EB8347

E²Brain Power QUICC II™ Pro
e300 Core Comminications Engine

Doc.ID: 33386, Rev. 1.0
July 18, 2006

User Guide

kontron

Revision History

Publication Title:	EB8347: E²Brain PowerQUICC II™ e300 Core Communications Engine	
Doc.ID:	33386	
Rev.	**Brief Description of Changes**	**Date of Issue**
1.0	Initial issue	18-JUL-2006

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 E-MAIL

+49 (0) 800-SALESKONTRON

sales@kontron.com

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

Disclaimer

Table of Contents

This page has been intentionally left blank.

List of Tables

This page has been intentionally left blank.

List of Figures

This page has been intentionally left blank.

Proprietary Note

This document contains information proprietary to Kontron. 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 or one of its authorized agents.

The information contained in this document is, to the best of our knowledge, entirely correct. However, Kontron 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 reserves the right to change, modify, or improve this document or the product described herein, as seen fit by Kontron without further notice.

Trademarks

This document may include names, company logos and trademarks, which are registered trademarks and, therefore, proprietary to their respective owners.

Environmental Protection Statement

This product has been manufactured to satisfy environmental protection requirements where possible. Many of the components used (structural parts, printed circuit boards, connectors, batteries, etc.) are capable of being recycled.

Final disposition of this product after its service life must be accomplished in accordance with applicable country, state, or local laws or regulations.

Explanation of Symbols

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.

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 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 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. In the event that the original packaging material is not available for storage or warranty shipments, packaging which complies with the standards indicated in section 1.8 may be used to ensure the proper protection of this product.

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 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.

Kontron 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 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, 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 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 liability to the customer shall not exceed the original purchase price of the item for which the claim exists.

Kontron 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 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 has been intentionally left blank.

Chapter 1

Introduction

This page has been intentionally left blank.

1. Introduction

The E²Brain™ module described in this guide is designed to Kontron Modular Computers' "E²Brain™ Module Specification", revision 01. Section 1.1 provides general information regarding this specification and the system environment as envisioned for E²Brain™ modules. For more detailed information regarding the "E²Brain™ Module Specification", please contact Kontron Modular Computers.

The remaining sub-sections of the Introduction provide more specific details of the EB8347 E²Brain™ module itself to familiarize the user with this product as a whole.

1.1 The E²Brain™ Concept

The E²Brain™ concept is a highly flexible approach to providing application developers with the ability to concentrate on the definition of application requirements without having to continuously factor in potential restrictions concerning available data processing and communications functionality.

More specific, data processing and communications requirements become a function of the application and not vice versa. This is possible through the implementation of the E²Brain™ concept. Unlike other approaches to providing application solutions, the E²Brain™ concept concentrates on the most essential aspects of providing data processing and communications without attempting to provide in one entity a complete, self-contained computer system.

The E²Brain™ specification first of all defines a PCB module with a form factor of 115 x 75 x 11.6 millimeters. For interfacing with applications, the specification calls for up to four connectors which provide not only interfacing capability for current industry standards but also for future standards or application specific requirements. The type, location, and usage of these connectors is also defined in the specification so as to guarantee standardized compatibility. The specification is open as to the data processing and communications functionality to be implemented which is by definition a function of the application requirements. In addition, the specification envisions considerations for thermal energy dissipation through the implementation of what are known as BrainCAP™s (E²Brain™ **C**ooling **A**ssembly, **P**rotector) which can range from heat spreaders to highly sophisticated heat sink cooling designs.

The key features of the E²Brain™ concept are:

- very compact and robust form factor
- independent of CPU architecture
- scalable, flexible, and open system interface
- PCI Master and Agent Mode
- PCI-64 and PCI-X capability
- versatile and very powerful communications interfaces
- complete thermal design concept

1.1.1 E²Brain™ Functionality

E²Brain™ (Embedded Electronic Brain) is a new platform architecture for advanced computer modules. The E²Brain™ specification defines a very compact mechanical form factor and a comprehensive set of functional interfaces which can be adapted to a wide variety of applications. E²Brain™ modules provide complete computer cores integrating a high-performance CPU, system memory and - typical for E²Brain™ - advanced communications controllers. E²Brain™ modules are plugged into customized backplanes or standardized carrier boards

which themselves provide the physical interfaces (PHYs) and connectors, power, and additional IO controllers. Through the use of E²Brain™ modules the system developer is relieved of the task of designing computers, and, instead, they permit him to concentrate on the specific product development.

E²Brain™ is a computer platform dedicated not just to one architecture like the PC and Windows architecture, but it is open for all architectures including PowerPC, ARM, SH, x86, and others. E²Brain™ modules are very suitable for “deeply” embedded applications requiring flexible computing power combined with versatile and high-performance communications power.

Although typical E²Brain™ modules are designed to be low power consumption devices, they are part of a well thought out thermal design concept which considers the thermal aspects right from the beginning. Where higher power consumption is unavoidable, E²Brain™ modules are fitted with appropriate BrainCAP™s. By utilizing BrainCAP™s, cooling, mechanical stabilization, and EMI protection are combined in a single concept to satisfy almost any application requirement.

1.1.2 Basic Architecture

The following figure illustrates the basic functional architecture of E²Brain™ modules. Common to all E²Brain™ modules are the data processing and communications core and the system and communications interfaces.

The application requirements determine the functionality required of the E²Brain™ module core which in turn mandates the functionality to be provided by the system and communications interfaces. Both of these interfaces are comprised of a base set and an extended set of functional features.

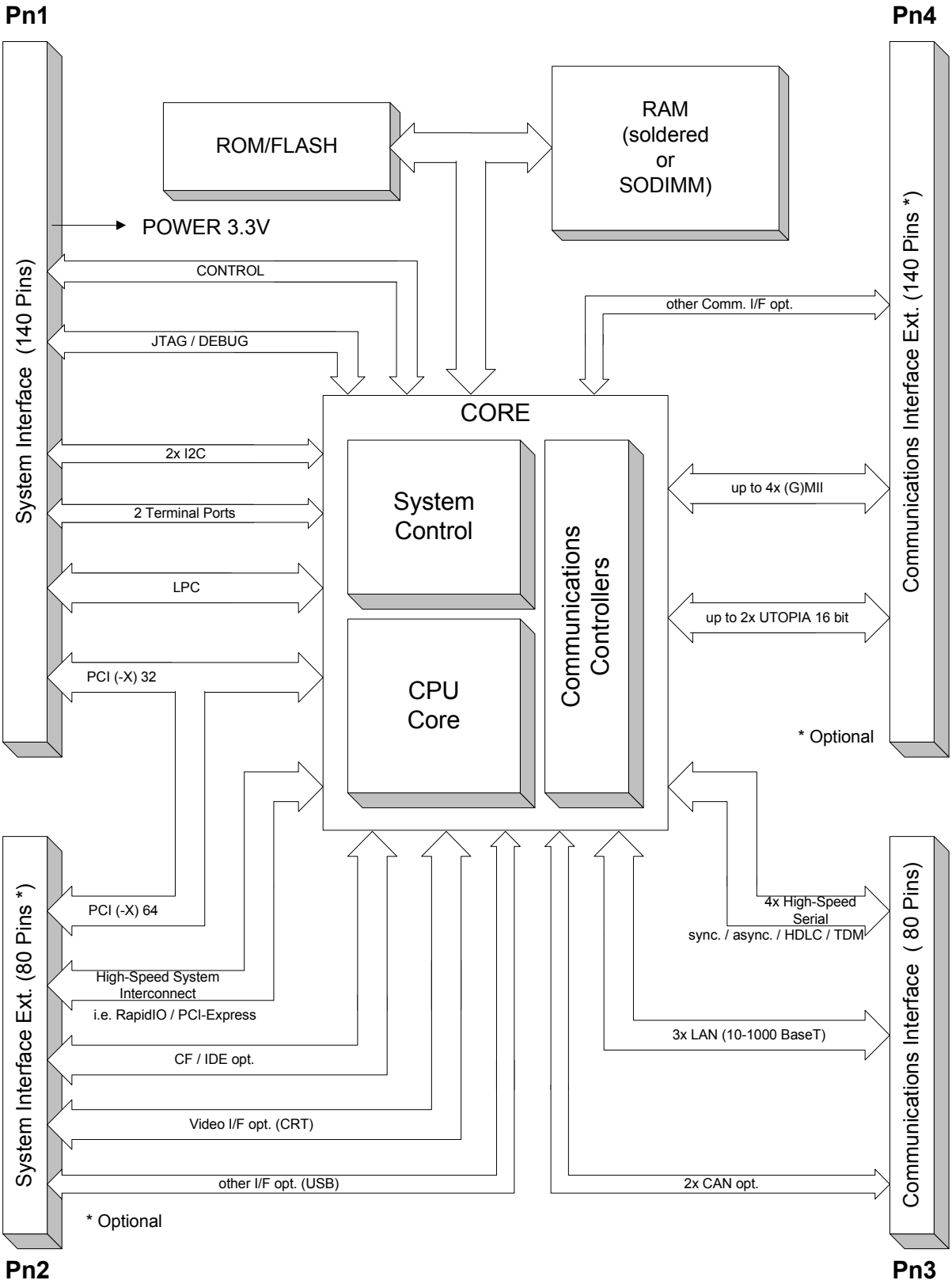
The system interface to the application is accomplished through connectors Pn1 and Pn2. Connector Pn1 provides the base set of system interfacing and Pn2 the extended set. If the application does not require extended system interfacing, it is not necessary to populate connector Pn2. The same analogy applies to the communications interfacing whereby connector Pn3 provides the base set of communications interfacing and Pn4 the extended set. Pn4 is not required to be populated if there is not an application requirement for it. This concept together with a corresponding core provides a maximum of scalability and flexibility to satisfy the most demanding of applications.

1.1.3 E²Brain™ System Environment

The E²Brain™ module form factor and mechanical and electrical interfacing are so conceived as to allow the use of E²Brain™ modules in practically any kind of system environment. These mezzanine modules can easily be integrated on most standardized carrier boards (VME, CompactPCI, PC PCI, etc.) as well as any other conceivable type of carrier board capable of providing the required mechanical and electrical infrastructure.

In addition to this infrastructure, thermal energy dissipation requirements must be taken into consideration when implementing applications using E²Brain™ modules. The E²Brain™ concept basically calls for modules to provide their own thermal energy dissipation. It may, however, be necessary to add additional thermal energy dissipation capability depending on the overall system environment. To satisfy such requirements, E²Brain™ modules may be equipped with specially designed cooling devices that are adapted to the specific system environment.

Figure 1-1: E²Brain™ Basic Architecture



1.2 EB8347 System Overview

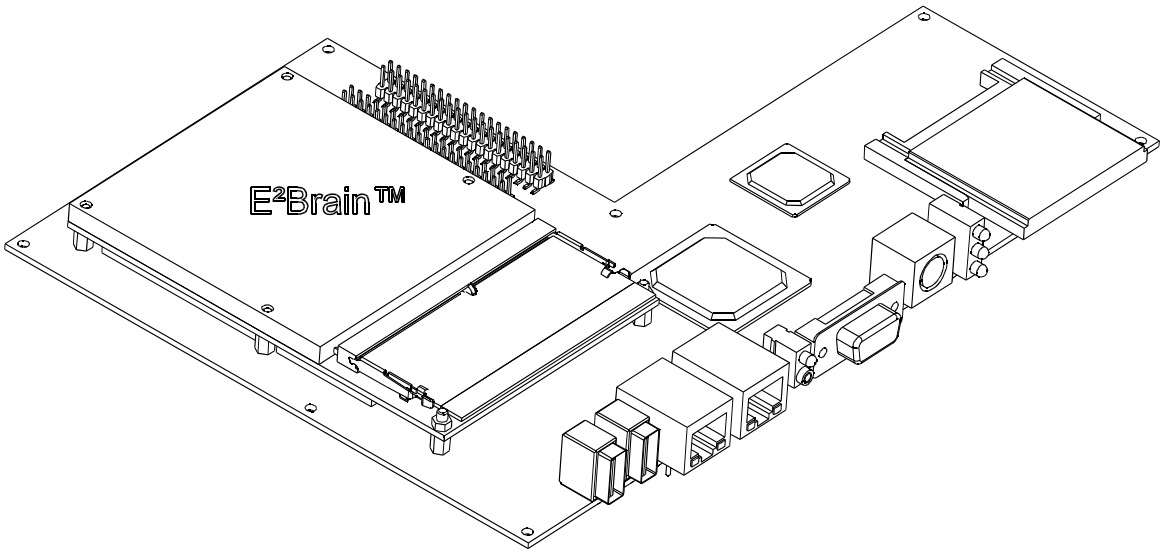
This E²Brain™ module is designed for applications requiring cost effective, high performance data processing capability where compact size, low power consumption, and interfacing flexibility are key factors for achieving a successful system design.

Specifically, system integrators are provided with a comprehensive set of industry standard interfaces from a range of low, medium, and high speed system level data exchange and monitor and control I/Os, and communications I/Os such as Ethernet, CAN, and high speed serial data I/Os.

System integration of an EB8347 E²Brain™ module requires a carrier board (industry standard or custom) which interfaces the application with the EB8347 module. All signal conditioning, mechanical, and electrical interfacing with the application must be accomplished by the carrier board.

In addition, physical packaging and thermal energy dissipation must be provided by the system. Integration on a carrier board is done via the four connectors and standoffs which allow for secure mounting to the carrier board. The following figure illustrates a typical integration of an E²Brain™ module with a BrainCAP™ on a custom carrier board.

Figure 1-2: E²Brain™ Module on a Custom Carrier Board



1.3 Product Overview

The EB8347 E²Brain™ High Performance PowerPC Processor Module is a part of an innovative concept to provide system integrators with a complete range of off-the-shelf as well as custom embedded computer cores for the most demanding of applications.

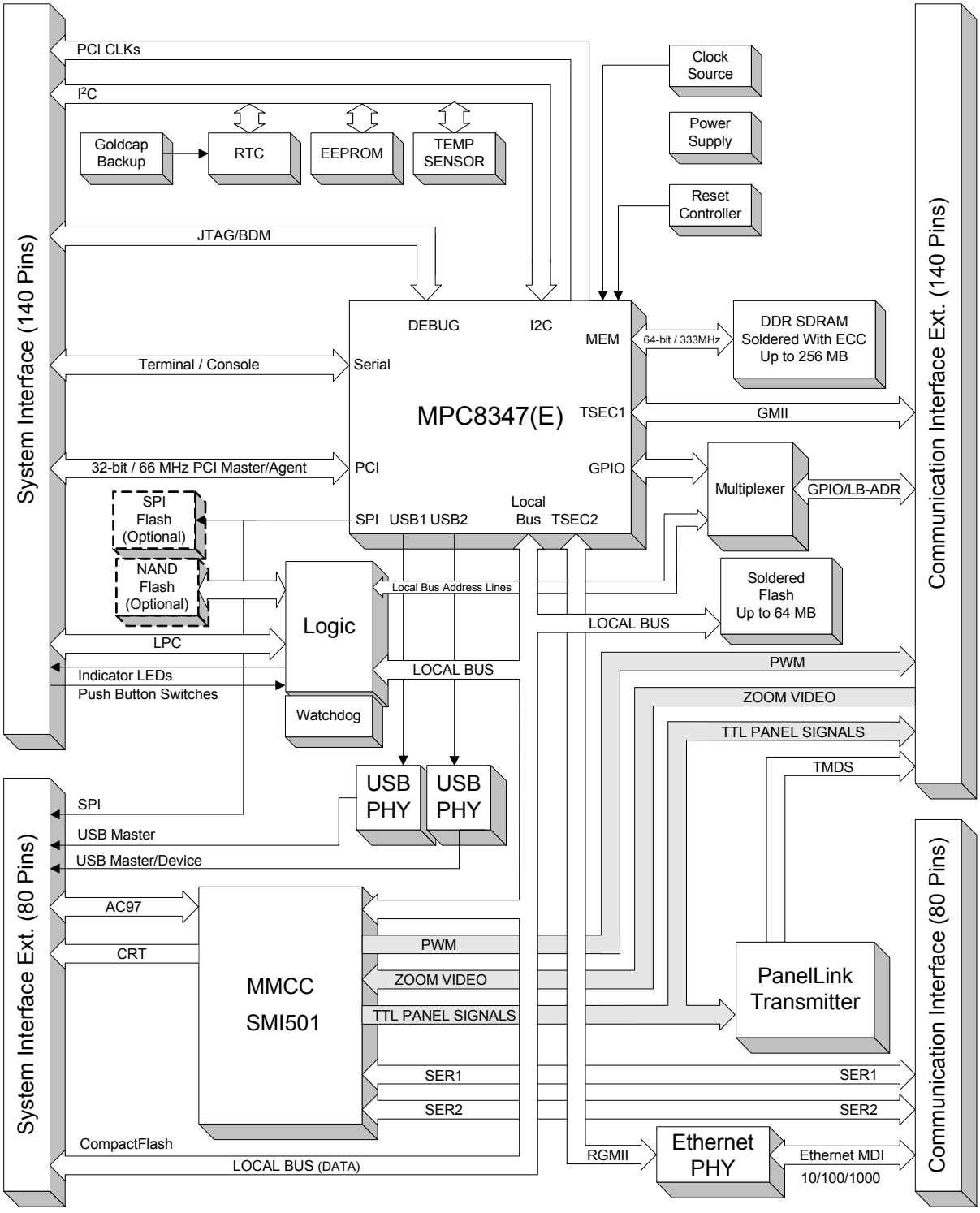
This module, designed around the Freescale MPC8347(E) Integrated Processor, provides a comprehensive set of standard computer functionality coupled with industry standard system and communications I/O capability. Realized on a compact, standardized, mezzanine board the EB8347 provides a complete embedded computer core which can be readily integrated into most any application.

