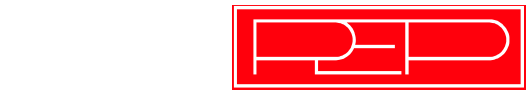
CP321

Power PC-based CPU Board
for CompactPCI Applications

Manual ID: 24977, Rev. Index 02
July 2003



The product described in this manual is in compliance with all applied CE standards.

Revision History

Manual/Product Title:	CP321		
Manual ID Number:	24977		
Rev. Index	**Brief Description of Changes**	**Board Index**	**Date of Issue**
01	Initial Issue	00	July 2002
02	Replace of BootstrapLoader chapter with NetBootLoader chapter	00	July 2003

Imprint

Kontron Modular Computers GmbH may be contacted via the following:

MAILING ADDRESS

Kontron Modular Computers GmbH
Sudetenstrasse 7
D - 87600 Kaufbeuren Germany

TELEPHONE AND EMAIL

+49 (0) 800-SALESKONTRON
sales@kontron.com

For further information about other Kontron Modular Computers' products, please visit our Internet web site: **www.kontron.com**

Copyright

Disclaimer:

Kontron Modular Computers GmbH rejects any liability for the correctnesss and completeness of this manual as well as its suitability for any particular purpose.

Table of Contents

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Annex A

Annex B

Annex C

Annex D

Annex E

List of Tables

List of Figures

This page was intentionally left blank.

Proprietary Note

This document contains information proprietary to Kontron Modular Computers GmbH. It may not be copied or transmitted by any means, disclosed to others, or stored in any retrieval system or media without the prior written consent of Kontron Modular Computers GmbH or one of its authorized agents.

The information contained in this document is, to the best of our knowledge, entirely correct. However, Kontron Modular Computers GmbH cannot accept liability for any inaccuracies or the consequences thereof, or for any liability arising from the use or application of any circuit, product, or example shown in this document.

Kontron Modular Computers GmbH reserves the right to change, modify, or improve this document or the product described herein, as seen fit by Kontron Modular Computers GmbH without further notice.

Trademarks

Kontron Modular Computers GmbH, the *PEP* logo and, if occurring in this manual, “CXM” are trade marks owned by Kontron Modular Computers GmbH, Kaufbeuren (Germany). In addition, this document may include names, company logos and trademarks, which are registered trademarks and, therefore, proprietary to their respective owners.

Explanation of Symbols

CE Conformity

This symbol indicates that the product described in this manual is in compliance with all applied CE standards. Please refer also to the section “Applied Standards” in this manual.

Caution, Electric Shock!

This symbol and title warn of hazards due to electrical shocks (> 60V) when touching products or parts of them. Failure to observe the precautions indicated and/or prescribed by the law may endanger your life/health and/or result in damage to your material.

Please refer also to the section “High Voltage Safety Instructions” on the following page.

Warning, ESD Sensitive Device!

This symbol and title inform that electronic boards and their components are sensitive to static electricity. Therefore, care must be taken during all handling operations and inspections of this product, in order to ensure product integrity at at all times.

Please read also the section “Special Handling and Unpacking Instructions” on the following page.

Warning!

This symbol and title emphasize points which, if not fully understood and taken into consideration by the reader, may endanger your health and/or result in damage to your material.

Note...

This symbol and title emphasize aspects the reader should read through carefully for his or her own advantage.

For Your Safety

Your new Kontron product was developed and tested carefully to provide all features necessary to ensure its compliance with electrical safety requirements. It was also designed for a long fault-free life. However, the life expectancy of your product can be drastically reduced by improper treatment during unpacking and installation. Therefore, in the interest of your own safety and of the correct operation of your new Kontron product, you are requested to conform with the following guidelines.

High Voltage Safety Instructions

Warning!

All operations on this device must be carried out by sufficiently skilled personnel only.

Caution, Electric Shock!

Before installing your new Kontron product into a system always ensure that your mains power is switched off. This applies also to the installation of piggybacks.

Serious electrical shock hazards can exist during all installation, repair and maintenance operations with this product. Therefore, always unplug the power cable and any other cables which provide external voltages before performing work.

Special Handling and Unpacking Instructions

ESD Sensitive Device!

Electronic boards and their components are sensitive to static electricity. Therefore, care must be taken during all handling operations and inspections of this product, in order to ensure product integrity at all times.

- Do not handle this product out of its protective enclosure while it is not used for operational purposes unless it is otherwise protected.
- Whenever possible, unpack or pack this product only at EOS/ESD safe work stations. Where a safe work station is not guaranteed, it is important for the user to be electrically discharged before touching the product with his/her hands or tools. This is most easily done by touching a metal part of your system housing.
- It is particularly important to observe standard anti-static precautions when changing piggybacks, ROM devices, jumper settings etc. If the product contains batteries for RTC or memory back-up, ensure that the board is not placed on conductive surfaces, including anti-static plastics or sponges. They can cause short circuits and damage the batteries or conductive circuits on the board.

General Instructions on Usage

- In order to maintain Kontron's product warranty, this product must not be altered or modified in any way. Changes or modifications to the device, which are not explicitly approved by Kontron Modular Computers GmbH and described in this manual or received from Kontron's Technical Support as a special handling instruction, will void your warranty.
- This device should only be installed in or connected to systems that fulfill all necessary technical and specific environmental requirements. This applies also to the operational temperature range of the specific board version, which must not be exceeded. If batteries are present their temperature restrictions must be taken into account.
- In performing all necessary installation and application operations, please follow only the instructions supplied by the present manual.
- Keep all the original packaging material for future storage or warranty shipments. If it is necessary to store or ship the board please re-pack it as nearly as possible in the manner in which it was delivered.
- Special care is necessary when handling or unpacking the product. Please, consult the special handling and unpacking instruction on the previous page of this manual.

Two Year Warranty

Kontron Modular Computers GmbH grants the original purchaser of Kontron's products a ***TWO YEAR LIMITED HARDWARE WARRANTY*** as described in the following. However, no other warranties that may be granted or implied by anyone on behalf of Kontron are valid unless the consumer has the express written consent of Kontron Modular Computers GmbH.

Kontron Modular Computers GmbH warrants their own products, excluding software, to be free from manufacturing and material defects for a period of 24 consecutive months from the date of purchase. This warranty is not transferable nor extendible to cover any other users or long-term storage of the product. It does not cover products which have been modified, altered or repaired by any other party than Kontron Modular Computers GmbH or their authorized agents. Furthermore, any product which has been, or is suspected of being damaged as a result of negligence, improper use, incorrect handling, servicing or maintenance, or which has been damaged as a result of excessive current/voltage or temperature, or which has had its serial number(s), any other markings or parts thereof altered, defaced or removed will also be excluded from this warranty.

If the customer's eligibility for warranty has not been voided, in the event of any claim, he may return the product at the earliest possible convenience to the original place of purchase, together with a copy of the original document of purchase, a full description of the application the product is used on and a description of the defect. Pack the product in such a way as to ensure safe transportation (see our safety instructions).

Kontron provides for repair or replacement of any part, assembly or sub-assembly at their own discretion, or to refund the original cost of purchase, if appropriate. In the event of repair, refunding or replacement of any part, the ownership of the removed or replaced parts reverts to Kontron Modular Computers GmbH, and the remaining part of the original guarantee, or any new guarantee to cover the repaired or replaced items, will be transferred to cover the new or repaired items. Any extensions to the original guarantee are considered gestures of goodwill, and will be defined in the "Repair Report" issued by Kontron with the repaired or replaced item.

Kontron Modular Computers GmbH will not accept liability for any further claims resulting directly or indirectly from any warranty claim, other than the above specified repair, replacement or refunding. In particular, all claims for damage to any system or process in which the product was employed, or any loss incurred as a result of the product not functioning at any given time, are excluded. The extent of Kontron Modular Computers GmbH liability to the customer shall not exceed the original purchase price of the item for which the claim exists.

Kontron Modular Computers GmbH issues no warranty or representation, either explicit or implicit, with respect to its products' reliability, fitness, quality, marketability or ability to fulfil any particular application or purpose. As a result, the products are sold "as is," and the responsibility to ensure their suitability for any given task remains that of the purchaser. In no event will Kontron be liable for direct, indirect or consequential damages resulting from the use of our hardware or software products, or documentation, even if Kontron were advised of the possibility of such claims prior to the purchase of the product or during any period since the date of its purchase.

Please remember that no Kontron Modular Computers GmbH employee, dealer or agent is authorized to make any modification or addition to the above specified terms, either verbally or in any other form, written or electronically transmitted, without the company's consent.

This page was intentionally left blank.

Chapter 1

Introduction

This page was intentionally left blank.

1. Introduction

1.1 System Overview

The CompactPCI board described in this manual operates with the PCI bus architecture to support additional I/O and memory-mapped devices as required by various industrial applications. For detailed information concerning the CompactPCI standard, please consult the complete Peripheral Component Interconnect (PCI) and CompactPCI Specifications. For further information regarding these standards and their use, visit the homepage of the *PCI Industrial Computer Manufacturers Group (PICMG)*.

Many system relevant CompactPCI features that are specific to Kontron Modular Computers CompactPCI systems may be found described in the Kontron CompactPCI System Manual. Please refer to the section “Related Publications” at the end of this chapter for the relevant ordering information.

The CompactPCI System Manual includes the following information:

- Common information that is applicable to all system components, such as safety information, warranty conditions, standard connector pinouts etc.
- All the information necessary to combine Kontron racks, boards, backplanes, power supply units and peripheral devices in a customized CompactPCI system, as well as configuration examples.
- Data on rack dimensions and configurations as well as information on mechanical and electrical rack characteristics.
- Information on the distinctive features of Kontron CompactPCI boards, such as functionality, hot swap capability. In addition, an overview is given for all existing Kontron CompactPCI boards.
- Generic information on the Kontron CompactPCI backplanes, such as the slot assignment, PCB form factor, distinctive features, clocks, power supply connectors and signalling environment, as well as an overview of the Kontron CompactPCI standard backplane family.
- Generic information on the Kontron CompactPCI power supply units, such as the input/output characteristics, redundant operation and distinctive features, as well as an overview of the Kontron CompactPCI standard power supply unit family.

1.2 Product Overview

The Kontron Modular Computers' single-height PowerPC CPU board CP321 is a comprehensive computing platform which brings together the latest advances in computing technology in a board designed for maximum performance, flexibility, and versatility within a rugged compact format.

The design centered on realizing a board which addresses the need for increased computing capacity while at the same time reducing the size and number of system components in order to reduce space requirements and optimize power dissipation.

The CP321 is based on the MPC8245, a highly integrated microprocessor containing a PowerPC MPC603e core with the initial version operating at 330 MHz and having a Floating Point Unit (FPU). One of the prime advantages of utilizing the established and proven MPC603e core is the associated broad infrastructure of support that has built up around it. All of the noteworthy third-party software tool vendors provide tools for the MPC8245.

The CP321 employs an OS-independent boot loader that enables the loading of any operating system available for the PowerPC. This boot loader makes an update of the Flash contents and automatically downloads from Flash to SDRAM before booting the OS. For performance reasons the OS and user programs are started from the SDRAM.

To satisfy increased demands for expansion flexibility, the CP321 design incorporates both local on-board PCI bus expansion as well as the standard external CPCI bus interfacing via CPCI backplanes. The local on-board PCI bus is extended to a special PCI expansion connector which currently allows the cascading of two additional modules which can either be carrier boards for other types of modules (e.g. PMC modules) or modules providing additional I/O capability. These features enable, for example, the connection of the widest range of system I/O components such as various field busses, a second Fast Ethernet, and Ultra 2 SCSI, to name just a few. A complete range of expansion possibilities is thus made available to the user by the CP321.

1.3 Board Overview

1.3.1 Board Introduction

The CP321 is a 3U CPCI CPU board featuring a powerful CPU (number cruncher). The design is based on the new highly integrated Motorola PowerPC processor MPC8245, which integrates a PCI interface and various peripherals inside one Chip.

Standard memory configurations of up to 256 MB of SDRAM are available. Flash memory for integrating the initial bootloader and ROMable operating systems are provided . Additionally, NVSRAM and Disk On Chip (by M-Systems) can be placed on two DIL sockets for memory expansion purposes.

The CP321 is also able to communicate with the environment through a Fast Ethernet interface and two serial interfaces at the front side of the board. One of the serial interfaces is a RS232 full modem interface while the other is replaceable with a RS232/RS485 port. These UARTS support baud rates up to 1.5 Mbps and are software compatible with the 16550 UART from National Semiconductor. They contain 128 Byte Transmit and 128 Byte Receive FIFO's for increasing the CPU availability for other operations.

The Ethernet is realized with the Intel 82559ER with full duplex support at both 10/100 Mbps possible. This Fast Ethernet controller with an integrated 10/100 Mbps physical layer device is the foremost solution for PCI board LAN designs. It combines low power consumption with a small package design which is ideal for power and space constrained environments.

Anticipating the CP321's use in data critical applications, the memory data path contains a selectable in-line ECC controller which can provide SDRAM single-bit error correct or double-bit error detect.

For mass data transmission a dual channel DMA controller is provided. It can be programmed directly or through the use of descriptor chains located in memory. Data can thus be moved from PCI to memory or vice versa, memory to memory, or PCI to PCI.

The MPC8245 supports processor control and visibility through the JTAG/COP (common on-chip processor) interface that is available on the CP321. Utilizing third party tools, the developer can access and control the processor. It also has standard IEEE 1149.1a-1993 compliant boundary scan capability.

Utilizing the local on-board PCI expansion connector, the CP321 supports up to 2 mezzanine modules. Currently there is a carrier board available for PMC modules, the CP320-IO1, which can carry a single PMC module. Given the wide range of PMC modules now available, this feature affords the user a very wide range of options with the possibility of low-cost system expansion without an additional PCI bridge or using the expansion capability of the backplane for ruggedized design applications such as the field of transportation.

Used as the system controller in a normal CPCI system, the CP321 supports the full range of expansion capabilities as provided via the external CPCI bus.

Board-Specific Information

The CP321 is a CPCI PowerPC-based single-board computer specifically designed for use in highly integrated platforms with solid mechanical interfacing for a wide range of industrial and transportation environment applications.

Some of the CP321's outstanding features are:

- 330 MHz PowerPC MPC8245 Kahlua II (603E core with an integrated FPU, combined with PCI interface and memory controller)
- 33 MHz CPCI bus (32-bit internal, 64-bit external) (32-bit Rear I/O version on request)
- 16 kB data cache
- 16 kB instruction cache
- up to 256 MB SDRAM (132MHz) with optional ECC support
- up to 8 MB onboard Flash
- one EEPROM for the system and one user EEPROM (8192 x 8)
- on-board interfaces:
- Fast Ethernet interface
- two RS232 serial I/O's, or one RS232 plus one RS485 opto-isolated (ESD protected and EMI compliant)
- memory expansion sockets for Flash memory, M-Systems' DiskOnChip, NVSRAM, or EPROM
- onboard PCI bus with expansion connector
- four counter/timers
- programmable watchdog timer
- real-time clock
- coding switch
- temperature sensing
- front panel LED status indicators
- debug interface, JTAG/COP
- double and triple-width versions via PMC carriers
- compliance with CPCI Specification PICMG 2.0 R 3.0
- operating system: VxWorks, Linux etc.

1.4 Optional Modules

1.4.1 CP320-TR1 Optoisolated RS485

This transition module provides a single, optoisolated RS485 interface for the second serial interface. It connects to CP321 via the CON3 connector and replaces the RS232 (Serial) interface. With this device installed it is only possible to have a 4HP board. See Appendix B for more details.

1.4.2 CP320-TR2 Optoisolated RS232

This transition module provides a single, optoisolated RS232 interface for the second serial interface. It connects to CP321 via the CON3 connector and replaces the onboard RS232 (Serial) interface. With this device installed it is only possible to have a 4HP board. See Appendix C for more details.

1.4.3 CP320-IO1

This module is a specially designed PMC carrier board for the CP321 system. The PCI signals are routed through the CON11 PCI Expansion Connector on the CP321 main board thus eliminating the need for a separate CPCI backplane connector. Up to two CP320-IO1 Modules can be cascaded allowing for two PMC modules to be added to a CP321 system. See Appendix A for more details.

1.4.4 CP-RIO3-01

Designed for use with a CP321 32-bit rear I/O variant and a backplane with system slot rear I/O capability, this module provides rear I/O interfacing to the two standard RS232 serial interfaces and the Fast Ethernet interface. In this configuration only the rear I/O interface is operational.

1.4.5 PMC-HDD1

The PMC-HDD1 module in conjunction with the CP320-IO1 module provides a cost-effective way to add a mass storage device. It is designed for use with one 2.5” IDE hard disk drive.

1.5 System Relevant Information

The following system relevant information is general in nature but should still be considered when developing applications using the CP321.

Table 1-1: System Relevant Information

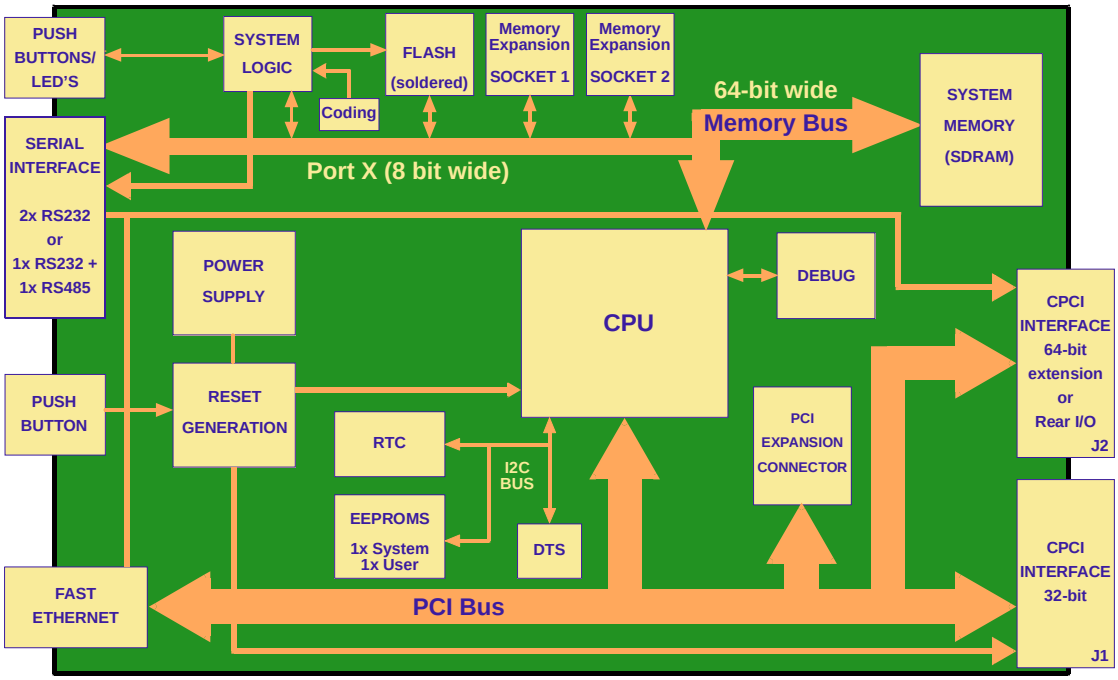
SUBJECT	INFORMATION
System Configuration	A CP321-based system is made up of at least one system controller (the CP321) and up to 7 other I/O boards can be located within one system.
Master/Slave Functionality	The CP321 can operate only as a system controller.
Board Location in the System	The CP321 board must be installed in the system controller slot of a CPCI back-plane.
Hot-Swap Compatibility	The CP321 supports all necessary signals to allow other peripheral boards to be removed or added with power on. The individual clocks for each slot and access to or interrupt on the backplane ENUM# signal are compliant to the PICMG 2.1 Hot-Swap specification.
Hardware Requirements	The CP321 can be installed in any CompactPCI 3U or 6U rack.
Operating Systems	The CP321 can operate under the following operating systems: VxWorks® Linux

1.6 Board Diagrams

The following diagrams provide additional information concerning board functionality and component layout.

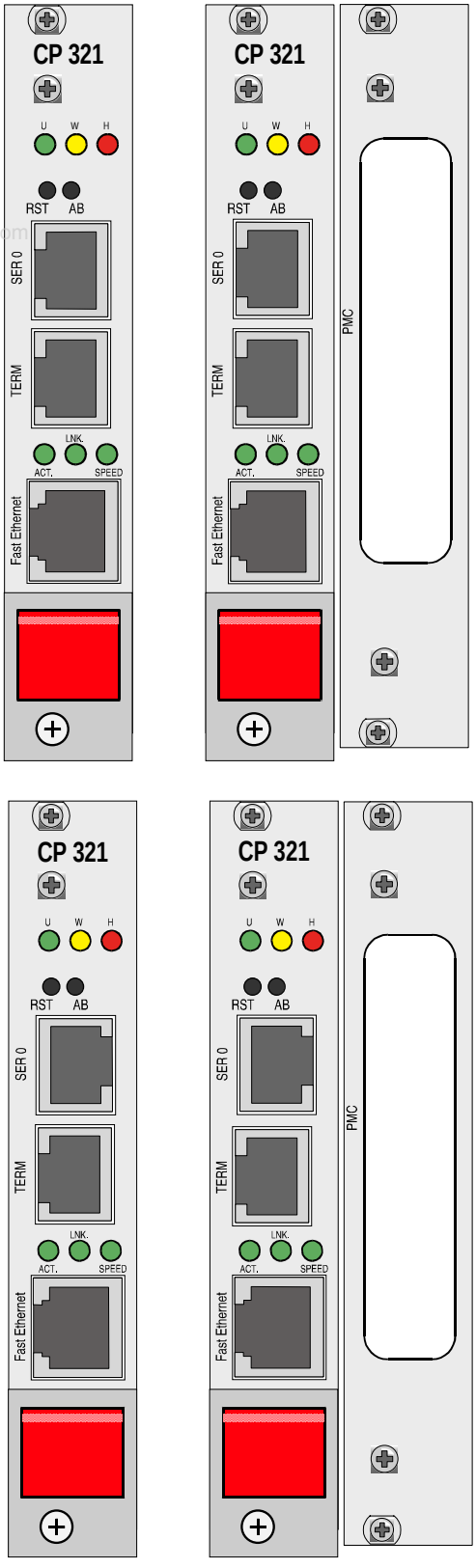
1.6.1 Functional Block Diagram

Figure 1-1: CP321 Functional Block Diagram



1.6.2 Frontpanels

Figure 1-2: CP321 Frontpanels



LEGEND:

Upper Figures

Left standard CP321

Right standard CP321 with a CP320-IO1 Module

Lower Figures

Left CP321 with a CP320-TR1/TR2 RS485 Optoisolation Module

Right CP321 with a CP320-TR1/TR2 and a CP320-IO1 Module

(Note the different orientation of the SER 0 connector.)

