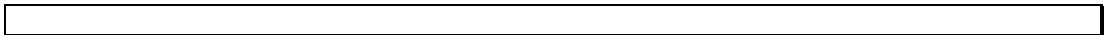


FEC

Front End Control unit for Embedded Slow Control

C. Ljuslin C. Paillard

DRAFT



1. DOCUMENT HISTORY

000322 TIMEOUT in status reg 1 added

000330 TTCRX_READY and RESET_TTCRX added

010820 RELEASE FEC added to control reg 1. Illegal symbol generates status in receive FIFO.

020212 Due to the new TTCRx the clock for the localbus interface has been changed to the PCI clock (33MHz).

020212 The new CERN made optical receivers generate a reset signal in the absence of input signal. The reset has thus been modified to generate a $\sim 0.4\mu\text{s}$ long absence of signal on the serial data outputs.

2. RELATED DOCUMENTS

1. Ccu Communication and Control Unit ASIC for Embedded Slow Control
2. APV6 User Manual
3. PCI 9080 Data Book
4. TTCRx User Manual
5. Lattice ispLSI 8840 Data Sheet
6. I2C documentation
7. LVDS documentation

3. GENERAL

The FEC is the Front End Control module for the slow control link in the CMS tracker. The FEC is implemented as a PMC module, i.e. a piggy-back board with a PCI interface. Since it uses a standard electrical interface it can be used in conjunction with any processor system that implements the PMC specification. In order to avoid electrical interference over the distance to the detector optical links are used. The slow control links of the CMS tracker are implemented as a ring architecture. A complete ring consists of several components, first the FEC which controls the ring, then the optical drivers and receivers that bring the clock and data to the CCU which contains the I2C masters that interface to the DCU, the APV and the PLL. The ring continues through several CCUs until the data is returned to the FEC where the status of an operation is returned

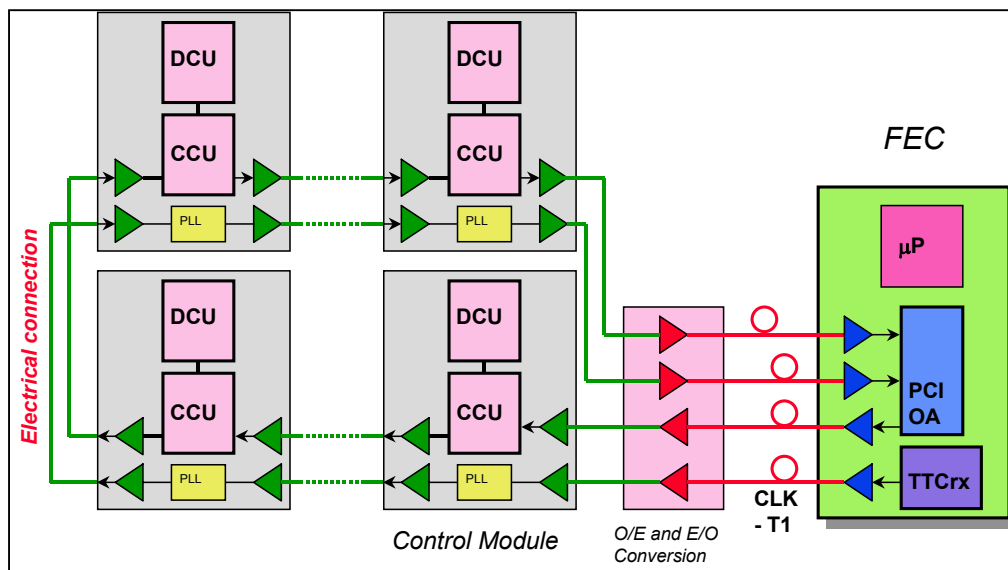


Figure 1 General block diagram

3.1. The FEC

The interface from PCI to the onboard logic is through a PLX9080 chip. The FEC controller is implemented in a Lattice 8040 chip which is a large CPLD. The FEC consists of a transmit FIFO for data to be sent out of the FEC onto the token-ring and a receive FIFO for data coming back from the ring. The 4 to 5 encoding and decoding is done in the FEC. The FEC also generates the CRC internally as data is transmitted, likewise the CRC is checked internally when received. This means that the FEC driver does not have to encode/decode the data neither generate nor check the CRC. There is a third FIFO used for data coming from the ring with destination FEC, this data is returned to the ring through the return FIFO. All three FIFOs can be read and written for testing purpose when the FEC is not ENABLED. When the FEC is ENABLED it is only possible to write into the transmit FIFO and read from the receive FIFO.

