes the self-test transaction by sending a TSTOP command, which is acknowledged by the SIU sending back a CTSTW to the DIU. This status word is transferred from DIU to RORC for termination of the transaction and error reporting. Figure 27 shows the self-test transaction.

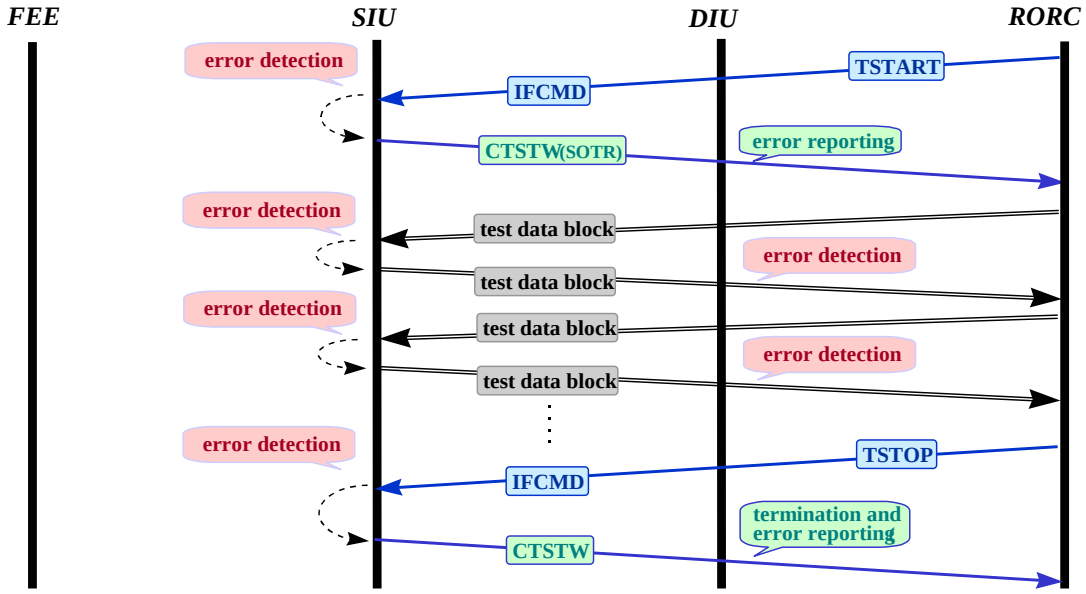


Figure 27 The self-test transaction

3.6.10 Transaction rules

Transaction types

Three main types of the transactions are described in the ICD: the basic transactions, the data block transmission transactions and the self-test transaction (see 3.6.9).

The following basic transactions are defined:

1. interface transactions:
   - DIU control (see 3.6.3);
   - SIU control (see 3.6.4);
   - DIU and SIU status read-out (see 3.6.5)
2. front-end transactions:
   - control (see 3.6.1);
   - status read-out (see 3.6.2).

The following data block transmission transactions are defined:

1. event data transmission (see 3.6.6);
2. data block downloading (see 3.6.7);
3. data block read (see 3.6.8).

Transaction rules

1. Interface transactions:
   - The TSTART and the TSOP interface commands are excluded from the interface transactions, because they control the self-test transaction.
   - A new DIU control or status read-out transaction can only be started, when the previous DIU control or status read-out transaction has already been accomplished.
   - A new SIU control or status read-out transaction can only be started, when the previous SIU control or status read-out transaction has already been accomplished.
   - A new SIU control or status read-out transaction can only be started, when the previous DIU control or status read-out transaction has already been accomplished.
   - New DIU control or status read-out transactions can be started any time during a SIU control or status read-out transaction.
   - New interface transactions can be started any time during a front-end transaction.
   - New interface transactions can be started any time during a data block transmission transaction.
   - New interface transactions can only be started, when the previous self-test transaction has already been accomplished.
   - The DIU immediately transmits all the interface commands through the DDL which are sent to the SIU.
2. Front-end transactions:
   - A new front-end transaction can only be started, when the previous interface transaction has already been accomplished.
   - A new front-end transaction can only be started, when the previous front-end transaction has already been accomplished.
   - A new front-end transaction can only be started, when the previous data block transmission transaction has already been accomplished. All of the data block transmission transactions are closed by a standard EOBTR front-end control command.
   - A new front-end transaction can only be started, when the previous self-test transaction has already been accomplished.
3. Data block transmission transactions:
   - A new data block transmission transaction can only be started, when the previous interface transaction has already been accomplished.
   - A new data block transmission transaction can only be started, when the previous front-end transaction has already been accomplished.
   - A new data block transmission transaction can only be started, when the previous data block transmission transaction has already been accomplished.
   - A new data block transmission transaction can only be started, when the previous self-test transaction has already been accomplished.
4. Self-test transaction:
   - No self-test transaction can be started during other transactions.