LED's

U	User	(green = LED1G)
W	Watchdog	(yellow = LED1Y)
H	Halt	(red = LED1R)
ACT	Active	(green = LED2AC)
LNK	Link	(green = LED2LN)
SPEED	Speed	(green = LED2SP)

Switches

RST	Reset
AB	Abort

1.6.3 Board Layouts

Figure 1-3: CP321 Board (Front View)

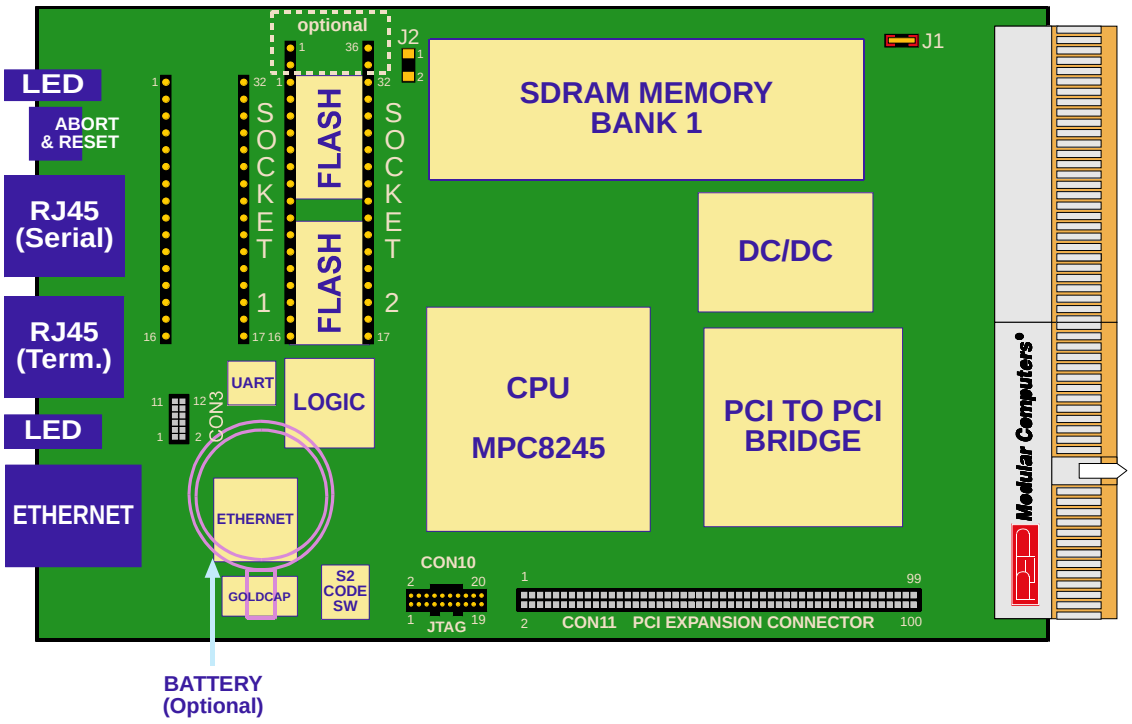
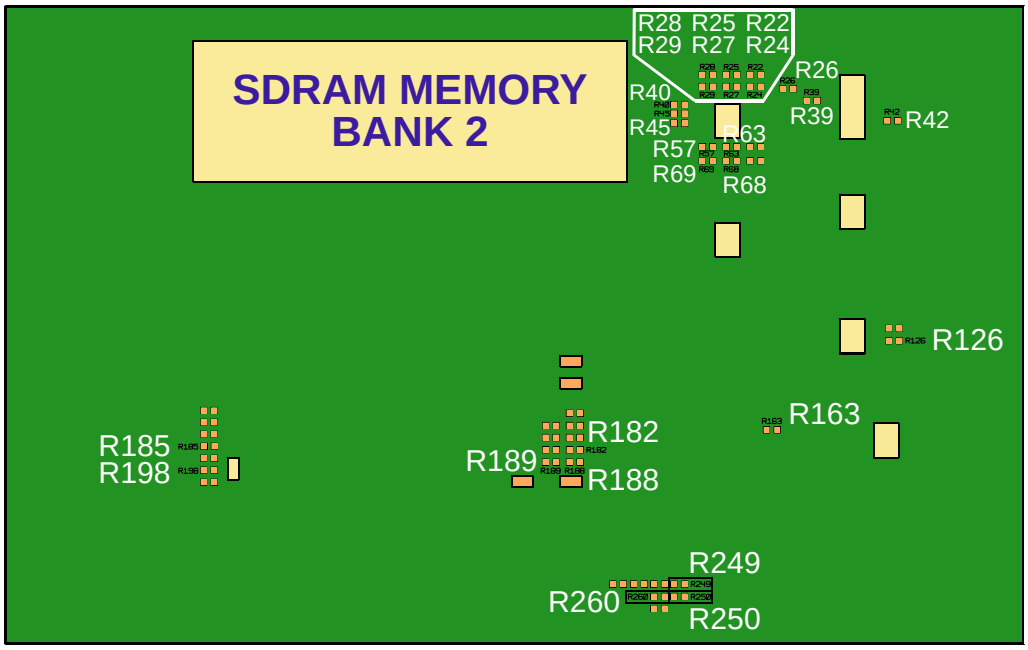


Figure 1-4: CP321 Board (Reverse View)



1.7 Technical Specifications

Table 1-2: CP321 Main Specifications

	CP321	Specifications
Processor and Related	Processor	Motorola MPC8245 with integrated PCI interface, 330MHz
	CPCI Interface	In accordance with PICMG 2.0 R 3.0
	Main Memory	Up to 256 MB of onboard SDRAM with ECC support available as standard
	Cache Structure	16K, 32 byte line, 4-way set associative instruction cache, and 16K, 32 byte line, 4-way set associative data cache
	Watchdog	Watchdog generates: Exception Condition / Reset, or NMI (software configurable)
	RTC	Realtime clock backed up using GoldCap with the data retention being about 5 days (optionally, a backup battery is available)
Peripheral Memory	Soldered, Onboard	
	Flash	8 MB for bootloader and ROMable OS (512 kB pages)
	EEPROM (I^2C Bus)	One System EEPROM, and one User EEPROM (4096 x 8) (User EEPROM write protection possible)
	Memory Expansion Sockets 1 and 2	
	SRAM	Up to 512 kB NV SRAM on the 32-pin DIL600 socket (2MB on request)
	EPROM	Up to 512 kB per memory expansion socket
	Flash	Up to 512 kB per memory expansion socket
	Flash Disk	DiskOnChip from M-Systems (refer to current data sheet of M-Systems for types available)
External Interfaces	Ethernet	10baseT / 100BaseTX via RJ45 connector
	Serial Ports	Standard: two RS-232 ports (TERM and SER 0), or Optionally: one RS232 port (TERM), and one RS485 optoisolated port (SER 0) All ports (standard or optional) use RJ45 connectors.
Internal Interfaces	PCI Expansion	Internal PCI bus extension for use with PMC carrier board: CP320-IO1 Uses a single Samtec SMT Board-to-Board connector, 100-pin; order number: FLE - 15 - 01 - G - DV
	Memory Expansion Sockets	Standard: two sockets (32-pin, DIL600) for Flash and SRAM extension Optional: one 32-pin DIL socket and one 36-pin DIL socket (See Peripheral Memory above for devices that can be used with these sockets.)
	Serial I/F Expansion	12-pin connector for extending UART B signals to optional external interfaces (i.e. the CP320-TR1, RS485 optoisolated serial interface and the CP320-TR2, RS232)
	Debug Interface	JTAG/COP interface for programming and testing purposes (Connector type: SAMTEC FTSH-110-01-L-DV-K)
Indicators / Switches	LED's	3 LED's for indicating system status (two user programmable, one fixed); 3 LED's indicating the status of the Fast Ethernet link
	Switches, Front Panel	Two, non-latching, push-button type switches for resetting or halting the system
	Switch, Coding	Freely selectable,16 position, rotary coding switch, the position of which is read out from the coding switch register

Table 1-2: CP321 Main Specifications (Continued)

	CP321	Specifications
General	Mechanical Conformance	Conforms with IEEE 1101.10
	PowerConsumption	5V, 1.1 Amp 3.3V, 0.84 Amp
	Temperature Range	0°Cto+70°CStandard -25°Cto+75°CE1 -40°Cto+85°CE2
	Humidity	0% to 95% non-condensing
	Dimensions	100 mm x 160 mm, single-height Eurocard
	Board Weight	CP321:233grams CP302-IO1:90grams CP320-TR-1:15grams CP320-TR-2:15grams
Software	Operating System Support	Initial boot loader with capability to load VxWorks operating system andLinux
Options	CP320-TR1	Transition module providing RS485 optoisolated serial interface (Replaces standard RS232 (SER 0) interface)
	CP320-TR2	Transition module providing RS232 optoisolated serial interface (Replaces onboard RS232 (SER 0) interface)
	CP320-IO1	Cascadable PMC carrier board for one PMC module
	PMC-HDD1	IDE hard drive module (in conjunction with CP320-IO1)
	Other	Via PCI expansion connector other cascadable I/O boards possible

1.8 Applied Standards

1.8.1 CE Compliance

The *Kontron Modular Computers'* CompactPCI systems comply with the requirements of the following CE-relevant standards:

- EmissionEN50081-1
- ImmissionEN50082-2
- Electrical SafetyEN60950

1.8.2 Mechanical Compliance

- Mechanical DimensionsIEEE 1101.10

1.8.3 Environmental Tests

- VibrationIEC68-2-6
  Random Vibration, BroadbandIEC68-2-64 (3U boards)
- Permanent ShockIEC68-2-29
- Single ShockIEC68-2-27

1.9 Related Publications

1.9.1 CompactPCI Systems/Boards

- CompactPCI Specification, V. 2.0, Rev. 3.0
- *PEP Modular Computers* CompactPCI System Manual, ID 19954

1.9.2 PMC Add-on Modules/Carriers

- Draft Standard for a Common Mezzanine Card Family, P1386/Draft 2.0
- Draft Standard Physical and Environment Layers for PCI Mezzanine Cards, P1386.1/Draft 2.0

Chapter 2

Functional Description

This page was intentionally left blank.

2. Functional Description

The following chapters provide more detailed board information covering the following: board general information, memory, board interfaces, special functions, and options.

2.1 General Information

The CP321 is based on the Motorola PowerPC processor MPC8245 which integrates a large number of peripherals, such as a FPU PCI interface, PCI arbiter, Interrupt Controller, Memory and DMA Controller and multiple Timers.

Important features of the CP321 are as follows:

603e Core:

- CPU speed is 330 MHz.
- high performance, superscalar 603e core
- 627 Dhrystone (2.1) MIPS
- integer unit (IU), floating point unit (FPU) (user enabled or disabled), load/store unit (LSU), system register unit (SRU), and a branch processing unit (BPU)
- 16 kB instruction cache
- 16 kB data cache
- lockable L1 cache - entire cache or on a per-way basis
- dynamic power management
- I²C controller with full master/slave support

Memory Interface:

- programmable timing support for SDRAM
  (The CP321 uses SDRAM at 132 MHz)
- high bandwidth bus (64-bit data bus) to SDRAM
- 2 memory banks with up to 128 MB each (64, 128 or 256 Mbit memory devices)
- supports 32, 64, 128 and 256 MB SDRAM
- contiguous memory mapping
- 8-bit ROM interface
- write buffering for PCI and processor accesses
- supports ECC
- SDRAM data path buffer
- low voltage transistor-to-transistor logic (LVTTL)
- Port X: 8-bit general-purpose I/O port using ROM controller interface with address strobe

32-bit PCI Interface:

- operates up to 33 MHz
- PCI Specification Revision 2.1 compatible
- universal board (3.3V or 5V signaling on CPCI)
- support for PCI-locked accesses to memory
- support for accesses to all PCI address spaces
- selectable big or little-endian operation (default on the CP321 is big-endian)
- store gathering of processor-to-PCI write and PCI-to-memory write accesses
- memory prefetching of PCI read accesses
- selectable hardware-enforced coherency

- PCI bus arbitration unit (five request/grant pairs)

PCI agent mode capability:

- address translation unit
- internal configuration registers accessible from PCI
- two-channel integrated DMA controller
- supports direct mode or chaining mode (automatic linking of DMA transfers)
- supports scatter gathering - read or write discontinuous memory
- interrupt on completed segment, chain, and error
- local-to-local memory
- PCI-to-PCI memory
- PCI-to-local memory
- local-to-PCI memory

Message Unit:

- I²O message controller
- two door-bell registers
- in-bound and out-bound messaging registers

Embedded Programmable Interrupt Controller (EPIC):

- five hardware interrupts (IRQs) or 16 serial interrupts
- four programmable timers

Programmable Memory and PCI Bus Output Drivers

Debug Features:

- watchpoint monitor
- address attribute and PCI attribute signals
- JTAG/COP - common onboard processor for in-circuit hardware debugging
- performance monitor

2.2 Memory

2.2.1 System Memory (SDRAM)

The main memory of the CP321 consists of 32, 64, 128 or 256 MB of SDRAM soldered onto the board for mechanical stability, provides ECC support with a maximum memory speed of 132 MHz.

2.2.2 Flash

Four or eight megabyte of soldered Flash memory accommodates the bootstrap loader software and can be used to store a ROMable operating system and user data. This Flash memory is 8-bit wide and windowed with window sizes of 512 kB.

2.2.3 EEPROM's

Two, 64-kbit serial EEPROM's are provided, organized 8192 x 8. One EEPROM is for system purposes; the other is for the user. Both EEPROM's can be write protected. These EEPROM's are connected to the I2C bus provided by the MPC8245.

2.2.4 Memory Expansion Sockets (DIL600)

The CP321 provides two, 32-pin DIL sockets on which to place Flash, SRAM, non-volatile SRAM, or other DIL600 devices on the board. Access to this memory is controlled by the on-board logic.

The following devices may be added to the CP321 via the 32-pin DIL600 socket:

- standard EPROM devices;
- standard Flash memory of up to 512 kB (e.g. the AMD29F010 and AMD29F040);
- the NV SRAM from Dallas Semiconductor;

These devices are available in the temperature range -40°C to +85°C for the industrial environment and guarantee a minimum data retention of 10 years (e.g. DS1250Y-100).

In addition, as an option, socket 2 also comes with 36 pins. This can be done to provide the possibility of using 1 MB and 2 MB NVSRAM from Dallas Semiconductor (DS1265/70). These devices can then be accessed in pages of 512 kB.

- DiskOnChip 2000 Flash memory.

  This type of Flash memory from M-Systems comes in versions with two different height profiles: low profile or high profile. The low profile modules can be used with all variants of the CP321 with their options. The high profile modules can only be used in conjunction with the CP320-IO1. If the CP320-TR1/TR2 is also installed with the CP320-IO1, then only one module can be installed and only on socket 2.

  Refer to the current M-Systems data sheets for types available.

To prevent these devices from dislodging from their sockets due to shock or vibration it is possible to secure them using a wire strap.

2.3 Board Interfaces

2.3.1 CPCI Interface and Pinout

The CPCI interface is based on the specification PICMG 2.0 R 3.0, 10/1/99.

The CP321 is available with one of two different versions of the CPCI interface:

64-bit / 33 MHz system controller interface (standard)

32-bit / 33 MHz system controller interface with REAR-IO functionality (All board interfaces are connected to the CPCI J2 connector

Tables showing the pinout of the CPCI connectors J1 and J2 appear on the following pages.

Figure 2-1: CPCI Connectors J1/J2

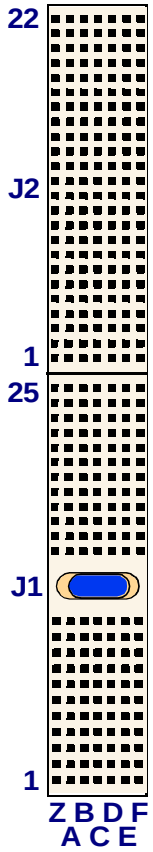


Table 2-1: CompactPCI Bus Connector J1

Pin	Row A	Row B	Row C	Row D	Row E	Row F
25	5V	REQ64#	ENUM#	3.3V	5V	GND
24	AD[1]	5V	V(I/O)	AD[0]	ACK64#	GND
23	3.3V	AD[4]	AD[3]	5V	AD[2]	GND
22	AD[7]	GND	3.3V	AD[6]	AD[5]	GND
21	3.3V	AD[9]	AD[8]	M66EN	C/BE[0]#	GND
20	AD[12]	GND	V(I/O)	AD[11]	AD[10]	GND
19	3.3V	AD[15]	AD[14]	GND	AD[13]	GND
18	SERR#	GND	3.3V	PAR	C/BE[1]#	GND
17	3.3V	SDONE	SBO#	GND	PERR#	GND
16	DEVSEL#	GND	V(I/O)	STOP#	LOCK#	GND
15	3.3V	FRAME#	IRDY#	GND	TRDY#	GND
12-14	Key Area					
11	AD[18]	AD[17]	AD[16]	GND	C/BE[2]#	GND
10	AD[21]	GND	3.3V	AD[20]	AD[19]	GND
9	C/BE[3]#	IDSEL	AD[23]	GND	AD[22]	GND
8	AD[26]	GND	V(I/O)	AD[25]	AD[24]	GND
7	AD[30]	AD[29]	AD[28]	GND	AD[27]	GND
6	REQ#	GND	3.3V	CLK	AD[31]	GND
5	BRSVP1A5	BRSVP1B5	RST#	GND	GNT#	GND
4	NC	GND	V(I/O)	INTP	INTS	GND
3	INTA#	INTB#	INTC#	5V	INTD#	GND
2	TCK	5V	TMS	TDO	TDI	GND
1	5V	-12V	TRST#	+12V	5V	GND

Table 2-2: CompactPCI Bus Connector J2 (64-bit version)

Pin	Row A	Row B	Row C	Row D	Row E	Row F
22	N/C	N/C	N/C	N/C	N/C	GND
21	CLK6	GND	RSV	RSV	RSV	GND
20	CLK5	GND	RSV	GND	RSV	GND
19	GND	GND	RSV	RSV	RSV	GND
18	BRSVP2A18	BRSVP2B18	BRSVP2C18	GND	BRSVP2E18	GND
17	BRSVP2A17	GND	PRST#	REQ6#	GNT6#	GND
16	BRSVP2A16	BRSVP2B16	DEG#	GND	BRSVP2E16	GND
15	BRSVP2A15	GND	FAL#	REQ5#	GNT5#	GND
14	AD[35]	AD[34]	AD[33]	GND	AD[32]	GND
13	AD[38]	GND	V(I/O)	AD[37]	AD[36]	GND
12	AD[42]	AD[41]	AD[40]	GND	AD[39]	GND
11	AD[45]	GND	V(I/O)	AD[44]	AD[43]	GND
10	AD[49]	AD[48]	AD[47]	GND	AD[46]	GND
9	AD[52]	GND	V(I/O)	AD[51]	AD[50]	GND
8	AD[56]	AD[55]	AD[54]	GND	AD[53]	GND
7	AD[59]	GND	V(I/O)	AD[58]	AD[57]	GND
6	AD[63]	AD[62]	AD[61]	GND	AD[60]	GND
5	C/BE[5]#	GND	V(I/O)	C/BE[4]#	PAR64	GND
4	V(I/O)	BRSVP2B4	C/BE[7]#	GND	C/BE[6]#	GND
3	CLK4	GND	GNT3#	REQ4#	GNT4#	GND
2	CLK2	CLK3	SYSEN#	GNT2#	REQ3#	GND
1	CLK1	GND	REQ1#	GNT1#	REQ2#	GND

Table 2-3: CompactPCI Bus Connector J2 (32-bit version)

Pin	Row A	Row B	Row C	Row D	Row E	Row F
22	N/C	N/C	N/C	N/C	N/C	GND
21	CLK6	GND	Eth. TD-	Eth. RD-	Eth. RD+	GND
20	CLK5	GND	Eth. TD+	GND	RSV	GND
19	GND	GND	RSV	RSV	RSV	GND
18	LED1	RSV	RSV	GND	RSV	GND
17	Eth. Speed LED	N/C	PRST# (Push button)	REQ6#	GNT6#	GND
16	Eth. Act. LED	LED (watchdog active)	DEG#	GND	RSV	GND
15	Eth. Link LED	N/C	FAL#	REQ5#	GNT5#	GND
14	RS485_DE	UART_B DSR	UART_B RTS	RSV	UART_B CTS	GND
13	UART_B RX	N/C	RSV	UART_B DTR	UART_B CD	GND
12	UART_A DSR	UART_A RTS	UART_A CTS	RSV	UART_B TX	GND
11	UART_A DTR	GND	RSV	UART_A CD	RS485_RE	GND
10	RSV	N/C	UART_A TX	RSV	UART_A RX	GND
9	RSV	N/C	RSV	RSV	RSV	GND
8	RSV	N/C	RSV	GND	RSV	GND
7	RSV	N/C	RSV	RSV	RSV	GND
6	RSV	N/C	RSV	RSV	RSV	GND
5	C/BE[5]#	GND	RSV	C/BE[4]#	RSV	GND
4	V(I/O)	LED2	C/BE[7]#	GND	C/BE[6]#	GND
3	CLK4	GND	GNT3#	REQ4#	GNT4#	GND
2	CLK2	CLK3	SYSEN#	GNT2#	REQ3#	GND
1	CLK1	GND	REQ1#	GNT1#	REQ2#	GND

Note...