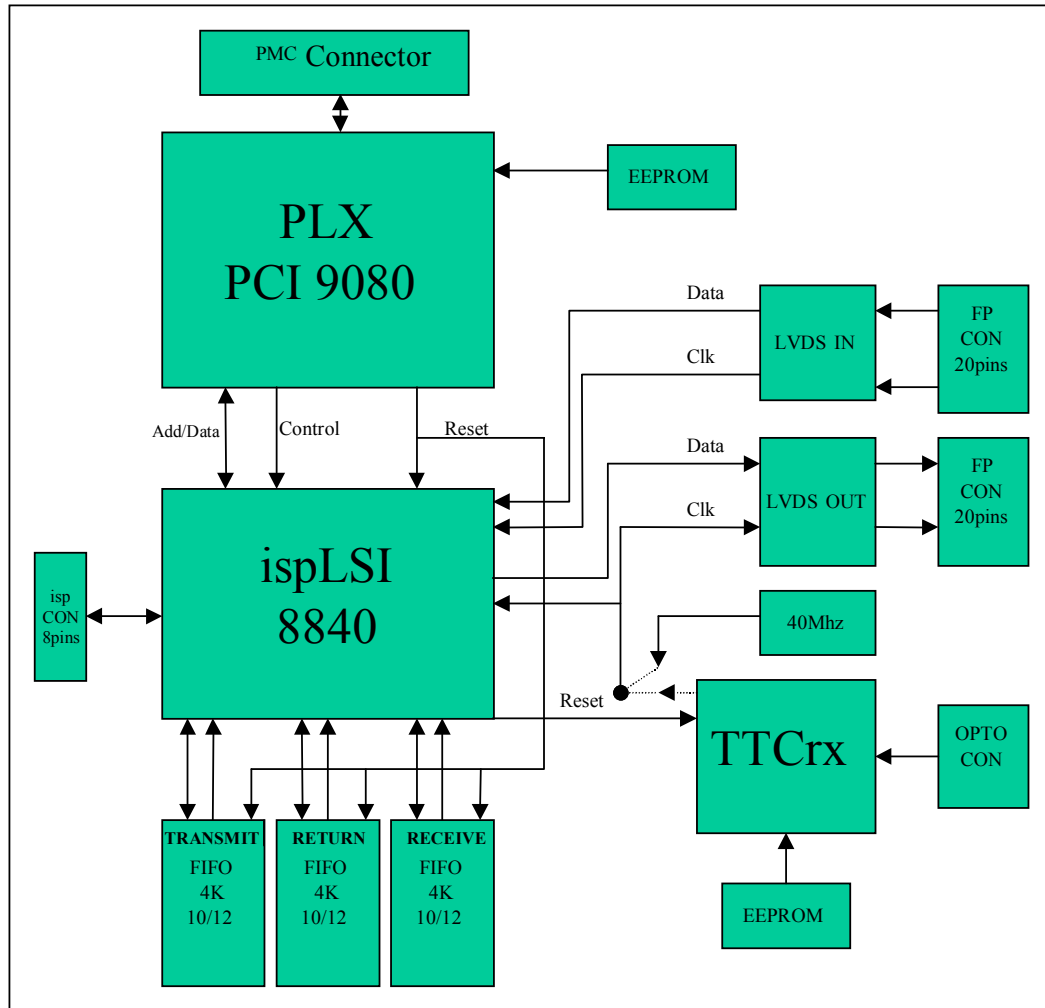


Figure 2 FEC block diagram

3.1.1. PLX PCI 9080

The PCI 9080 chip is an interface chip between the host PCI bus in our case physically implemented as a PMC module, and a local bus. The PCI 9080 is compliant with PCI specification v2.1. There is a choice of configuration possibilities on the local bus side, in our case it is configured as a multiplexed 32bit address/32bit data bus. For the addressing the whole 32bits are decoded. Data is 16bit wide for writing and reading in the FEC. For the moment the FEC only supports single word reads and writes but the PCI 9080 chip also contains all what is necessary to transfer data in DMA mode. The exploitation of this mechanism is however not yet implemented in the Lattice 8040 chip. The default configuration of the internal registers of the PCI 9080 are stored in a serial EEPROM. This configuration can be overwritten from the host computer but care must be taken when changing the parameters.