	new transaction				
current transaction	DIU	SIU	front-end	data block transmission	self-test
DIU	–	–	–	–	–
SIU	•	–	–	–	–
front-end	•	•	–	–	–
data block transmission	•	•	EOBTR only	–	–
self-test	•	•	–	–	–

Table 7 The transaction rules

Table 7 shows the new transactions which can be started (•) or can not be started (-) before the current transaction has been accomplished. The transaction rule check is implemented by the DDL control software.

3.7 Flow control

If the PCI bandwidth available to one RORC does not allow data transfer at full DDL speed, the RORC can use flow control by activating the BUSY line at the RORC-DIU interface. If the FEE is not fast enough to receive data, it can use flow control by activating the BUSY line at the FEE-SIU interface. Both the SIU and the DIU card will include small FIFO memories to temporally store data words, while the flow control is active on the their interface. Should any of those buffers get almost full, the corresponding interface card will suspend the transfer by sending XOFF signals to the remote end. Once the buffer is free again, XON signals will be sent to revoke the flow control.

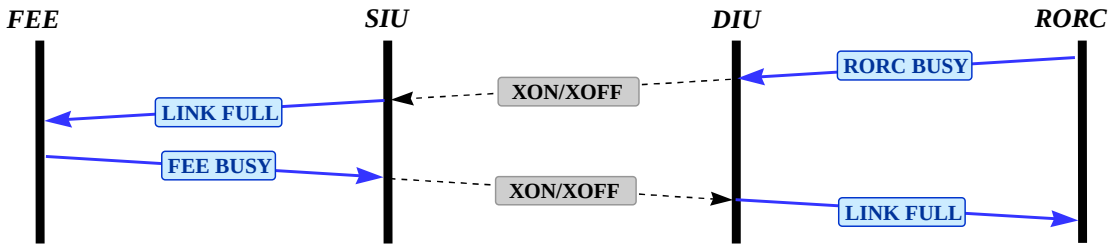


Figure 28 The flow control protocol

3.8 Interface timing

3.8.1 DIU reset

The DIU reset cycle can be initiated by the RORC, activating the roRST_N interface line at least during 4 roCLK cycles (see 3.3.4). This cycle will be automatically initiated by the DIU after each power on. Figure 29 shows the timing diagram of the DIU reset cycle.

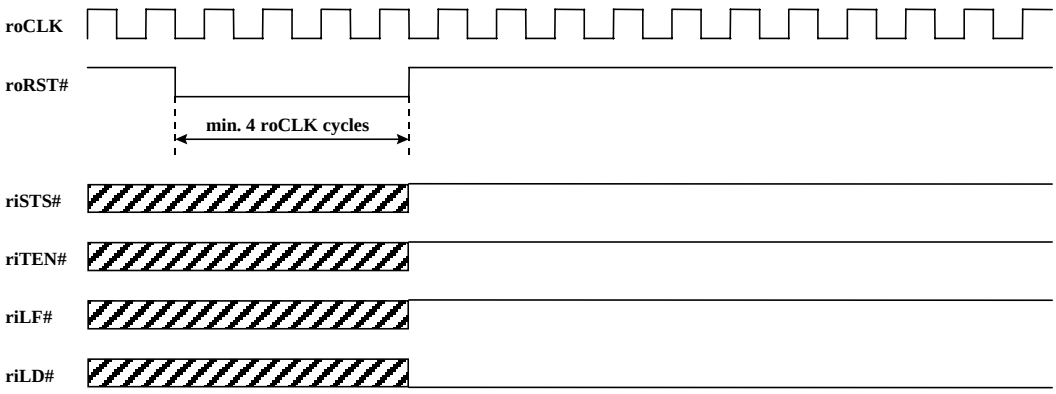


Figure 29 The timing diagram of the DIU reset cycle

3.8.2 Physical link initialisation

The physical link of the DDL is managed by the link manager state-machines implemented in the SIU and the DIU. There are three stable states of the link managers on both side: low power state, off-line state and on-line state (see Figure 30).