The signal IPMI_PWR (on J1) is routed to 3.3V as it was defined in the preliminary version of the CompactPCI specification PICMG 2.0 R3.0.

2.3.2 Ethernet Connector and Pinout

The Ethernet interface is based on a PCI device from Intel: the Ethernet Controller 82559ERS.

The main features of the Ethernet are as follows:

- integrated IEEE 802.3 10baseT and 100BaseTX compatible PHY
- glueless 32-bit PCI master interface
- compatible with driver software of the 82558 and 82557
- full duplex support at both 10 and 100 Mbps
- IEEE 802.3u Auto-Negotiation support
- 4 kB transmit FIFO
- 3 kB receive FIFO

The connector used for the 100BaseTX Ethernet interface is an RJ45 connector. The signals on this connector are as follows.

Figure 2-2: Ethernet Connector

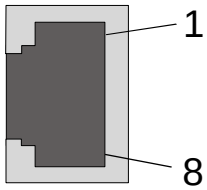


Table 2-4: Ethernet Connector Pinout

PIN NUMBER	SIGNAL
1	TX+
2	TX-
3	RX+
4	N/C
5	N/C
6	RX-
7	N/C
8	N/C

2.3.3 Serial Interfaces and Pinouts

Two serial ports: TERM (UART A) and SER 0 (UART B) are provided by means of 8-pin RJ45 connectors.

The two serial interfaces are 16C550 compliant and have 128-byte transmit and receive buffers. The TERM port is used to interface with the bootstrap loader, the operating system, and the application as required. The SER 0 port is used for data transfers as called for by the operating system or the application.

In the case that SER 0, the upper serial interface, is configured as RS232, the two serial ports are identical and they provide a complete set of handshaking and modem control signals, maskable interrupt generation and data transfer of up to 115.2 KBaud.

A separate transition module, the CP320-TR1, is also available from *Kontron* which provides an optoisolated half/full duplex RS485 interface. When installed, this module replaces the standard onboard SER 0 interface along with its associated RJ45 connector. See Figure 1-2 and Appendix B for more details of this module.

Figure 2-2: RS-232 Serial Connector

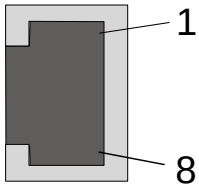


Table 2-5: Serial Port Pinout

PIN	RS-232 SIGNALS	RS-485 SIGNALS	
		HALF-DUPLEX	FULL-DUPLEX
1	DSR	N/C	-RxD
2	RTS	N/C	N/C
3	GND	GND	GND
4	TXD	+TRXD	-TxD
5	RXD	N/C	N/C
6	DCD	N/C	+RxD
7	CTS	-TRXD	+TxD
8	DTR	N/C	N/C

2.3.4 Serial Interface Expansion Connector and Pinout

The serial interface expansion connector provides the capability to add different front end interfaces to the UART B signals. For example, the available opto-isolated RS422/485 module, CP320-TR1, may be plugged into this connector.

Figure 2-2: Serial I/F (CON3) Connector

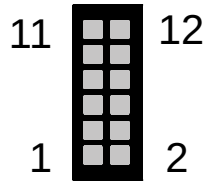


Table 2-6: Serial I/F Expansion Connector (CON3) Pinout

SIGNAL	PIN	PIN	SIGNAL
+3.3V	11	12	VCC
SCL	9	10	SDA
CTS	7	8	DTR
RxD	5	6	TxD
RE	3	4	DE
GND	1	2	RTSB

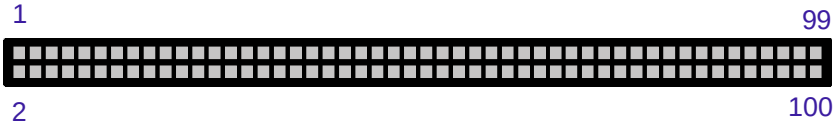
2.3.5 PCI Expansion Connector and Pinout

The PCI Expansion Connector (CON11) provides the possibility to mount several PCI mezzanine boards above the CP321 for adding additional functionality which is not provided on the CP321 main board or on the CPCI bus. All the PCI signals of the onboard PCI bus (32-bit, 33 MHz) are routed to this connector so that a complete PCI bus is provided to the mezzanine board with almost the same number of ground and power pins (3.3V, 5V, and V(I/O)) as are on a CPCI J1 or PMC connector. In addition to the PCI signals, I²C signals are also routed to this connector.

Examples of PCI expansion boards are:

- PMC carrier
- PC-MIP carrier
- IO board with second Ethernet interface, SCSI etc.

Figure 2-2: PCI Expansion Connector (CON11)



A table showing the pinout of the PCI Expansion connector appears on the following pages.

Table 2-7: PCI Expansion Connector Pinout

SIGNAL	PIN	PIN	SIGNAL
GND 1)	1	2	SCL (I2C)
RST#	3	4	+3.3V 2)
+3.3V 2)	5	6	CLK2
CLK3	7	8	GND 1)
GND 1)	9	10	CLK4
INTB#	11	12	INTA#
INTD#	13	14	INTC#
+5V 3)	15	16	GNT#2
GNT#3	17	18	V(I/O) 6)
+3.3V 2)	19	20	GNT#4
GND 1)	21	22	REQ#2
REQ#3	23	24	GND 1)
+5V 3)	25	26	REQ#4
AD31	27	28	AD30
AD29	29	30	+5V 3)
GND 1)	31	32	AD28
AD27	33	34	AD26
AD25	35	36	GND 1)
+3.3V 2)	37	38	AD24
C/BE3#	39	40	SDA (I2C)
AD23	41	42	+3.3V 2)
GND 1)	43	44	AD22
AD21	45	46	AD20
AD19	47	48	GND 1)
V(I/O) 6)	49	50	AD18
AD17	51	52	AD16
C/BE2#	53	54	+5V 3)
GND 1)	55	56	FRAME#
IRDY#	57	58	GND 1)
+3.3V 2)	59	60	TRDY#
DEVSEL#	61	62	reserved
GND 1)	63	64	STOP#
LOCK#	65	66	+3.3V 2)
PERR#	67	68	V(I/O) 6)
SERR#	69	70	GND 1)
+5V 3)	71	72	PAR
C/BE1#	73	74	AD15
AD14	75	76	+3.3V 2)
GND 1)	77	78	AD13
AD12	79	80	AD11
AD10	81	82	GND 1)
GND 1)	83	84	AD9

Table 2-7: PCI Expansion Connector Pinout (Continued)

SIGNAL	PIN	PIN	SIGNAL
AD8	85	86	C/BE0#
AD7	87	88	+5V 3)
+3.3V 2)	89	90	AD6
AD5	91	92	AD4
AD3	93	94	GND 1)
reserved	95	96	AD2
AD1	97	98	AD0
+12V 4)	99	100	-12V 5)

Key

1) Ground	*4) +12V*
2) +3.3V	*5) -12V*
3) +5V	*6) V(I/O)*

2.3.6 Memory Expansion Connectors

Two, 32-pin DIL600 sockets are provided for the addition of various types memory expansion devices with access times of less than 150ns.

The devices which have been tested and approved for these connectors are as follows:

- DIL type Flash memory (up to 512 kB)
- DIL SRAM (up to 512 kB) e.g. Samsung KM684000BLP-7
- NVSRAM (up to 512 kB) e.g. DALLAS DS1250Y-100)
- EPROM (up to 512 kB) e.g. 27C040
- M-Systems DiskOnChip 2000

Optionally, Socket 2 can be expanded to 36 pins to allow the use of 1 MB or 2 MB NVSRAM from Dallas Semiconductor (DS1265/70).

Figure 2-3: Memory Expansion Sockets 1 and 2

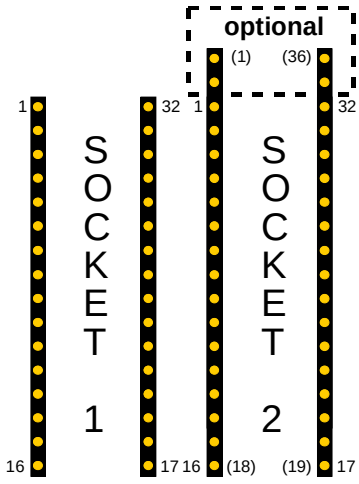


Table 2-8: DIL 32 Pinout for Various Devices

Pin	4M-bit Flash	Disk OnChip	NV SRAM	4M-bit Eprom		4M-bit Eprom	NV SRAM	Disk OnChip	4M-bit Flash	Pin
1	A18	N/C	A18	VPP		VCC	VCC	VCC	VCC	32
2	A16	N/C	A16	A16		A18	A15	WE_	WE	31
3	A15	N/C	A14	A15		A17	A17	N/C	A17	30
4	A12	A12	A12	A12		A14	WE	N/C	A14	29
5	A7	A7	A7	A7		A13	A13	N/C	A13	28
6	A6	A6	A6	A6		A8	A8	A8	A8	27
7	A5	A5	A5	A5		A9	A9	A9	A9	26
8	A4	A4	A4	A4		A11	A11	A11	A11	25
9	A3	A3	A3	A3		OE_	OE_	OE_	OE_	24
10	A2	A2	A2	A2		A10	A10	A10	A10	23
11	A1	A1	A1	A1		CE_	CE_	CE_	CE_	22
12	A0	A0	A0	A0		D7	D7	D7	D7	21
13	D0	D0	D0	D0		D6	D6	D6	D6	20
14	D1	D1	D1	D1		D5	D5	D5	D5	19
15	D2	D2	D2	D2		D4	D4	D4	D4	18
16	GND	GND	GND	GND		D3	D3	D3	D3	17

Table 2-9: DIL 36 Pinout for 1 MB and 2 MB NVSRAM Devices

(Dallas Semiconductor 12654 and 1270Y)

Pin	NV SRAM		NV SRAM	Pin
1	N/C		VCC	36
2	A20		A19	35
3	A18		N/C	34
4	A16		A15	33
5	A14		A17	32
6	A12		WE	31
7	A7		A13	30
8	A6		A8	29
9	A5		A9	28
10	A4		A11	27
11	A3		OE_	26
12	A2		A10	25
13	A1		CE_	24
14	A0		D7	23
15	D0		D6	22
16	D1		D5	21
17	D2		D4	20
18	GND		D3	19

2.4 Special Board Functions

2.4.1 Watchdog Timer

A watchdog timer is available which (when enabled) on timeout forces either a non-maskable interrupt (NMI) to be generated or causes a system reset to occur (refer to chapter 4 for configuration details). The watchdog timing has four possible settings: 0.5, 1.0, 1.5, and 2.0 seconds. After selecting the timeout value and routing (NMI or reset) the watchdog can be enabled. Once enabled, the watchdog must be continuously retriggered or a timeout will occur. When the watchdog timer is enabled, it cannot be stopped or reprogrammed except by resetting the system. The yellow watchdog LED (W) indicates the enabling status of the watchdog. Prior to the watchdog being enabled it is off. After enabling it comes on and remains on until a system reset occurs.

2.4.2 Realtime Clock (STC M41T56)

A separate hardware realtime clock (RTC) is incorporated on the CP321 board which provides clock information via the I²C bus for application use. An eight byte wide register (refer to chapter 4 for description) is available for accessing, setting, and starting the RTC. The RTC must be initialized prior to its use whereby settings are possible for seconds, minutes, hours, day, date, month, year, and calibration information. Continuous clock operation (even with system power off) is possible through the use of a rechargable Gold Cap, or alternately, lithium battery buffering is possible. Accuracy of the RTC is 35 ppm whereby temperature compensation can be adjusted in steps of +4.068 or -2.034 ppm per software using the onboard digital temperatur sensor (LM75).

For calibration purposes the RTC can also generate a 512 Hz test signal which is made available at test jack J2 (figure 1-3 indicates the location of J2 on the board). Please refer to the datasheet of the ST M41T56 for more information concerning calibration.

Figure 2-4: RTC J2 Pinout

1 FT GND

2 FT OUT

2.4.3 Reset/Abort

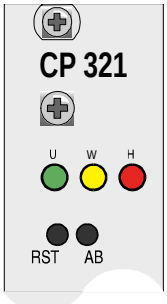
On the CP321 front panel there are two push button switches for interacting with the system: RST for reset and AB for abort.

Pressing the RST button initiates an immediate hardware reset of the system.

During normal operation pressing the AB button causes a non-maskable interrupt (NMI) to be generated. In addition it is latched into a bit in the System Logic, the purpose of which is to differentiate between the NMI initiated from the ABORT Button and the NMI initiated from the Watchdog Timer.

Pressing the AB button during system startup when the U LED (green) is blinking causes the bootstrap loader to enter interactive command mode. Commands can then be entered for processing by the bootstrap loader. Refer to chapter 5 for Bootstrap Loader information.

Figure 2-5: CP321 Front Panel



2.4.4 System Status Indicators

Six system status indicators divided into two groups of three LED's each are provided on the front panel of the CP321. The first group (LED1) is application oriented whereas the second group (LED2) is dedicated to and controlled by the Ethernet interface. The table below provides an overview of the functionality associated with these indicators.

Table 2-10:System Status Indicators

G	FP DES.	COLOR	NAME	DESCRIPTION
LED1	U	GREEN	LED1G	This LED blinks during startup indicating that the system is being initialized (bootstrap loading in progress). After system initialization has been completed it is on steady and is available for use by the application program. Refer to chapter 4 (Control Register) for setting of this LED.
	W	YELLOW	LED1Y	This LED is used to indicate the status of Watchdog Timer enabling. When on, the Watchdog Timer is enabled.
	H	RED	LED1R	This LED is available for general use and is application dependent. It is off if not used. Refer to chapter 4 (Control Register) for setting of this LED.
LED2	ACT	GREEN	LED2AC	This LED indicates that data are being transmitted or received via the Ethernet link. It blinks when there is traffic on the link.
	LNK	GREEN	LED2LN	This LED indicates the integrity of the Ethernet link. When on the link is established in both directions.
	SPEED	GREEN	LED2SP	This LED indicates the data rate of the Ethernet link. When on the speed is 100 Mbps; off it is 10 Mbps.

2.4.5 Coding Switch

The CP321 provides a 16-position, rotary coding switch (S2), which is available to the user for general programming purposes. The setting and usage of this switch is a function of the application (e.g. To configure software or change functionality of the board depending on the position of the switch without reprogramming the configuration via interface access or software). The actual position of the switch is read out of the coding switch register as a hexadecimal value. Refer to chapter 4, Coding Switch Register, for details of the register.

2.4.6 Digital Temperature Sensor (LM75)

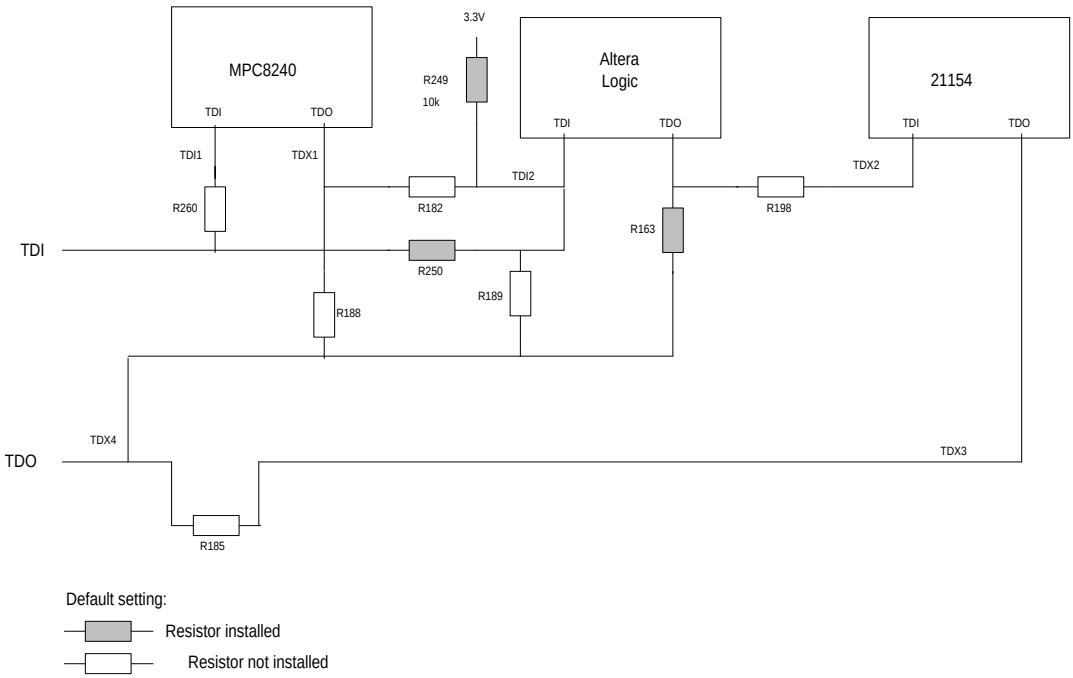
For purposes of temperature surveillance and a means of determining the current board temperature there is an digital temperature sensor (National Semiconductor LM75) installed on the CP321. Used as a thermal watchdog, the LM75 can generate a maskable interrupt which can be used by an application. In addition, the actual temperature can be read out of LM75 via the I²C bus. This is can be used, for example, to maintain the calibration of the onboard RTC over a wide operational temperature range. Refer to chapter 4, DTS Register, for further details.

2.4.7 DEBUG Interface and Pinout

A JTAG/COP interface is provided on the CP321 for the manufacturer's use (logic programming, JTAG test) or for software debugging. All the JTAG capable devices on the CP321 can be accessed through the onboard JTAG chain. If EMULATOR access to the MPC8245 is required it must be ensured that R260 and R188 are set and also that R163 and R250 are removed (all resistors are 0 ohm). When using this interface with emulator probes please use the signals on pins: 7, 8, 11, 12, 13, 15, 17, and 19. These are the standard signals defined by Motorola for the MPC8245 JTAG/COP port. All other signals are used for factory purposes.

The following figure illustrates the layout of the JTAG chain.

Figure 2-6: JTAG Chain Layout



Note...

As shipped, only the Altera onboard logic can be detected by means of the JTAG interface. If the JTAG interface requires to be reconfigured for software debugging, please contact Support at Kontron Modular Computers for assistance.

Figure 2-7: JTAG Connector (CON11)

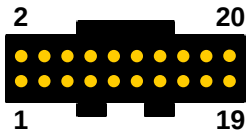


Table 2-11: JTAG Connector (CON11) Pinout

SIGNAL	PIN	PIN	SIGNAL
TEST	1	2	E_TO
E_TI	3	4	TEXEC
E_TCK	5	6	TDO
TDI	7	8	TRST#
NC	9	10	3.3V
TCK	11	12	CKSTP_IN
TMS	13	14	RY1
SRESET#	15	16	RY2
HRESET#	17	18	F_WE
CHKSTP#	19	20	GND

Note...

The connector used on the board is the SAMTEC FTSH- 110-01-L-DV-K. It is recommended to use a cable assembly (Type: FFSD-10-S-5-01-N) for connecting an emulator to this connector.

The factory setting of the chain is such that only the onboard logic is in the chain. If it is required to access the Processor via the JTAG chain a different setting must be used (some resistors must be reset). Refer to the table below for configuration information.

Table 2-11: JTAG Chain Resistor Settings

DEVICES	SETTING	RESISTORS								
		R163	R182	R185	R188	R189	R198	R249	R250	R260
CPU	Installed				X			X		X
	Not Installed	X	X	X		X	X		X	
ALTERA	Installed	X						X	X	
	Not Installed		X	X	X	X	X			X
CPU + ALTERA	Installed	X	X					X		X
	Not Installed			X	X	X	X		X	
ALTERA + 21154	Installed			X			X	X	X	
	Not Installed	X	X		X	X				X
CPU + ALTERA + 21154	Installed		X	X			X	X		X
	Not Installed	X			X	X			X	

2.5 Options

The following options are currently available for the CP321:

- the CP320-IO1 mezzanine carrier board for a single PMC module, and
- the CP320-TR1 RS485 optoisolated transition module.
- the CP320-TR2 RS232 optoisolated transition module
- PMC-HDD1 IDE hard disk module

Both of these options are described in Appendices A and B respectively.

In addition to the above options there is a special test/debug adapter, the CP320-Post, and a rear I/O module, the CP-RIO3-01, which can be obtained on request. Please contact your nearest sales representative for further information.

Chapter 3

Installation

This page was intentionally left blank.

3. Installation

The CP321 has been designed for easy installation. However, the following standard precautions and installation information/procedures must be observed.