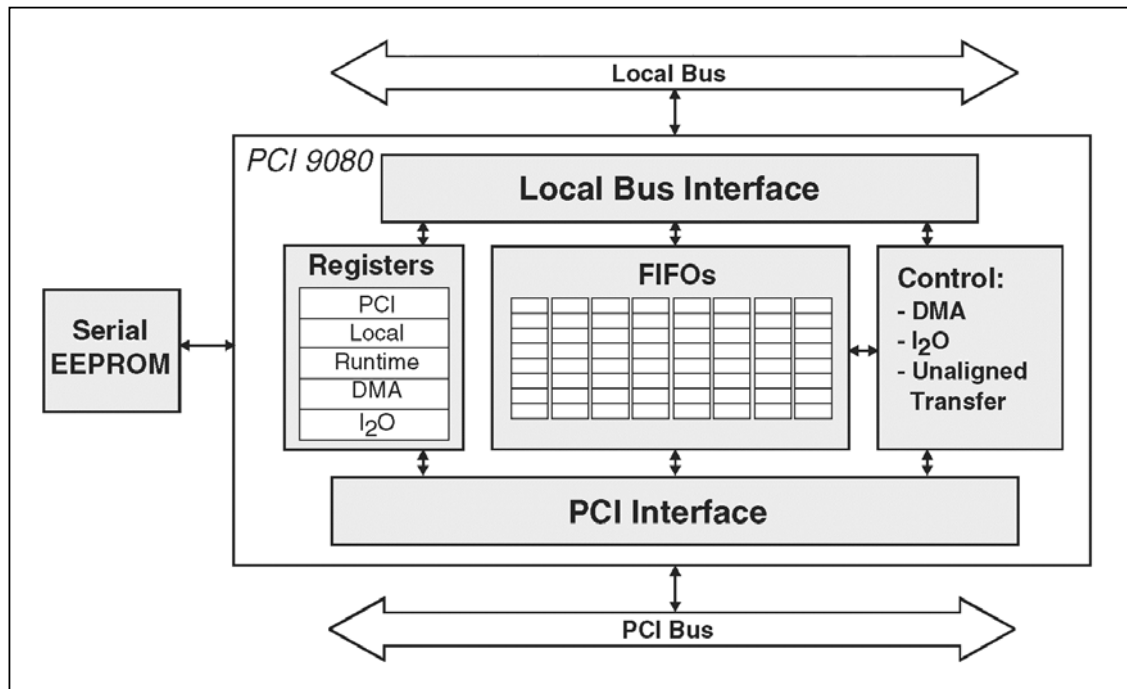


Figure 3 PCI 9080 block diagram

3.1.2. TTCRx

On the FEC card there is also a TTCRx module with its associated fiberoptic receiver. The default configuration for the mode of operation of the TTCRx is stored in a serial EEPROM which is loaded into the TTCRx upon power up or a reset generated from the control register 0 in the FEC. The TTCRx is used to generate the clock for the token ring as well as to insert the trigger information received on the fiber. When a trigger is generated a "hole" is inserted instead of a clockpulse. There is a maximum of 3 consecutive missing clock pulses allowed in the system. Two consecutive missing pulses is used to generate a calibration cycle in the APV. A sequence of missing, present, missing pulses generates a reset in the APV. The whole chain of devices connected to a ring see these missing pulses, but since the ring is a synchronous system, it is transparent to the ring devices. The local bus of the PLX 9080 chip is also driven from this clock. For devices not directly connected to the ring needing a continuous clock, a PLL is inserted to recover the continuous clock and generate trigger pulses instead of the missing pulses. When no optical fiber is present, an onboard crystal oscillator @ 40MHz can be used by changing the jumper position J7 on the FEC.

3.1.3. Addressing of the FEC

The addresses for the various registers and FIFOs are decoded from the 32 bit address on the local bus side of the PLX 9800. Seen from the PCI side these addresses are mapped into the memory space through a register called PCI Base Address Register for Local Space X (PCIBARX). The data words on the local bus are always 16 bits wide in the FEC.

REGISTER	LOCAL_ADDR	PCI_ADDR_OFFSET
CONTROL0	32x0000_0000	16x0000
CONTROL1	32x0000_0004	16x0001
STATUS0	32x0000_0008	16x0002
STATUS1	32x0000_000C	16x0003
TRA-FIFO	32x0000_1xxx	16x0400
RET-FIFO	32x0000_2xxx	16x0800
REC-FIFO	32x0000_3xxx	16x0C00