The basic functions of this board are to provide high performance data processing capability as well as flexible and comprehensive system and communications I/O. The major elements involved in these processes are: the MPC8347(E) Integrated Processor, the BPCC logic controller which is realized in a CMOS programmable logic device (CPLD), a mobile multimedia companion chip (SM501), a Gigabit Ethernet PHY, and two system and two communications interfacing connectors. The EB8347 provides various types of system I/Os: I²C, LPC, a PCI-bus, USB, SPI, CRT, TMDS, audio, and serial terminal and console. In addition, there are several onboard memory elements available: DDR-SDRAM, FLASH, and EEPROM as well as a provision for accessing off-board CompactFlash. For test and programming purposes there is also a JTAG/TAP interface available.The following table and figure provide a quick overview of the EB8347 module.

Table 1-1: EB8347 Product Overview

EB8347 FEATURES	DESCRIPTION
Product Type	High performance PowerPC embedded computer core: • Processor: Freescale MPC8347(E) Integrated Processor (PowerQUICC™ II Pro) • Memory: DDR-SDRAM, FLASH, EEPROM • Security engine (only with the MPC8347E CPU) • Multiple system and communications I/Os • Form factor: E²Brain™ standard: 115 x 75 x 11.6 mm (minimum height) • Complies with the E²Brain™ specification
I/Os	System: • I²C • LPC • PCI-bus • USB • SPI • CRT • TMDS • Audio (AC97) • Serial: terminal and console • CompactFlash Communications: • High speed serial UART • Ethernet (Fast and Gigabit)
Other	Test and Programming: • JTAG/TAP (test access port) Monitor and Control: • Reset • GPIOs (General Purpose IOs) • Switches • Watchdog • Temperature sensing

Figure 1-3: EB8347 Basic Architecture



1.4 Board Overview

1.4.1 Board Introduction

The EB8347 is a high performance mezzanine computer module compliant to the Kontron Modular Computers' E²Brain™ Module Specification, revision 02.

This E²Brain™ module is comprised of a computer core, the Freescale MPC8347(E) Integrated Processor, and a standard set of E²Brain™ system and communications interfaces as well as onboard memory.

The computer core provides direct interfacing for the system interfaces: I²C, PCI, SPI, USB, terminal and console, and test and programming. The remaining memory, system, and communications interfaces are realized using the core's controller interfaces (DDR-SDRAM, ROM/Flash, GPIO, local bus, and Gigabit Ethernet) in conjunction with corresponding interfacing, and the BPCC (Board Process/Communications Controller), the MMCC, and multiplexer devices (Ethernet (Fast or Gigabit), LPC, CRT, TMDS, sound, serial, GPIO, monitor and control).

1.4.2 Board Specific Information

Major board components of the EB8347 E²Brain™ module are:

- Computer core: Freescale's MPC8347(E) Integrated Processor with PowerPC e300 core
- Board Process/Communications Controller (BPCC, realized in a CPLD)
- One Mobile Multimedia Companion Chip from Silicon Motions(SM501)
- One PanelLink transmitter
- One Gigabit Ethernet PHY
- One GPIO multiplexer
- Two USB PHYs
- System memory: DDR-SDRAM, soldered
- Soldered FLASH device(s)
- Serial EEPROM device
- RTC
- Temperature sensor: LM75
- Reset controller
- GoldCap (SRAM and RTC backup) (only on special request)
- Two System Interface connectors
- Two communications Interface connectors
- BrainCAP™ heat sink (for extended temperature applications)

1.5 System Relevant Information

The following system relevant information is general in nature but should still be considered when developing applications using the EB8347.

Table 1-2: System Relevant Information

SUBJECT	INFORMATION
System Configuration	The EB8347 operates with a PCI system clock frequency of 33 or 66 MHz. The number of EB8347s which can be installed in any one system depends solely on the number of carrier interfaces available.
Master/Slave Functionality	The EB8347 can function either as a PCI Master or PCI Agent.
System Controller	The EB8347 can function as a system controller.
Application Interfacing	The application interfacing to the EB8347 must comply with the specifications set forth in this manual.

1.5.1 System Configuration

System configuration is solely a function of the application, however, when implementing applications, precautions must be taken to ensure that the signals of the EB8347 are properly terminated in accordance with the specifications set forth in this manual. For this reason it will be necessary for system integrators to ensure proper signal conditioning for their applications before interfacing with the EB8347. In addition, it is imperative that signal interference be kept to a minimum. Refer to chapter 2 for further information.

1.5.2 Operating Software

The EB8347 is supplied with appropriate operating system and board support software for board operation.

1.6 Board Diagrams

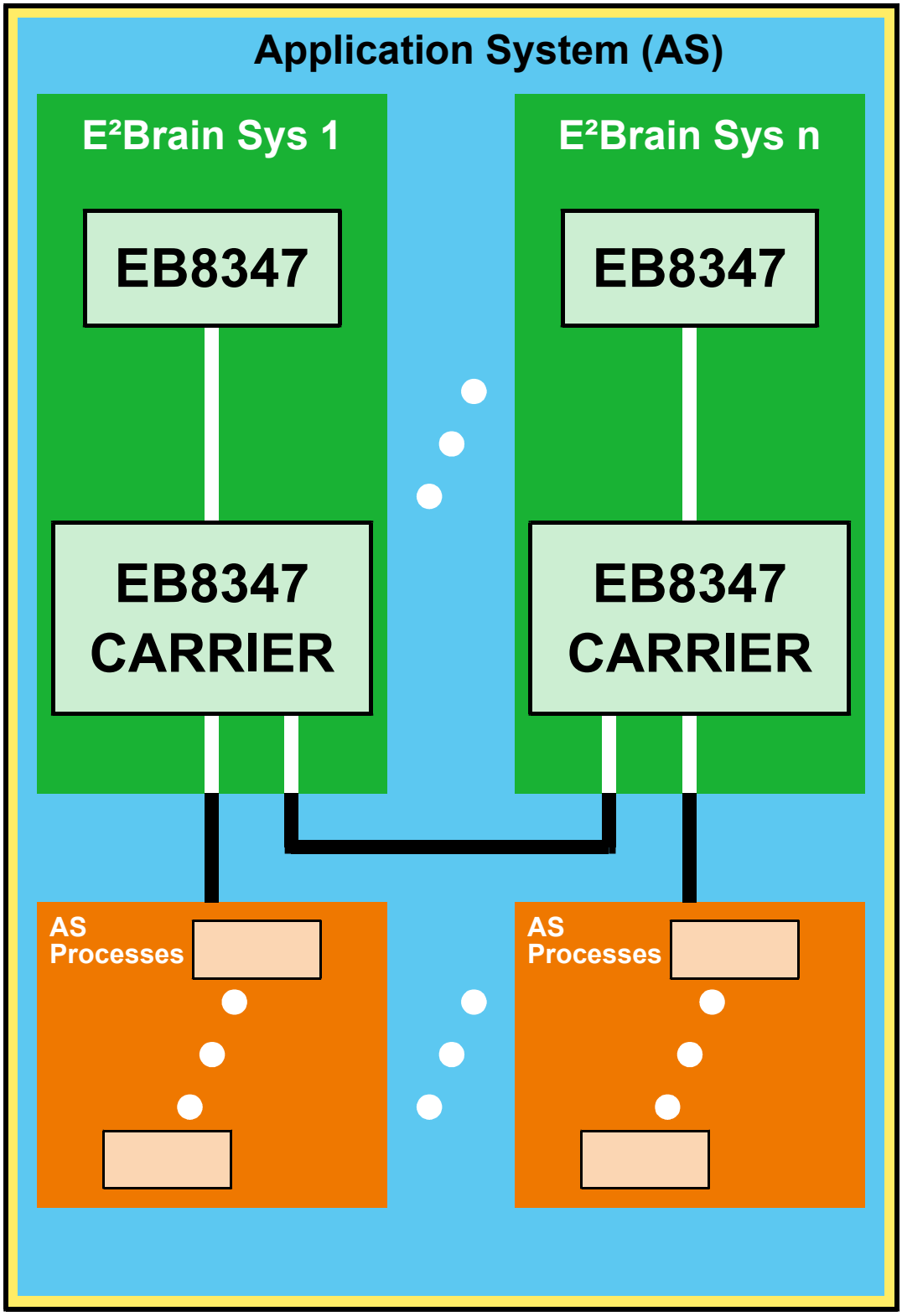
The following diagrams provide additional information concerning board functionality and component layout.

LEGEND FOR FIGURE 1-5:

AC97 Sound
CF CompactFlash
CRT Display (analog)
FE Fast Ethernet
GE Gigabit Ethernet
GPIO General Purpose IO
HSS High Speed Serial
I²C Inter-Integrated Circuit
LB(D/A) Local Bus (Data or Address)
LCD LCD Display (TTL Panel)
LPC Low Pin Count
M/C Monitor and Control
PCI Peripheral Component Interface
PWM Pulse Width Modulation (Brightness)
SPI Serial Peripheral Interface
T/C Terminal/Console (Serial Interface)
T/P Test/Programming
TMDS Transition Minimized Differential Serial
USB M USB Master
USB M/D USB Master/Device
ZV Zoom Video

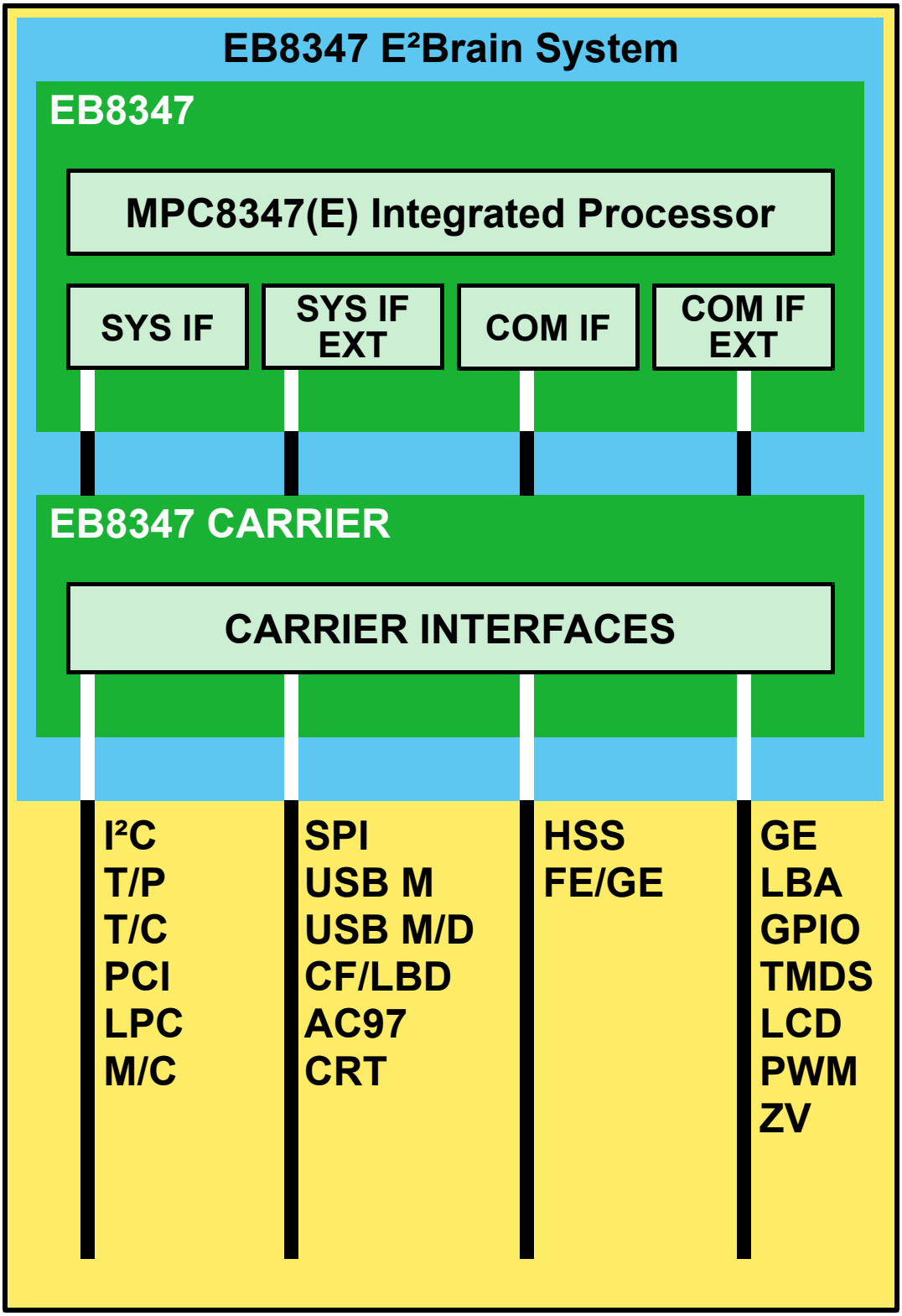
1.6.1 Application System Interfacing

Figure 1-4: EB8347 Application System Interfacing Diagram



1.6.2 System Level Interfacing

Figure 1-5: EB8347 System Level Interfacing Diagram



1.6.3 Board Layout

Figure 1-6: EB8347 Board (Top View)