3.1 Board Installation

Caution, Electric Schock!

If your board type is not specifically qualified as hotswap capable, please switch off the CompactPCI system before installing the board in a free CompactPCI slot. Failure to do so could endanger your life/health and may damage your board or system.

Note...

Certain CompactPCI boards require bus master and/or rear I/O capability. If you are in doubt whether such features are required for the board you intend to install, please check your specific board and/or system documentation to make sure your system is provided with an appropriate free slot to insert the board.

ESD Equipment!

This CompactPCI board is sensitive to static electricity discharges (ESD). Please observe the following precautions to avoid damage to your board:

- Discharge your clothing before touching the assembly. Tools must be discharged before use.
- Do not touch components, connector-pins, or conductive circuits.
- If working at an anti-static workbench with professional discharging equipment, please do not omit to use it.

3.2 Placement of the CP321

The *Kontron* CompactPCI system configuration is characterized by the fact that its system slot (slot “1”) is on the right end of the backplane, thus allowing for physical CPU growth (heat-sink, cooling fan, PCI expanson modules, etc.) associated with higher-performance processors.

Note...

Prior to inserting this controller board, please make sure it is being fitted into the system slot.

3.3 Front Panel I/O Connectors

Warning!

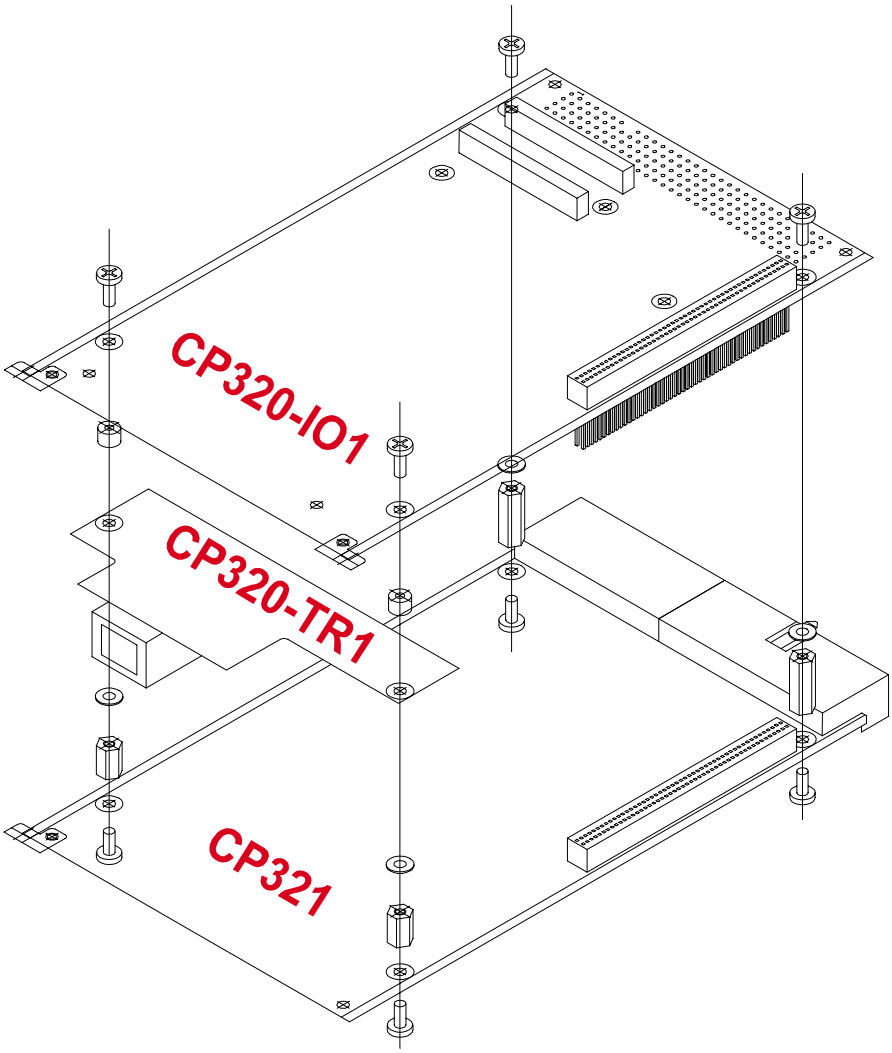
Due care should be exercised when connecting cabling in order to avoid damage to your connected device and/or the CP321 board.

For pinouts of the Front Panel connectors, please see Chapter 2: Functional Description.

3.4 Assembly of the CP321 and Options

The following diagram illustrates how the CP320-IO1 mezzanine carrier board and the CP320-TR1 RS485 optoisolated transition module are assembled with the CP321 main board. Assembly of the second carrier board follows on top of first carrier board.

Figure 3-1: CP321 and Options Assembly



3.5 Software Installation

Software installation is a function of the Bootstrap Loader and is described in chapter 5 of this manual.

This page was intentionally left blank.

Chapter 4

Configuration

This page was intentionally left blank.

4. Configuration

4.1 Jumper and Resistor Settings

Please see Figures 1-3 and 1-4 in Chapter 1 to view the positions of the jumpers and resistors on the board.

4.1.1 J1 - Bootstrap Loader / Socket Jumper

The jumper J1 is used to select the memory position from which the CP321 fetches its boot code. It establishes the address location of the onboard Flash window and the memory expansion socket 1 (DIL600, 32-pin). Refer to the Memory Configuration Register chapter for further information.

Note...

The MPC8245 initially fetches its boot code from address 0xFFF0 0100

Table 4-1: J1 - Bootstrap Loader / Socket Jumper Settings

J1	DESCRIPTION	ADDRESS ASSIGNMENT	
Open	CP321 fetches boot code from onboard Flash	Socket 1: Onboard Flash Window:	0xFFF8 0000 - 0xFFFF FFFF 0xFFF0 0000 - 0xFFF7 FFFF
Closed	CP321 fetches boot code from socket 1	Socket 1: Onboard Flash Window:	0xFFF0 0000 - 0xFFF7 FFFF 0xFFF8 0000 - 0xFFFF FFFF

4.1.2 J2 - Realtime Clock (RTC) Calibration Output

J2 is a test point for calibration measurement of the frequency of the RTC and is as such not a jumper. Refer to the datasheet of the ST M41T56 for further information on the use of this output signal.

Warning!

At NO TIME is J2 to be jumpered (short circuited). This is a test point and operation with a jumper installed will cause damage to the RTC.

4.1.3 Resistor Settings for Non-standard Socket Devices

The default pinouts of sockets 1 and 2 are designed for use with standard DIL Flashes and M-Systems DiskOnChip. However, in order to accommodate the various possible devices it is necessary to install resistors as jumpers to configure the board for proper operation.

Table 4-2: Resistor Settings for Socket 1

USED SOCKET DEVICE	R42	R40	R63	R57	R38	R68	R69
Flash / DiskOnChip (default)	Open	Open	Open	Set	Set	Open	Set
NVSRAM	Open	Open	Set	Open	Set	Set	Open
4 Mbit EPROM	Set	Set	Open	Open	Open	Open	Set

Table 4-3: Resistor Settings for Socket 2

USED SOCKET DEVICE	R26	R28	R25	R22	R24	R27	R29
Flash / DiskOnChip (default)	Open	Open	Open	Set	Set	Open	Set
NVSRAM	Open	Open	Set	Open	Set	Set	Open
4 Mbit EPROM	Set	Set	Open	Open	Open	Open	Set

Note...

All resistors are 0 ohm.

4.2 Board Address Map

The following figures illustrate the address mapping of the CP321. Where the first figure describes the overall map, the second figure provides a more detailed map of the uppermost address area. The upper area address map depends on the configuration of the CP321 memory expansion sockets and the requirements of the application.

Figure 4-1: CP321 Address Map

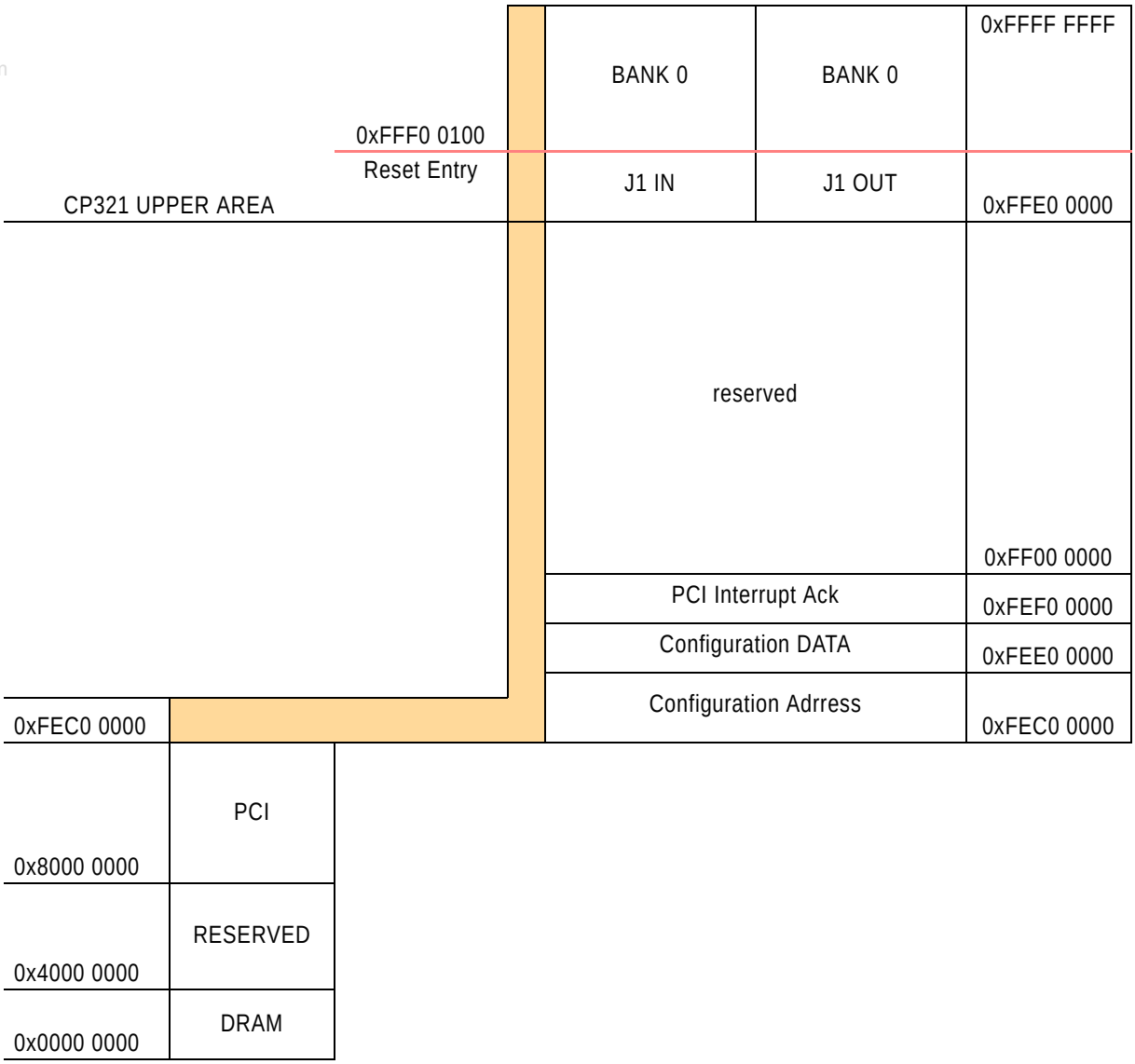
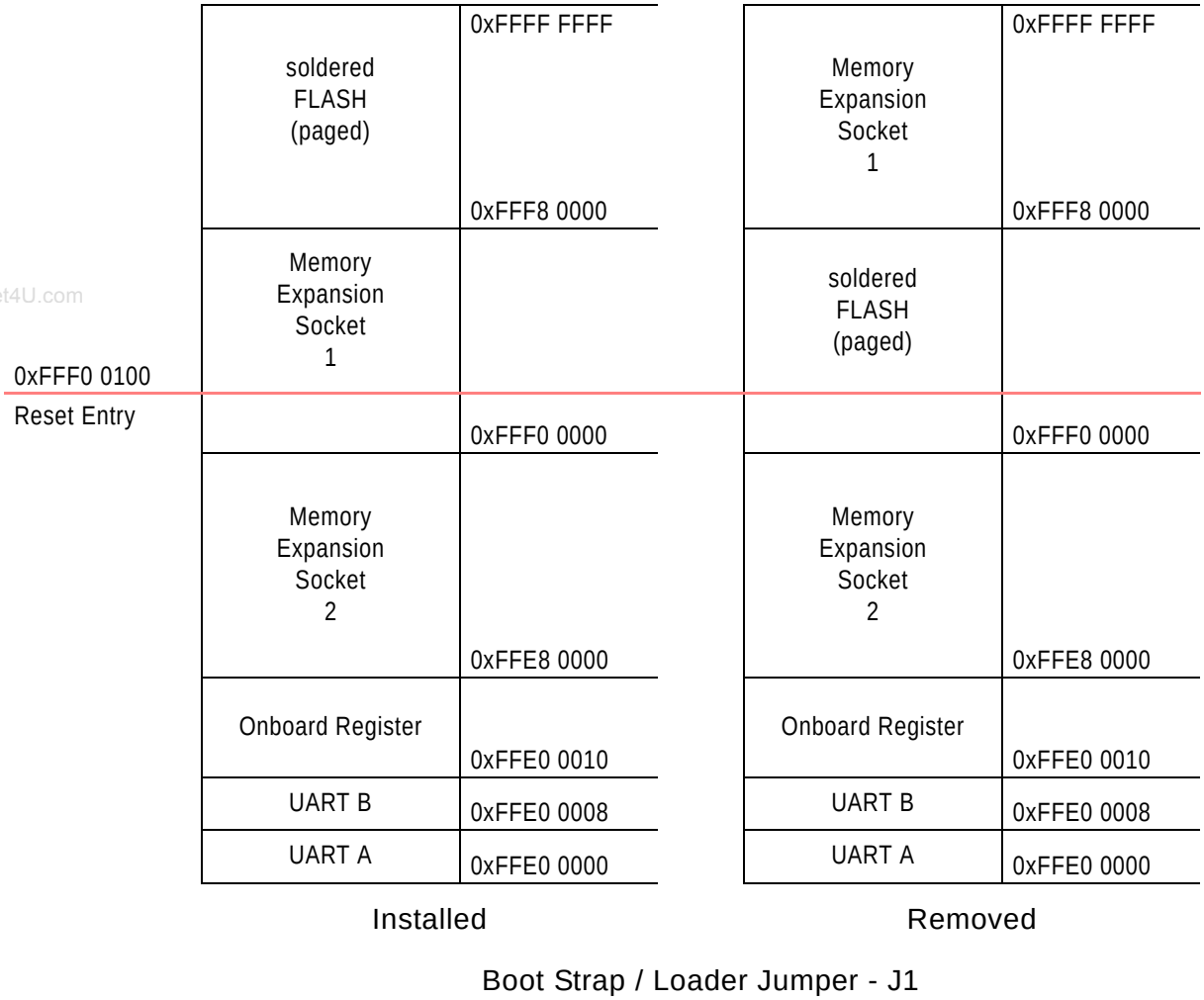


Figure 4-2: CP321 Upper Area Address Map



Note...

Write access to the upper area addresses is only possible using byte-wide write commands.

4.3 Board Control Registers

The Board Control registers may be accessed through byte-wide read and write operations to the address space 0xFFE0 0000 - 0xFFE7 FFFF

Figure 4-3: Board Control Registers

REGISTER	ADDRESS	ACCESS	
		READ	WRITE
Board-ID	0xFFE0 0010	X	
Software Compatibility ID	0xFFE0 0012	X	
Memory Configuration	0xFFE0 0014	X	
Flash Bank Select	0xFFE0 0016	X	X
SRAM Bank Select	0xFFE0 0017	X	X
Watchdog Control Register	0xFFE0 0018	X	X
Interrupt Enable Register	0xFFE0 0019	X	X
Control Register	0xFFE0 001A	X	X
Coding Switch Register	0xFFE0 001B	X	
Event Register	0xFFE0 001C	X	X
Board/Logic Revision	0xFFE0 001E	X	

4.3.1 Board ID Register

The Board ID is used to identify the CP321 in a CPCI system. The value for the CP321 is 0x83 which is factory set and cannot be changed.

Table 4-4: Board ID Register

REGISTER NAME	BOARD ID						ACCESS	
ADDRESS	0xFFE0 0010						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	BID7	BID6	BID5	BID4	BID3	BID2	BID1	BID0
DEFAULT	1	0	0	0	0	0	0	0

4.3.2 Software Compatibility ID

The Software Compatibility ID will signal to the software when differences in hardware require different handling by the software. It starts with the value 0x00 and will be incremented with each change in hardware (software sensitive only). This register is set at the factory and is for use only by the Boot Strap Loader and BSP software, and as such, is not user relevant.

Table 4-5: Software Compatibility ID

REGISTER NAME	SOFTWARE COMPATIBILITY ID						ACCESS	
ADDRESS	0xFFE0 0012						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SC7	SC6	SC5	SC4	SC3	SC2	SC1	SC0
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

4.3.3 Memory Configuration Register

The Memory Configuration register provides basic information concerning the amount of installed main memory, whether or not ECC is enabled, and from where the operating system is to obtain the boot strap loader.

Table 4-6: Memory Configuration Register

REGISTER NAME		MEMORY CONFIGURATION						ACCESS	
ADDRESS		0xFFE0 0014						R	
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		BJ	res.	res.	ECC	res.	res.	SZ1	SZ0
DEFAULT		n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BIT	NAME	VAL	DESCRIPTION						
0	SZ0	0	Settings: SZ1 SZ0 0 0 32 MB (64 Mbit chips, 1 bank equipped) 0 1 64 MB (64 Mbit chips, 2 banks equipped) 1 0 256 MB (256 Mbit chips, 2 banks equipped) 1 1 128 MB (128 Mbit chips, 2 banks equipped)						
		1							
1	SZ1	0							
		1							
2		0	reserved						
		1							
3		0	reserved						
		1							
4	ECC	0	ECC disabled						
		1	ECC enabled						
5		0	reserved						
		1							
6		0	reserved						
		1							
7	BJ	0	Boot Jumper J1 closed (CP321 fetches boot code from socket 1)						
		1	Boot Jumper J1 open (CP321 fetches boot code from onboard flash)						

4.3.4 Flash Bank Select Register

The Flash bank select register is used to select the appropriate soldered Flash bank. As 8-bit wide Flash memory may only be accessed through a 512 kB window this is the only way to address a larger size Flash memory. Using bits 0 to 3 (FBn), 16 Flash banks can be selected (16x512 kB = 8 MB). The default value on startup of the CP321 is 0x00.

Table 4-7: Flash Bank Select Register

REGISTER NAME	FLASH BANK SELECT						ACCESS	
ADDRESS	0xFFE0 0016						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	FB3	FB2	FB1	FB0
DEFAULT	n/a	n/a	n/a	n/a	0	0	0	0

4.3.5 SRAM Bank Select Register

This register is for usage in conjunction with the memory expansion socket 2 and NVSRAM Types DS1265Y or DS1270Y. Due to the fact that there is only a page size of 512 kB available for the memory expansion socket 2, the additional address lines are provided by writing to the bits 0 and 1 (SBn) of the register:

- a 1 in bit 0 (SB0) provides address A19 for the NVSRAM,
- a 1 in bit 1 (SB1) provides address A20 for the NVSRAM.

Table 4-8: SRAM Bank Select Register

REGISTER NAME	SRAM BANK SELECT						ACCESS	
ADDRESS	0xFFE0 0017						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	res.	res.	SB1	SB0
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	0	0

4.3.6 Watchdog Control Register

The Watchdog Control register is the interface between applications and the operating system for controlling the functioning of the Watchdog. Together with the Event Register, bit 0 (WD) and bit 2 (PB2), the possibility is provided for either hardware (Abort switch) or software (Watchdog timer) intervention in the execution of the application.

Table 4-9: Watchdog Control Register

REGISTER NAME		WATCHDOG CONTROL						ACCESS	
ADDRESS		0xFFE0 0018						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		WD_EN	WD_R	res.	WD_TRG	res.	res.	WDT1	WDT0
DEFAULT		0	0	n/a	0	n/a	n/a	n/a	n/a
BIT	**NAME**	**VAL**	**DESCRIPTION**						
0	WDT0	0	Settings: WDT1 WDT0 0 0 0.5 seconds Watchdog timeout time 0 1 1.0 seconds Watchdog timeout time 1 0 1.5 seconds Watchdog timeout time 1 1 2.0 seconds Watchdog timeout time						
		1							
1	WDT1	0							
		1							
2		0	reserved						
		1							
3		0	reserved						
		1							
4	WD_TRG	0	When WD-EN (bit 7) set to 1, indicates that Watchdog timer has not been retriggered.						
		1	Causes the Watchdog to be retriggered (Resets Watchdog timer to value indicated by bits 0 and 1, and WD_TRG (bit 4) to 0)						
5	WD_CCD	0	Normal watchdog functionality						
		1	Cascade mode: when watchdog timout occurs, an NMI will be generated, the watchdog timer resets, a further timeout will result in a system reset (when WD_R is first set to 1)						
6	WD_R	0	Causes hardware reset of system upon Watchdog timeout						
		1	Causes generation of a non-maskable interrupt upon Watchdog timeout						
7	WD_EN	0	Watchdog timer disabled						
		1	Watchdog timer enabled  ***Note...*** Once the Watchdog timer is enabled it cannot be disable except by resetting the system.						