Figure 4 Addressing of the FEC

3.2. Mode of operation

The normal mode of operation for the FEC is as follows. When an action is to be performed on a unit connected to the ring one first makes sure that the link is initialized and that no pending operation is active. A data frame is then prepared and written byte by byte into the transmit FIFO, the last word in the frame should have bit 15 set to one. When the frame has been written the SEND bit in the control register 0 is toggled. The Fec then waits for a token to be replaced by the the data frame present in the transmit FIFO. Once the token is grabbed the status register 0 indicates TRANSMIT RUNNING. Depending on the ring delay the transmitted data will then start returning to the FEC at which point the status register 0 indicates RECEIVE RUNNING. Data will then be written into the receive FIFO. Normal ring delay is around 8x25x#ofCCU ns so the delays are normally rather short, i.e. in the order of microseconds. Once the whole message is returned the last byte is written together with the status of the frame, i.e. Error, Address seen and Data copied. If errors have been detected in the FEC the status is signalled as CRC error, Illegal sequence or Illegal data as well. In case packets return with the Data copied bit set to 0 that indicates that the addressed channel was busy and the packet should be retransmitted. The retransmission is not done automatically in hardware but should be taken care of in the software. When the last word has been written into the receive FIFO the FEC sends an IRQ to the PCI 8090 which can be fed through to the host computer if this function is enabled in the PCI 8090 chip. If the interrupt function is not enabled it is possible to poll the status register0. Once the IRQ has been seen the CLEAR IRQ has to be sent to control register 1 to indicate that the IRQ has been taken care of. The two status bits mentioned before are active only during the time needed for transmitting or receiving the data frame so if they are permanently on there is a problem somewhere. Together with the IRQ there is an indication in the status register if the frame is a returned frame or a frame destined for the FEC.

3.3. Control registers.

The operation of the FEC is controlled through two control registers.

3.3.1. Control register 0.

The main operation of the FEC is controlled through the control reg0 at address 0. For the time being there are 7 bits used. As more functionality is added more bits could be implemented.

The control register 0 is read-write. The following bits are defined:

15	6	5	4	3	2	1	0
RESET OUT	RESET TTCRX	SELECT IRQ MODE	SELECT SERIAL IN	SEL SERIAL OUT	RESEND	SEND	ENABLE FEC

Figure 5 Defined bits in control register 0

The function of each bit in the control register 0 is explained in the table below.

Bit	Name	Function
15	RESET_OUT	By setting this bit a 0.4 μ s reset pulse is transmitted over the LVDS bus to the connected CCUs and PLLs. If an optical link with the CERN developed receivers I used, the receiver generates the reset pulse during absence of signal on the input. This bit is or-ed with the LRESETO from the PCI8090. The default value is "0" (no reset).
6	RESET_TTCrx	By toggling this bit the TTCRX is reset. After the TTCrx is reset the CCUs and PLLs should be reset after the TTCRX READY bit is set. This bit is or-ed with the LRESETO from the PCI8090. The default value is "0" (no reset).
5	SEL_IRQ_MODE	Selects the IRQ mode. In mode "0" the transmission of data is not interrupted when an IRQ is generated. In mode "1" the transmission of data and tokens is interrupted until the IRQ is cleared. The default value is "0".
4	SEL_SER_IN	Enables one of the two serial data inputs used for redundancy. Writing a "1" in this bit enables SER_IN[1]. The default value is "0" (SER_IN[0] enabled).
3	SEL_SER_OUT	Enables one of the two serial data outputs used for redundancy. The non-enabled output is at a fixed "0" level. Writing a "1" in this bit enables SER_OUT[1]. The default value is "0" (SER_OUT[0] enabled).
2	RESEND	By toggling this bit, the read pointer in the TRANSMIT FIFO is reset and the data present in the TRANSMIT FIFO is retransmitted. The default value is "0".
1	SEND	By toggling this bit the data present in the TRANSMIT FIFO is transmitted. The data is assumed to be logically and syntactically correct since no further checking is undertaken in the FEC. The default value is "0".
0	EN_FEC	Writing a "1" to this bit enables the transmission of data already present in the TRANSMIT FIFO. At the same time WRITE operations from PCI to the RETURN FIFO and the RECEIVE FIFO are disabled like the READ operations from PCI to the TRANSMIT FIFO and the RETURN FIFO. This bit must be set to "1" before or at the same time as the SEND or RESEND bits are set to "1". The default value is "0".

Figure 6 Function of control register 0

3.3.2. Remark on RESET

At power up or booting of the host CPU a PCI reset is generated which causes activation of LRESETO. LRESETO is the master reset for the Lattice 8040 which in its turn generates the two signals RESET_OUT and RESET_TTCRX. To reset the FEC and the FIFOs when the host CPU is up and running the PCI8060 CNTRL register (PCI:6Ch bit 30) should be toggled. This also generates the LRESETO signal. After toggling PCI:6Ch bit 30 the default values from the EPROM are loaded into the PCI8090 which imposes a pause before accessing the FEC again. Any special configurations not included in the default EPROM of the PCI8090 must subsequently be repeated by writing into the corresponding configuration registers. In

order to be sure that the PLLs are properly locked RESET_OUT (CONTROL0[15]) should be toggled only when STATUS0[14] indicates TTC_READY.

