3 ANSWERS TO FREQUENTLY ASKED QUESTIONS

3.1 General questions

Q1) Are reference designs available for the FREEDM?

A1) Yes, there is the Frame Relay Port Card reference design (PMC-1990533) available. This reference design shows an application of the FREEDM-84P672.

Beside this the customer can download from the PMC-Sierra's web site the CABGA TOCTL with FREEDM-32 reference design (PMC-1980942). This is an add-in card that connects the FREEDM-32 to a host processor and packet memory.

Q2) Are Programmer's guides available for the FREEDM?

A2) Yes, the FREEDM-32P672, FREEDM-32A672, FREEDM-84P672 and FREEDM-84A672 Programmer's guides are available on the PMC-Sierra's web site www.pmc-sierra.com. The document numbers are PMC-1990545, PMC-1990639, PMC-1990715 and PMC-1990716 respectively.

Q3) Do the FREEDM devices support industrial temperature range?

A3) Yes, the FREEDM-32P672, FREEDM-32A672, FREEDM-84P672 and FREEDM-84A672 devices support industrial temperature range (-40C to 85C).

Q4) Are there any software drivers available for the FREEDM?

A4) Yes, there are software drivers available for the FREEDM-32P672, FREEDM-32A672, FREEDM-84P672 and FREEDM-84A672. The software drivers can be downloaded after registering at PMC-Sierra's web site.

Q5) Are IBIS models available for the FREEDM?

A5) Yes, IBIS models for the above mentioned FREEDM devices are available. The IBIS models can be downloaded after registering at PMC-Sierra's Web site.

3.2 Questions regarding the FREEDM-32P672

Q6) What is the total power consumption of the FREEDM-32P672 for different operating modes?

A6) Table 1 shows different operating modes of the FREEDM-32P672. Please note that for 3.3V power supply the power should be calculated with the following equation: $V_{DD} \times V_{DD} \times C_{LOAD} \times f$. The power will vary in each customer's system.

Table 1 Operating modes of the FREEDM-32P672

Mode	FREEDM	Core current for $V_{DD2.5}$ = 2.7V (mA) **	Ptotal (W)
32 Channelized E1 links	33 MHz PCI	220	0.548
3 Unchannelized HSSI at 52 MHz each	33 MHz PCI	300	0.778
3 Unchannelized HSSI at 52 MHz each	66 MHz PCI	330	0.826

** Core current is highest when the $V_{DD2.5}$ power supply is at its upper limit of 2.7V

Q7) Has the FREEDM-32P672 PCI retry cycle been validated for conformance to the PCI Rev. 2.1 Specification?

A7) Yes, PMC-Sierra has completed a PCI feature test plan for the FREEDM-32P672. The device operates as stated in the PCI Specification Rev. 2.1 with regards to target retry.

For the target retry test, the test observation is the following:"the GPIC responds correctly to a Target Retry including the re-initiation of the transaction. The read/write data equals the original data. The PCI command/status register is correct. No error bits are set."

Q8) What is the effect of giving a software reset to the FREEDM-32P672 after initializing all descriptor base address registers and all free and ready queue registers (start, end, write, read). Do these registers return the last programmed values or they will go to default states?

A8) A software reset will reset all the normal mode register bits to default values as described in the FREEDM data sheet. It will not affect the values of the PCI configuration registers – they will remain in the last programmed state.

Q9) In some configurations of our design framers may be depopulated, leaving the RCLK and RDAT inputs on the FREEDM floating. In PMC-Sierra's datasheet is not stated they need to be pulled up or down if unused. Do we have to put pull-ups on them?

A9) It is always a good practice to terminate the unused input signals properly. It is recommended to pull both RCLK and RDAT to high if unused.

Q10) The width of the internal registers in the FREEDM-32P672 is 32 bits. Is this little endian or big endian mode?

A10) The FREEDM-32P672 always runs in 32 bit little endian mode as a target. As a master, it is able to swizzle packet data using the LENDIAN bit in the GPIC control register.

Q11) What is the allowable range of the PCI clock for the FREEDM-32P672?

A11) The PCI clock can run from 25 MHz to 66 MHz, with PCICLK >=1/2 SBICLK for the FREEDM-32P672. This is stated in the Table 37 (PCI Interface) of the device data sheet.

Q12) When we set CBI Memory Base Address Register (0x10h) to "0000 1000" we can't see the register values inside the FREEDM-32P672. When we change the value to "0000 2000", we can see them. Why is this happening?

A12) Earlier FREEDM devices only required 4 kbytes of CBI address space while the FREEDM-32P672 requires 8 kbytes of the CBI address space.