Following the power-on initialisation of the hardware, the link manager of the DIU will wake-up in the power-on reset state. While the DIU is in this state, it will send special characters on the physical medium to the remote end, in order to facilitate the synchronization at the level of the physical layer (PL). After the expiration of the power-on reset timer, the DIU will enter in the off-line state and start acquire the signals from PL device. If the received signals are good for sufficient amount of time, the state-machine will advance into the on-line state.

The link manager state-machine of the SIU is similar to the one of the DIU. However, the SIU state-machine wakes up in the SIU reset state. This state is also entered upon receiving the SIU reset signal from the DIU. Depending on the presence of the optical signal at the receiver, the SIU will enter into the power-on reset state (optical signal detected), or it will switch to low-power mode (optical signal is not detected).

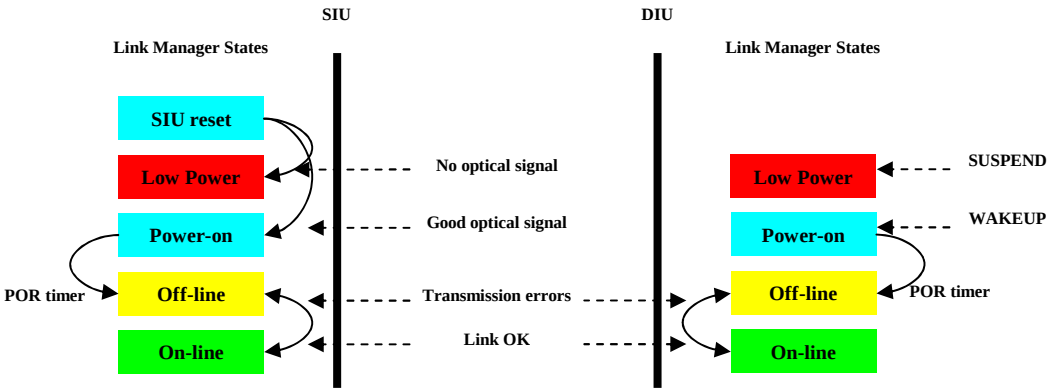


Figure 30 DIU and SIU link managers

3.8.3 SIU reset

The SIU reset cycle can be initiated by the RORC transmitting a SRST interface command to the DIU (see 3.4.2), which will send the reset signal to the SIU. The reset will be automatically initiated by the SIU after each power on. When the SIU receives the signal, it activates the fiBEN_N interface line and put in inactive state the fbCTRL_N, the fbTEN_N and the fiLF# interface lines. The SIU sets the fiDIR interface line in logical '0' level, so the front-end bus is directed from the SIU towards the FEE (see Figure 31). After sending the reset signal, the DIU sends back a CTSTW to the RORC for the acknowledgement of the SRST interface command.

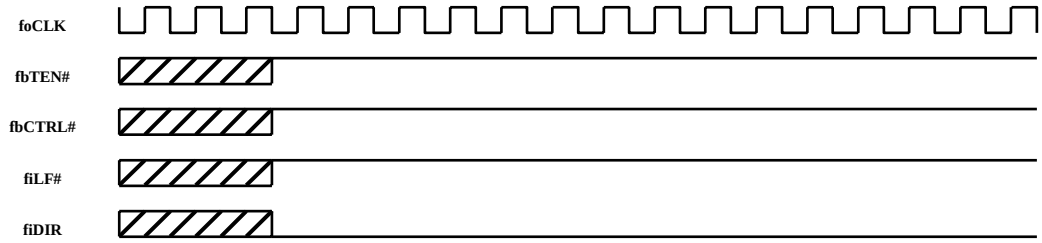


Figure 31 Timing diagram of the SIU reset cycle

3.8.4 Front-end command transmission

The FECMD is executed by the SIU by activating the fbTEN_N and fbCTRL_N interface lines simultaneously and providing the command on the fbD data bus. If the command starts a read operation on the FEE-SIU interface (i.e. status or data is expected from the FEE), the SIU will change the direction of the bus (FEE to SIU). Following the proper termination of the transaction, or upon receiving the SIU reset signal from the RORC in case of error, the SIU will change the direction of the bus to its default state (SIU to FEE). Consult [3] for more information.