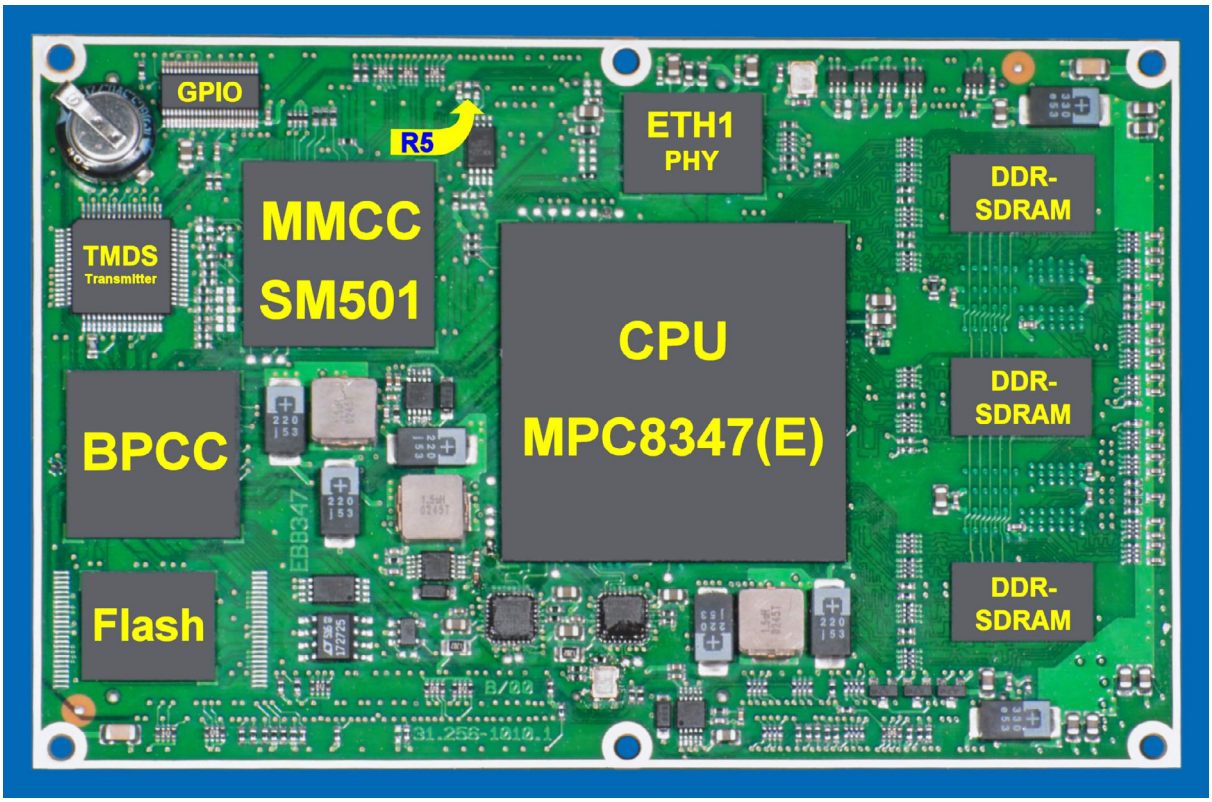
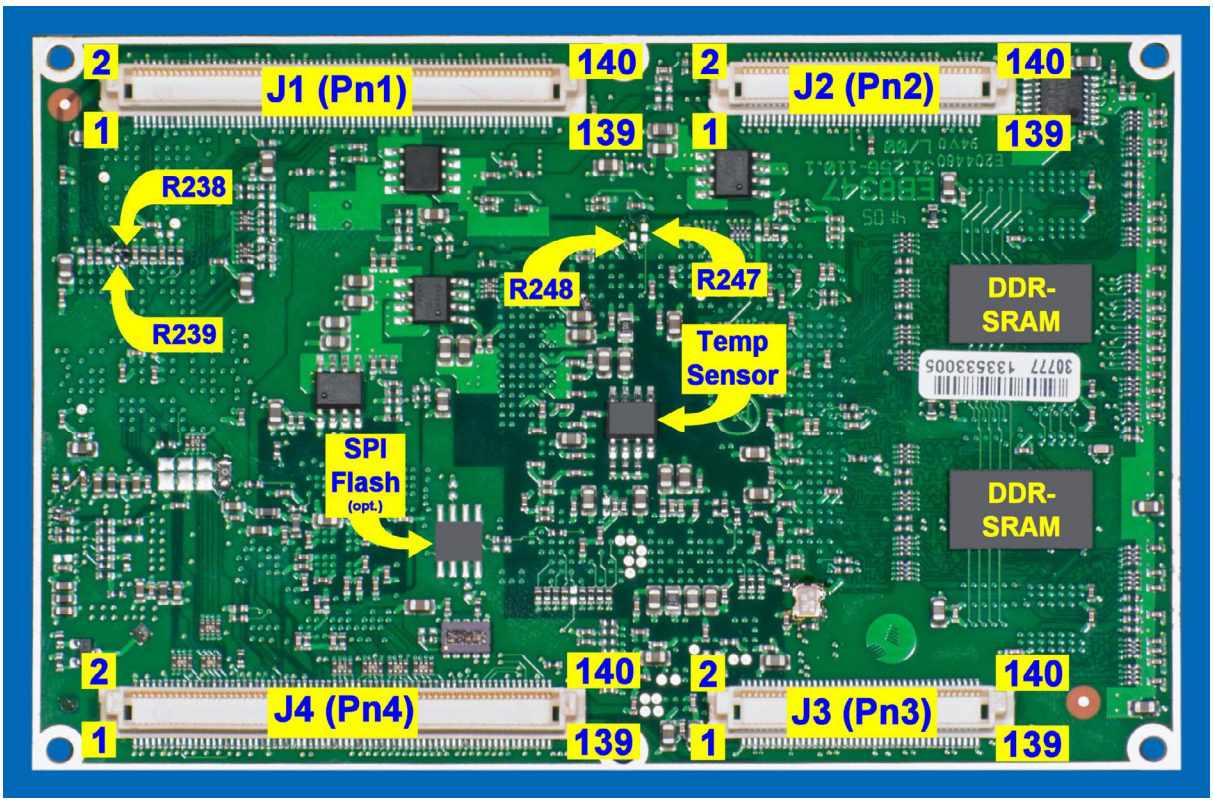


Figure 1-7: EB8347 Board (Bottom View)



1.7 Technical Specifications

Table 1-3: EB8347 Main Specifications

	EB8347	SPECIFICATIONS
Processor and Related	Processor	Freescale MPC8347(E) Integrated Processor PowerPC, e300 core
	Cache Structure	16 KB instruction cache and 16 KB data cache
	On Chip Controllers	DDR-SDRAM, Local Bus, PCI, TSE, USB, GPIO
	Main Memory	Soldered DDR-SDRAM: up to 256 MB, 64-bit, 264/330 MHz with or without ECC (must be specified when ordering)
	Watchdog	Watchdog generates exception condition: system reset, NMI, or cascading
	RTC	Real-time clock, optionally backed up using GoldCap with the data retention being typically about 6 to 7 days or via auxiliary power
	BPCC	Board Process/Communications Controller: controls Local Bus interfacing (MMCC, Multiplexer, CompactFlash), provides LPC interface and monitor and control functions
	MMCC	Mobile Multimedia Companion Chip: provides graphic, zoom video, sound, and high speed serial interfacing
Peripheral Memory	EEPROM (I^2C)	64 kBit soldered, serial access
	Flash	Minimum of 4 MB and up to 64 MB soldered (in addition, NOR or SPI/NAND Flash optionally available)
External Interfaces	PCI	On-chip controller, 32-bit, 33/66 MHz, PCI System Master and PCI Agent modes
	LPC	Low Pin Count sub-set, 8-bit IO and memory space access, realized in BPCC
	I^2C	Two on-chip, message interface, full master/slave
	CF	CompactFlash (Local Bus), PIO modes
	Fast Ethernet	Two 10baseT, 100baseTX, FCCs integrated in MPC8347(E), Intel PHY: LXT 972
	Gigabit Ethernet	Two three speed Ethernet (TSE): 10/100/1000 Mbit. One interface uses MDI with a BCM5461S PHY. The other is realized as a GMII interface.
	USB	Two USB 2.0, one master and one master device
	Sound	AC97 audio interface
	Graphic	One CRT analog interface One TMDS/PanelLink interface (DVI) One TTL panel interface (up to 8 bits pro color) One zoom video interface
	Serial Ports	Terminal and Console: Two serial ports: TTL: TERM (Terminal) TTL: SER0 (Console) High Speed Serial: Two ports : SER1 (both provided by the MMCC chip) : SER2

Table 1-3: EB8347 Main Specifications (Continued)

	EB8347	SPECIFICATIONS
T/P Interfaces	JTAG/Debug	JTAG/Debug interface for programming and testing purposes
M/C Interfaces	Inputs	Seven system/application control inputs available Three inputs available for general purpose (with interrupt capability)
	Outputs	Five system/application monitor outputs available Four outputs available for application use
	GPIO	General purpose IO: • provides one independent programmable chip select • configurable as GPIO or Local Bus address lines • LGTA handshake possible
General	Mechanical	Conforms with Kontron Modular Computers' "E²Brain™ Module Specification", revision 01
	Connector Types	The following connector types are standard on the EB8347: System Interface: J1 (Pn1) HiRose FX8C - 140P - SV (Header) System Interface Extension: J2 (Pn2) HiRose FX8C - 80P - SV (Header) Communications Interface: J3 (Pn3) HiRose FX8C - 80P - SV (Header) Communications Interface Extension: J4 (Pn4) HiRose FX8C - 140P - SV (Header)
	Power Consumption	Source: 3.3 V: consumption: 5.0 watts @ 528 MHz
	Temperature Range	Operational: 0ºC to +70ºC Standard -40ºC to +85ºC E2 (on request) Storage: -55ºC to +125ºC
	Climatic Humidity	93% relative humidity at 40°C, non-condensing
	Dimensions	115 mm L x 75 mm W x 11.6 mm H (minimum height)
	Board Weight	59 g (without heatsink)

1.8 Applied Standards

The Kontron Modular Computers' E²Brain™ module EB8347 complies with the requirements of the following standards:

Table 1-4: Applied Standards

	TYPE	STANDARD
CE	Emission	EN55022, EN61000-6-3
	Immunity, Industrial Environment	EN61000-6-2
	Immunity, IT Equipment	EN55024
	Electrical Safety	EN60950-1 Note: The EB8347 is specified I/O only for: SELV and EVL. It is **NOT SPECIFIED** for "Hazardous"
MECHANICAL	Mechanical Dimensions	E²Brain™ Module Specification, Rev. 01
ENVIRONMENTAL	Vibration, Sinusoidal	IEC 60068-2-6
	Random Vibration, Broadband	IEC 60068-2-64
	Bump	IEC 60068-2-29
	Single Shock	IEC 60068-2-27
	Temperature Tests A: Cold	IEC 60068-2-1
	Temperature Tests B: Dry Heat	IEC 60068-2-2
	Climatic Humidity	IEC 60068-2-78

1.9 Related Publications

Table 1-5: Related Publications

	ISSUED BY	DOCUMENT
PCI	PCI-SIG	PCI Local Bus Specification, R.2.2
LPC	Intel®	Intel® Low Pin Count (LPC) Interface Specification, Rev. 1.1
I²C	Philips	I2C-BUS SPECIFICATION, Rev. 2.1
E²Brain	Kontron Modular Computers	E²Brain™ Module Specification, Rev. 01
USB	USB Implementers Forum, Inc.	Universal Serial Bus Revision 2.0 specification
AC97	Intel®	AC97 codec specification, version 2.1
DVI	DDWG	Digital Visual Interface, rev. 1.0

Chapter 2

Functional Description

This page has been intentionally left blank.

2. Functional Description

The following chapters present more detailed, board level information about the EB8347 E²Brain™ High Performance PowerPC Processor Module whereby the board components and their basic functionality are discussed in general.

2.1 General Information

The EB8347 is comprised basically of the following:

- Freescale MPC8347(E) Integrated Processor, PowerQUICC™ II Pro
  - PowerPC™ e300 processor core
  - Security Engine
  - Peripheral logic
    - DDR memory
    - Local bus
    - Dual TSE
    - PCI interface
    - Dual I²C interfaces
    - Dual USB interfaces
    - SPI interface
    - GPIO interfacing
    - Dual UART serial interface
    - Debug and JTAG/TAP interfaces
- Board Process/Communications Controller (BPCC)
  - Realized in an FPGA device
  - Controls Local Bus interfacing (FLASH, MMCC, and addressing)
  - Provides LPC interface
  - Provides system monitor and control functions
  - Provides additional GPIO interfacing
- Mobile Multimedia Companion Chip (MMCC)
  - multifunctional graphic and GPIO controller
  - supports TMDS DVI (PanelLink), flat panel displays, AC97 sound, zoom video, high speed serial interfacing, analog CRT displays, display data channel interfacing
- System interfaces for:
  - I²C bus
  - LPC bus
  - PCI bus
  - SPI
  - USB
  - Local bus data, CompactFlash (CF) interface
  - Terminal and Console (T/C) serial interfacing
  - Monitor and Control (M/C) interfacing
  - CRT analog
  - AC97
  - DDC interface
  - Test and Programming (T/P) interfacing

- Communications interfaces for:
  - High Speed Serial (HSS) communications
  - Gigabit Ethernet (ETH1, TSEC1)
  - TMDS DVI
  - TTL flat panel interface
  - GPIO
  - Local bus addressing
  - Zoom video
  - PWM
- Memory
  - Main memory DDR-SDRAM, soldered
  - Soldered FLASH
  - Serial EEPROM
  - SPI Flash or NAND Flash (optional, exclusive)
- Monitor and Control
  - Interfacing for LED's, operator switches, system monitor/control signals (Reset, FAL, etc.)
  - RTC
  - Temperature sensing
  - Watchdog timer
  - Hardware delay timer
  - Clock generation
  - Reset generation
  - Board control registers
- Test and Programming
  - Debugging interface
  - JTAG/TAP interface
- Software
  - Operating system
  - Board support package
  - Boot strap loader (NetBootLoader)

2.1.1 Freescale MPC8347(E) Integrated Processor

The EB8347 is based on Freescale's PowerPC processor MPC8347(E) which integrates a large number of peripherals. Important features of the MPC8347(E) are as follows:

- BooK E e300 processor core
  - High-performance, 32-bit Book E enhanced core that implements the PowerPC architecture
  - High-performance, superscalar processor core
  - Floating-point, integer, load/store, system register, and branch processing units
  - 32-Kbyte instruction cache, 32-Kbyte data cache
  - Lockable portion of L1 cache
  - Dynamic power management
  - Software-compatible with the other Freescale processor families that implement the PowerPC architecture
  - Performance monitor

- Security engine is optimized to handle all the algorithms associated with IPSec, SSL/TLS, SRTP, IEEE Std. 802.11i™, iSCSI, and IKE processing. The security engine contains four crypto-channels, a controller, and a set of crypto execution units (EUs): The Execution Units are:
  - Public Key Execution Unit (PKEU)
  - Data Encryption Standard Execution Unit (DEU)
  - Advanced Encryption Standard Unit (AESU)
  - ARC Four execution unit (AFEU)
  - Message Digest Execution Unit (MDEU)
  - Random Number Generator (RNG)
  - Four crypto-channels, each supporting multi-command descriptor chains
- Double data rate, DDR SDRAM memory controller
  - Programmable timing for DDR-1 SDRAM
  - 32- or 64-bit data interface, up to 333-MHz data rate
  - Four banks of memory, each up to 1 Gbyte
  - DRAM chip configurations from 64 Mbit to 1 Gbit with x8/x16 data ports
  - Full error checking and correction (ECC) support
  - Page mode support (up to 16 simultaneous open pages)
  - Contiguous or discontiguous memory mapping
  - Read-modify-write support
  - Sleep mode for self-refresh SDRAM
  - Auto refresh
  - On-the-fly power management using CKE
  - Registered DIMM support
  - 2.5-V SSTL2 compatible I/O
- PCI interface
  - Data bus width: 32-bit data PCI interface operating at up to 66 MHz
  - PCI 3.3-V compatible
  - PCI host bridge capabilities
  - PCI agent mode on PCI interface
  - PCI-to-memory and memory-to-PCI streaming
  - Memory prefetching of PCI read accesses and support for delayed read transactions
  - Posting of processor-to-PCI and PCI-to-memory writes
  - On-chip arbitration supporting five masters on PCI
  - Accesses to all PCI address spaces
  - Parity supported
  - Selectable hardware-enforced coherency
  - Address translation units for address mapping between host and peripheral
  - Dual address cycle for target
  - Internal configuration registers accessible from PCIPCI 3.3V compatible
- Local bus controller (LBC)
  - Multiplexed 32-bit address and data operating at up to 133 MHz
  - Four chip selects support four external slaves
  - Up to eight-beat burst transfers
  - 32-, 16-, and 8-bit port sizes controlled by an on-chip memory controller
  - Three protocol engines on a per chip select basis:
    - General-purpose chip select machine (GPCM)
    - Three user-programmable machines (UPMs)