4.3.7 Interrupt Enable Register

Table 4-10:Interrupt Enable Register

REGISTER NAME		INTERRUPT ENABLE						ACCESS	
ADDRESS		0xFFE0019						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		res.	FAL_EN	DEG_EN	res.	res.	res.	res.	res.
DEFAULT		n/a	0	0	n/a	n/a	n/a	n/a	n/a
BIT	NAME	VAL	DESCRIPTION						
0		0	Reserved						
		1							
1		0	Reserved						
		1							
2		0	Reserved						
		1							
3		0	Reserved						
		1							
4		0	Reserved						
		1							
5	DEG_EN	0	Assertion of the power supply derate signal DEG cannot result in an interrupt						
		1	Assertion of the power supply derate signal DEG results in an interrupt						
6	FAL_EN	0	Assertion of the power supply failure signal FAL cannot result in an interrupt						
		1	Assertion of the power supply failure signal FAL results in an interrupt						
7		0	Reserved						
		1							

4.3.8 Control Register

The Control register provides access to the front panel general purpose LED's (LED1R and LED1G), allows for the generation of a software reset of the system, and is used to control the configuration of the SER 0 (UART B) either for RS232 or RS485 operation.

Table 4-11: Control Register

REGISTER NAME		CONTROL						ACCESS	
ADDRESS		0xFFE0 001A						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		RS_CTL	res.	res.	S_RST	res.	res.	LED1R	LED1G
DEFAULT		n/a	n/a	n/a	n/a	n/a	n/a	0	0
BIT	**NAME**	**VAL**	**DESCRIPTION**						
0	LED1G	0	LED1G (green) off						
		1	LED1G (green) on						
1	LED1R	0	LED1R (red) off						
		1	LED1R (red) on						
2		0	reserved						
		1							
3		0	reserved						
		1							
4	S_RST	0	no operation						
		1	Causes a software reset (S_RST) to be initiated						
5		0	reserved						
		1							
6		0	reserved						
		1							
7	RS_CTL	0	Indicates that the serial interface, SER 0 (UART B), is to be configured for RS232 operation						
		1	Indicates that the serial interface, SER 0 (UART B), is to be configured for RS485 operation  ***Warning!*** When setting bit 7 care must be taken to ensure that the installed interface corresponds to the bit setting. A mismatch may cause damage to the CP321 or the application.						

4.3.9 Coding Switch Register

The Coding Switch Register is used to indicate the actual position of the onboard general purpose coding switch. The position is shown in binary form.

Table 4-12:Coding Switch Register

REGISTER NAME	CODING SWITCH						ACCESS	
ADDRESS	0xFFE0 001B						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	SW3	SW2	SW1	SW0
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

4.3.10 Board Logic / Revision Register

The Board Revision Register may be used to identify the hardware (BRn) and logic status of the board by the software (LRn). It is set at the factory and starts with the value 0x00 for the initial board prototypes and will be incremented with each redesign / logic release.

Table 4-13:Board Logic / Revision Register

REGISTER NAME	BOARD LOGIC/REVISION						ACCESS	
ADDRESS	0xFFE0 001E						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	LR3	LR2	LR1	LR0	BR3	BR2	BR1	BR0
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

4.3.11 Event Register

The Event register is used to indicate the origin of the generation of the non-maskable interrupts caused either by a Watchdog timeout or the pressing of the Abort switch.

Table 4-14:Event Register

REGISTER NAME		EVENT						ACCESS	
ADDRESS		0xFFE0 001C						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		res.	res.	res.	res.	res.	PB2	res.	WD
DEFAULT		n/a	n/a	n/a	n/a	n/a	0	n/a	0
BIT	NAME	VAL	DESCRIPTION						
0	WD	0	Indicates that no Watchdog timeout has occurred						
		1	Indicates that a Watchdog timeout has occurred						
1		0	reserved						
		1							
2	PB2	0	Indicates that the Abort switch has not been pressed						
		1	Indicates that the Abort switch has been pressed						
3		0	reserved						
		1							
4		0	reserved						
		1							
5		0	reserved						
		1							
6		0	reserved						
		1							
7		0	reserved						
		1							

4.4 UART Registers Address Mapping

4.4.1 UART A

The following table indicate the address mapping of the UART A. For a more detailed description please refer to the EXAR XR16C2850 DUART manual.

Table 4-15:UART A General Register Set

READ MODE	WRITE MODE	ADDRESS
Receive Holding Register	Transmit Holding Register	0xFFE0 0000
n/a	Interrupt Enable Register	0xFFE0 0001
Interrupt Status Register	FIFO Control Register	0xFFE0 0002
n/a	Line Control Register	0xFFE0 0007
n/a	Modem Control Register	0xFFE0 0004
Line Status Register	n/a	0xFFE0 0005
Modem Status Register	n/a	0xFFE0 0006
Scratchpad Register	Scratchpad Register	0xFFE0 0007

Table 4-16:UART A Baud Rate Register Set

READ MODE	WRITE MODE	ADDRESS
LSB of divisor latch	LSB of divisor latch	0xFFE0 0000
MSB of divisor latch	MSB of divisor latch	0xFFE0 0001

Table 4-17:UART A Enhanced Register Set

READ MODE	WRITE MODE	ADDRESS
Trigger Level Register	Trigger Level Register	0xFFE0 0000
Feature Control Register	Feature Control Register	0xFFE0 0001
Enhanced Feature Register	Enhanced Feature Register	0xFFE0 0002
Enhanced Mode Select Register	Enhanced Mode Select Register	0xFFE0 0007
Xon-1	Xon-1	0xFFE0 0004
Xon-2	Xon-2	0xFFE0 0005
Xoff-1	Xoff-1	0xFFE0 0006
Xoff-2	Xoff-2	0xFFE0 0007

4.4.2 UART B

The following table indicate the address mapping of the UART B. For a more detailed description please refer to the EXAR XR16C2850 DUART manual.

Table 4-18:UART B General Register Set

READ MODE	WRITE MODE	ADDRESS
Receive Holding Register	Transmit Holding Register	0xFFE0 0008
n/a	Interrupt Enable Register	0xFFE0 0009
Interrupt Status Register	FIFO Control Register	0xFFE0 000A
n/a	Line Control Register	0xFFE0 000B
n/a	Modem Control Register	0xFFE0 000C
Line Status Register	n/a	0xFFE0 000D
Modem Status Register	n/a	0xFFE0 000E
Scratchpad Register	Scratchpad Register	0xFFE0 000F

Table 4-19:UART B Baud Rate Register Set

READ MODE	WRITE MODE	ADDRESS
LSB of divisor latch	LSB of divisor latch	0xFFE0 0008
MSB of divisor latch	MSB of divisor latch	0xFFE0 0009

Table 4-20:UART B Enhanced Register Set

READ MODE	WRITE MODE	ADDRESS
Trigger Level Register	Trigger Level Register	0xFFE0 0008
Feature Control Register	Feature Control Register	0xFFE0 0009
Enhanced Feature Register	Enhanced Feature Register	0xFFE0 000A
Enhanced Mode Select Register	Enhanced Mode Select Register	0xFFE0 000B
Xon-1	Xon-1	0xFFE0 000C
Xon-2	Xon-2	0xFFE0 000D
Xoff-1	Xoff-1	0xFFE0 000E
Xoff-2	Xoff-2	0xFFE0 000F

4.4.3 IRQ Routing

The IRQ routing of the CP321 is serial as opposed to being parallel. Hence the IRQ names are prefixed with S_ to indicate that they are serial.

Table 4-21:IRQ Routing

IRQ NAME	SOURCE
S_IRQ0	Reserved
S_IRQ1	UART-A
S_IRQ2	UART-B
S_IRQ3	INTA# (PCI)
S_IRQ4	INTB# (PCI)
S_IRQ5	INTC# (PCI)
S_IRQ6	INTD# (PCI)
S_IRQ7	TEMP_INT (Temperature Interrupt)
S_IRQ8	reserved
S_IRQ9	ENUM
S_IRQ10	reserved
S_IRQ11	reserved
S_IRQ12	reserved
S_IRQ13	DEG
S_IRQ14	FAL
S_IRQ15	reserved

4.4.4 Real-time Clock

Access to the real-time clock (RTC) is effected via the I2C bus. The RTC uses address 0xD0. For more detailed information please refer to the manuals for the ST - Microelectronics M41T56 and the Motorola MPC 8245 (I2C - Bus).

Table 4-22:Register Map RTC M41T56

REG. BYTE	ADDRESS BITS								FUNCTION RANGE IN BCD FORMAT
	D7	D6	D5	D4	D3	D2	D1	D0	
0	ST	10 Seconds			Seconds				Seconds: 00 - 59
1	X	10 Minutes			Minutes				Minutes: 00 - 59
2	CEB	CB	10 Hours		Hours				Century: 0 - 1 Hours: 00 - 23
3	X	X	X	X	X	Day			Day: 00 - 07
4	X	X	10 Date		Date				Date: 01 - 31
5	X	X	X	10M.	Month				Month: 01 - 12
6	10 Years				Years				Year: 00 - 99
7	OUT	FT	S	Calibraton					Control

Legend for Table 4-22:

CEB = Century enable bit
CB = Century bit
FT = Frequency test bit
OUT = Output level
ST = Stop bit
S = Sign bit

Note...

When the RTC has once been stopped due to low voltage, it is necessary to re-initialize the “Seconds” “Minutes” and “Hours” registers before it will run again.

4.5 EEPROM’s

Access to the EEPROM’s is effected via the I2C bus of the MPC8245. The EEPROM’s use the I2C address 0xA0 (System) and the address 0xA2 (User). Write protection is achieved by installing 0 ohm resistors R126 (System) and R45 (User). Default is unprotected.

For more detailed information please refer to the manuals for the MICROCHIP 24LC16B and the MOTOROLA MPC8245 (I2C bus).

4.6 Digital Temperature Sensor, LM75

Access to the onboard digital temperature sensor (DTS) is effected via the I2C bus of the MPC8245. The DTS uses the I2C address 0x90.

For more detailed information please refer to the manuals for the National Semiconductor LM75 and the MOTOROLA MPC8245 (I2C bus).

This page was intentionally left blank.

Chapter 5

NetBootLoader

26172.01.VC.030730/163444

This page has been intentionally left blank.

26172.01.VC.030730/163444

5. NetBootLoader

This E²Brain™ module is delivered with the NetBootLoader software already programmed into the onboard soldered Flash memory. The NetBootLoader itself is a software utility which initializes the module for operation before turning control over to either an application or to an operator. This software also provides the capability to monitor and control the operation of the NetBootLoader itself, display system status information, to program executable code and data to the Flash memory, and to load and start application software.

To attain full operational capability, the NetBootLoader FLASH must be programmed by the user with application software. Once the application has been programmed to Flash memory, the NetBootLoader will support the complete boot operation. The following chapters describe the functioning of the NetBootLoader and how to program the Flash memory.

Note...

The following description assumes a standard CPU board with appropriate hardware. In the event such hardware is not available, disregard the text that applies to the missing hardware and proceed as appropriate.

5.1 General Operation

Upon power on or a system reset, the NetBootLoader is started. The CPU board is configured for operation and control is either passed to an application or an operator. In the event a valid application has been programmed into the Flash memory and no operator intervention takes place, the application is copied from FLASH into SDRAM and control is passed to the application. If the NetBootLoader does not find a valid application or operator intervention has occurred, control is passed to the operator. The operator now has control to determine the system status, make configuration changes, read or program the Flash memory, or to restart or shut down the system.

The operator command interfacing with the NetBootLoader is accomplished either via the TERM serial port or the Ethernet port. During the boot operation a command interpreter is started which allows the operator to input commands to the NetBootLoader. Prior to interfacing via the Ethernet port the network must be configured. This is done via the TERM port.

5.2 NetBootLoader Interfaces

There are four possibilities to interface with the NetBootLoader:

- Via the MC1 (Abort) signal
- Via the TERM serial interface
- Via the SER0 serial interface
- Via the Ethernet interface

Gaining access to the NetBootLoader is a function of the contents of the Flash memory and the “BootWaitTime” setting. If there is no valid application programmed into the Flash memory, the boot operation automatically terminates after the module has been initialized and control is passed to the command interpreter. If there is a valid application in the Flash memory the boot operation is delayed according to the setting of the boot wait time, and the MC6 (LED1) output signal is alternately asserted indicating that the boot operation is in a wait state. During this time the operator may intervene in the boot operation either by asserting the MC1 (Abort) signal, entering the “abort” command via the TERM interface, or by performing a successful telnet login via the Ethernet interface. If the operator does not intervene, the boot operation is continued after the boot wait time has been exceeded.

26172.01.VC.030730/163444

5.2.1 MC1 (Abort) Signal

The MC1 (Abort) signal is routed to the CP321 carrier board via the System Interface (CON1 connector) and, if made available on the carrier, provides the operator with the ability to directly terminate the boot operation during the boot wait time which is indicated by the alternately asserted MC6 (LED1) signal. This is the sole purpose of the MC1 (Abort) signal during the NetBootLoader operation.

5.2.2 TERM Serial Interface

The TERM serial port, if realized on the carrier board, is used to provide direct operator interfacing to the NetBootLoader. As soon as the CPU board has been initialized this port is activated and the operator may input commands. During the boot wait time the operator may terminate the boot operation and take control of the NetBootLoader. Once the boot wait time is exceeded the command interpreter is deactivated and the operator no longer has access to the NetBootLoader.

The TERM serial interface may either be directly connected to a terminal device or may interface with a terminal emulator.

5.2.3 SER0 Serial Interface

The SER0 serial port is used to provide the NetBootLoader with the ability to access Motorola S-Records for programming an application to FLASH. No command interpreter is available for this interface.

5.2.4 Ethernet Interface

The Ethernet interface provides the capability of remotely interfacing with the NetBootLoader. Prior to using this interface it is necessary to configure the NetBootLoader network settings. This is accomplished via the TERM interface. Once the network settings have been made, the remote operator has the same capabilities as with the TERM interface. During the boot wait time the operator gains control of the NetBootLoader by logging into it via the Ethernet interface. This causes the boot operation to be terminated and gives control to the remote operator.

The Ethernet interface uses the telnet protocol for operator interfacing with the NetBootLoader. In addition to the operator interface via Ethernet, the NetBootLoader also uses the Ethernet interface for ftp server access.

5.3 NetBootLoader Functions

In addition to initializing the CPU board for operation and the loading and starting of applications, the NetBootLoader provides the following operator monitor and control functions:

- NetBootLoader control
- system status monitoring
- ftp server access
- FLASH reading and programming operations
- Motorola S-Record acquisition

These functions are described in detail in the following chapters.

NOTE ...

The command title (CMD TITLE) is expressed in capital letters and is not the same as the syntax of the command. The command syntax is always written using small letters

26172.01.VC.030730/163444

5.3.1 NetBootLoader Control

The NetBootLoader provides various functions for controlling the operation of the NetBoot-Loader itself as well as the setting of operational parameters. The following table provides an overview of available NetBootLoader control functions.

Table 5-1: NetBootLoader Control Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
ABORT	-	Terminate boot wait	
BW	Boot Wait	Set or display BootWaitTime	
HELP or ?	-	Display online HELP pages	
LOGOUT	-	Terminate telnet session	
NET	-	Set network parameters	Must be set before attempting telnet login
PASSWD	Password	Set telnet password	Must be set before attempting telnet login
PF	Port Format	Set serial port parameters	Used for both TERM and SER0 ports
RS	Reset	Resets system	

5.3.2 System Status Monitoring

The NetBootLoader provides various functions for monitoring the overall status of the system during the operation of the NetBootLoader. The following table provides an overview of avail-able system status monitoring functions.

Table 5-2: System Status Monitoring Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
CHECK	-	Application validation	Verifies validity of user image programmed to FLASH
INFO	-	Display system information	
MD	Memory Display	Display memory contents	Applies to all visible memory
PCI	-	Display PCI device information	
PING	-	Verify network status	
VER	Version	Display version number of NetBootLoader	

26172.01.VC.030730/163444

5.3.3 ftp Server Access

The NetBootLoader provides various functions for interfacing with an ftp server. The following table provides an overview of available ftp server functions.

Table 5-3: ftp Server Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
BYE	-	Terminate session with ftp server	
CD	Change Directory	Change ftp server directory	
GET	-	Download a file from ftp server	Only for executable applications. Data buffer is target.
LOGIN	-	Login to ftp server	
LS	List Directory	List ftp server directory	Lists contents of directory.
PUT	-	Upload a file to ftp server	Data buffer is source.
PWD	Print Working Directory	Display current ftp server directory	Lists name of directory

5.3.4 FLASH Operation

The NetBootLoader provides various functions for performing operations with Flash memory. The following table provides an overview of available FLASH operation functions.

Table 5-4: FLASH Operation Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
CLONE	-	Program NetBootLoader to FLASH	Uses data buffer or socket as source
LF	Load FLASH	Program application to FLASH	Uses data buffer as source
SF	Store FLASH	Reads FLASH to data buffer	Uses data buffer as target

5.3.5 Motorola S-Records

The NetBootLoader provides one function for acquiring Motorola S-Records. The following table provides an overview of this function.

Table 5-5: Motorola S-Records Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
SL	SLoad	Download Motorola S-Records	Uses data buffer as target

26172.01.VC.030730/163444

5.4 Operating the NetBootLoader

5.4.1 Initial Setup

The CPU board is delivered with the NetBootLoader already installed in the onboard soldered FLASH and is ready for operation. However, in order for the CPU board to be used in a system, application software must be made available for use. This is accomplished by programming the application also to the onboard soldered Flash memory where the NetBootLoader is located.

Upon initial power up the NetBootLoader is started automatically. As soon as the NetBootLoader has completed initialization of the CPU board, it checks to see if there is a valid application programmed in FLASH and at the same time initiates a command interpreter which the operator can access either via the TERM or telnet interfaces. If there is no valid application in memory, the NetBootLoader terminates the boot operation, and waits for operator intervention. As this is the case when the CPU board is first powered up, the operator now has the opportunity to program an application.

Prior to programming an application it may be necessary to configure the NetBootLoader or perform other functions depending on the user's application development environment or application requirements. Once this has be accomplished and the application has been programmed, the CPU board is ready for operation.

The following chapters provide information on how to set up and operate the NetBootLoader itself, initiation of the telnet interface, and how to program an application to FLASH.

5.4.2 Accessing the NetBootLoader

Initial access to the NetBootLoader can only be achieved via the TERM interface. Prior to using the telnet interface, the Ethernet parameters must be set and this can only be accomplished initially via the TERM interface. Once valid Ethernet parameters and the telnet login password have be set, the telnet interface is available for operation.

Use of the TERM interface requires either a terminal or a terminal emulator. Use of the telnet interface requires a remote telnet login to the NetBootLoader.

Availability of the command interpreter depends on the system status. If there is no valid application programmed, the command interpreter is available as long as the operator requires it. If a valid application is programmed, the command interpreter is only available for the duration of the boot wait time. If the operator requires the command interpreter for a longer time he must terminate the boot operation before the boot wait time is exceeded.

Upon initiation of the command interpreter, a prompt is sent to the TERM interface and commands may be entered. To gain access to the NetBootLoader from a remote location via Ethernet a telnet login must be performed. If the boot wait time has not been exceeded, a telnet login automatically terminates the boot operation and a command prompt is sent to the telnet remote interface.

Once the operator has control of the NetBootLoader, he may perform any required action. To continue with the operation of the CPU board, the system must either be cold started or the operator must issue a "reset" command. In either event, the NetBootLoader is restarted and the boot operation begins anew.

26172.01.VC.030730/163444

5.4.3 NetBootLoader Configuration

There are several NetBootLoader commands which provide the operator with the capability to configure specific parameters which are used by the NetBootLoader for interfacing operations. These commands are:

- BW (BootWait)
- NET
- PASSWD
- PF (Port Format)

Default settings are available for all the above commands except for “net” which is dependent on the application environment.

5.4.3.1 BW

This command is used to display or set the actual boot wait time used by the NetBootLoader to delay the boot operation before proceeding with the loading and starting of an application. If this time is set too short it may only be possible to gain access to the NetBootLoader via the MC1 (Abort) signal.

The BootWaitTime value is stored in the boot section of the serial EEPROM. This section is validated with a CRC code to avoid the setting of random parameters.

Note ...

If the CRC of the boot section is not valid, changing the BootWaitTime will have no effect because the “bw” command does not validate an invalid CRC. In this case, a default timing of 5 seconds is always used.

To validate an invalid CRC, an operating system utility must be used, or, alternatively, the “-f” option of the “bw” command must be issued.

Warning !!!

Using the “bw -f” command to validate invalid entries may adversely impact the operation of the operating system.

5.4.3.2 NET

This command is used to set or display the parameters for the configuration of the Ethernet interface of the CPU board. The Ethernet interface is only available after these settings have been made. Once these settings have been made, the system must be cold started or reset for them to take effect.

5.4.3.3 PASSWD

This command is used to set the password used by the NetBootLoader for the operation of the telnet interface. No password is required for access from the TERM interface.

26172.01.VC.030730/163444

5.4.3.4 PF

This command is used to set the port parameters for the TERM and SER0 serial interfaces only for the current operator session. The next system restart will cause these settings to revert to the default settings of: 9600 Baud, 8 bits per character, 1 stop bit, and no parity. This is done to preclude a system lockout when restarting due to incompatible settings.

5.4.4 telnet Login

A telnet login to the NetBootLoader is only possible during the boot wait time and only after the Ethernet network parameters have been set.