3.3.3. Control register 1.

Control reg1 at address1 is used to clear the level sensitive interrupt generated by the FEC and the error indicators in status register1.

The control register 1 is write only since the bits are automatically cleared.The following bits are defined:

2	1	0
RELEASE FEC	CLEAR ERROR BITS	CLEAR INTERRUPT

Figure 7 Defined bits in control register 1

The function of each bit in the control register 1 is explained in the table below

Bit	Name	Function
2	RELEASE FEC	By writing a 1 to this register the FEC generates a token and stops waiting for a return packet. This is intended to be used in case the ring hangs due to redundancy switching for example.
1	CLEAR ERROR BITS	By writing a 1 to this register the bits [2:0] in STATUS REG1 are cleared. The control register is automatically reset after the operation
0	CLEAR INTERRUPT	By writing a 1 to this register the bits PENDING IRQ and DATA TO FEC (bits [13:12]) are cleared as well as bits [6:3] in STATUS REG1. The control register is automatically reset after the operation.

Figure 8 Function of control register 1

3.4. Status registers.

The result of an operation in the FEC can be determined through the status registers.

3.4.1. Status register 0.

The status register0 at address 2 contains 14 bits right now, but more bits might be added in the future if necessary.

The status register is read only. The following bits are defined:

14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TTCRX READY	DATA TO FEC	PENDING IRQ	LINK INITIALIZED	TRANSMIT FIFO EMPTY	TRANSMIT FIFO FULL	TRANSMIT FIFO HALF FULL	RETURN FIFO EMPTY	RETURN FIFO FULL	RETURN FIFO HALF FULL	RECEIVE FIFO EMPTY	RECEIVE FIFO FULL	RECEIVE FIFO HALF FULL	RECEIVE RUNNING	TRANSMIT RUNNING

Figure 9 Status register 0

The function of each bit in the status register 0 is explained in the table below

Bit	Name	Function
14	TTCRX READY	This bit is set when the TTCRX is locked
13	DATA TO FEC	This bit is set when the last byte of an incoming message destined for the FEC is written into the RECEIVE FIFO. Since it is destined to the FEC it is also written into the RETURN FIFO to be circulated back to the source. This makes it possible to distinguish between messages destined to the FEC and messages destined to a CCU and returning to the FEC. The bit is cleared by writing a "1" to CONTROL REGISTER1[0].
12	PENDING IRQ	This bit follows the local interrupt transmitted to the PLX9080. An interrupt is generated when the last byte of an incoming message is written into the RECEIVE FIFO. The interrupt is cleared by writing a "1" to CONTROL REGISTER1[0].
11	LINK INITIALIZED	This bit is "0" until the FEC has transmitted 512 bytes of "IDLE " symbols in order to initialize the optical link and received the first TOKEN back.
10	TRANSMIT FIFO EMPTY	This bit follows the empty flag from the TRANSMIT FIFO but is high active. PCI read operations when empty produce FFFF as data.
9	TRANSMIT FIFO FULL	This bit follows the full flag from the TRANSMIT FIFO but is high active
8	TRANSMIT FIFO HALF FULL	This bit follows the half full flag from the TRANSMIT FIFO but is high active
7	RETURN FIFO EMPTY	This bit follows the empty flag from the RETURN FIFO but is high active. PCI read operations when empty produce FFFF as data.
6	RETURN FIFO FULL	This bit follows the full flag from the RETURN FIFO but is high active.
5	RETURN FIFO HALF FULL	This bit follows the half full flag from the RETURN FIFO but is high active
4	RECEIVE FIFO EMPTY	This bit follows the empty flag from the RECEIVE FIFO but is high active. PCI read operations when empty produce FFFF as data.
3	RECEIVE FIFO FULL	This bit follows the full flag from the RECEIVE FIFO but is high active.
2	RECEIVE FIFO HALF FULL	This bit follows the half full flag from the RECEIVE FIFO but is high active
1	RECEIVE RUNNING	This bit is "1" when data is being written into the RECEIVE FIFO. The status bit goes to "0" between frames and during tokens and idle characters.
0	TRANSMIT RUNNING	This bit is "1" if EN FEC is set and a SEND or RESEND command has been issued. TRANSMIT RUNNING stays "1" until the TRANSMIT FIFO is empty and the EOF has been transmitted. The TRANSMIT FIFO should not be refilled before TRANSMIT RUNNING has gone to "0" unless the writing speed to the TRANSMIT FIFO is guaranteed to exceed the reading

		speed of the FEC.
--	--	-------------------

Figure 10 Function of status register 0

3.4.2. Status register 1.

The status register1 at address 3 contains 6 bits right now, but more bits might be added in the future if necessary.