Q13) How is the power sequencing resolved for the FREEDM-32P672?

A13) 3.3V needs to be powered up before 2.5V to ensure that 3.3V is always equal or higher than 2.5V and to prevent an external device from driving the FREEDM-32P672 I/O before 3.3V is supplied to the FREEDM-32P672.

Q14) What are the major differences between the FREEDM-32 PM 7364 and the FREEDM-32P672?

A14)
- The FREEDM-32P672 is a 329 pin PBGA packaged device and the FREEDM-8 is either a 256 pin SBGA or a 272 pin PBGA packaged device. The pin connections between the two parts are different.
- The FREEDM-32P672 is a 329 pin PBGA packaged device and the FREEDM-8 is either a 256 pin SBGA or a 272 pin PBGA packaged device. The pin connections between the two parts are different.
- Besides serial clock and data signals, the FREEDM-32P672 supports H-MVIP interface.
- The FREEDM-32P672 can run up to 66 MHz PCI bus while the FREEDM-32 only runs up to 33 MHz PCI bus.
- The FREEDM-32/8 software can be reused on new FREEDM-PCI devices with the following changes:
    - Increase of the addressable HDLC channels from 128 to 672 in Transmit and Receive Descriptors.
    - Increase of maximum number of addressable descriptors from 16 kbytes to 32 kbytes.
    - Increase of the addressable number of partial packet blocks from 512 to 2048.
- Changes of the registers and the descriptors are described in the Appendix of the FREEDM-32P672 Programmer's Guide (PMC-990545).

Q15) What is the relationship between Status[5:0] in RPD and Status[1:0] in RPDR in the FREEDM-32P672? For example, when Status[1:0] in RPDR is set to 11, what is Status[5:0] set to ?

A15) When the STATUS[1:0] bits of the RPDR are set to 01, one of the bits of the Status[5:0] of RPD must have been set to 1, because the bits of the Status[5:0] are wired OR together to indicate whether a packet has been received successfully or not.

When the STATUS[1:0] bits of the RPDR are set to 11, it indicates the packet has been returned as the RAWMAX limit is reached. The RAXMAX limit is only used for transparent channel, i.e., no HDLC processing will be performed. When

the STATUS[1:0] bits are set to 11, it is not related to the Status[5:0] field of the RPD at all.

3.3 Questions regarding the FREEDM-32A672

Q16) We would like to use the FREEDM-32A672 for passing non-HDLC data between the T1 links and the Any-PHY bus. All the data should arrive on the T1 link transparently. Will we be able to set transmit and receive DELIN bits to 0 and the MODE[2:0] bits of the RCAS and TCAS to 000?

A16) Yes, for transparent channel configuration, DELIN bit should be set to 0, and for unchannelized application, MODE[2:0] should be set to 000. Please note that if the links are framed T1, the link clock must be gapped out at the framing bit so that the FREEDM will not sample the framing bit as part of the data flow. For unframed T1, there is no need to gap out the link clock at the framing bit and the FREEDM-32A672 will take all 193 bits as part of the data flow.

Q17) The application we intend to support with the FREEDM-32A672 device is unchannelized packet-over-DS3 using PPP encapsulation or Frame Relay. The data sheet for PM7381 FREEDM-32A672 discusses High Level Data Link Control Protocol and describes the bit-destuffing operation. Could you detail this operation with a diagram of the protocol and a description of how it is used in our application?

A17) Please find details of the HDLC protocol in International Standard ISO/IEC 3309 “Information technology – Telecommunications and information exchange between systems – High level data link control (HDLC) procedures – Frame Structure”. The bit stuffing information could be found in 4.5.1 “Synchronous transmission” of the above document.

3.3 Questions regarding the FREEDM-84P672

Q18) According to the datasheet of the FREEDM-84P672 I/O pin’s tolerant high voltage level range is 2.0 – 2.5V (“V_{T+}” parameter) for Schmitt triggered input. Is this the range for only Schmitt triggered input? If an input signal is not Schmitt triggered, what voltage is the maximum level?

A18) Yes, the range of 2.0 to 2.5V is only applicable to Schmitt triggered input. All input pins with the exception of the PCI inputs are Schmitt triggered. The PCI inputs should use the range for PCI Input High Voltage $V_{IH.}$ The range of V_{IH} is $0.5*V_{DD3.3}$ to $V_{DD3.3}+0.5$.

Q19) Why was the CBI register space increased from 4 kbytes to 8 kbytes for the FREEDM-84P672?

A19) The reason the register space increased is due to an increase in the number of test mode registers. They are located in the extra 4 kbytes.

While test mode registers may not be used in normal operation, access to them may be useful during debug of a FREEDM design.