    - Dedicated single data rate SDRAM controller
  - Parity support
  - Default boot ROM chip select with configurable bus width (8-, 16-, or 32-bit)
- Dual three-speed (10/100/1000) Ethernet controllers (TSECs)
  - Dual controllers designed to comply with IEEE Std. 802.3™, 802.3u, 820.3x, 802.3z, 802.3ac
  - Ethernet physical interfaces:
    - 1000 Mbps IEEE Std. 802.3 GMII/RGMII, IEEE Std. 802.3z TBI/RTBI, full-duplex
    - 10/100 Mbps IEEE Std. 802.3 MII full- and half-duplex
  - Buffer descriptors are backward-compatible with MPC8260 and MPC860T 10/100 programming models
  - 9.6-Kbyte jumbo frame support
  - RMON statistics support
  - Internal 2-Kbyte transmit and 2-Kbyte receive FIFOs per TSEC module
  - MII management interface for control and status
  - Programmable CRC generation and checking
- Programmable interrupt controller (PIC)
  - Functional and programming compatibility with the MPC8260 interrupt controller
  - Support for 8 external and 35 internal discrete interrupt sources
  - Support for 1 external (optional) and 7 internal machine checkstop interrupt sources
  - Programmable highest priority request
  - Four groups of interrupts with programmable priority
  - External and internal interrupts directed to host processor
  - Redirects interrupts to external INTA pin in core disable mode.
  - Unique vector number for each interrupt source
- Universal serial bus (USB) dual role controller
  - USB on-the-go mode with both device and host functionality
  - Complies with USB specification Rev. 2.0
  - Can operate as a stand-alone USB device
    - One upstream facing port
    - Six programmable USB endpoints
  - Can operate as a stand-alone USB host controller
    - USB root hub with one downstream-facing port
    - Enhanced host controller interface (EHCI) compatible
    - High-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) operations
  - External PHY with UTMI, serial and UTMI+ low-pin interface (ULPI)
- Universal serial bus (USB) multi-port host controller
  - Can operate as a stand-alone USB host controller
    - USB root hub with one or two downstream-facing ports
    - Enhanced host controller interface (EHCI) compatible
    - Complies with USB specification Rev. 2.0
  - High-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) operations
  - Direct connection to a high-speed device without an external hub
  - External PHY with serial and low-pin count (ULPI) interfaces
- DUART
  - Two 4-wire interfaces (RxD, TxD, RTS, CTS)
  - Programming model compatible with the original 16450 UART and the PC16550D

- Dual industry-standard I²C interfaces
  - Two-wire interface
  - Multiple master support
  - Master or slave I²C mode support
  - On-chip filtering rejects spikes on the bus
  - System initialization data optionally loaded from I²C-1 EPROM by boot sequencer embedded hardware
- Serial peripheral interface (SPI) for master or slave
- General-purpose parallel I/O (GPIO), 52 parallel I/O pins multiplexed on various chip interfaces
- System access port
  - Uses JTAG interface and a TAP controller to access entire system memory map
  - Supports 32-bit accesses to configuration registers
  - Supports cache-line burst accesses to main memory
  - Supports large block (4 Kbyte) uploads and downloads
  - Supports continuous bit streaming of entire block for fast upload and download

2.1.2 Board Process/Communications Controller (BPCC)

The BPCC provides extensive interfacing and monitor/control functionality for the EB8347. It provides control and addressing for the Local Bus devices, control of the CPU configuration and board, and an onboard register set. The LPC interface, the Watchdog timer, and the hardware delay timer are also realized in the BPCC as well as the monitor and control interfaces. In addition, the BPCC provides a set of GPIOs.

2.1.3 System Interfacing

The EB8347 E²Brain™ module is supplied with a comprehensive set of system interfacing capabilities. The standard set of system interfaces is routed through the System Interface connector J1 (Pn1). An extended set of system interfacing is routed through the System Interface Extension connector J2 (Pn2).

The System Interface connector J1 (Pn1) provides interfacing for the following:

- Dual I²C bus
- LPC bus
- PCI bus
- I²C bus
- Terminal and Console (T/C) serial interfacing (TERM and SER0)
- Monitor and Control (M/C) interfacing
- Test and Programming (T/P) interfacing

The System Interface Extension connector J2 provides interfacing for the following:

- Local bus data, CompactFlash (CF) interface
- USB master
- USB master/device
- SPI
- CRT analog
- AC97

2.1.4 Communications Interfacing

The EB8347 E²Brain™ module is also supplied with a comprehensive set of communications interfacing capabilities which are routed through the Communications Interface connectors J3 (Pn3) and J4 (Pn4).

The Communications Interface connector J3 provides interfacing for the following:

- High Speed Serial (HSS) interfacing (TTL-level) (SER1, SER2)
- Gigabit Ethernet (MDI interface) (ETH1)

The Communications Interface Extension connector J4 provides interfacing for the following:

- Three speed Ethernet (TSEC1) GMII interface (10/100/1000 Mbits)
- Twenty-four GPIOs plus CS and LGTA handshake
- TMDS DVI
- Flat panel TTL-level signals
- Zoom video
- PWM brightness control signal output

2.1.5 Memory

Main memory for the EB8347 is provided by soldered DDR-SDRAM up to 256 MB. Up to 64 MB of soldered NOR Flash is available for ROMable operating systems and boot strap loaders. Finally, there is a 64 kBit (8 x 8 kbit) serial EEPROM connected to the I²C bus for system use.

2.1.6 Monitor and Control (M/C)

Various monitor and control functions are available for use with the EB8347 E²Brain™ module. Twelve M/C signals are available on the System Interface for application usage. In addition, the EB8347 provides a RTC, a Watchdog timer, a hardware delay timer, a digital temperature sensor, clock generators, a reset controller, and variety of board control registers.

2.1.7 Test and Programming

The EB8347 supports the comprehensive set of MPC8347(E) debugging and JTAG/TAP functionality. Interfacing for this functionality is available on the System Interface.

2.1.8 Software

The EB8347 is supported by various operating systems. In addition, board support packages are available as well as the “NetBootLoader” bootstrap loader.

2.2 Board-Level Interfacing Diagram

The following figures demonstrate the interfacing structure between the internal processing modules of the EB8347 and other major EB8347 module components. Where EB8347 system elements have common interfacing they are grouped into a block. Interfacing common to only one element of a block is indicated with a direct connecting line. The interfacing lines are shown in white where they are on board and in black for board external interfacing.

LEGEND FOR FIGURES 8, 9, and 10

AC97	Audio Control 97
ADD	Address signals (Local bus)
BPCC	Board Process/Communications Controller
CF	CompactFlash
CI	Communications Interface (J3 (Pn3))
CIE	Communications Interface Extension (J4 (Pn4))
CRT	Cathode Ray Tube (display, analog)
DDC	Display Data Channel
ETH1	ETHernet 1
FCC	Fast Communications Controller
FE	Fast Ethernet
GMII1	Gigabit Media Independent Interface 1
GPIO	General Purpose IO
HSS	High Speed Serial
I²C	Inter-Integrated Circuit
LB	Local Bus
LBC	Local Bus Controller
LPC	Low Pin Count
M/C	Monitor/Control
MMCC	Mobile Multimedia Companion Chip
opt.	optional
PCI	Peripheral Component Interface
PHY	PHY (Ethernet or USB)
PWM	Pulse Width Modulation
RTC	Real Time Clock
SERn	Serial interface n
SI	System Interface (J1 (Pn1))
SIE	System Interface Extension (J2 (Pn2))
SPI	Serial Peripheral Interface
T/C	Terminal/Console (Serial Interface)
T/P	Test/Programming
TEMP	Digital Temperature Sensor
TMDS	Transition Minimized Differential Signal
TSEC	Three Speed Ethernet Controller
TTL-PS	TTL Panel Signals
USB(-n)	Universal Serial Bus (-n)
ZV	Zoom Video

Figure 2-1: EB8347 Board Level Interfacing (Sheet 1 of 3)

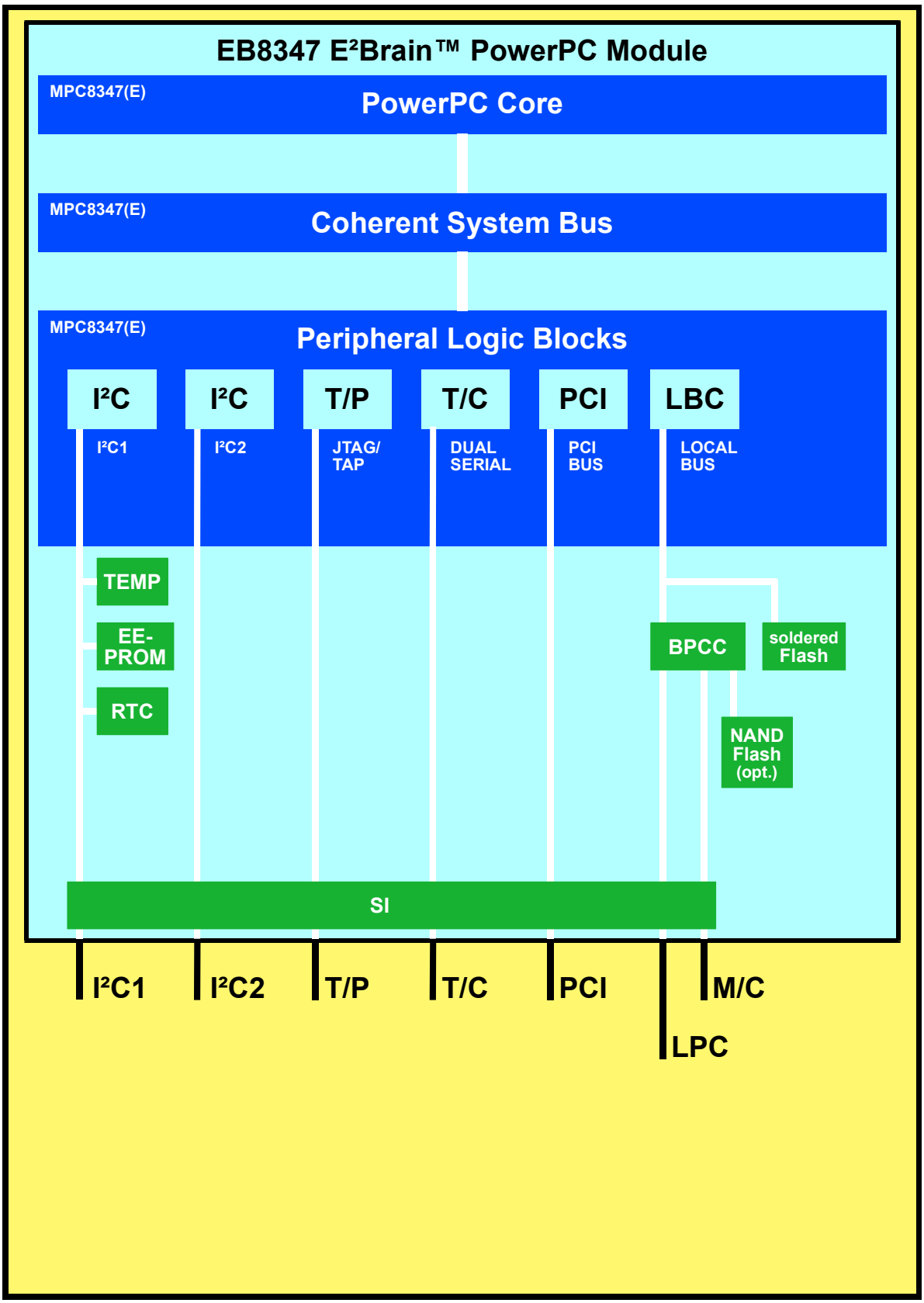


Figure 2-2: EB8347 Board Level Interfacing (Sheet 2 of 3)

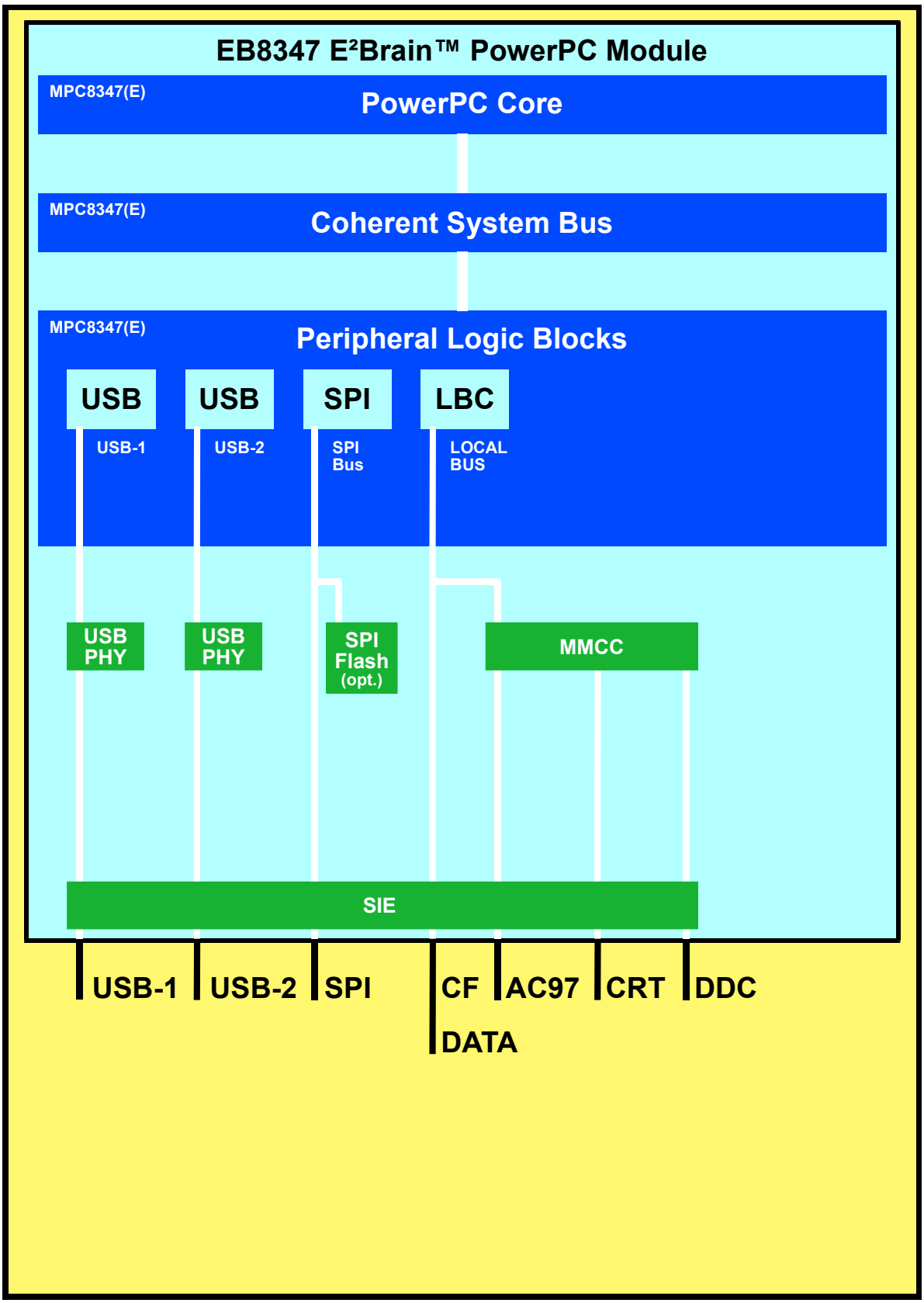
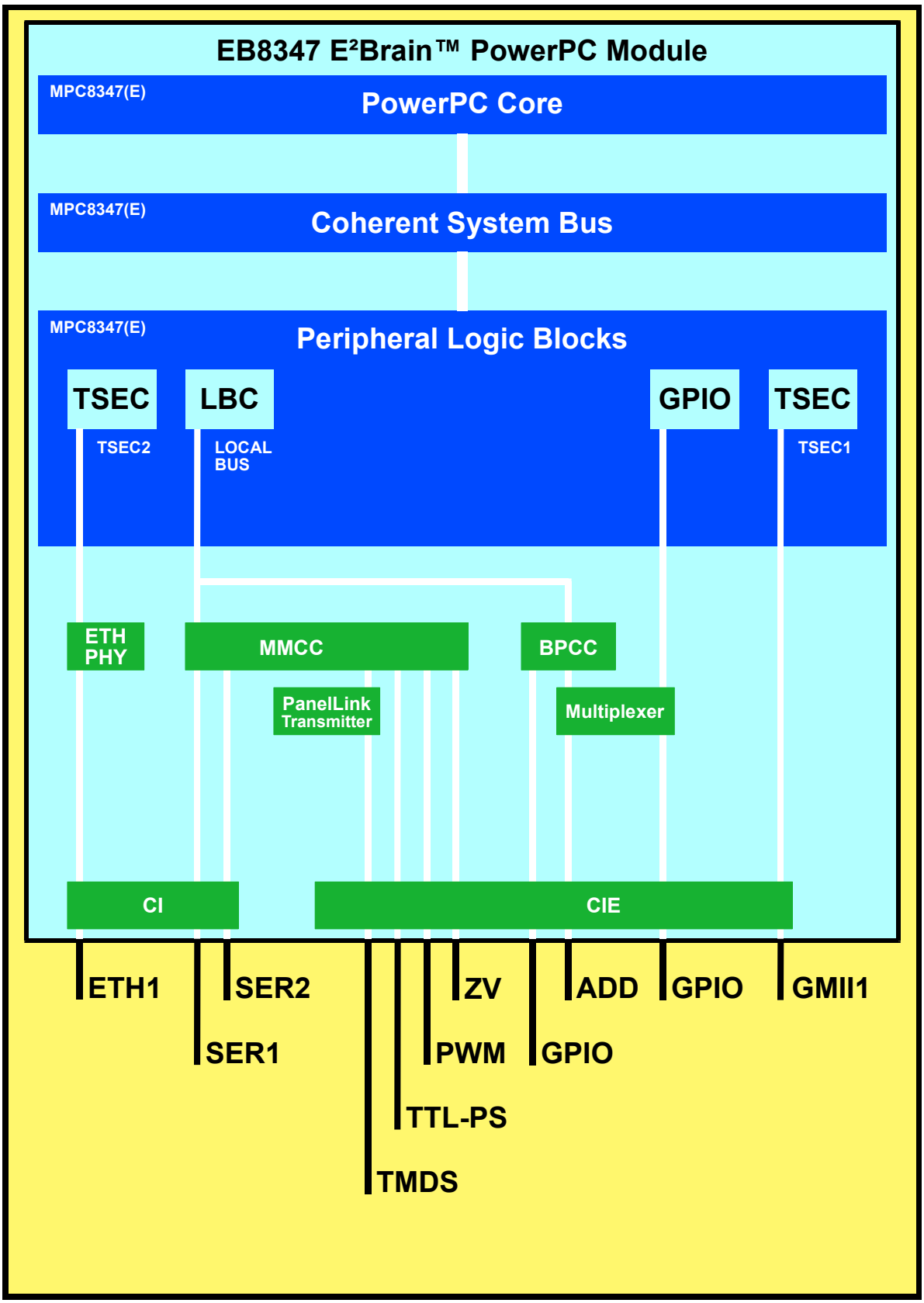