The status register 1 is read-only. The following bits are defined.

6	5	4	3	2	1	0
TIMEOUT	ERROR	ADDRESS SEEN	DATA COPIED	CRC ERROR	ILLEGAL SEQUENCE	ILLEGAL DATA

Figure 11 Defined bits in status register 1

The function of each bit in the status register 1 is explained in the table below.

Bit	Name	Function
6	TIMEOUT	This bit is set if a TIMEOUT has been detected, i.e. the FEC has not seen a token or data frame during the timeout period which is ~500µs. This bit is cleared by writing a "1" to CONTROL REGISTER 1[1].
5	ERROR	This bit corresponds to the error symbol in the End of Frame that is transmitted after data (TRRR). It indicates that a CRC error or some other error has been detected in a CCU. This bit is cleared together with the IRQ.
4	ADDRESS SEEN	This bit corresponds to the address seen symbol in the End of Frame that is transmitted after data (TRRR). It indicates that the destination has been addressed. This bit is cleared together with the IRQ.
3	DATA COPIED	This bit corresponds to the data copied symbol in the End of Frame that is transmitted after data (TRRR). It indicates that the destination has accepted the frame. This bit is cleared together with the IRQ.
2	CRC ERROR	Indicates that a CRC error has been detected in the FEC. If set, this bit stays on until intentionally cleared. This bit is cleared by writing a "1" to CONTROL REGISTER 1[1].
1	ILLEGAL SEQUENCE	Indicates that an illegal symbol sequence has been detected in the FEC. If set, this bit stays on until intentionally cleared. This bit is cleared by writing a "1" to CONTROL REGISTER 1[1].
0	ILLEGAL DATA	Indicates that an illegal symbol has been detected in the FEC. If set, this bit stays on until intentionally cleared. This bit is cleared by writing a "1" to CONTROL REGISTER 1[1].

Figure 12 Function of status register 1

3.5. Token ring packet Format

Two basic packet types are foreseen, one for signaling availability of the ring and the second to actually carry data.

The network token format is defined as:

SOF [1 B]	EOF [2 B]
-----------------------	-----------------------

Figure 13 Token packet format

The network data packet format is defined below:

SOF [1B]	Destination DAddress [1 B]	Source SAddress [1 B]	Length [1/2 B]	Data [<128/32K B]	CRC-16 [2 B]	EOF [2 B]
--------------------	-------------------------------------------	--------------------------------------	--------------------------	-----------------------------	------------------------	---------------------

Figure 14 Data packet format

The Start of Frame (SOF), End of Frame (EOF), Source and Destination fields are mandatory for all circulating data packets.

The SOF field is defined as the unique “J-H/K” sequence, using the two special characters defined in the “Figure 20 Control characters in network”.

The SOF field is defined as follows:

Symbol	Name	Comment
0	J	Special symbol used for synchronisation
1	H/K	K used to mark token packet H used to mark normal data packet

Figure 15 Frame Header format

The End of Frame (EOF) field consists of two bytes and is defined as follows:

End Delimiter [0.5 B]	Frame Status [1.5 B]
---------------------------------	--------------------------------

Figure 16 EOF definition

The End Delimiter consists of a single “T” character as defined in the Figure 20 Control characters in network. The Frame status is generated by the transmitter as three “R” characters and contains the three following sub-fields:

Symbol	Value	Comment
ER	R/S	Error symbol
AR	R/S	Address recognised
DC	R/S	Data copied

Figure 17 The EOF frame

These symbols are set by the receiving node; they are generated as “R” symbols from the source and modified to “S” by the receiver.

The Length field can be one or two bytes long and gives the length of the data payload, excluding the two length bytes themselves and the CRC field.

When the high bit [bit 7] of the first byte is ‘0’ the length of the field is one byte only and the maximum payload can be 0-127 bytes long. When the high bit is a ‘1’, the length field is two bytes long and the data payload can be 0-32K long.

The Data field is not interpreted by the ring protocol and is used exclusively by the channel adapter for internal addressing and data. This fields are defined explicitly by the functionality of each channel later in this document.

The data portion of the packet is shown below:

CH# [1 B]	TR# [1 B]	Channel Specific Command [(Length-2) B]
---------------------	-----------------------	-----------------------------------------------------

Figure 18 Data portion of packet

Two data bytes are mandatory as payload at the beginning of each data packet:

- a channel number (single byte field), used to identify a device channel within a node
- a transaction number (single byte field with wrap-around) used to assure correct identification of operation within a given channel. This field is always generated by the initiator of a transaction.