3.8.5 Flow-control

Figure 32 shows the timing diagram of the flow control on both the RORC-DIU and the FEE-SIU interfaces.

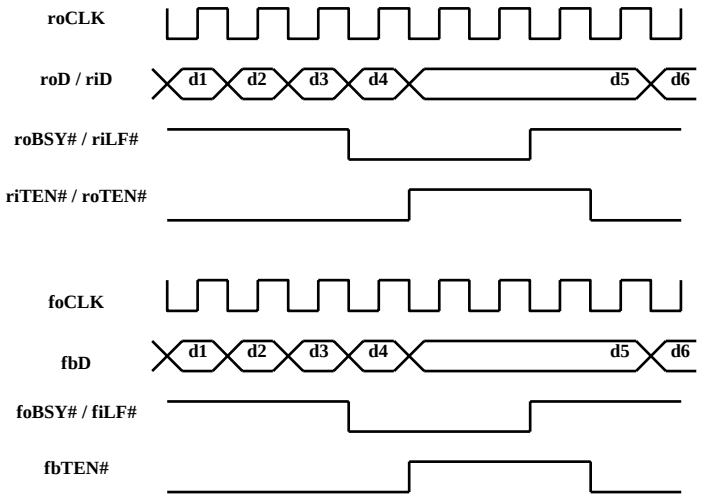


Figure 32 The timing diagram of the flow control

On the RORC-DIU interface, the RORC can use the roBSY_N signal to suspend the transfer from the DIU, when its internal data buffer is almost full. During data download, the DIU uses the riLF_N signal to indicate link full.

On the FEE-SIU interface, the FEE can use the foBSY_N signal if it can not accept more data from the SIU. The link full is indicated by the SIU using the fiLF_N signal.

In all cases, the transfer must be suspended max. one clock cycle after the assertion of the flow control signal. The transfer can resume at any time after the deassertion of the flow control line.

3.8.6 Start of the event data transmission from the FEE to the RORC

Figure 33 shows the timing diagram of the start of the event data transmission from the FEE to the RORC. The event data transmission is initiated by the RORC by sending a RDYRX front-end command (see 3.4.2). This command prepares only the event data transmission executing the following steps: first the RDYRX command is transferred from the SIU to the FEE, then the direction of the front-end bus is changed and a CTSTW is sent back by the SIU to the RORC. The transmission of the event data can only be started, when a read-out command from the trigger system is received by the FEE.