To effect a telnet login the operator performs the standard telnet login procedure during the boot wait time. The NetBootLoader responds by suspending the boot wait and requests a login password. The operator then enters a password. If the password is valid, the boot wait is terminated and the operator can now access the NetBootLoader. If the password is invalid, the telnet login procedure is terminated and the boot operation continues.

In the case of an invalid password, the login procedure may be repeated as often as required within the boot wait time. Once the boot wait time is exceeded, a telnet login is no longer possible.

5.4.5 FLASH Operations

To achieve an operable system for an application, the application software must be programmed to FLASH. The NetBootLoader supports the programming of the application to FLASH. In addition to this, it also supports the updating of the NetBootLoader itself as well as data transfer from the FLASH to the data buffer and from the data buffer to an ftp server. The following chapters provide information on performing the various types of FLASH operations.

5.4.5.1 FLASH Offsets

All FLASH is treated as one uniform FLASH, regardless of the physical addresses of the devices involved. All offsets are based from the beginning of the FLASH area. This means that 0x0 is the beginning of the first FLASH bank. The NetBootLoader itself is located at the beginning of the FLASH area and for this reason this area cannot be used for application image programming. To display an overview of the current FLASH organization use the “info” command.

If the application image is an operating system (which is the default case), it must be programmed without an offset. When such an image is programmed to FLASH, the image length and CRC information is also programmed along with the image to FLASH. This information is used by the NetBootLoader to determine the validity of the image during the boot operation. During system startup, a valid image is copied to SDRAM address 0x0 and started at offset 0x100 after the boot wait time is exceeded.

If an offset is specified, the image will be programmed exactly at this offset without adding length or CRC information. This option is intended for the storing of configuration information which is required to be located in FLASH.

5.4.5.2 Programming an Application

The application image itself must be compiled and linked to run from the SDRAM base address 0x0 of the CPU. The image must contain executable PPC code at offset 0x100 which is the usual case with ROM/Flash images.

26172.01.VC.030730/163444

Gaining access to the image for programming to FLASH depends on where it is located. The NetBootLoader can access three different sources for images:

- ftp server
- Motorola S-Records
- memory within the visible address range of the CPU board

The NetBootLoader uses a single data buffer for downloading an image from an ftp server or an image as Motorola S-Records. These images must first be downloaded to the data buffer prior to being programmed to FLASH. An image located within the visible address range of the CPU board is directly accessible for programming.

To access an image located on an ftp server, the “get” command is used. To perform Motorola S-Record acquisition, the “sl” (SLoad) command is used. Once the image is in the data buffer, the FLASH is programmed using the “lf” (Load Flash) command. For an image within visible memory, the “lf” (LoadFlash) command is used to program directly to FLASH.

5.4.5.3 ftp Server Access

To gain access to an application image file stored on an ftp server the Ethernet interface is used. Images are downloaded to the data buffer using the ftp protocol. To use this interface the Ethernet parameters must first be set and then the system must be restarted. During boot wait the operator must gain control of the NetBootLoader and perform an ftp server login. After a successful login, the operator then locates the image file required and downloads it to the data buffer. As with any type of server session, the operator should logout when the session is finished.

Note ...

The commands “get” and “ls” use the same data buffer. Therefore if an “ls” command is issued after a “get” command the data buffer will be overwritten. If an “lf” command follows the “ls” the NetBootLoader refuses to program the overwritten data buffer to the FLASH.

5.4.5.4 Motorola S-Records

The NetBootLoader will also accept Motorola S-Records as an application image. The “sl”’ command accepts S1, S2 and S3 records. Operation is terminated by the appropriate S9, S8 or S7 record. Other types of records are ignored.

The checksum of every record except end records is checked. Bad records are rejected by the NetBootLoader. The address range of every record is also checked. Records which fall outside of the internal buffer are rejected.

The records must be 0-based. This means that it’s address must correspond to the address where they will be loaded in the data buffer relative to its start. If necessary, the base address can be modified with the -o option of the “sl” command.

Note ...

If the data buffer is programmed to FLASH without the -o option (program a startable image) the downloaded image is copied to RAM during startup and is executed there. For this reason application images which require to be programmed must start at the address 0x0.

26172.01.VC.030730/163444

The image must start at the absolute address 0x0 and must contain executable PPC code at the absolute address 0x100. If S1 or S2 record input is preferred, please note that these records only include 16 and 24-bit wide addresses. If no switch to another record type is included it must be ensured that the code is not larger than the address range covered.

Note ...

Neither the “sl” nor “lf” command can be used to program Motorola S-Records to RAM areas.

For accessing the Motorola S-Records, both the TERM and SER0 interfaces can be used. The MC6 (LED1) signal is asserted alternately at a low rate while downloading indicating that the transfer is in progress. The transfer itself may take several minutes to complete.

Ensure that the XON/XOFF protocol is used on the host side. This is a fixed setting and cannot be changed. Additionally, ensure that the host does not stop transmission after a number of lines (e.g. OS-9: use the ‘nopause’ attribute).

The TERM and SER0 serial interface parameters can be modified with the “pf” command.

5.4.6 Updating the NetBootLoader

In addition to programming an application to FLASH, the NetBootLoader itself can be updated. The new version of the image is made available via an ftp server.

5.4.6.1 Updating With an Image Loaded Via an ftp Server

The image is downloaded in the same way as an application image (refer to chapter 5.4.5.3). The new version of NetBootLoader image is then programmed using the “clone -n” command.

5.4.7 Uploading a FLASH Area

The NetBootLoader also has the possibility to upload certain areas of the FLASH to a host using the Ethernet interface. To use this interface the Ethernet parameters must first be set and then the system must be restarted. During boot wait the operator must gain control of the NetBootLoader and perform an ftp server login. After a successful login, the operator then stores the FLASH area to be uploaded to the local data buffer using the “sf” command. Using the “put” command transfers the contents of the data buffer to the ftp server. As with any type of server session, the operator should logout when the session is finished.

5.5 Plug and Play

On the CPU board the NetBootLoader includes “Plug and Play” functionality. This ensures that the board is completely initialized and that all resources necessary for PCI devices (addresses, interrupts etc.) are assigned automatically. This important feature has the advantage that conflicts do not arise when PCI devices are added or removed. Furthermore, the operating system itself does not include the board initialization code.

26172.01.VC.030730/163444

5.6 Porting an Operating System to the CPU Board

The image for the absolute address 0x0 should be linked with an entry point at the absolute address 0x100.

One should not attempt to reassign the PCI BAR registers. The assigned values should be read back and these should always be used in the drivers.

The “interrupt line” field in the PCI configuration header is initialized with the IRQ line number to which the INTA of the device is routed.

It is not necessary to rewrite the “EUMBBAR” field in the KAHLUA (MPC 8240) configuration space as this has already been done by the NetBootLoader. The existing value should be used.

Downloaded images are never executed from the FLASH due to the fact that on the CPU board it is paged. The programmed image is always downloaded to SDRAM, the absolute address 0x0 being downloaded first. There is no configuration option available to amend this process. If it is necessary to relocate the image to another address after download, simply add a small assembly routine at the beginning of the code which will move the image to the correct address.

26172.01.VC.030730/163444

5.7 Commands

The following commands are available with the NetBootLoader. Where an ellipsis (…) appears in the command syntax it means that the command is continued from the previous line. Observe any spaces that may be between the ellipsis and the remainder of the command.

ABORT

FUNCTION:	Terminate the NetBootLoader boot operation
SYNTAX:	**abort**
DESCRIPTION:	This command is used by the operator to to terminate the boot operation during the boot wait time to allow the operator to perform other NetBootLoader operations. To be asserted it must be issued during the boot wait time which is indicated by the alternating assertion of the MC6 (LED1) signal.

BW

FUNCTION:	Set or display the parameters of the boot wait function of the NetBootLoader
SYNTAX:	**bw [<time>| -f]** where: bw command <time> parameter: value: seconds 1, 2, 5, 10, 20, 50 -f option: force CRC update

BW

DESCRIPTION:	The command “bw” displays the parameter “<time>” setting. The parameter “<time>” stipulates the waiting time in seconds that the boot operation is delayed before the application is loaded and started. No values other than these are supported. Bear in mind when setting the boot wait time that the MC6 (LED1) signal is asserted alternately at the rate of two times a second. Therefore, if the boot wait is set to 1 second the MC6 signal will only be alternately asserted two times. The option “-f” is used to force updating of the CRC value of boot section of the EEPROM. For further information refer to chapter 5.4.3.1.
USAGE:	Display setting of “<time>” parameter COMMAND / RESPONSE: **bw** **WaitTime: 20**
	Set boot wait time to 50 seconds COMMAND / RESPONSE (none): **bw 50**

BYE

FUNCTION:	Terminate an ftp server session
SYNTAX:	**bye**
DESCRIPTION:	An ftp server session which has been established with the command “login” is terminated with the command “bye”.

26172.01.VC.030730/163444

CD

FUNCTION:	Change the current ftp server directory
SYNTAX:	**cd <new-path>** where: cd command <new-path> parameter: string new directory path
DESCRIPTION:	If an ftp server session has been established with the “login” command, the command “cd” is used to change the current ftp server directory. The argument “<new-path>” may be an absolute or relative path. The format depends on what the server accepts. For example, UNIX hosts require that the directory names must be entered exactly in the same case.

CHECK

FUNCTION:	Verify validity of application programmed to FLASH
SYNTAX:	**check**
DESCRIPTION:	When an application is programmed to FLASH, a CRC is performed and the results are stored in FLASH along with the application. The “check” command is used to verify that the current application image in FLASH is valid.
USAGE:	Veriy valid application is stored in FLASH COMMAND / RESPONSE: **check** **Check userimage CRC: ok**

26172.01.VC.030730/163444

CLONE

FUNCTION:	Program the NetBootLoader to FLASH
SYNTAX:	**clone [-n]** where: clone command -n option: program from data buffer
DESCRIPTION:	To update the NetBootLoader itself, the command “clone” is used. The application image source for programming is the data buffer. The image must first be downloaded to the data buffer from an ftp server. To program from the data buffer, the command “clone -n” is used. The new image is checked for validity. If an image is invalid, the update is aborted. Additionally, the operation must be confirmed by typing the word “yes”. Any other or no input will cancel the operation.
USAGE:	Program NetBootLoader (normal operation) COMMAND / RESPONSE: **NetBtLd> clone -n** **clone: Fixup FLASH info from ftp buffer** **This will overwrite the current ... NetBootLoader, are you sure? [no] yes** **clone: System transferred; Start again, ... assure that Bootjumper is removed.** **NetBtLd>** Note: When responding to the overwrite query, “yes” must be spelled out. Any other response will terminate the cloning operation.

CLONE

Program NetBootLoader (image not valid)

COMMAND / RESPONSE:

```
NetBtLd> clone -n
clone: Fixup FLASH info from ftp buffer
Image length invalid, image is damaged, abort.
NetBtLd>
```


GET

FUNCTION:	Download file from ftp server
SYNTAX:	**get <filename>** where: get command <filename> parameter: string name of image file to be downloaded, or path and name of image file to be downloaded
DESCRIPTION:	To download a file from the ftp server to the local data buffer, the command “get” is used. A successful ftp server login must be carried out before a file can be downloaded and the file must be in binary format. The argument “<filename>” must refer to an existing and accessible file on the server and the syntax must follow the requirements on the server, e.g. case sensitiveness. The argument may also include a path specification, if the server supports this.

26172.01.VC.030730/163444

HELP or ?

FUNCTION:	Display online help pages
SYNTAX:	**help** **?**
DESCRIPTION:	This command displays the online help pages. The display of the help text varies between the different CPU's reflecting their differences. The syntax of every command and a brief description is shown. The display output pauses after every page. The output can be continued with any key. Entering a "." (period) aborts the help function.

INFO

FUNCTION:	Display system information
SYNTAX:	**info**
DESCRIPTION:	The command "info" is used to display an information summary for the running system. The CPU type, the board type, and the detected FLASH layout are displayed.

26172.01.VC.030730/163444

LF

FUNCTION:	Load Flash
SYNTAX:	**`lf [-o[=]<offset> [-k]]`** **`… [-m[=]<adr> -l[=]<len>]`** where: lf command -o option: offset <offset> parameter: value: hexadecimal program to FLASH offset of ... -k option: keep retain surrounding contents -m option: memory (address) <adr> parameter: value: hexadecimal absolute address of image to be programmed -l option: length <len> parameter: value: hexadecimal length of image to be programmed
DESCRIPTION:	Without parameters, the FLASH is programmed using the contents of the data buffer. If no image is available in the data buffer, the FLASH programming is terminated. If no offset option (“-o”) is specified the image is considered to be valid and is therefore added along with CRC and length information. If the CRC is determined to be valid during the next startup, the image is copied to the absolute address 0x0 and started at 0x100 after the boot wait time has been exceeded. Normally, the local data buffer holds the image to be programmed. However, if the “-m” and “-l” parameters are specified, the image is programmed from the absolute address specified. If “<offset>” is specified, the contents are programmed exactly at this offset in FLASH. No length and no CRC information is added. The “-k” option can be specified to prevent deletion of the surrounding FLASH contents.

26172.01.VC.030730/163444

LF

DESCRIPTION:	FLASH memory can only be erased sector-wise. If an image is programmed to a certain offset with the “-o” option, at least this sector (and maybe one or more of the following sectors depending on the size of the image) will be erased. The “-k” option can be used to retain the surrounding data, however, this slows down the operation significantly. To achieve fast programming of parameter images without destroying other FLASH contents, the data should be placed at a sector boundary and the sector(s) must not contain any other data or executable images. If organized this way, use of the “-k” option can be avoided. Note: The “lf” command cannot be used to program the NetBootLoader.
USAGE:	Program FLASH from data buffer and add CRC and image length COMMAND / RESPONSE (none): **`lf`**
	Program FLASH from data buffer to offset 0xF4240 COMMAND / RESPONSE (none): **`lf -o=f4240`**
	Program FLASH from visible address at 0x87000000 for length of 0x123456 COMMAND / RESPONSE (none): **`lf -m=87000000 -l=123456`**
	Program FLASH from data buffer to offset 0xF4240 and retain adjacent FLASH contents COMMAND / RESPONSE (none): **`lf -o=f4240 -k`**

26172.01.VC.030730/163444

LOGIN

FUNCTION:	Initiate ftp server session
SYNTAX:	**login <ip-of-host> <username> [<password>]** where: login — command <ip-of-host> — parameter: value: numerical string IP address of host: nnn.nnn.nnn.nnn <username> — parameter: value: string ftp server “username” <password> — parameter: value: string user’s password
DESCRIPTION:	The command “login” is used to establish an ftp server session. The “<ip-of-host>” must be specified as four numbers separated by single dots. The “<password>” parameter is not necessary if the server does not request one.
USAGE:	Initiate ftp server session COMMAND / RESPONSE: **login 192.168.47.12 johndoe mypassword** (Response is dependent on the server accessed)

LOGOUT

FUNCTION:	Terminate telnet session with NetBootLoader
SYNTAX:	**logout**
DESCRIPTION:	A remote telnet session will be terminated with the command “logout”. No application is loaded and started if the session is terminated with “logout”. The NetBootLoader waits for a new session to be initiated or for a command entry from the serial console.

26172.01.VC.030730/163444

LS

FUNCTION:	Display listing of the current ftp server directory
SYNTAX:	**ls**
DESCRIPTION:	To display a listing of the current ftp server directory the command “ls” is used. This command downloads the listing to the data buffer and then the listing is displayed. Any previously loaded image in the data buffer is overwritten. If an attempt is then made to program the FLASH after the “ls” command has been issued it will fail.

MD

FUNCTION:	Display visible memory
SYNTAX:	**md [<adr>]** where: md command <adr> parameter: value: hexadecimal starting address of a visible memory area
DESCRIPTION:	To display a visible memory area the command “md” is used. The first time the command “md” is issued, visible memory contents starting at the address 0x0 are displayed if no “<adr>” parameter is used. If issued again without the “<adr>” parameter, the display starts with the end address of the previous display. Data is displayed as hexadecimal 32-bit words and as ASCII dump.

26172.01.VC.030730/163444

NET

FUNCTION:	Set or display the parameters for the Ethernet interface
SYNTAX:	**net [<ip-addr>][-netmask <netmask>]** **…[-gw <gateway>][-f]** where: net — command <ip-addr> — parameter: value: numerical string IP address of CPU board: nnn.nnn.nnn.nnn -netmask — option: netmask <netmask> — parameter: value: numerical string netmask of CPU board: nnn.nnn.nnn.nnn -gw — option: gateway <gateway> — parameter: value: numerical string gateway address for network: nnn.nnn.nnn.nnn -f — option: force CRC update
DESCRIPTION:	To set or display the parameters of the Ethernet interface the command “net” is used. Initially the CPU board does not have a valid Ethernet interface configuration, and, therefore, this interface is inoperable. The initial configuration must be done from the TERM interface using the command “net ... -f”. Using the “-f” option forces a CRC to be performed and stored along with the other configuration parameters in the serial EEPROM. Once the initialization of the Ethernet interface is done, the CPU board must be restarted for the parameters to take effect. Later changes to the parameters do not require the use of the “-f” option to force a CRC. This is done automatically. Only in the event that the Ethernet interface does not properly initialize, may it be necessary to re-enter the parameters using the “-f” option.

26172.01.VC.030730/163444

PASSWD

FUNCTION:	Set the telnet password
SYNTAX:	**passwd [-f | -d]** where: passwd command -f option: if password is not known -d option: disable disable telnet login (remote access)
DESCRIPTION:	To set the password for telnet sessions with the NetBootLoader the command “passwd” is used. This command is interactive, meaning that after it is issued, the NetBootLoader responds with an appropriate request to the operator which must be properly acknowledged or the operation fails (refer to USAGE below). To set the password in the event it is unknown, use the option “-f”. This is can only be accomplished from the TERM interface and not from the Ethernet interface. With the option “-d”, the remote telnet login can be disabled by invalidating the password.
USAGE:	Set password COMMAND / RESPONSE: **NetBtLd> passwd** **Old Password: ***** New Password: ***** Type again : ***** NetBtLd>** (The old password must be known)
	Set password when the old password is not known COMMAND / RESPONSE: **NetBtLd> passwd** **New Password: ***** Type again : ***** NetBtLd>**

PCI

FUNCTION:	Display PCI information
SYNTAX:	**pci**
DESCRIPTION:	The command “pci” is used to display detailed information on all detected PCI devices. The bus number, device number, function number, vendor, and device ID’s are displayed together with the configured base addresses and the assigned IRQ number.

PF

FUNCTION:	Set or display the serial port parameters (format)
SYNTAX:	**pf [<port> [<baud>][/[<bitschar>]** **…[/[<parity>][/<stops>]]]]**

where:

pf	command
<port>	parameter: string: “term” or “ser0” defines serial port to be configured
<baud>	parameter: value: numeric: “50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, 38400, 115200” defines the baud rate for the port
<bitschar>	parameter: value: numeric: “7” or “8” defines the number of bits per character
<parity>	parameter: string: “n” (none), “o” (odd), “e” (even) defines parity to be used
<stops>	parameter: value: number: “1”, “2” defines number of stop bits

26172.01.VC.030730/163444

PF

DESCRIPTION:	To set or display the operational parameters for the available serial interfaces the command “pf” is used. At startup the settings for the “TERM” and “SER0” interfaces are always set to the default values (9600/8/n/1). This is to avoid a possible system lockout. If other settings are required during operation of the NetBootLoader they may be made. If changes are made, it must be ensured that corresponding parameters are used for the operator consle. Issuing this command without parameters being specified will display the current serial port settings. Syntax-wise, no spaces are permitted between the parameters and they must be separated with a slash. Not all parameters must be specified, but the “/” characters must be present to distinguish the different parameters from each other. The sequence can be aborted after every option.
USAGE:	Set “TERM” to 300 Baud, 7 Bits/char, odd parity, and 2 stop bits COMMAND / RESPONSE (none): **`pf term 300/7/o/2`**
	Set the bits per character parameter of “SER0” to 7 COMMAND / RESPONSE (none): **`pf ser0 //7`**
	Set the stop bits parameter of “SER0” to 2 COMMAND / RESPONSE (none): **`pf ser0 ///2`**

PING

FUNCTION:	Verify operability of the Ethernet interface
SYNTAX:	**ping <ip_addr> [-c <count>] [-s <size>]** **… [-w <wait>]** where: ping — command <ip-addr> — parameter: value: numerical string IP address of target: nnn.nnn.nnn.nnn -c — option: count <count> — parameter: value: numeric: “[n ...]n” number of packets to send -s — option: size <size> — parameter: value: numeric: “[n ...]n”: bytes size of packet to send -w — option: wait <wait> — parameter: value: numeric: “[n ...]n”: seconds wait time between packets
DESCRIPTION:	To verify the operational status of the Ethernet interface the command “ping” is used. This command tests the network connection and target server’s ability to respond. If no other parameters are specified, four requests will be sent. This can be changed with the parameter “-c”. The typical size of a ping packet can be changed with the parameter “-s” and the time between requests, which is typically one second, can be changed with the parameter “-w”. Reponses to the “ping” command are dependent on the performance of the network.
USAGE:	Send four packets COMMAND / RESPONSE: **ping 192.192.158.7**
	Send ten packets, 100 bytes long, and wait two seconds between packets COMMAND / RESPONSE: **ping 192.192.158.7 -c 10 -s 100 -w 2**

26172.01.VC.030730/163444

PUT

FUNCTION:	Upload contents of the data buffer to the ftp server.
SYNTAX:	**put <filename>** where: put command <filename> parameter: string file name to be used for contents of data buffer to be uploaded
DESCRIPTION:	To upload the contents of the data buffer to a file on an ftp server, the command “put” is used. The file indicated by the parameter “<filename>” is created on the server. In the event that a file with this name already exists, its contents will be overwritten.