Figure 2-3: EB8347 Board Level Interfacing (Sheet 3 of 3)



2.3 System Interface

As the name implies, this interface provides the basic application connection functionality required to integrate the EB8347 E²Brain™ module either as a high performance core or as a dedicated, special purpose subsystem within a comprehensive data processing and handling system.

The System Interface is realized using a 140-pin, HIROSE FX8C-140P-SV connector designated as J1 (Pn1) which is designed to mate with a 140-pin, HIROSE FX8C-140S-SV connector on the EB8347 carrier board. The following table provides an overview of the signal types and a brief description of the interfacing realized on this connector. The ensuing sections provide more detailed information concerning the signal specification for this interface.

Table 2-1: System Interface J1 (Pn1) Signal Types

SIGNAL TYPE	DESCRIPTION
Power	EB8347 E²Brain™ module input power, grounds, battery backup power, PCI signaling voltage V(I/O)
Monitor And Control (M/C)	Control signals for E²Brain™ module operation, configuration, and additional GPIO interfacing
Test And Programming (T/P)	JTAG/TAP signals for Emulator interfacing
Terminal And Console (T/C)	Two 2-wire serial interfaces: RxD1/TxD1:Used by the boot loader during startup as a terminal interface; once the system has been booted, is available as general purpose serial interface (Terminal) RxD2/TxD2:general purpose serial interface (Console)
I2C	Two I2C standard interfaces for low speed, serial, inter-chip communications
LPC	One LPC standard interface for (GP)IOs and simple memory interfacing
PCI	One PCI standard interface for PCI-bus interfacing

2.3.1 J1 (Pn1) Connector Pinout

Because the E²Brain™ specification defines signal interfacing which is at the physical component level, the actual electrical characteristics of signals are for the most part different from those which are specified using accepted industry standards which apply more to unit-to-unit level signals. Only in those cases where the industry standard for such signals is at the physical component level are the characteristics of the signals specified compliant with the standard indicated, for example: PCI compliant signals.

The following table provides signal pinouts along with information concerning signal characteristics for connector J1 (Pn1) of EB8347 E²Brain™ module.

Table 2-2: Pinout of J1 (Pn1) Connector

PIN	SIGNAL	TYPE	SIGNAL GROUP	REMARKS
1	GND	I	POWER	
2	AUX-Power	I	POWER	Used to supply power to RTC and SRAM
3	SDA1	I/O	I2C	Internal pull-up: 4.7kΩ
4	SCL1	I/O	I2C	Internal pull-up: 4.7kΩ
5	MC6	O	M/C	High, if Control Register 0 = 0; low, if Control Register 0 = 1; can be used to drive an LED; this signal toggles during NetBootLoader startup
6	+3.3V	I	POWER	
7	MC0	I	M/C	Debounced Reset input, active low
8	MC7	O	M/C	Used for Watchdog activity LED, low if WD enabled
9	MC2	I	M/C	Internal pull-up: 8.2kΩ
10	MC1	I	M/C	Latched input; internal pull-up: 8.2kΩ (refer to the Event Register); ABORT functionality during bootloader startup
11	+3.3V	I	POWER	
12	GND	I	POWER	
13	MC3	I	M/C	Internal pull-up: 8.2kΩ
14	MC11	O	M/C	High, if Control Register 1 = 0; low, if Control Register 1 = 1; can be used to drive an LED
15	MC4	I	M/C	Internal pull-up, this pin should be left open (reserved for factory use)
16	MC5	I	M/C	This pin requires an external pull-up: 2kΩ - 10kΩ
17	MC9	O	M/C	Open collector: driven low if Control Register 3 = 0; high impedance if Control Register 3 = 1
18	MC8	I	M/C	Internal pull-up: 8.2kΩ; leave this signal open if used in Standalone/ Master configuration; connect to GND to force Slave-Mode
19	GND	I	POWER	
20	MC10	O	M/C	Open collector: driven low if Control Register 5 = 0; high impedance if Control Register 5 = 1
21	LPCCLK	O	LPC	
22	RESERVED		LPC	This pin must be left unconnected
23	LAD0	I/O	LPC	Internal pull-up: 8.2kΩ
24	LAD1	I/O	LPC	Internal pull-up: 8.2kΩ
25	LAD2	I/O	LPC	Internal pull-up: 8.2kΩ
26	+3.3V	I	POWER	
27	LFRAME#	O	LPC	
28	GND	I	POWER	
29	SERIRQ#	I/O	LPC	Internal pull-up: 10kΩ

Table 2-2: Pinout of J1 (Pn1) Connector

PIN	SIGNAL	TYPE	SIGNAL GROUP	REMARKS
30	LAD3	I/O	LPC	Internal pull-up: 8.2kΩ
31	TxD1	O	T/C	TTL-Level
32	RESERVED			This pin is not connected internally
33	+3.3V	I	POWER	
34	TxD2	O	T/C	TTL-Level
35	GND	I	POWER	
36	RxD2	I	T/C	Internal pull-up: 100kΩ; input signal levels must not exceed 3.3V + 5%
37	RxD1	I	T/C	Internal pull-up: 100kΩ; input signal levels must not exceed 3.3V + 5%
38	CHKSTP_IN	I	T/P	Internal pull-up: 10kΩ
39	EMU_VCC	O	T/P	Provides the suitable IO voltage to an Emulation probe(3.3V)
40	TAP-HRST	I	T/P	Internal pull-up: 10kΩ
41	CHKSTP_OUT	O	T/P	Internal pull-up: 10kΩ
42	GND	I	POWER	
43	SRST	I	T/P	Internal pull-up: 10kΩ
44	JTAG-TMS	I	T/P	Internal pull-up: 10kΩ
45	JTAG-TRST	I	T/P	Internal pull-up: 10kΩ
46	+3.3V	I	POWER	
47	JTAG-TCK	I	T/P	Internal pull-down: 10kΩ
48	JTAG-TDO	O	T/P	
49	SLC2	I/O	I2C	Internal pull-up: 1kΩ
50	JTAG-TDI	I	T/P	Weak internal pull-up: > 10kΩ
51	GND	I	POWER	
52	SDA2	I/O	I2C	Internal pull-up: 1kΩ
53	PCI-CLK-OUT-0	O	PCI	
54	V(I/O)	I	POWER	This pin defines the PCI Signaling Voltage. This pin must be connected to 3.3V.
55	+3.3V	I	POWER	
56	PCI-CLK-IN	I	PCI	Clock input used for PCI Agent mode; not used in Master mode configuration
57	PCI-CLK-OUT-1	O	PCI	
58	GND	I	POWER	
59	PCI-RST#	O	PCI	Can also be used to reset other carrier devices
60	PCI-CLK-OUT-2	O	PCI	
61	INTA#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up

Table 2-2: Pinout of J1 (Pn1) Connector

PIN	SIGNAL	TYPE	SIGNAL GROUP	REMARKS
62	INTC#	I	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
63	INTB#	I	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
64	GNT#0	I/O	PCI	
65	INTD#	I	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
66	+3.3V	I	POWER	
67	GND	I	POWER	
68	GNT#1	O	PCI	
69	AD31	I/O	PCI	
70	GNT#2	O	PCI	
71	AD29	I/O	PCI	
72	REQ#0	I/O	PCI	Master mode variant: internal pull-up: 10kΩ; Slave mode variant: no pull-up
73	AD27	I/O	PCI	
74	REQ#1	I	PCI	Internal pull-up: 8.2kΩ
75	+3.3V	I	POWER	
76	GND	I	POWER	
77	AD25	I/O	PCI	
78	REQ#2	I	PCI	Internal pull-up: 10kΩ
79	C/BE3#	I/O	PCI	
80	AD30	I/O	PCI	
81	AD23	I/O	PCI	
82	AD28	I/O	PCI	
83	GND	I	POWER	
84	AD26	I/O	PCI	
85	AD21	I/O	PCI	
86	AD24	I/O	PCI	
87	V(I/O)	I	POWER	This pin defines the PCI Signaling Voltage. This pin must be connected to 3.3V.
88	+3.3V	I	POWER	
89	AD19	I/O	PCI	
90	IDSEL	I/O	PCI	Leave unconnected for Master mode variant; IDSEL input in Agent mode variant
91	AD17	I/O	PCI	

Table 2-2: Pinout of J1 (Pn1) Connector

PIN	SIGNAL	TYPE	SIGNAL GROUP	REMARKS
92	GND	I	POWER	
93	C/BE2#	I/O	PCI	
94	AD22	I/O	PCI	
95	+3.3V	I	POWER	
96	AD20	I/O	PCI	
97	IRDY#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
98	AD18	I/O	PCI	
99	DEVSEL#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
100	AD16	I/O	PCI	
101	GND	I	POWER	
102	FRAME#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
103	PCI-X-CAP	I/O	PCI	Install a pull-down resistor of 100 ohm or less for PCI or leave open for PCI-X functionality
104	GND	I	POWER	
105	LOCK#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
106	TRDY#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
107	PERR#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
108	STOP#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
109	SERR#	I/O	PCI	Master mode variant: internal pull-up: 8.2kΩ; Slave mode variant: no pull-up
110	+3.3V	I	POWER	
111	GND	I	POWER	
112	PAR	I/O	PCI	
113	C/BE1#	I/O	PCI	
114	AD15	I/O	PCI	
115	AD14	I/O	PCI	
116	V(I/O)	I	POWER	This pin defines the PCI Signaling Voltage. This pin must be connected to 3.3V.
117	AD12	I/O	PCI	
118	AD13	I/O	PCI	

Table 2-2: Pinout of J1 (Pn1) Connector

PIN	SIGNAL	TYPE	SIGNAL GROUP	REMARKS
119	AD10	I/O	PCI	
120	AD11	I/O	PCI	
121	+3.3V	I	POWER	
122	AD9	I/O	PCI	
123	M66EN	I/O	PCI	Install a pull-down resistor of 100 ohm or less for 33 MHz or leave open for 66 MHz
124	GND	I	POWER	
125	AD8	I/O	PCI	
126	C/BE0#	I/O	PCI	
127	AD7	I/O	PCI	
128	AD6	I/O	PCI	
129	AD5	I/O	PCI	
130	+3.3V	I	POWER	
131	GND	I	POWER	
132	AD4	I/O	PCI	
133	AD3	I/O	PCI	
134	AD2	I/O	PCI	
135	AD1	I/O	PCI	
136	AD0	I/O	PCI	
137	ACK64#	I/O	PCI	Master mode variant: internal pull-up 10kΩ; Slave mode variant: no pull-up, signal not used on this board
138	REQ64#	I/O	PCI	Master mode variant: internal pull-up 10kΩ; Slave mode variant: no pull-up, signal not used on this board
139	+3.3V	I	POWER	
140	GND	I	POWER	

2.3.2 I2C Interfaces

The EB8347 E²Brain™ module provides two I2C serial interface for supporting direct interfacing of EB8347 and carrier board devices. These interfaces are two signals wide and fully support the I2C specification.

The following table provides a listing of the I2C interface signals and a brief description.

Table 2-3: I2C Interface Signal Description

SIGNAL	DESCRIPTION
SCL1, SCL2	Serial Clock line
SDA1, SDA2	Serial Data line

The onboard I2C devices of the EB8347 are:

- RTC (RealTime Clock)
- Temperature sensor
- User EEPROM

Refer to the Freescale Reference Manual for the MPC8347 and Application Note: AN2919 for I2C programming information.

2.3.3 Power Interface

For the EB8347 E²Brain™ module, a single power supply voltage of 3.3 VDC is specified. The following table summarizes the power specifications.

Table 2-4: EB8347 Power Interface Requirements

VOLTAGE	DESCRIPTION
+ 3.3 VDC	Input voltage tolerance: +5% to -3% Supply voltage ripple: 100 mV peak-to-peak; 0 to 20 MHz For further information concerning power, refer to chapter 6.3, “Power Requirements”.
GND	Ground voltage reference input
AUX-Power	Optional auxiliary power input for battery backup of CMOS memory devices; input voltage must be =< 3.3V, + 5% tolerance
V(I/O)	PCI signalling voltage selection.  ***WARNING!*** The EB8347 may only be operated using 3.3V signalling voltage. Use of 5V signalling voltage will result in damage to the EB8347. Kontron Modular Computers disclaims any and all liability for damage caused by the use of 5V signalling voltage.

2.3.4 Monitor and Control Interface

This interface is comprised of a set of twelve IO signals which can be used to facilitate system integration.

The following table provides a listing of Monitor and Control signals along with a brief description.

Table 2-5: Monitor and Control Interface Signal Description

SIGNAL	DESCRIPTION
MC0	Reset input, debounced (e.g. for push button switch)
MC1	Input, non-maskable interrupt request, debounced (e.g. for ABORT switch)
MC2	Input, suitable for maskable interrupt operation, active low
MC3	Input, suitable for maskable interrupt operation, active low
MC4	Input, factory mode, used as boot control signal on the EB8347; low = boot from LPC device; high or open = boot from onboard FLASH

Table 2-5: Monitor and Control Interface Signal Description

SIGNAL	DESCRIPTION
MC5	Input, suitable for maskable interrupt operation
MC6	Output, can be used as a general purpose output after the bootloader startup (e.g. general purpose LED or as a high load driver)
MC7	Output, can be used as a general purpose output after the bootloader startup (e.g. general purpose LED or as a high load driver)
MC8	Input, AGENT, low = Agent mode, high = Master mode
MC9	Output, open collector
MC10	Output, open collector
MC11	Output, can be used as a general purpose output after the bootloader startup (e.g. general purpose LED or as a high load driver)

2.3.5 Test and Programming Interface

The Test and Programming interface supports JTAG/TAP operations. This interface can be used for connecting hardware emulators and debuggers (e.g. BDM, COP, …). It is comprised of a set of ten signals whereby some are common to both interfaces and some are dedicated to only one.