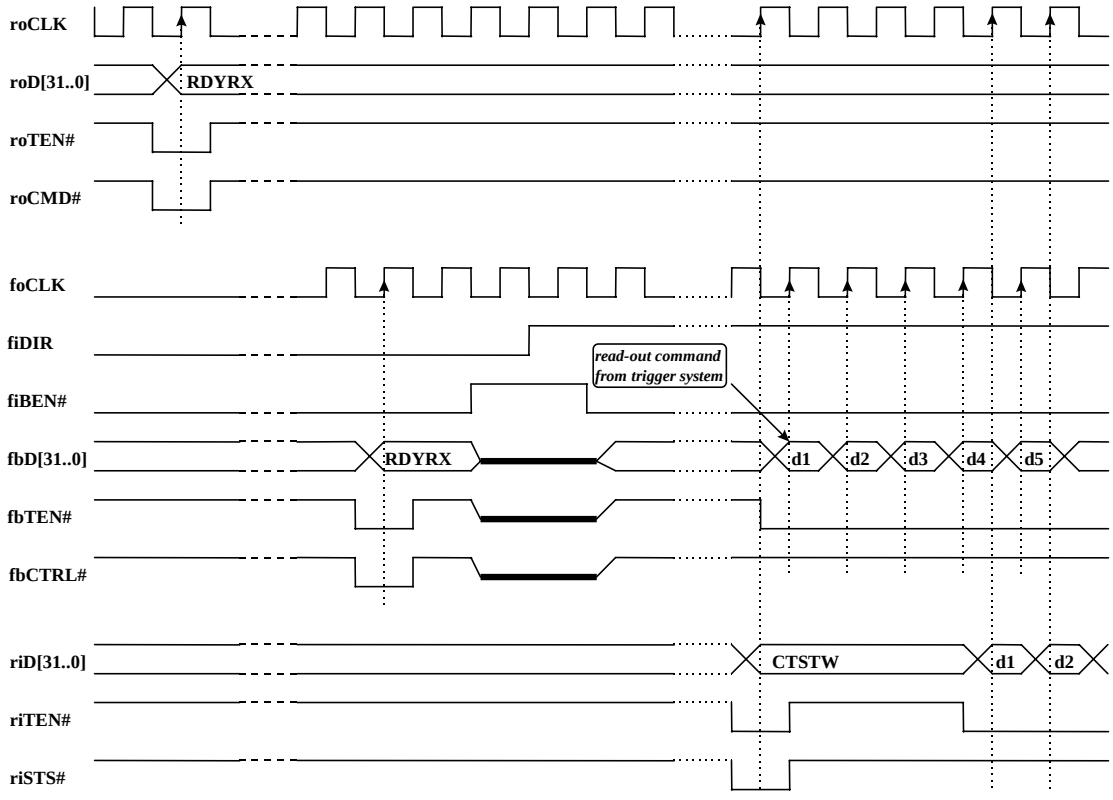


Figure 33 The timing diagram of the start of the event data transmission from the FEE to the RORC

4. Annexes

4.1 Memory bank organisation

The data words can be transmitted through the DDL in data blocks, with a maximum block size of 2 MByte[4] (512 kWord). We propose to use bank organisation for the FEE memories (see Figure 34). This organisation allows to read and write data words sequentially from/to the FEE memories without generating address cycle on the front-end bus.

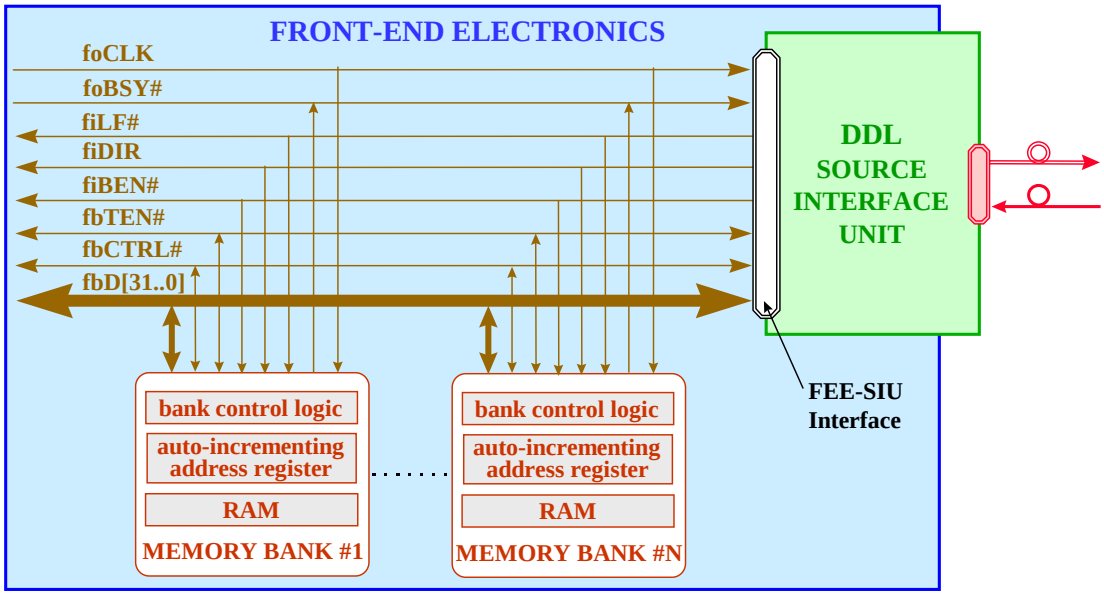


Figure 34 The bank organisation of the FEE memories

Memory bank write operation

A memory bank is selected for write operation, when it receives a STBWR#address command from the RORC. This command opens a data block downloading transaction in the DDL (see 3.6.7). Furthermore, STBWR command prepares the control logic of selected memory bank for the write operation and sets the address register as well. After opening the transaction, data words can be written into the memory bank until it is deselected, receiving a EOBTR command. The address register is automatically incremented after each memory write cycle. If the memory is not fast enough for the write operation, the FEE can use flow control by activating the foBSY_N interface signal at the FEE-SIU interface. This action will activate the riLF_N interface signal at the RORC-DIU interface.