PWD

FUNCTION:	Display the current ftp server directory.
SYNTAX:	**pwd**
DESCRIPTION:	If a ftp connection has been established with the “login” command, the command “pwd” is used to display the complete path of the current directory on the ftp server.

RS

FUNCTION:	Reset the system
SYNTAX:	**rs**

26172.01.VC.030730/163444

RS

DESCRIPTION:	To permit the operator to force a restart of the system, the command “rs” is used. This command terminates the NetBootLoader command interpreter and resets the entire system, generating a system reset with the onboard watchdog. If this command is issued over a remote telnet connection, the telnet session is terminated prior to the generation of the reset.

SF

FUNCTION:	Store FLASH contents to data buffer
SYNTAX:	**sf -o[=]<offset> -l[=]<length>** where: sf — command -o — option: offset <offset> — parameter: value: hexadecimal relative offset to start of FLASH contents to be stored to the data buffer -l — option: length <length> — parameter: value: hexadecimal length of FLASH contents to be stored to the data buffer
DESCRIPTION:	With the command “sf” a selected portion of the FLASH contents may be copied to the local data buffer, e.g. for a subsequent upload to the ftp server with the “put” command. The “<offset>” parameter refers to the relative offset within the FLASH area similar to the “lf” command. The parameter “<length>” specifies the length to store.
USAGE:	Store 64 kB of FLASH contents to the data buffer beginning at an offset of 1 MB COMMAND / RESPONSE (none): **sf -o=100000 -l=10000**

SL

FUNCTION:	Download Motorola S-Records to data buffer
SYNTAX:	**sl [-o[=]<offset>] [-u]** where: sl command -o option: offset <offset> parameter: value: hexadecimal: unsigned offset to be subtracted from each record's address -u option: upper
DESCRIPTION:	With the command “sl” Motorola S-Records are downloaded to the data buffer and the record addresses modified accordingly as required for SDRAM operation (for copying to 0x0). The “<offset>” parameter may be used to change the record base to 0x0. The “-u” option selects the SER0 interface as source for the S-Records.
USAGE:	Download S-Records to data buffer and reduce each record’s address by 0x10000. COMMAND / RESPONSE (none): **sl -o=10000**

VER

FUNCTION:	Display version number
SYNTAX:	**ver**
DESCRIPTION:	The command “ver” displays the actual version number of the NetBootLoader.

26172.01.VC.030730/163445

Chapter A

CP320-IO1 Module

This page was intentionally left blank.

A. CP320-IO1 Module

A.1. Overview

The optional *Kontron* CP320-IO1 module has been designed to provide the CP321 user with an effective gateway to the world of PMC modules. This additional capability opens up the broadest range of expansion possibilities.

PMC modules are renowned for their flexibility and versatility of use. They afford the user wide ranging system-independent solutions by means of easily interchanged or upgraded add-on modules. The *Kontron* CP320-IO1 has been designed to maximize the advantages provided by PMC modules in a 3U environment.

A special feature of the CP321 is the ability to cascade two of these IO1 modules on top of one another. This means that the CP321 is able to carry any two PMC modules. Tremendous advantages in terms of expandability and flexibility are thus made available to the user as a result of the addition of this capability to the board's many outstanding features.

The CP320-IO1 is a non-intelligent, passive 3U CPCI carrier board with one PMC slot.

Some of the Outstanding Features of the CP320-IO1

- 32 Bit / 33MHz PCI Bus on the PMC side
- IO voltage, V(I/O), 5V, or 3.3V, can be configured on the base board (CP321)
- It supports the Interrupts INTA, INTB, INTC and INTD
- It supports all the signals of the PCI Bus on its connectors Jn1 (CON4), Jn2 (CON5)
- The connectors which connect the mezzanine board with the carrier include all the signals of a 33MHz, 32-bit, multi-master PCI bus; the power rails for 5V, 3.3V, and V(I/O); and other specialized signals for board detection.

Features of the *Kontron Modular Computers*' PMC modules

Kontron Modular Computers' PMC modules are operable in CompactPCI systems and VME systems which support PCI busses. They offer all the key benefits of PC I/O technology, namely:

- low cost solutions
- high performance
- a processor independent local I/O bus
- a broad range of I/O peripheral devices

Kontron Modular Computers' PMC modules may be installed on a variety of different carrier boards, including:

- CompactPCI 3U/6U: CPU CP302, CP600, CP602, CP610, CP611, CP612
- CompactPCI PMC carrier boards such as the CP390 and CP690
- VME 3U: VMP1 by means of the VMP1-IO1 module

A.2. Board Interfaces

PCI Expansion Connector

The PCI expansion connector CON2/3 provide all the necessary signals for data transfer as defined by PCI Specification Rev. 2.1. This connector combination allows for board stacking (cascading) with CON2 providing the connection to the carrier board or previous CP320-IO1 and CON3 providing connection to the next CP320-IO1 carrier. The pinouts of CON2 and CON3 are the same.

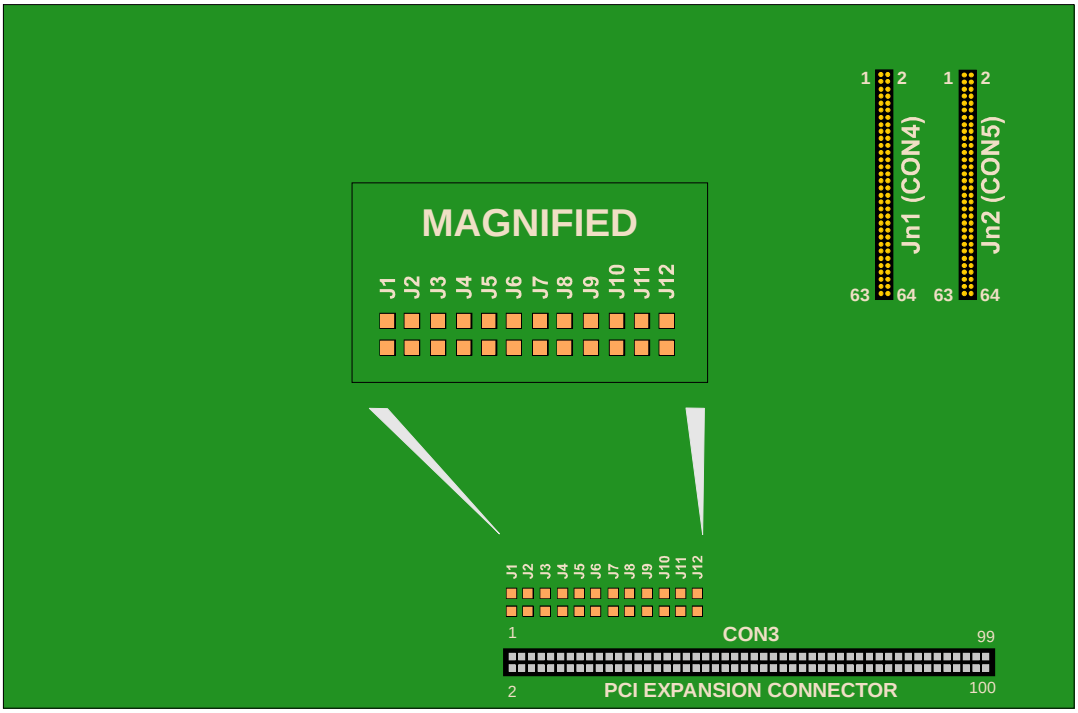
PMC Interface

The PMC interface provides an easy way to extend the CP321 via the wide array of interfaces and functions which are available on PMC modules produced by the entire range of PMC vendors. PMC connectors provide a 32-bit wide PCI data path with a speed of up to 33MHz which is routed to the onboard connectors Jn1 and Jn2. These connectors also provide the power supply for the PMC module. The interface has been designed to comply with the IEEE 1386.1 specification which defines a PCI electrical interface for the CMC (Common Mezzanine Card) form factor.

A.3. Board Layout

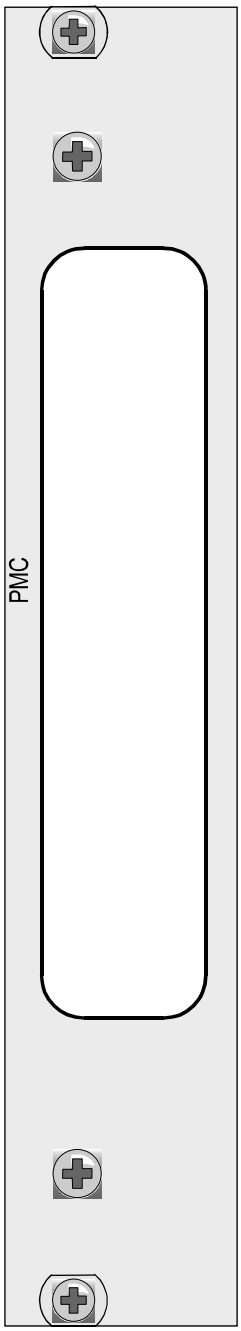
The CP320-IO1 has two onboard connectors (CON4 and CON5) which provide all the PCI signals and the power supply for the PMC module. The CON3 connector in conjunction with the CON2 connector (on reverse side of board) make board stacking possible

Figure A-1: Board Layout (Front View)



A.4. CP320-IO1 Front Panel

Figure A-2: CP320-IO1 Front Panel



The CP320-IO1 front panel is provided with a window for the insertion of a PMC module bezel.

A.5. Pinouts

Table A-1: Jn1 (CON4), 32-bit PCI

PIN	SIGNAL	SIGNAL	PIN
1	TCK	-12V	2
3	Ground	INTA#	4
5	INTB#	INTC#	6
7	BUSMODE1#	+5V	8
9	INTD#	PCI-RSVD	10
11	Ground	PCI-RSVD	12
13	CLK	Ground	14
15	Ground	GNT#	16
17	REQ#	+5V	18
19	V(I/O)	AD[31]	20
21	AD[28]	AD[27]	22
23	AD[25]	Ground	24
25	Ground	C/BE[3]#	26
27	AD[22]	AD[21]	28
29	AD[19]	+5V	30
31	V(I/O)	AD[17]	32
33	FRAME#	Ground	34
35	Ground	IRDY#	36
37	DEVSEL#	+5V	38
39	Ground	LOCK#	40
41	SDONE#	SBO#	42
43	PAR	Ground	44
45	V(I/O)	AD[15]	46
47	AD[12]	AD[11]	48
49	AD[09]	+5V	50
51	Ground	C/BE[0]#	52
53	AD[06]	AD[05]	54
55	AD[04]	Ground	56
57	V(I/O)	AD[03]	58
59	AD[02]	AD[01]	60
61	AD[00]	+5V	62
63	Ground	REQ64#	64

Table A-2: Jn2 (CON5), 32-bit PCI

PIN	SIGNAL	SIGNAL	PIN
1	+12V	TRST#	2
3	TMS	TDO	4
5	TDI	Ground	6
7	Ground	PCI-RSVD	8
9	PCI-RSVD	PCI-RSVD	10
11	BUSMODE2#	+3.3V	12
13	RST#	BUSMODE3#	14
15	3.3V	BUSMODE4#	16
17	PCI-RSVD	Ground	18
19	AD[30]	AD[29]	20
21	Ground	AD[26]	22
23	AD[24]	+3.3V	24
25	IDSEL	AD[23]	26
27	+3.3V	AD[20]	28
29	AD[18]	Ground	30
31	AD[16]	C/BE[2]#	32
33	Ground	PMC-RSVD	34
35	TRDY#	+3.3V	36
37	Ground	STOP#	38
39	PERR#	Ground	40
41	+3.3V	SERR#	42
43	C/BE[1]#	Ground	44
45	AD[14]	AD[13]	46
47	Ground	AD[10]	48
49	AD[08]	+3.3V	50
51	AD[07]	PMC-RSVD	52
53	+3.3V	PMC-RSVD	54
55	PMC-RSVD	Ground	56
57	PMC-RSVD	PMC-RSVD	58
59	Ground	PMC-RSVD	60
61	ACK64#	+3.3V	62
63	Ground	PMC-RSVD	64

Table A-3: PCI Expansion Connector (CON2/3) Pinout

SIGNAL	PIN	PIN	SIGNAL
GND 1)	1	2	SCL (I2C)
RST#	3	4	+3.3V 2)
+3.3V 2)	5	6	CLK2
CLK3	7	8	GND 1)
GND 1)	9	10	CLK4
INTB#	11	12	INTA#
INTD#	13	14	INTC#
+5V 3)	15	16	GNT#2
GNT#3	17	18	V(I/O) 6)
+3.3V 2)	19	20	GNT#4
GND 1)	21	22	REQ#2
REQ#3	23	24	GND 1)
+5V 3)	25	26	REQ#4
AD31	27	28	AD30
AD29	29	30	+5V 3)
GND 1)	31	32	AD28
AD27	33	34	AD26
AD25	35	36	GND 1)
+3.3V 2)	37	38	AD24
C/BE3#	39	40	SDA (I2C)
AD23	41	42	+3.3V 2)
GND 1)	43	44	AD22
AD21	45	46	AD20
AD19	47	48	GND 1)
V(I/O) 6)	49	50	AD18
AD17	51	52	AD16
C/BE2#	53	54	+5V 3)
GND 1)	55	56	FRAME#
IRDY#	57	58	GND 1)
+3.3V 2)	59	60	TRDY#
DEVSEL#	61	62	reserved

Table A-3: PCI Expansion Connector (CON2/3) Pinout (Continued)

SIGNAL	PIN	PIN	SIGNAL
GND 1)	63	64	STOP#
LOCK#	65	66	+3.3V 2)
PERR#	67	68	V(I/O) 6)
SERR#	69	70	GND 1)
+5V 3)	71	72	PAR
C/BE1#	73	74	AD15
AD14	75	76	+3.3V 2)
GND 1)	77	78	AD13
AD12	79	80	AD11
AD10	81	82	GND 1)
GND 1)	83	84	AD9
AD8	85	86	C/BE0#
AD7	87	88	+5V 3)
+3.3V 2)	89	90	AD6
AD5	91	92	AD4
AD3	93	94	GND 1)
reserved	95	96	AD2
AD1	97	98	AD0
+12V 4)	99	100	-12V 5)

Key

1) Ground	***4) +12V***
2) +3.3V	***5) -12V***
3) +5V	***6) V(I/O)***

A.6. Technical Specifications

Table A-4: CP320-IO1 Specifications

CP320-IO1	SPECIFICATIONS
PCI-Standard	Compliant with PCI 2.1
Signaling Voltage	Depends on the internal base board signaling voltage (5V default)
Connectors	PMC Jn1 (CON4) and Jn2 (CON5) connectors; and PCI Expansion Connectors (CON2/3)
Mechanical Compliance	IEEE 1101.10 CMC IEEE P1386/Draft 2.0
Temperature Range	0°C to +70°C Standard -25°C to +75°C E1 -40°C to +85°C E2
Operating Humidity	0% to 95% non-condensing
Vibration and Broad-Band Random Vibration	IEC68-2-6 compliant IEC68-2-64
Shock: Permanent Shock Single Shock	 IEC68-2-29 IEC68-2-27
Board Dimensions	Single-height Eurocard: 100 mm x 160 mm 1 x 4 HP slot
Board Weight	122 grams without PMC module

A.7. Board Installation

In order to keep the installation process as simple and easy as possible please follow the recommended order of work:

ESD Equipment!

Your carrier board and PMC module contain electrostatically sensitive devices. Please observe the necessary precautions to avoid damage to your board:

- Discharge your clothing before touching the assembly. Tools must be discharged before use.
- Do not touch components, connector-pins or traces.
- If working at an anti-static workbench with professional discharging equipment, please do not omit to use it.

PMC Module Installation

1. Place the EMC gasket on the bezel of your PMC-Module.
2. Push the PMC bezel into the window of the front panel of the CP320-IO1 and plug the connectors together.
3. Use four screws (M2.5 × 6mm) to secure the module to the board.

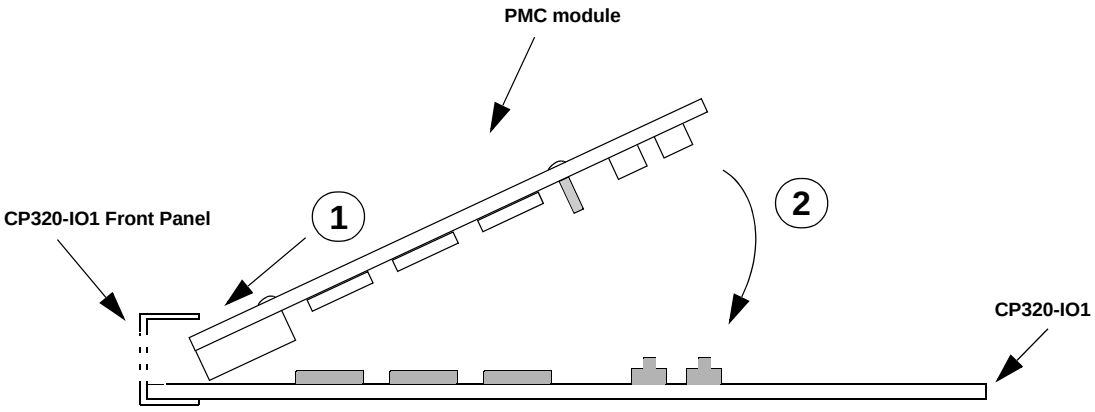
Installation of the CP320-IO1 Module on the CP321 Baseboard

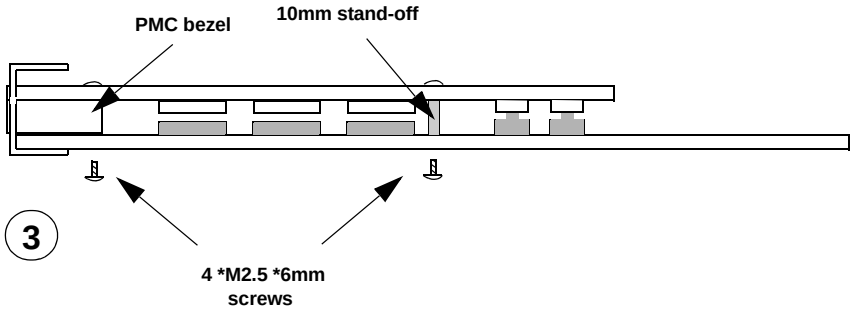
1. Place the CP320-IO1 exactly above the CP321
2. Plug them togethor
3. Use 4 screws (2.5 × 6 mm) to secure the board to the CP321

Note...

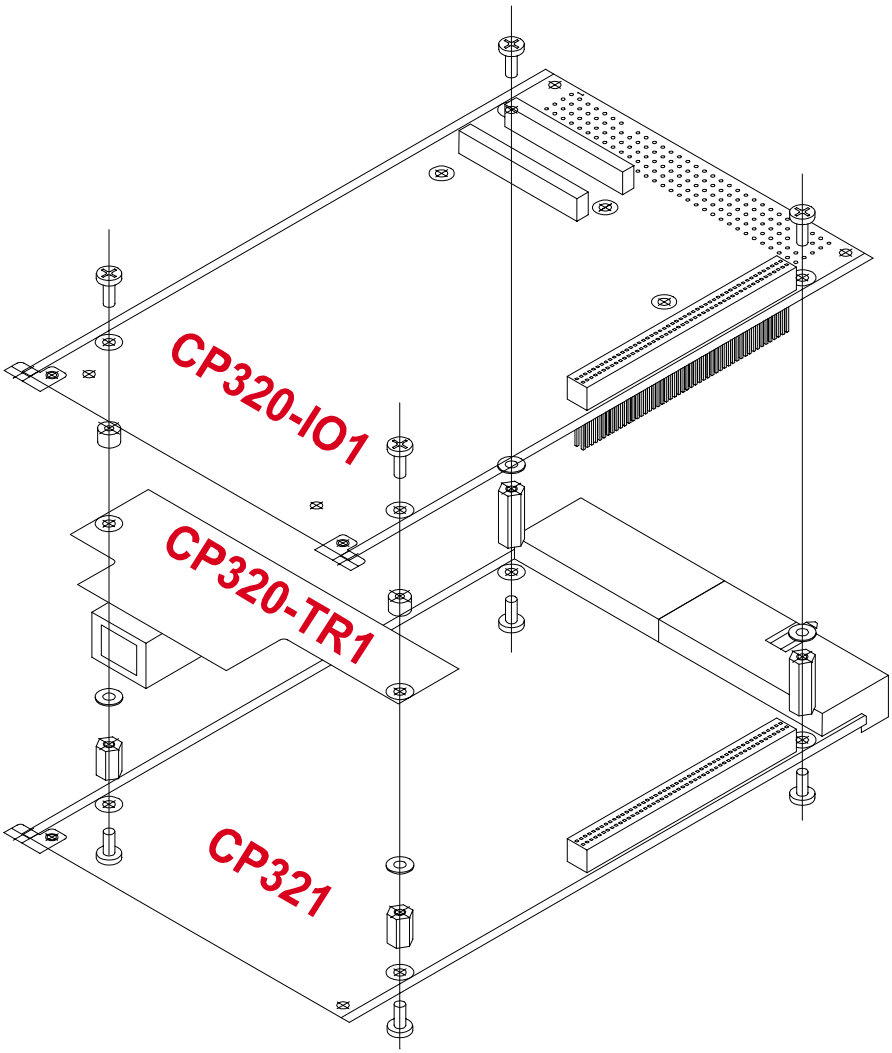
For further information regarding the installation of the CP320-IO1 board refer to the CP321 Installation chapter.