The following table provides a listing of the Test and Programming interface signals and a brief description.

Table 2-6: Test and Programming Interface Signal Description

SIGNAL	DESCRIPTION
JTAG_TCK	Test Clock in for JTAG and emulator/debugger
JTAG_TDI	Test Data In for JTAG and emulator/debugger
JTAG_TDO	Test Data Out JTAG and emulator/debugger
JTAG_TMS	Test Mode Select, input for JTAG and emulator/debugger
JTAG_TRST	Test Reset, input for JTAG and emulator/debugger
HRST	Hard Reset, emulator/debugger hard reset input
SRST	Soft Reset, emulator/debugger reset input
CHKSTP_IN	Checkstop input
CHKSTP_OUT	Checkstop output
EMU_VCC	Reference Voltage of the JTAG/TAP core

2.3.6 Terminal and Console Interface

The EB8347 provides two serial interfaces for supporting a terminal port and a low speed communications interface (console) for firmware updating. These interfaces are realized using the MPC8347(E) on-chip dual UART, and as such provide only a two wire interface without hardware handshake signals.

The following table provides a listing of the Terminal and Console interface signals and a brief description.

Table 2-7: Terminal and Console Interface Signal Description

SIGNAL	DESCRIPTION
TxD1, RxD1	Terminal serial interface
TxD2, RxD2	Console serial interface

Note ...

The corresponding serial signals on the EB8347 are TTL logic level signals. Therefore, the transceivers for RS232 must be provided by the carrier board.

WARNING!

The signal level on the receive lines must not exceed 3.3V. Transients and signal levels higher than 3.3V may damage the board.

2.3.7 LPC Interface

One Low Pin Count (LPC) interface for supporting simple IOs, simple static memory devices, and IO controllers is available with the EB8347.

The controller is completely integrated in the BPCC and offers a 8-bit data access port to devices which use LPC IO or memory access protocols. I/O and memory area are selected using different address spaces.

The I/O address space is 64 kByte in size, whereas the memory area offers 16 MByte address space. DMA is not supported by this interface.

In addition, a serial IRQ controller is also implemented in the BPCC, controlling and collecting the serial LPC IRQs and converting and processing them to IRQs for the CPU.

The serial IRQ controller is realized according to the “Serialized IRQ Support for PCI Systems” Specification, Rev. 6.0, Sept. 1, 1995

The following table provides a listing of the LPC interface signals and a brief description.

Table 2-8: LPC Interface Signal Description

SIGNAL	DESCRIPTION
LAD[3:0]	Multiplexed Command, Address, and Data lines
LFRAME#	Indicates start of a new cycle, termination of broken cycle
LPCCLK	33 MHz clock
SERIRQ	Serialized IRQ, optional for peripherals that need interrupt

2.3.8 PCI Interface

The EB8347 is capable of either PCI Master mode or PCI Agent mode operation. There are two variants available: one for Master mode and one for Agent mode operation.

The PCI Master mode variant operates as the host, initializing and controlling up to three PCI devices on the carrier, whereas, the Agent mode variant operates as a PCI target.

Note ...

For proper operation of the EB8347, the monitor and control signal, MC8, must be set to the correct PCI operation mode: low = Agent, high = Master.

The following table identifies the EB8347 PCI bus signals and provides a short description of each signal.

Table 2-9: PCI Interface Signal Description

SIGNAL	DESCRIPTION
AD [31:0]	PCI multiplexed address and data bus
INT [A, B, C, D]#	PCI interrupt requests
C/BE [3:0]#	PCI multiplexed bus command and byte enable
IRDY#	Initiator Ready indicates the current bus master is ready to complete the current data phase.
TRDY#	Target Ready indicates the selected device is ready to complete the current data phase.
PCI-RST#	PCI Reset signal, is also used for LPC devices and other devices on the carrier board
PCI-CLK-OUT-[2:0]	PCI clock Outputs for up to 3 external bus mastering PCI devices. All PCI signals except PCI_RST#, and INT [A, B, C, D] # are sampled on the rising edge.
FRAME#	Indicate the beginning and duration of a PCI access.
STOP#	Indicates the target is requesting the master to stop the current transaction
DEVSEL#	Device select generated by the target when cycle refers to its own address.
REQ [2:0]#	PCI Arbiter requests
GNT [2:0]#	PCI Arbiter grants
PAR	Calculated/Checked Parity
PERR#	Parity Error
LOCK#	PCI Lock resource signal
SERR#	System Error
REQ64#	PCI request for a 64-bit access (not supported)
ACK64#	PCI grant for a 64-bit access (not supported)
PCI-CLK-IN	PCI clock input, used by the PCI Agent mode variant
AGENT (MC8)	PCI-agent mode logic input, 0 = PCI agent mode, 1 = system controller (required for CPU configuration)

2.4 System Interface Extension

The System Interface Extension is realized using an 80-pin, HIROSE FX8C-80P-SV connector designated as J2 (Pn2) which is designed to mate with an 80-pin, HIROSE FX8C-80S-SV connector on the EB8347 carrier board, and it is used to provide CPU architecture specific system interfaces.

Table 2-10: System Interface Extension J2 (Pn2) Signal Types

SIGNAL TYPE	DESCRIPTION
Power	EB8347 E²Brain™ module input power and grounds
Local bus	Supports up to 16-bit data IO for external devices such as CompactFlash in true IDE mode for example and is user configurable
SPI bus	Single Serial Peripheral Interface bus
USB	Dual USB 2.0 interface; one master/device and one master
CRT analog	Standard analog CRT display signals with H/V sync
AC97	AC97 sound interface
DDC bus (I²C)	Display Data Channel bus (I²C) for readout of LCD/TFT display information for display driver support

The following table provides signal pinouts along with information concerning signal characteristics for connector J2 (Pn2) of EB8347 E²Brain™ module.

Table 2-11: Pinout of J2 (Pn2) Connector

PIN	SIGNAL	I/O	REMARKS
1	GND	I	
2	NC		
3	CF_D3 (XD3)	I/O	Internal serial resistor 33Ω
4	CF_D11 (XD11)	I/O	Internal serial resistor 33Ω
5	CF_D4 (XD4)	I/O	Internal serial resistor 33Ω
6	+3.3V	I	
7	+3.3V	I	
8	CF_D12 (XD12)	I/O	Internal serial resistor 33Ω
9	CF_D5 (XD5)	I/O	Internal serial resistor 33Ω
10	CF_D13 (XD13)	I/O	Internal serial resistor 33Ω
11	CF_D6 (XD6)	I/O	Internal serial resistor 33Ω
12	GND	I	
13	GND	I	
14	CF_D14 (XD14)	I/O	Internal serial resistor 33Ω
15	CF_D7 (XD7)	I/O	Internal serial resistor 33Ω
16	CF_D15 (XD15)	I/O	Internal serial resistor 33Ω
17	CF_CS0	O	Internal serial resistor 33Ω
18	+3.3V	I	

Table 2-11: Pinout of J2 (Pn2) Connector

PIN	SIGNAL	I/O	REMARKS
19	GND	I	
20	CF_CS1	O	Internal serial resistor 33Ω
21	CF_RD (XRD)	O	Internal serial resistor 33Ω
22	CF_WR (XWR)	O	Internal serial resistor 33Ω
23	CF_INTRQ	I	Internal serial resistor 33Ω (mapped to CPU-IRQ11)
24	GND	I	
25	GND	I	
26	CF_RST	O	
27	CF_IORDY	?	Internal serial resistor 33Ω
28	CF_A2	O	Internal serial resistor 33Ω
29	CF_A1	O	Internal serial resistor 33Ω
30	GND	I	
31	+3.3V	I	
32	CF_A0	O	Internal serial resistor 33Ω
33	NC		
34	CF_D0 (XD0)	I/O	Internal serial resistor 33Ω
35	CF_PDIAG	?	Internal pullup 4.7 kOhm
36	+3.3V	I	
37	GND	I	
38	CF_D1	I/O	Internal serial resistor 33Ω
39	CF_D8	I/O	Internal serial resistor 33Ω
40	CF_D2	I/O	Internal serial resistor 33Ω
41	CF_D9	I/O	Internal serial resistor 33Ω
42	GND	I	
43	V(I/O)	I	Additional V(I/O) pin for PCI signaling (must be connected to 3.3V)
44	NC		
45	CF_D10	I/O	Internal serial resistor 33Ω
46	NC		
47	NC		
48	GND	I	
49	+3.3V	I	
50	NC		
51	AC97_BIT_CLK	I	Internal pull-up: 10kΩ
52	AC97_SYNC	O	Internal pull-down: 4.7kΩ
53	AC97_SDATA_IN	I	
54	GND	I	
55	GND	I	
56	AC97_SDATA_OUT	O	

Table 2-11: Pinout of J2 (Pn2) Connector

PIN	SIGNAL	I/O	REMARKS
57	USB_OC#	O	Internal pull-up: 10kΩ (Over current condition indicator for Host ports, active low)
58	AC97_RST	O	Internal pull-up: 10kΩ
59	USB1_PO	I	Power ON control for Host ports (active high)
60	+3.3V	I	
61	GND	I	
62	Reserved	-	
63	CRT_R	O	Internal termination 75Ω to GND plus ESD protection
64	Reserved	-	
65	CRT_G	O	Internal termination 75Ω to GND plus ESD protection
66	GND	I	
67	CRT_B	O	Internal termination 75Ω to GND plus ESD protection
68	SPI_MOSI	I/O	Master out, slave in
69	CRT_VSYNC	O	
70	SPI_MISO	I/O	Master in, slave out
71	+3.3V	I	
72	SPI_SEL	I	Internal pull-up: 1kΩ
73	CRT_HSYNC	O	
74	SPI_CLK	O	
75	USB2_PO	O	Power ON control for Host ports (active high)
76	GND	I	
77	USB1+	I/O	Differential USB1 +
78	USB2+	I/O	Differential USB2 +
79	USB1-	I/O	Differential USB1 -
80	USB2-	I/O	Differential USB2 -

2.4.1 CompactFlash Interface

The following table provides a summary of the CompactFlash signals implemented on the System Interface Extension connector.

Table 2-12: CompactFlash Interface Signals

SIGNAL	DESCRIPTION
CF_D[15:0]	Compact Flash data bus – 16-bit wide
CF_CS[1:0]	Compact Flash chip select
CF_RD	Compact Flash IO read strobe
CF_WR	Compact Flash IO write strobe
CF_RST	Compact Flash reset, active low
CF_INTRQ	Compact Flash interrupt request, active high

Table 2-12: CompactFlash Interface Signals

SIGNAL	DESCRIPTION
CF_IORDY	Compact Flash IO ready
CF_A[2:0]	Compact Flash address lines

Note ...

The CompactFlash interface is realized as a true IDE interface (PIO mode). The PC CARD Memory Mode and the PC Card I/O Mode of the Compact Flash Specification are not supported.

In addition, the pinout for the Compact Flash signals have been optimized for routing to a CompactFlash socket.

2.4.2 AC97 Interface

The following table provides a summary of the AC97 signals implemented on the System Interface Extension connector.

Table 2-13: AC97 Interface Signals

SIGNAL	DESCRIPTION
AC97_BIT_CLK	Serial data clock (12.288MHz)
AC97_RST	Codec reset
AC97_SDATA_IN	Serial data in
AC97_SDATA_OUT	Serial data out
AC97_SYNC	Fixed rate sample sync (48kHz)

2.4.3 SPI Interface

The following table provides a summary of the SPI signals implemented on the System Interface Extension connector.

Table 2-14: SPI Interface Signals

SIGNAL	DESCRIPTION
SPI_CLK	Input/output serial clock
SPI_MISO	Master in, slave out
SPI_MOSI	Master out, slave in
SPI_SEL	SPI slave select (input)

2.4.4 CRT Interface

The following table provides a summary of the CRT signals implemented on the System Interface Extension connector.

Table 2-15: CRT Interface Signals

SIGNAL	DESCRIPTION
CRT_R	CRT red signal (analog): red color information signal
CRT_G	CRT green signal (analog): green color information signal
CRT_B	CRT blue signal (analog): blue color information signal
CRT_HSYNC	CRT horizontal synchronization pulse
CRT_VSYNC	CRT vertical synchronization pulse

2.4.5 USB Interface

The following table provides a summary of the USB signals implemented on the System Interface Extension connector.

Table 2-16: USB Interface Signals

SIGNAL	DESCRIPTION
USB_OC#	Over current condition indicator for Host ports (active low)
USB1_PO	Power ON control for Host port 1 (active high)
USB2_PO	Power ON control for Host port 2 (active high)
USB1-	Differential USB1 - (USB port 1, master only)
USB1+	Differential USB1 + (USB port 1, master only)
USB2-	Differential USB2 - (USB port 2), master or device)
USB2+	Differential USB2 + (USB port 2, master or device)

2.5 Communications Interface

The Communications Interface Connector J3 (Pn3), is used to provide a set of standard communications interfaces. In the case of the EB8347, there are two types of interfaces provided: two high speed serial interfaces and one Gigabit Ethernet interface.

Table 2-17: Communications Interface J3 (Pn3) Signal Types

SIGNAL TYPE	DESCRIPTION
High Speed Serial (HSS)	Two high speed serial interfaces (SER1, SER2)
Gigabit Ethernet	One Gigabit Ethernet 10/100/1000 Mbps interface

All of these interfaces are provided on the Communications Interface J3 (Pn3) connector (an 80-pin, HIROSE FX8C-80P-SV connector) which is designed to mate with an 80-pin, HIROSE FX8C-80S-SV connector on the EB8347 carrier board.

The following table provides pinout information for J3 (Pn3).

Table 2-18: Pinout of J3 (Pn3) Connector

PIN	SIGNAL	REMARKS
1	GND	
2	GND	
3	NC	
4	NC	
5	SER1_RTS	
6	SER1_CTS	Input signal levels must not exceed: – 0.3V (low) or 3.3V (high)
7	SER1_TXD	
8	SER1_RXD	Internal pull-up: 10kΩ; input signal levels must not exceed: – 0.3V (low) or 3.3V (high)
9	NC	
10	NC	
11	GND	
12	NC	
13	NC	
14	SER2_CTS	Input signal levels must not exceed: – 0.3V (low) or 3.3V (high)
15	SER2_RTS	
16	SER2_RXD	Internal pull-up 10kΩ; input signal levels must not exceed: – 0.3V (low) or 3.3V (high)
17	SER2_TXD	
18	NC	
19	NC	
20	NC	
21	NC	
22	GND	
23	NC	
24	NC	
25	NC	
26	NC	
27	NC	
28	NC	
29	NC	
30	NC	
31	NC	
32	NC	
33	NC	

Table 2-18: Pinout of J3 (Pn3) Connector

PIN	SIGNAL	REMARKS
34	NC	
35	NC	
36	NC	
37	NC	
38	NC	
39	NC	
40	NC	
41	NC	
42	NC	
43	NC	
44	NC	
45	GND	
46	GND	
47	NC	
48	NC	
49	NC	
50	NC	
51	NC	
52	NC	
53	NC	
54	GND	
55	NC	
56	NC	
57	NC	
58	NC	
59	NC	
60	NC	
61	NC	
62	NC	
63	2.5V	Usable for center tap of Gigabit Ethernet magnetic
64	2.5V	Usable for center tap of Gigabit Ethernet magnetic
65	ETH1_BIDC-	Internal termination, MDI interface, only magnetics required
66	ETH1_BIDD-	Internal termination, MDI interface, only magnetics required
67	ETH1_BIDC+	Internal termination, MDI interface, only magnetics required
68	ETH1_BIDD+	Internal termination, MDI interface, only magnetics required
69	ETH1_T- / BIDA-	Internal termination, MDI interface, only magnetics required
70	ETH1_R- / BIDB-	Internal termination, MDI interface, only magnetics required
71	ETH1_T+ / BIDA+	Internal termination, MDI interface, only magnetics required

Table 2-18: Pinout of J3 (Pn3) Connector

PIN	SIGNAL	REMARKS
72	ETH1_R+ / BIDB+	Internal termination, MDI interface, only magnetics required
73	GND	
74	ETH1_SD	Signal detect for fiber mode
75	ETH1_LINK_LED	Active low, drives LED from source to sink, drive capacity 10mA
76	ETH1_1000_LED	Active low, drives LED from source to sink, drive capacity 10mA
77	ETH1_ACT_LED	Active low, drives LED from source to sink, drive capacity 10mA
78	ETH1_100_LED	Active low, drives LED from source to sink, drive capacity 10mA
79	GND	
80	GND	

2.5.1 High Speed Serial Interfaces

Two, full modem, serial ports (SER1 and SER2) are available on the EB8347 E²Brain™ module. Four signals per port are provided to realize asynchronous high speed serial links interfaced using dedicated controlling/handshaking. The EB8347 uses the dual UARTs provided by the MMCC device which are 16550 compatible and provide the RTS and CTS handshake signals.