Q20) In the FREEDM-84P672 the transmit descriptors (TDs) define a More (M) field, which can be used to chain multiple packets within the list of TDs. Why is such an arrangement not followed in RPDs?

A20) TDs are put together by the user while RPDs are built by the FREEDM-84P672.

Q21) The normal mode registers in the FREEDM-84P672 all have 32 bits. In most of the registers only 4 or 5 bits are used. The remaining bits are unused. Why?

A21) Our chips use megacell blocks where one megacell block may be used in a number of different PMC products. Some megacell blocks have 16 or 8-bit registers because they are used in PMC products which have only 16 or 8-bit micro ports. Such blocks are used in the FREEDM resulting in many registers having a large number of unused bits.

3.4 Questions regarding the FREEDM-84A672

Q22) Is it possible to connect the TEMUX PM8315 to the FREEDM-84A672 (using SBI or any other method) and set the FREEDM-84A672 up so that it will take each T1 frame and pack all 193 bits into one Any-PHY packet?

A22) Yes, the SBI bus can be configured for unframed T1 tributaries so that all 193 bits of a T1 contain valid data. The related register bits are in the RCAS, TCAS, SBI Extract and SBI Insert blocks.

Q23) Our design includes the SPECTRA-622 PM5313, the TEMUX and the FREEDM-84A672. Why does the FREEDM-84A672 have 16 bit processor interface but it is in a 32 bit boundary? How do we connect the 16 bit processor data bus to the 32 bit Motorola 860 processor in big endian mode?

A23) The data bus can be connected to the lower 16 data bits of the microprocessor. As the register offsets are DWORD aligned, A[0] and A[1] of the FREEDM are not used. You can connect A[2] of the FREEDM-84A672 to A[0] of the

microprocessor if you would like to use only the least significant 10 bits of the address bus.

Q24) There are 5 possible errors on the receive side that can be reported for an errored packet status on the last byte of an Any-PHY transfer of the FREEDM-84A672. Is it possible to have more than one error reported for a given errored packet?

A24) Yes, it is possible that more than one error is reported for a given packet. However, the Receive HDLC processor does mask certain errors if a more serious error occurs, as explained in section 9.5 of the datasheet (Section 9.4 of the FREEDM-32A672 datasheet).

Q25) Is there a line loopback for the three independent timed bidirectional clock/data links? The documentation states that the line loopback will connect the transmit and receive of the SBI, but it does not mention the clock/data links.

A25) Yes, line loopback is available for the serial data links. The following information can be found in the Issue 2 and later of the FREEDM-84A672 data sheet:

Register 0x014: FREEDM-84A672 Master Line Loopback

Bit 15 to Bit 3 R/W Reserved Default 0000h

Bit 2 R/W LLBEN[2] Default 0

Bit 1 R/W LLBEN[1] Default 0

Bit 0 R/W LLBEN[0] Default 0

This register controls line loopback for the three serial data links (enabled when SPEn_EN is low).

Q26) When there is a receive channel FIFO overrun within the partial packet RAM, what happens to the existing data within the channel FIFO?

A26) For example, there is a channel FIFO that completely contains 2 packets (A and B) and is in the process of receiving a third packet (C). Before packet C is completely received, the channel FIFO overruns. The FREEDM-84A672 will be able to transfer packets A and B to the host correctly. For packet C, once the FREEDM detects overrun, it will discard the rest of the packet. Therefore a portion of packet C will be transferred to the host with the status on RXDATA[7:0] if STATEN bit is set to 1 in register 0x580 RAPI control. When a FIFO overrun occurs, the RFOVRI bit in register 0x8 will be set to 1. You can also enable the overrun interrupt by setting the RFOVRE bit in register 0x4 to 1.

Note that, when a FIFO overrun occurs, the contents of the FIFO just prior to the overrun is not corrupted in any way. Any complete packets received prior to the overrun will be transferred to the Any-PHY interface as normal.

Q27) What is the significance of packet ID field in both transmit and receive packet structures?

A27) The packet ID field is not used by the FREEDM at all, it is simply a field that the host uses for debugging purposes.

4 NOTES

5 CONTACTING PMC-SIERRA, INC.

PMC-Sierra, Inc.
105-8555 Baxter Place Burnaby, BC
Canada V5A 4V7

Tel: (604) 415-6000

Fax: (604) 415-6200

Document Information: document@pmc-sierra.com
Corporate Information: info@pmc-sierra.com
Technical Support: apps@pmc-sierra.com
(604) 415-4533
Web Site: http://www.pmc-sierra.com

PMC- 2001119 Issue date: August 2000