Figure A-3: Installation of PMC Module





The following diagram illustrates how the CP320-IO1 mezzanine carrier board and the CP320-TR1 RS485 optoisolated transition module are assembled with the CP321 main board. Assembly of the second carrier board follows on top of first carrier board.



A.8. Jumper Setting

The jumper settings of the IO1 module depend on the module's position relative to the CP321 and other modules, if any (please see Figure A4 below).

Table A-5: IO1 Jumper Settings for Different Module Positions

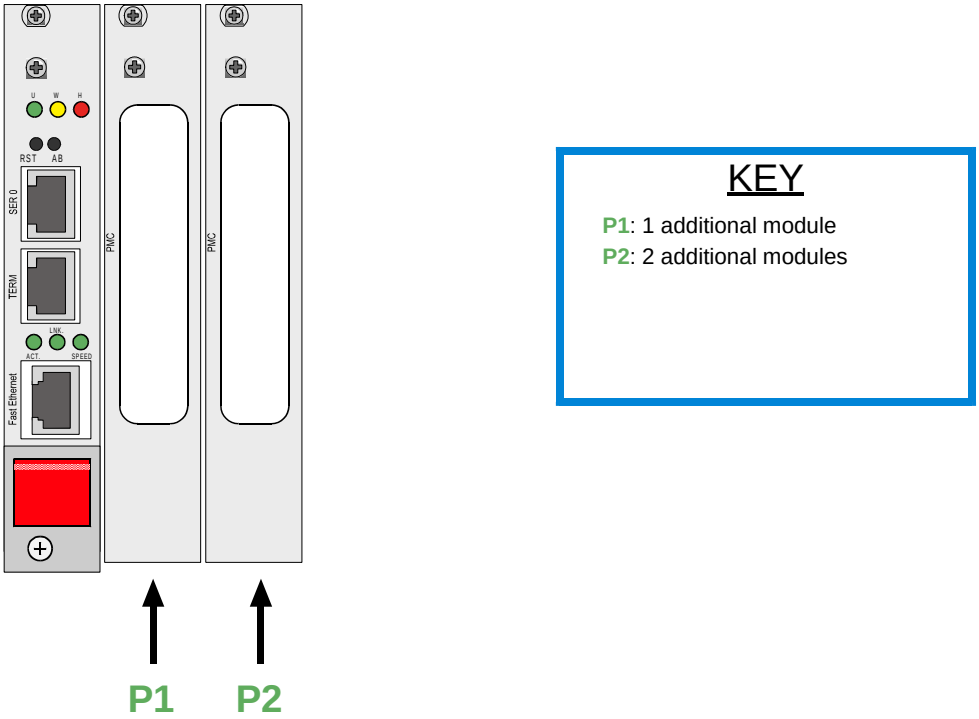
POSITION	IDSEL			CLOCK			GNT#			REQ#		
	J12	J11	J10	J1	J2	J3	J4	J5	J6	J7	J8	J9
P1	Set	Open	Open	Set	Open	Open	Set	Open	Open	Set	Open	Open
P2	Open	Set	Open	Open	Set	Open	Open	Set	Open	Open	Set	Open

Note...

Position P1 refers to the settings applicable for a module attatched to the CP321 in position P1.

Position P2 refers to the settings applicable for a module attatched to the CP321 in position P2.

Figure A-5: Cascading of IO1 (or other) Modules onto the CP321



Chapter B

CP320-TR1 (Optional)

This page was intentionally left blank.

B. CP320-TR1 (Optional)

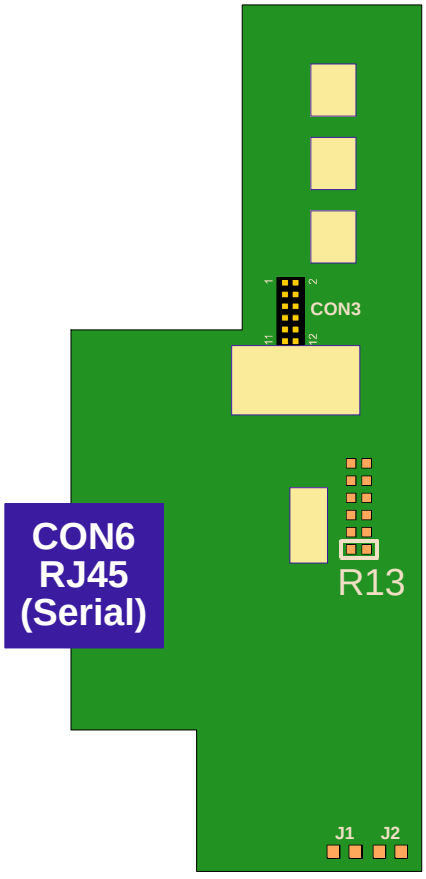
B.1 Board Description

On the CP321 it is possible to utilize a transition module which provides optoisolated RS485 functionality either half or full duplex where half duplex is the default.

Users who require a CP321 with an opto-isolated serial interface are supplied with a customized CP321 on which the standard RJ45 connector is omitted (SER) and also with this module which comes with a substitute RJ45 connector routed through optoisolation circuitry on the module.

The module has been designed so that it does not increase the board width, which remains unchanged at 4HP with the module in place.

Figure B-1: View of Underside of the CP320-TR1 Module



Refer to the Functional Description chapter for the pinout of CON3.

Table B-1: Duplex Configuration

DUPLEX FUNCTION	R13 SETTING
Full	Open
Half	Set

Table B-2: Serial Port Pinout

RS485		
PIN	SIGNAL	
	HALF-DUPLEX	FULL-DUPLEX
1	N/C	-RxD
2	N/C	N/C
3	GND	GND
4	+TRXD	-TxD
5	N/C	N/C
6	N/C	+RxD
7	-TRXD	+TxD
8	N/C	N/C

Table B-3: CP320-TR1 Jumper Settings

FUNCTION	JUMPER SETTING	
	J1	J2
120 ohm termination, full-duplex	Set	Set
120 ohm termination, half-duplex	Set	Open
No termination	Open	Open

Chapter C

CP320-TR2 (Optional)

This page was intentionally left blank.

C. CP320-TR2 (Optional)

C.1 Board description

On the CP321 it is possible to utilize a transition module which provides optoisolated RS232 functionality for the SER0-Interface. All signals for utilizing hardware handshake protocol are available in optoisolated form.

Supported Signals

The following signals are provided by the CP320-TR2:

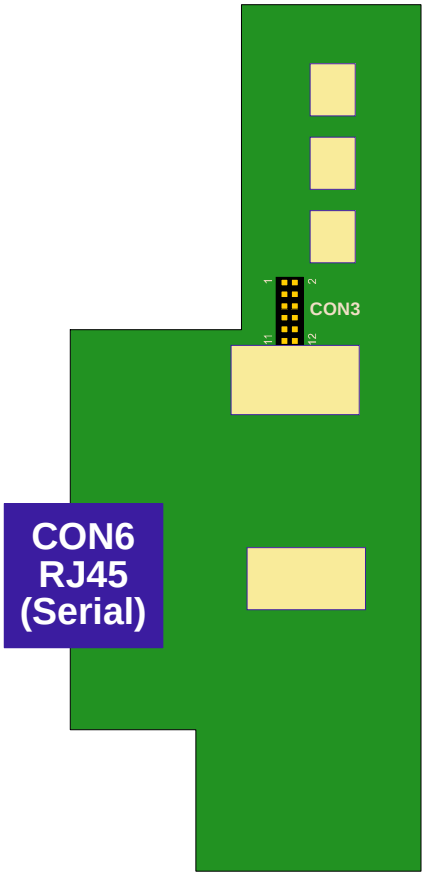
- TxD - Transmit Data
- RxD - Receive Data
- RTS - Request to Send (used on PC based systems for hardware handshaking)
- CTS - (used on PC and *Kontron* systems for hardware handshaking)
- DTR - (used on *Kontron* systems for hardware handshaking)

The board itself is available in the E2 temperature range.

Users who require a CP321 with an opto-isolated serial interface are supplied with a customized CP321 on which the standard RJ45 connector is omitted (SER) and also with this module which comes with a substitute RJ45 connector routed through optoisolation circuitry on the module.

The module has been designed so that it does not increase the board width, which remains unchanged at 4HP with the module in place. View of Underside of the CP320-TR2 Module.

Figure C-1: View of Underside of CP320-TR2 Module



Serial Port Pinout

Table C-1: Serial Port Pinout

RS232	
Pin	Signal
1	NC
2	RTS
3	ISO-GND
4	TxD
5	RxD
6	NC
7	CTS
8	DTR

Please note that this diagram is not to scale with other board diagrams

Chapter D

Post Module

This page was intentionally left blank.

D. Post Module

D.1 Board description

The CP320-Post is an optionally available tool which is used for hardware and software debugging. During the startup process of the CP321 it provides the user with information about the status of the boot process by means of a message code similar to the POST codes on the Intel PC. When the board has completed the startup process, the CP320-Post may be used to provide debug information for software development. The programmer can, therefore, define his own debug code and send it to the CP320-Post by making a byte write command to the first address of the socket memory area. The address for accessing the Post module depends on the DIL socket (1 or 2) in which it is inserted and the setting of jumper J1 on the CP321.

Table D-1: Access Addresses for CP320-Post

MEMORY EXPANSION POSITION	J1 SETTING	ADDRESS
Socket 1	Open	0xFFF8 0000
	Set	0xFFF0 0000
Socket 2	not applicable	0xFFE8 0000

Note...

The CP320-Post will only work properly on sockets which are configured for Flash or DiskOnChip modules.

Warning!

TURN POWER OFF

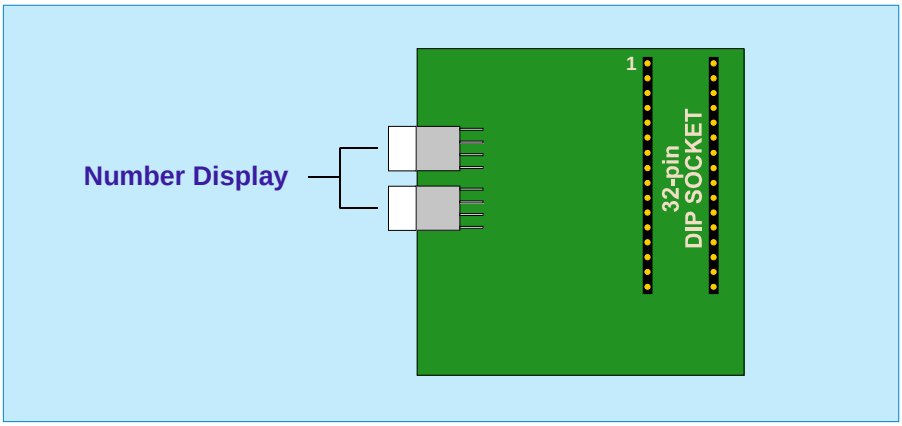
Always ensure that power is switched off before installing the CP320-Post module.

SHORT CIRCUIT DANGER

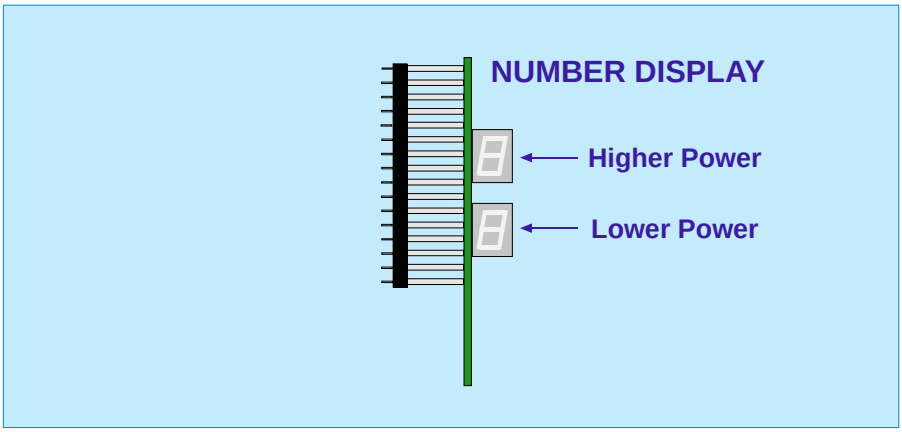
When installing and operating the CP320-Post module ensure that the module does not make contact with the front panel or any other portions of the CP321 except the connections to the DIL socket.

Figure D-1: Plan and Profile Views of CP320-Post Module

PLAN



PROFILE



D.2 POST Codes

The following is a list of POST codes currently in use.

Table D-2: POST Code LIsting

POST CODE	DESIGNATOR	DESCRIPTION
1	PR_COLD_ON	Reset entry reached
2	PR_INIT_ON	Board initialization reached
3	PR_INIT_FLUSH	Board initialization, caches flushed & disabled
4	PR_INIT_CSOFF	Board initialization, CS lines switch off
5	PR_INIT_CS0	Board initialization, CS0 done
6	PR_INIT_CS1	Board initialization, CS1 done
7	PR_INIT_CSON	Board initialization, CS lines on
8	PR_INIT_MEMINZ	Board initialization, memory controller initialized
9	PR_INIT_PRIF	Board initialization, processor interface
10	PR_COLD_BOARD	Board initialization done
11	PR_COPREL	Copy and relocations done, about to jump
12	PR_ICAON	I cache switched on
30	PR_PPC_GO	Start in ‘C’ part with ppcGo
31	PR_MMU_INIT	MMU initialization start
32	PR_MMU_REGSW0	MMU registers 0 written
33	PR_MMU_REGSW1	MMU registers 1 written
34	PR_MMU_REGSW2	MMU registers 2 written
35	PR_MMU_REGSW3	MMU registers 3 written
36	PR_MMU_TREN	MMU translation enabled
40	PR_PCI_INIT	PCI start initialisation
41	PR_PCI_PREBR	PCI bridges preset
42	PR_PCI_TABLE	PCI fill table
43	PR_PCI_UNIF	PCI universe found
44	PR_PCI_WRITE	PCI write configuration to devices
45	PR_PCI_BWRITE	PCI write configuration to bridges
46	PR_PCI_CIRQ	PCI write IRQ info to devices
47	PR_PCI_UNII	PCI configure universe 2
50	PR_CROOT_ENTER	CROOT entry
51	PR_CROOT_EXPT	CROOT exception handler installed
52	PR_CROOT_FLDET	CROOT flash detected
53	PR_CROOT_KEYF	CROOT startkey search

This page was intentionally left blank.

Chapter E

PMC-HDD1 Module

This page was intentionally left blank.

E. PMC-HDD1 Module

E.1 Board description

The optional PMC-HDD1 module provides the *Kontron* PowerPC-based CPU boards with a cost-effective way to add substantial mass storage capacity. It is designed to connect a 2.5” IDE hard disk drive to the PCI bus of those boards. It is based on the silicon image IDE controller SiI0680, which provides the interface between the 32 bit wide, 33 MHz PCI bus and a standard IDE hard disk drive. It is able to handle transfer rates up to the ATA-133 speed standard.

Note...

- The maximum transfer rate which can be achieved with this module is restricted by the hard drive in use.
- The capacity of the module is defined by the hard drive in use.

Figure E-1: PMC-HDD1 Module with Hard Disk Drive Attached

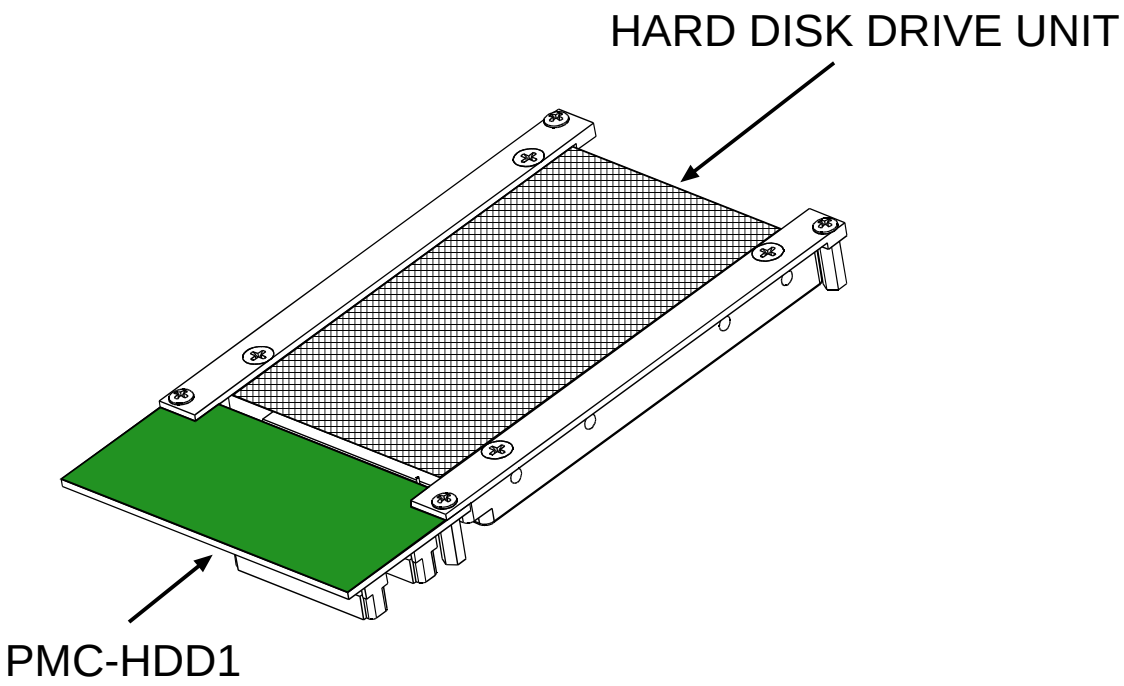


Table E-1: Pinout of the PMC Connectors

PN1/JN1 (CON1)				PN2/JN2 (CON2)			
Pin #	Signal Name	Signal Name	Pin #	Pin #	Signal Name	Signal Name	Pin #
1	Signal	-12V	2	1	+12V	Signal	2
3	Ground	Signal	4	3	Signal	Signal	4
5	Signal	Signal	6	5	Signal	Ground	6
7	BUSMODE1#	+5V	8	7	Ground	Signal	8
9	Signal	Signal	10	9	Signal	Signal	10
11	Ground	Signal	12	11	BUSMODE2#	+3.3V	12
13	Signal	Ground	14	13	Signal	BUSMODE3#	14
15	Ground	Signal	16	15	+3.3V	BUSMODE4#	16
17	Signal	+5V	18	17	Signal	Ground	18
19	V (I/O)	Signal	20	19	Signal	Signal	20
21	Signal	Signal	22	21	Ground	Signal	22
23	Signal	Ground	24	23	Signal	+3.3V	24
25	Ground	Signal	26	25	Signal	Signal	26
27	Signal	Signal	28	27	+3.3V	Signal	28
29	Signal	+5V	30	29	Signal	Ground	30
31	V (I/O)	Signal	32	31	Signal	Signal	32
33	Signal	Ground	34	33	Ground	Signal	34
35	Ground	Signal	36	35	Signal	+3.3V	36
37	Signal	+5V	38	37	Ground	Signal	38
39	Ground	Signal	40	39	Signal	Ground	40
41	Signal	Signal	42	41	+3.3V	Signal	42
43	Signal	Ground	44	43	Signal	Ground	44
45	V (I/O)	Signal	46	45	Signal	Signal	46
47	Signal	Signal	48	47	Ground	Signal	48
49	Signal	+5V	50	49	Signal	+3.3V	50
51	Ground	Signal	52	51	Signal	Signal	52
53	Signal	Signal	54	53	+3.3V	Signal	54
55	Signal	Ground	56	55	Signal	Ground	56
57	V (I/O)	Signal	58	57	Signal	Signal	58
59	Signal	Signal	60	59	Ground	Signal	60
61	Signal	+5V	62	61	Signal	+3.3V	62
63	Ground	Signal	64	63	Ground	Signal	64

.

Table E-2: IDE Hard Disk Drive Connector Pinout

Pin Number	Signal	Function	In/Out
1	IDERESET	Reset HD	Out
2	GND	Ground signal	--
3	HD7	HD data 7	In/Out
4	HD8	HD data 8	In/Out
5	HD6	HD data 6	In/Out
6	HD9	HD data 9	In/Out
7	HD5	HD data 5	In/Out
8	HD10	HD data 10	In/Out
9	HD4	HD data 4	In/Out
10	HD11	HD data 11	In/Out
11	HD3	HD data 3	In/Out
12	HD12	HD data 12	In/Out
13	HD2	HD data 2	In/Out
14	HD13	HD data 13	In/Out
15	HD1	HD data 1	In/Out
16	HD14	HD data 14	In/Out
17	HD0	HD data 0	In/Out
18	HD15	HD data 15	In/Out
19	GND	Ground signal	--
20	N/C	--	--
21	IDEDRQ	DMA request	In
22	GND	Ground signal	--
23	IOW	I/O write	Out
24	GND	Ground signal	--
25	IOR	I/O read	Out
26	GND	Ground signal	--
27	IOCHRDY	I/O channel ready	In
28	GND	Ground signal	--
29	IDEDACKA	DMA Ack	Out
30	GND	Ground signal	--
31	IDEIRQ	Interrupt request	In
32	N/C	--	--
33	A1	Address 1	Out
34	N/C	--	--
35	A0	Address 0	Out
36	A2	Address 2	Out
37	HCS0	HD select 0	Out
38	HCS1	HD select 1	Out
39	NC	--	In
40	GND	Ground signal	--
41	VCC	5V power	--
42	VCC	5V power	--
43	GND	Ground signal	--
44	N/C	--	--

This page was intentionally left blank.