Table 2-19: High Speed Serial Interface Signal Type and Description

SIGNAL	DESCRIPTION
SERn_TXD	Transmit data output
SERn_RXD	Receive data input
SERn_RTS	Request to send output
SERn_CTS	Clear to send input

Note ...

All signals are available and supplied at 3.3V TTL levels. Further signal conditioning via appropriate transceivers on the carrier board is required to support the respective communications standards.

2.5.2 Gigabit Ethernet Interface

The EB8347 module provides one Gigabit Ethernet interface whose signals are already at copper Ethernet transmission voltage levels (physical levels) for CAT5 cabling. So the carrier board needs to add only the galvanic isolation (magnetics) function and the appropriate transmission connector type.

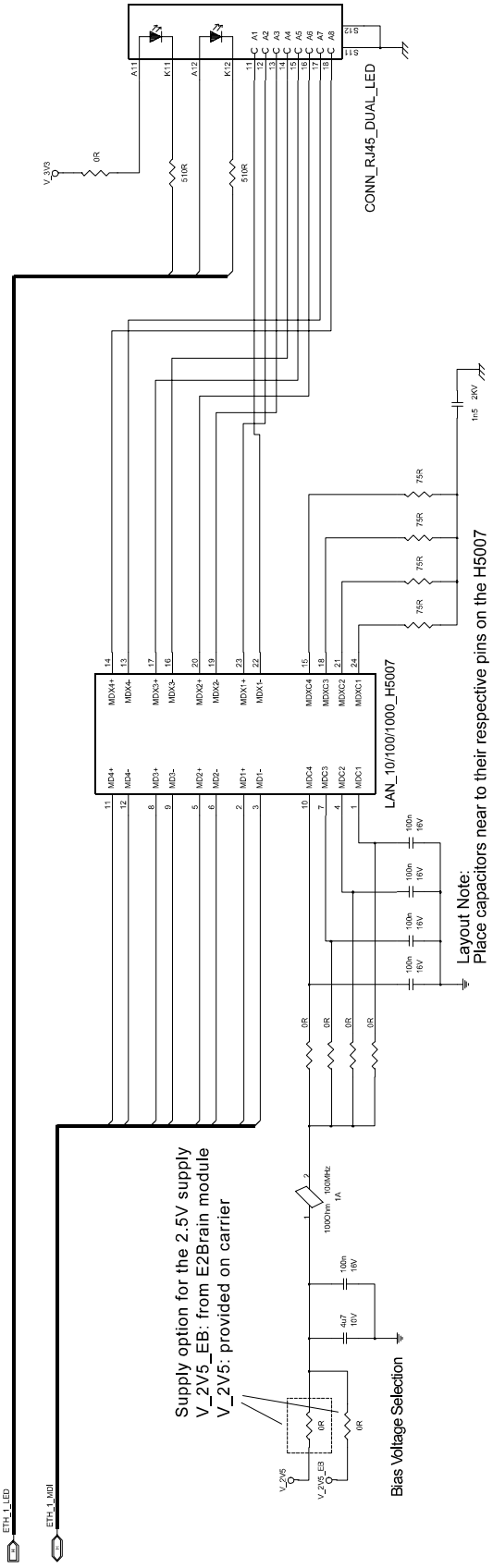
Additionally, for monitoring and control purposes, LED functionality is provided to indicate activity, link, and speed status information for the respective ports.

Table 2-20: Gigabit Ethernet Port Signal Type and Description

SIGNAL	DESCRIPTION
ETH1_T+ / BIDA+	Transmit pair in 10BaseT/100BaseTX configuration Channel A pair of Media Dependent Interface in 1000BaseT configuration
ETH1_T- / BIDA-	
ETH1_R+ / BIDB+	Receive pair in 10BaseT/100BaseTX configuration Channel B pair of Media Dependent Interface in 1000BaseT configuration
ETH1_R- / BIDB-	
ETH1_BIDC+	Channel C pair of Media Dependent Interface in 1000BaseT configuration
ETH1_BIDC-	
ETH1_BIDD+	Channel D pair of Media Dependent Interface in 1000BaseT configuration
ETH1_BIDD-	
2.5V	2.5V for magnetics transformer center tap
ETH1_LINK_LED	Indicates that link is present Steady on: link is present
ETH1_ACT_LED	Indicates that link is actively transmitting data Blinking: traffic on link
ETH1_100_LED	Indicates link speed Out: 10 Mb On: 100 Mb
ETH1_1000_LED	Indicates link speed Out: 10/100 Mb On: 1000 Mb

A typical magnetics circuit design where the LEDs are integrated in the RJ-45 connector is provided in the following figure.

Figure 2-4: Typical Magnetics Circuit Design



2.6 Communications Interface Extension

The Communications Interface Extension is realized using a 140-pin, HIROSE FX8C-140P-SV connector designated as J4 (Pn4) which is designed to mate with a 140-pin, HIROSE FX8C-140S-SV connector on the EB8347 carrier board. The following table provides an overview of the signal types and a brief description of the interfacing realized on this connector.

Table 2-21: Communications Interface Extension J4 (Pn4) Signal Type

SIGNAL TYPE	DESCRIPTION
GPIO(Address)	General Purpose Inputs and Outputs (Local bus addressing)
TMDS	Transition Minimized Differential Signal PanelLink DVI interface
ZV	Zoom Video port
PWM	Pulse Width Modulation flat panel, TFT or LCD, brightness control output
FP	TTL Flat Panel interface
TSEC1	Three Speed Ethernet Controller GMII interface (Gigabit Ethernet)

The following table provides signal pinouts along with information concerning signal characteristics for connector J4 (Pn4) of EB8347 E²Brain™ module.

Table 2-22: Pinout of J4 (Pn4) Connector

PIN	SIGNAL	REMARKS
1	GND	
2	GND	
3	GPIO25 (XCS1)	Provides the local bus chipselect of the MPC8347
4	LCLK	Local bus clock
5	GPIO23 (XA23)	May be configured as GPIO23 (MPC8347 GPIO11) or local bus address line A23
6	GPIO22 (XA22)	May be configured as GPIO22 (MPC8347 GPIO10) or local bus address line A22
7	GPIO21 (XA21)	May be configured as GPIO21 (MPC8347 GPIO9) or local bus address line A21
8	GPIO20 (XA20)	May be configured as GPIO20 (MPC8347 GPIO8) or local bus address line A20
9	GPIO19 (XA19)	May be configured as GPIO19 (MPC8347 GPIO7) or local bus address line A19
10	GPIO18 (XA18)	May be configured as GPIO18 (MPC8347 GPIO6) or local bus address line A18
11	GND	
12	GND	
13	GPIO17 (XA17)	May be configured as GPIO17 (MPC8347 GPIO5) or local bus address line A17
14	GPIO16 (XA16)	May be configured as GPIO16 (MPC8347 GPIO4) or local bus address line A16
15	GPIO15 (XA15)	May be configured as GPIO15 (MPC8347 GPIO3) or local bus address line A15
16	GPIO14 (XA14)	May be configured as GPIO14 (MPC8347 GPIO2) or local bus address line A14
17	GPIO13 (XA13)	May be configured as GPIO13 (MPC8347 GPIO1) or local bus address line A13
18	GPIO12 (XA12)	May be configured as GPIO12 (MPC8347 GPIO0) or local bus address line A12
19	GPIO11 (XA11)	May be configured as GPIO11 (BPCC GPIO11) or local bus address line A11
20	GPIO10 (XA10)	May be configured as GPIO10 (BPCC GPIO10) or local bus address line A10
21	GPIO9 (XA9)	May be configured as GPIO9 (BPCC GPIO9) or local bus address line A9

Table 2-22: Pinout of J4 (Pn4) Connector

PIN	SIGNAL	REMARKS
22	GND	
23	GPIO7 (XA7)	May be configured as GPIO7 (BPCC GPIO7) or local bus address line A7
24	GPIO8 (XA8)	May be configured as GPIO8 (BPCC GPIO8) or local bus address line A8
25	GPIO5 (XA5)	May be configured as GPIO5 (BPCC GPIO5) or local bus address line A5
26	GPIO6 (XA6)	May be configured as GPIO6 (BPCC GPIO6) or local bus address line A6
27	GPIO3 (XA3)	May be configured as GPIO3 (BPCC GPIO3) or local bus address line A3
28	GPIO4 (XA4)	May be configured as GPIO4 (BPCC GPIO4) or local bus address line A4
29	GPIO1 (XA1)	May be configured as GPIO1 (BPCC GPIO1) or local bus address line A1
30	GPIO2 (XA2)	May be configured as GPIO2 (BPCC GPIO2) or local bus address line A2
31	LGTA	MPC8347 LGTA signal
32	GPIO0 (XA0)	May be configured as GPIO0 (BPCC GPIO0) or local bus address line A0
33	NC	
34	GND	
35	GND	
36	HTPLG	
37	DVI_TXCLK-	TMDS (PanelLink) differential clock
38	DVI_TxOUT2-	TMDS (PanelLink) differential signal
39	DVI_TxCLK+	TMDS (PanelLink) differential clock
40	DVI_TxOUT2+	TMDS (PanelLink) differential signal
41	DVI_TxOUT1-	TMDS (PanelLink) differential signal
42	DVI_TxOUT0-	TMDS (PanelLink) differential signal
43	DVI_TxOUT1+	TMDS (PanelLink) differential signal
44	DVI_TxOUT0+	TMDS (PanelLink) differential signal
45	GND	
46	GND	
47	PWM0 (GPIO29)	GPIO29 signal of the MMCC, can be used as PWM0 output
48	PWM2 (GPIO31)	GPIO31 signal of the MMCC, can be used as PWM2 output
49	NC	
50	PWM1 (GPIO30)	GPIO30 signal of the MMCC, can be used as PWM1 output
51	ZV15 (GPIO63)	Zoom Video Port provided by MMCC SM501
52	ZV_CLK	Zoom Video Port provided by MMCC SM501
53	ZV14 (GPIO62)	Zoom Video Port provided by MMCC SM501
54	ZV_HREF	Zoom Video Port provided by MMCC SM501
55	ZV7 (GPIO23)	Zoom Video Port provided by MMCC SM501
56	GND	
57	ZV6 (GPIO22)	Zoom Video Port provided by MMCC SM501
58	ZV_VSYNC	Zoom Video Port provided by MMCC SM501
59	ZV5 (GPIO21)	Zoom Video Port provided by MMCC SM501

Table 2-22: Pinout of J4 (Pn4) Connector

PIN	SIGNAL	REMARKS
60	ZV4 (GPIO20)	Zoom Video Port provided by MMCC SM501
61	ZV3 (GPIO19)	Zoom Video Port provided by MMCC SM501
62	ZV2 (GPIO18)	Zoom Video Port provided by MMCC SM501
63	ZV1 (GPIO17)	Zoom Video Port provided by MMCC SM501
64	ZV0 (GPIO16)	Zoom Video Port provided by MMCC SM501
65	reserved	
66	GND	
67	NC	
68	FP23	TTL Flat Panel interface
69	GND	
70	FP22	TTL Flat Panel interface
71	SM501_BIAS	TTL Flat Panel interface
72	FP21	TTL Flat Panel interface
73	FP_VDEN	TTL Flat Panel interface
74	FP20	TTL Flat Panel interface
75	NC	
76	FP19	TTL Flat Panel interface
77	FP_EN	TTL Flat Panel interface
78	FP18	TTL Flat Panel interface
79	FP17 (GPIO61)	TTL Flat Panel interface
80	GND	
81	GND	
82	FP16 (GPIO60)	TTL Flat Panel interface
83	FP15	TTL Flat Panel interface
84	FP14	TTL Flat Panel interface
85	FP13	TTL Flat Panel interface
86	FP12	TTL Flat Panel interface
87	FP11	TTL Flat Panel interface
88	FP10	TTL Flat Panel interface
89	FP9 (GPIO59)	TTL Flat Panel interface
90	GND	
91	FP7	TTL Flat Panel interface
92	FP8 (GPIO58)	TTL Flat Panel interface
93	FP5	TTL Flat Panel interface
94	FP6	TTL Flat Panel interface
95	FP3	TTL Flat Panel interface
96	FP4	TTL Flat Panel interface
97	FP1 (GPIO57)	TTL Flat Panel interface

Table 2-22: Pinout of J4 (Pn4) Connector

PIN	SIGNAL	REMARKS
98	FP2	TTL Flat Panel interface
99	FP_HSYNC	TTL Flat Panel interface
100	FP0 (GPIO56)	TTL Flat Panel interface
101	FP_VSYNC	TTL Flat Panel interface
102	GND	
103	FP_CLK	TTL Flat Panel interface
104	FP_DISP	TTL Flat Panel interface
105	GND	
106	NC	
107	NC	
108	TSEC1_TXD7	Three speed Ethernet controller GMII interface
109	TSEC1_TXD6	Three speed Ethernet controller GMII interface
110	TSEC1_TXD5	Three speed Ethernet controller GMII interface
111	TSEC1_TXD4	Three speed Ethernet controller GMII interface
112	TSEC1_TXD3	Three speed Ethernet controller GMII interface
113	TSEC1_TXD2	Three speed Ethernet controller GMII interface
114	GND	
115	GND	
116	TSEC1_TXD1	Three speed Ethernet controller GMII interface
117	TSEC1_TXD0	Three speed Ethernet controller GMII interface
118	TSEC1_RXD1	Three speed Ethernet controller GMII interface
119	TSEC1_RXD0	Three speed Ethernet controller GMII interface
120	TSEC1_RXD3	Three speed Ethernet controller GMII interface
121	TSEC1_RXD2	Three speed Ethernet controller GMII interface
122	TSEC1_RXD5	Three speed Ethernet controller GMII interface
123	TSEC1_RXD4	Three speed Ethernet controller GMII interface
124	GND	
125	TSEC1_RXD6	Three speed Ethernet controller GMII interface
126	TSEC1_RXD7	Three speed Ethernet controller GMII interface
127	TSEC1_COL	Three speed Ethernet controller GMII interface
128	TSEC1_CRS	Three speed Ethernet controller GMII interface
129	TSEC1_TX_ER	Three speed Ethernet controller GMII interface
130	TSEC1_TX_EN	Three speed Ethernet controller GMII interface
131	TSEC1_RX_DV	Three speed Ethernet controller GMII interface
132	TSEC1_RX_ER	Three speed Ethernet controller GMII interface
133	EC_MDIO	Ethernet Management Interface
134	TSEC1_RxClk	Three speed Ethernet controller GMII interface
135	TSEC1-GTX-CLK	Three speed Ethernet controller GMII interface

Table 2-22: Pinout of J4 (Pn4) Connector

PIN	SIGNAL	REMARKS
136	GND	
137	EC_MDC	Ethernet Management Interface
138	TSEC1_TxClk	Three speed Ethernet controller GMII interface
139	GND	
140	GND	

Note ...

For more information concerning the three speed Ethernet interface, refer to the MPC8347 PowerQUICC™ II Pro Integrated Host Processor Reference Manual from Freescale.

2.6.1 GPIO/Address Interfaces

The EB8347 module provides up to twenty-four GPIO signals or address lines depending on application requirements. These signals are user configurable and provide interfacing to carrier devices for monitor and control functions or data transfer to/from the EB8347 module.