For transactions initiated by the FEC, the transaction number should always be in the range 1-255, as the special transaction number 0 is used for Alarms generated by the CCUs.

The CRC-16 field covers the packet content from the Destination address to the end of the Data field. The polynomial used for the CRC-16 calculations is:

$$X^{16} + X^{15} + X^2 + 1$$

3.5.1. Data encoding

Data on the data line are qualified by the rising edge of the clock signal. In addition, the data bits on the data line will be transmitted using 4bit to 5bit encoding, using a NRZI (Non Return to Zero with Invert 1 on change) signaling scheme as follows:

4 bit Binary	Hex Value	5 bit Symbol
0000	0	11110
0001	1	01001
0010	2	10100
0011	3	10101
0100	4	01010
0101	5	01011
0110	6	01110
0111	7	01111
1000	8	10010
1001	9	10011
1010	A	10110
1011	B	10111
1100	C	11010
1101	D	11011
1110	E	11100
1111	F	11101

Figure 19 4B/5B encoding

This encoding scheme has been chosen because it requires very limited hardware resources and can be easily implemented in hardware. The worst case DC unbalance deriving from the usage of this coding is estimated to be about 10%.

The control symbols are defined as follows:

Control Symbol	Code	Comment
Idle	11111	Idle
J	11000	In SOF field
K	10001	in SOF field
H	00100	Special
R	00111	Reset
S	11001	Set
T	01101	Termination

Figure 20 Control characters in network

3.5.2. The NRZI signaling scheme

Basically, this scheme uses a line transition to represent a “1” and no transition to represent a “0”. The 4B/5B symbols used above are made such that there are never more than 3 consecutive zeros in the line, assuring an adequate number of line transitions, to avoid large DC shifts.

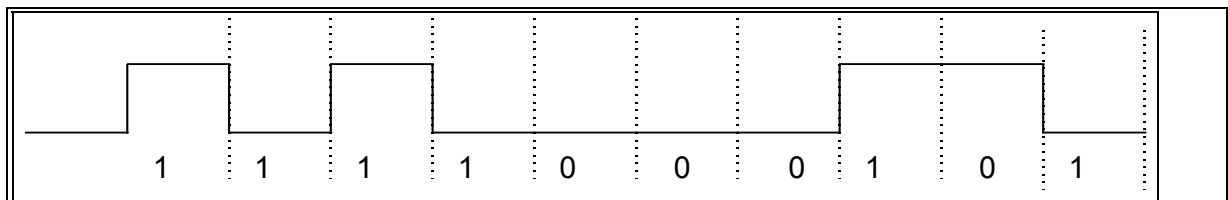


Figure 21 NRZI encoding format

3.6. Transmit FIFO data structure

The organisation of data packets for the TRANSMIT FIFO is shown below. The last word in a message should have bit [15] = 1. Several packets can be written one after the other, provided each has bit [15] set in the last word. The data packets are embedded into a token ring frame by the FEC through adding the control symbols for SOF and EOF. While sending out a frame the CRC is calculated and inserted after the last data byte.

3.6.1. Transmit data structure

BITS [15:8]	BITS [7:0]
8x00	DESTINATION
8x00	SOURCE
8x00	LENGTH
8x00	DATA
8x00	"
8x00	"
8x80	DATA
8x00	DESTINATION
8x00	SOURCE
8x00	LENGTH
8x00	DATA
8x00	"
8x00	"
8x80	DATA

Figure 22 Transmit FIFO data structure

3.7. Receive FIFO data structure

The last word in the receive fifo contains the last byte in bits [7:0] and status in bits [15:8]. Several messages can be stored consecutively in the receive FIFO.

3.7.1. Receive data structure

BITS [15:8]	BITS [7:0]
8x00	DESTINATION
8x00	SOURCE
8x00	LENGTH
8x00	DATA
8x00	"
8x00	"
8x80 + STATUS	DATA
8x00	DESTINATION
8x00	SOURCE
8x00	LENGTH
8x00	DATA
8x00	"
8x00	"
8x80 + STATUS	DATA

Figure 23 Receive FIFO data structure

3.7.2. Received data status

The bits [14:12] should be 011 for a frame that is returning to the FEC, i.e it was originated in the FEC. In a frame destined for the FEC the bits [14:12] should be 000. Bits [11:9] correspond to errors in the data stream detected in the FEC.

If an illegal symbol or an illegal sequence of symbols is encountered, an end of data word (8x80 +STATUS) is immediately written to the receive fifo i.e. the receive fifo will contain either a truncated message or just the status. When the status has been written the FEC waits for a new message.