Memory bank read operation

A memory bank is selected for read operation, when it receives a STBRD#address command from the RORC. This command opens a data block read transaction in the DDL (see 3.6.8). Furthermore, STBRD command prepares the control logic of selected memory bank for the read operation and sets the address register as well. After opening the transaction, the memory bank automatically sends data words to the RORC until it is empty or it is deselected. When the memory bank becomes empty, the FEE must transmit to the SIU a FESTW in with the EOB bit is set to '1'. The address register is automatically incremented after each memory read cycle.

[4] **The maximum size is 2 MB – 4 bytes**

4.2 HLT interface

The interface between the DAQ and HLT systems is based on the DDL (Figure 35). Every DAQ RORC can host two DIUs. One of the on-board DIUs can be connected to the front-end electronics (1) while the other one can transfer a copy of all the raw data to the HLT RORC sitting in one of the HLT nodes (2).

Despite this configuration is similar to a DIU - DIU configuration described in section 3.5, the communication between the two partners (DAQ and HLT) is strictly limited to the transfer of raw data blocks. The transaction management (e.g. opening and closing front-end transactions) is controlled by the ALICE Experimental Control System and implemented by the DAQ computers. If the error bit in the DTSTW is set, the HLT RORC will have to read the DIU Interface Status Word (see 3.4.3 and Table 6) in order to discover the source of the problem.

The results of the processing in the HLT system are transferred back to the DAQ system using other DDL links (3). These links work in SIU - DIU configuration, which ensures that the HLT nodes are seen as any other detectors from the DAQ point of view.

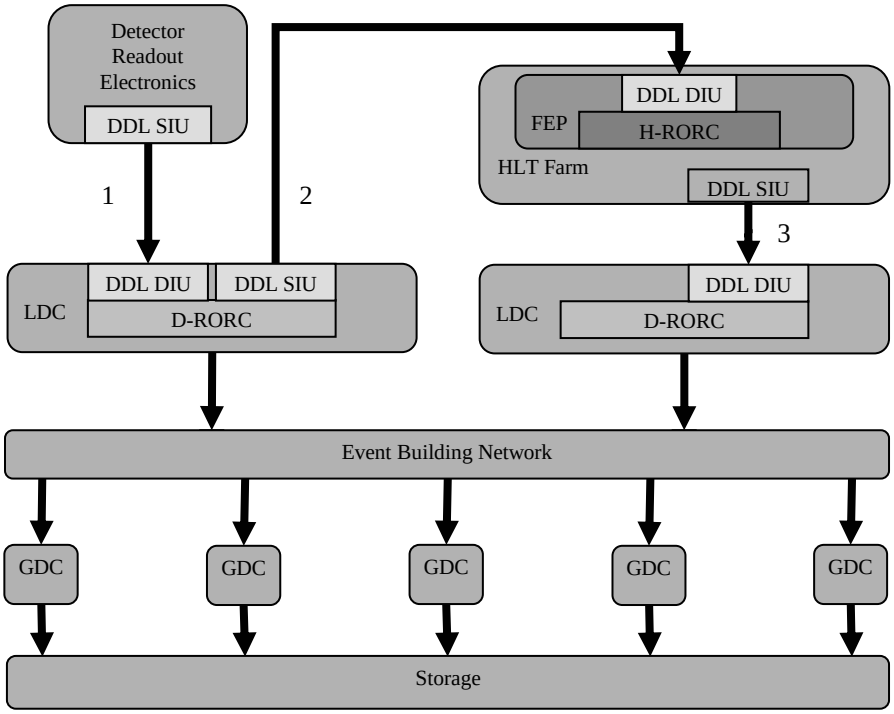


Figure 35 DAQ-HLT dataflow overview

5. Notes