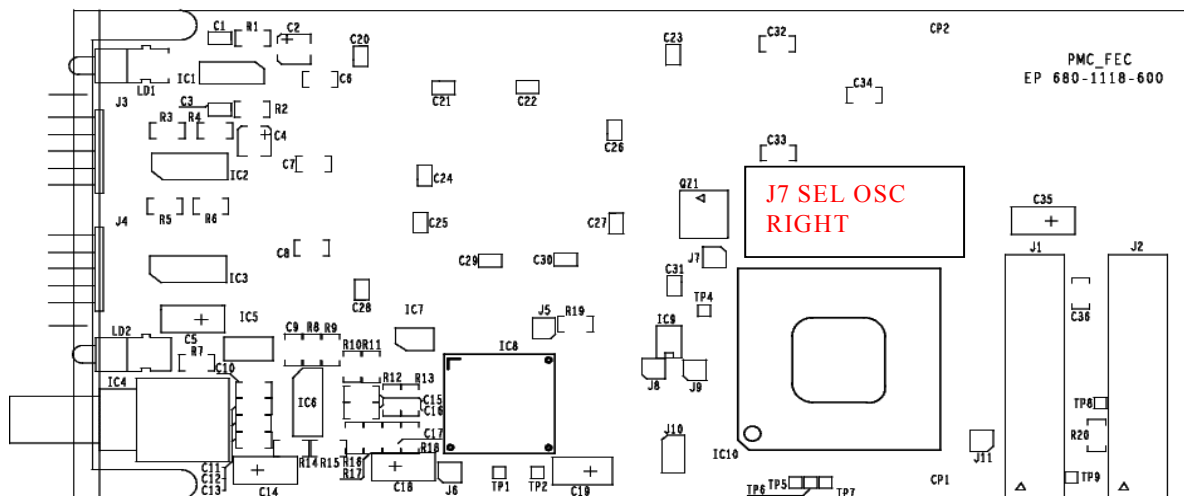
BIT	FUNCTION
15	HIGH (End Of Message)
14	ERROR
13	ADDR SEEN
12	DATA COPIED
11	CRC ERROR
10	ILLEGAL SEQUENCE
9	ILLEGAL DATA
8	NOT USED (LOW)

Figure 24 Receive status

3.8. FEC power consumption

The power consumption of the FEC is ~1.1 A on the +5V supply.

3.9. FEC physical layout



J11	J9	J8	J7	J6	J5
TEST	EECS	EEPROM	SEL OSC	I2C	TOE LATTICE
..	l:	:l	:l
—				—	—

Figure 25 Default jumper configuration

4. LIST OF FIGURES

Figure 1 General block diagram	4
Figure 2 FEC block diagram	5
Figure 3 PCI 9080 block diagram	6
Figure 4 Addressing of the FEC	7
Figure 5 Defined bits in control register 0	8
Figure 6 Function of control register 0	9
Figure 7 Defined bits in control register 1	10
Figure 8 Function of control register 1	10
Figure 9 Status register 011	
Figure 10 Function of status register 0	13
Figure 11 Defined bits in status register 1	13
Figure 12 Function of status register 1	14
Figure 13 Token packet format	14
Figure 14 Data packet format	15
Figure 15 Frame Header format	15
Figure 16 EOF definition	15
Figure 17 The EOF frame	15
Figure 18 Data portion of packet	15
Figure 19 4B/5B encoding	16
Figure 20 Control characters in network	17
Figure 21 NRZI encoding format	17
Figure 22 Transmit FIFO data structure	18
Figure 23 Receive FIFO data structure	18
Figure 24 Receive status	19
Figure 25 Default jumper configuration	20

5.	INDEX	
1.	Document History	2
2.	Related documents	3
3.	General	4
3.1.	The FEC	4
3.1.1.	PLX PCI 9080	5
3.1.2.	TTCRx	6
3.1.3.	Addressing of the FEC	6
3.2.	Mode of operation	7
3.3.	Control registers.	7
3.3.1.	Control register 0.	7
3.3.2.	Remark on RESET	9
3.3.3.	Control register 1.	10
3.4.	Status registers.	10
3.4.1.	Status register 0.	10
3.4.2.	Status register 1.	13
3.5.	Token ring packet Format	14
3.5.1.	Data encoding	16
3.5.2.	The NRZI signaling scheme	17
3.6.	Transmit FIFO data structure	17
3.6.1.	Transmit data structure	18
3.7.	Receive FIFO data structure	18
3.7.1.	Receive data structure	18
3.7.2.	Received data status	19
3.8.	FEC physical layout	19
4.	LIST OF FIGURES	21
5.	INDEX	22