Table 2-23: GPIO/Address Signal Type and Description

SIGNAL	I/O	DESCRIPTION
GPIO[23:0]	I/O	General purpose digital input/output signals (12 CPU IOs; 12 BPCC IOs)
XA[23:0]	O	Address signals, up to 24-bit
XCS1	O	Chip Select 1
LCLK	O	Local bus clock
LGTA	O	Local bus gate

2.6.2 TMDS DVI Interface

The EB8347 module provides all of the signals required for a complete TMDS PanelLink™ DVI interface. The electrical characteristics of these signals are low voltage differential and therefore care must be taken when connecting them to the DVI connector on the carrier board.

Table 2-24: TMDS DVI Signal Type and Description

SIGNAL	I/O	DESCRIPTION
DVI_TxCLK+/-	O	TMDS PanelLink clock channel
DVI_TxOUT0+/-	O	TMDS PanelLink output data channel 0 (differential signal)
DVI_TxOUT1+/-	O	TMDS PanelLink output data channel 1 differential signal)
DVI_TxOUT2+/-	O	TMDS PanelLink output data channel 2 (differential signal)
HTPLG	O	Hot plug signal

2.6.3 Zoom Video Interface

The EB8347 MMCC Zoom Video interface can interface with video decoders, such as NTSC/PAL decoders, MPEG-2 decoders, and JPEG codecs. The ZV interface supports resolutions up to 1280 x 1024 pixels. It directly accepts digitized RGB or YUV signals, but does not accept analog signals.

Table 2-25: Zoom Video Signal Type and Description

SIGNAL	I/O	DESCRIPTION
ZV[15:0] (GPIOnn)	I	Zoom video input signals (up to 16-bit) (the GPIOnn refers to GPIO signal pin of the MMCC).
ZV_CLK	I	Zoom video clock
ZV_HREF	I	Zoom horizontal reference
ZV_VSYNC	I	Zoom video vertical synchronization

2.6.4 Flat Panel Display Interface

The EB8347 module provides also TTL-level flat panel signals for connecting either a 16- or 24-bit pro pixel parallel TFT display. This interface includes the display control signals and the display power management signals which are all provided by the MMCC.

Table 2-26: Flat Panel Display Signal Type and Description

SIGNAL	I/O	DESCRIPTION
FP[23:0]	O	Flat panel display data
SM501_BIAS	O	On/off switching control signal for panel backlighting
FP_VDEN	O	Flat panel VDD enable
FP_EN	O	Flat panel enable
FP_HSYNC	O	Flat panel TFT horizontal sync
FP_VSYNC	O	Flat panel TFT vertical sync
FP_CLK	O	Flat panel pixel clock
FP_DISP	O	Flat panel display enable

Table 2-27: Flat Panel Color Mapping

DATA OUTPUT	TFT			
	24-BIT	18-BIT	12-BIT	9-BIT
FP23	R7	R5	R3	R2
FP22	R6	R4	R2	R1
FP21	R5	R3	R1	R0
FP20	R4	R2	R0	
FP19	R3	R1		
FP18	R2	R0		
GPIO61	R1			
GPIO60	R0			
FP15	G7	G5	G3	G2
FP14	G6	G4	G2	G1
FP13	G5	G3	G1	G0
FP12	G4	G2	G0	
FP11	G3	G1		
FP10	G2	G0		
GPIO59	G1			
GPIO58	G0			
FP7	B7	B5	B3	B2
FP6	B6	B4	B2	B1
FP5	B5	B3	B1	B0
FP4	B4	B2	B0	
FP3	B3	B1		
FP2	B2	B0		
GPIO57	B1			
GPIO56	B0			

2.6.5 PWM Interface

The EB8347 MMCC PWM interface serves to provide brightness control for flat panel displays, TFT or LCD.

Table 2-28: Zoom Video Signal Type and Description

SIGNAL	I/O	DESCRIPTION
PWM[2:0] (GPIOnn)	I	Three independent pulse width modulation signals for flat panel display brightness control (the GPIOnn refers to GPIO signal pin of the MMCC).

2.6.6 Gigabit Ethernet Interface

The EB8347 module provides two three speed Ethernet (GMII) interfaces (TSEC1 and TSEC2) whose signals are already at TTL level. The carrier board must provide the PHY, galvanic isolation (magnetics) function, and the appropriate transmission connector type for each Ethernet interface.

In addition, if LED functionality is required, it must be provided by the PHY on the carrier board for the respective port.

Table 2-29: Gigabit Ethernet Signal Type and Description

SIGNAL	I/O	DESCRIPTION
TSECn_COL	I	Collision
TSECn_CRS	I	Carrier sense
TSECn_GTX_CLK	O	Transmit clock
EC_GTX_CLK125	I	Oscillator source for GMII
EC_MDC	O	Management clock
EC_MDIO	I/O	Management data
TSECn_RX_CLK	I	Receive clock
TSECn_RX_DV	I	Receive data valid
TSECn_RXD[7:0]	I	Receive data bits 7:0
TSECn_RX_ER	I	Receive error
TSECn_TX_CLK	I	Transmit clock
TSECn_TXD[7:0]	O	Transmit data bits 7:0
TSECn_TX_EN	O	Transmit data valid
TSECn_TX_ER	O	Transmit error
TSECn_TX_ER	O	Transmit error

2.7 Monitor and Control (M/C)

Monitor and Control functions are divided essentially into Pre-operation and Operation. Pre-operation M/C deals with board configuration and system requirements. Operation M/C covers direct operational interfaces. For further information regarding Monitor and Control functions refer to chapters 2.3.4 and 4.

2.7.1 Pre-Operation M/C

Pre-operation M/C is a direct function of the application and the system requirements. These requirements dictate the EB8347 configuration as well as the overall system integration. Overall system integration and compliance with its requirements is beyond the scope of this manual.

2.7.2 Operation M/C

Operation M/C is primarily a function of the EB8347 driver software and the application. M/C signals are available, and, if implemented as part of the application, the operator as well as application software has access to these functions.

This page has been intentionally left blank.

Chapter 3

Installation

This page has been intentionally left blank.

3. Installation

The EB8347 has been designed for easy installation. However, the following standard precautions, installation procedures, and general information must be observed to ensure proper installation and to preclude damage to the board or injury to personnel.

3.1 Hardware Installation

The product described in this manual may only be mounted on an appropriate E²Brain™ carrier board which is specifically designed for this E²Brain™ module.

3.1.1 Safety Requirements

The module must be securely fastened to the carrier board using the mounting standoffs and screws provided with the module.

In addition the following electrical hazard precautions must be observed.

CAUTION, ELECTRIC SHOCK!

Ensure that the system main power is removed prior to installing or removing this board. Ensure that there are no other external voltages or signals being applied to this board or other boards within the system. Failure to comply with the above could endanger your life or health and may cause damage to this board or other system components including process-side signal conditioning equipment.

ESD Sensitive Device!

This Kontron board contains electrostatically sensitive devices. 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 any on board components, connector pins, or board conductive circuits.

If working at an anti-static workbench with professional discharging equipment, ensure compliance with its usage when handling this product.

3.1.2 Installation Procedures

To install this E²Brain™ module proceed as follows:

1. Ensure that the handling and safety requirements indicated in chapter 3.1 are observed.

WARNING!

Failure to comply with the instruction below may cause damage to the board or result in improper system operation. Please refer to chapters 4 and 5 for configuration information.

2. Ensure that the board is properly configured for operation before installing.

Note ...

Care must be taken when applying the procedures below to ensure that when the module is mounted on the carrier board it is not damaged through contact with other boards in the system.

3. To install the E²Brain™ module perform the following:
   1. Orient the E²Brain™ module as appropriate to the carrier board and engage it with the carrier board.
   2. Fasten all mounting screws provided with the E²Brain™ module ensuring that the mounting standoffs are also properly fastened.
   3. As appropriate, install the carrier board in the application system.

3.1.3 Removal Procedures

To remove this module proceed as follows:

1. Ensure that the handling and safety requirements indicated in chapter 3.1 are observed.

WARNING!

Care must be taken when applying the procedures below to ensure that when the board is removed it is not damaged through contact with other boards in the system.

2. Disconnect any interfacing cable(s) that may be connected to the module.
3. Remove all module mounting screws.
4. Disengage the module from the carrier board.
5. Reinstall the module mounting screws in the module standoffs.

6. Dispose of the module as required observing applicable environmental regulations governing the handling and disposition of this type of product.

Note ...

If the removed module is to be returned to the manufacturer, ensure that the packaging and ESD requirements are observed as specified by the following sections of this guide:

- page xvii, "Special Handling and Unpacking Instructions"
- page xviii, "General Instructions on Usage"
- section 1.8, "Applied Standards"

3.2 Software Installation

Installation of the EB8347 driver software is a function of the application operating system. For further information refer to the appropriate software documentation.

This page has been intentionally left blank.

Chapter 4

Configuration

This page has been intentionally left blank.

4. Configuration

The following sections provide system integrators with detailed information for configuring the EB8347 module for operation

4.1 Board Address Map

The following table illustrates the address mapping of the EB8347.

Table 4-1: Board Address Map

	AREA NAME	SIZE BYTE	START ADDRESS
OVERLAP	MPC8347(E) REGs	4 M	0xFFC0 0000
	PCI MEM	2 G	0x8000 0000
	Reserved		0x4F00 0000
	CUS SPE	16 M	0x4E00 0000
	ONBOARD REGs	8 M	0x4D80 0000
	LPC IO	8 M	0x4D00 0000
	LPC MEM	16 M	0x4C00 0000
	GFX	64 M	0x4800 0000
	NOR FLASH	64 M	0x4400 0000
	CF	64 M	0x4200 0000
	BOOT ROM		0x4000 0000
	PCI IO	64 K	0x3FFF 0000
	Reserved		0x2000 0000
	DDR-SRAM	2 G	0x0000 0000

START ADDRESS	REGISTER NAME	ACC.
0x4D80 001D	Device Interrupt Mask	RW
0x4D80 001C	Device Interrupt Pending	R
0x4D80 001B	GPIO TxData2	RW
0x4D80 001A	GPIO TxData1	RW
0x4D80 0019	GPIO RxData2	R
0x4D80 0018	GPIO RxData1	R
0x4D80 0017	GPIO DIR2	RW
0x4D80 0016	GPIO DIR1	RW
0x4D80 0015	GPIO CFG2	RW
0x4D80 0014	GPIO CFG1	RW
0x4D80 0013	Delay Timer Control/Status	RW
0x4D80 0012	Device Disable Register	R
0x4D80 0011	Serial Interrupt Polarity 2	RW
0x4D80 0010	Serial Interrupt Polarity 1	RW
0x4D80 000F	Serial Interrupt Mask Register 2	RW
0x4D80 000E	Serial Interrupt Mask Register 1	RW
0x4D80 000D	Serial Interrupt Pending 2	R
0x4D80 000C	Serial Interrupt Pending 1	R
0x4D80 000B	Reserved	-
0x4D80 000A	Reserved	-
0x4D80 0009	Board/Logic Revision	RW
0x4D80 0008	Watchdog Control Register	RW
0x4D80 0007	SYS Info Register	R
0x4D80 0006	Reserved	-
0x4D80 0005	Interrupt Config Register	RW
0x4D80 0004	Event Register	RW
0x4D80 0003	Control Register	RW
0x4D80 0002	Memory Configuration Register	R
0x4D80 0001	Software Compatibility ID	R
0x4D80 0000	Board ID	R

4.2 Board Control Registers

The Board Control registers may be accessed through byte-wide read and write operations.

4.2.1 Board ID Register

The Board ID is used to identify the EB8347 in a E²Brain™ system. The value for the EB8347 is 0x46 which is factory set and cannot be changed.

Table 4-2: Board ID Register

REGISTER NAME	BOARD ID						ACCESS	
ADDRESS	0x4D80 0000						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	BID7	BID6	BID5	BID4	BID3	BID2	BID1	BID0
DEFAULT	0	1	0	0	0	1	0	1

4.2.2 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 “NetBootLoader” and the BSP software, and, as such, is not user relevant.

Table 4-3: Software Compatibility ID

REGISTER NAME	SOFTWARE COMPATIBILITY ID						ACCESS	
ADDRESS	0x4D80 0001						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

4.2.3 Memory Configuration Register

The memory configuration register is used by the NetBootLoader to detect the right memory configuration. This information is then used to configure the Memory controller.

Table 4-4: Memory Configuration Register

REGISTER NAME		MEMORY CONFIGURATION						ACCESS	
ADDRESS		0x4D80 0002						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		MEM_ TECH	res.	MEM BUS CLOCK	ECC	res.	MEM_ SZ1	MEM_ SZ0	BUS_ WIDTH
DEFAULT		n/a	n/a	1	0	n/a	n/a	n/a	n/a
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	BUS_ WIDTH	0	32-bit interface						
		1	64-bit interface						
1	MEM_SZ0	0	Memory size: Bit 2 Bit 1 0 0 128 MB (256 Mbit chips) 0 1 256 MB (512 Mbit chips) 1 0 512 MB (1 Gbit chips) 1 1 1 GB (2 Gbit chips)						
		1							
2	MEM_SZ1	0							
		1							
3	reserved	0							
		1							
4	ECC	0	ECC not available						
		1	ECC available						
5	MEM BUS CLOCK	0	264 MHz (Maximum possible memory bus clock)						
		1	330 MHz (Maximum possible memory bus clock)						
6	reserved	0							
		1							
7	MEM_TECH	0	Boot Flash = NOR Flash						
		1	Boot Flash = SPI + NAND Flash						

4.2.4 Control Register

The Control register provides output interfacing to the application software. Assertion of the appropriate bits by the application software will cause the BPCC to generate outputs accordingly.

During startup, the state of Bit 0 is controlled by the NetBootLoader software. After the startup is completed, the NetBootLoader sets Bit 0 to 1.

The SPI_SEL0 is used for chipselect of the “on-module” SPI Flash. Writing a “1” enables this device.

Table 4-5: Control Register

REGISTER NAME	CONTROL						ACCESS	
ADDRESS	0x4D80 0003						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	SPI_ SEL0	MC10	S_RST	MC9	res.	MC11	MC6
DEFAULT	n/a	n/a	0	n/a	0	n/a	0	n/a

BIT	CONTENT	STATE	DESCRIPTION
0	MC6	0	Logical high on the MC6 output pin
		1	Logical low on the MC6 output pin
1	MC11	0	Logical high on the MC11 output pin
		1	Logical low on the MC11 output pin
2	reserved	0	
		1	
3	MC9	0	Logical low on the MC9 output pin
		1	High impedance (Z) on the MC9 output pin
4	S_RST	0	no operation
		1	Causes a complete system reset (S_RST) to be initiated
5	MC10	0	Logical low on the MC10 output pin
		1	High impedance (Z) on the MC10 output pin
6	SPI_SEL0	0	SPI Flash disabled
		1	SPI Flash enabled
7	reserved	0	
		1	

4.2.5 Event Register

The Event register provides status information about the Watchdog timer and various monitor and control inputs. Depending on the type of event which occurs, interrupts may be generated automatically which then require servicing. The application software is responsible for servicing the interrupts as well as the other events, and, where applicable, the resetting of the event bits.

Table 4-6: Event Register

REGISTER NAME		EVENT						ACCESS	
ADDRESS		0x4D80 0004						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		MC5	MC3	MC4	MC2	MC8	MC1	res.	WD
DEFAULT		n/a	n/a	0	n/a	n/a	0	n/a	0
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	WD	0	No interrupt has occurred						
		1	An interrupt has occurred						
1	reserved	0							
		1							
2	MC1	0	Indicates that the MC1 signal has not been asserted						
		1	Indicates that the MC1 signal has been asserted (This will result in a NMI being generated which can be cleared by writing a'1'.)						
3	MC8	0	Indicates that the EB8347 is being operated in Agent mode						
		1	Indicates that the EB8347 is being operated in Master mode						
4	MC2	0	Case: MC2_INT_EN = 0 Indicates the status of the MC2 input: electrical: low = 0, high = 1 Case: MC2_INT_EN = 1 (default = 0)						
		1	The falling edge of the signal on MC2 pin sets this bit to '1' and generates the MCInt on the CPU provided it is enabled there (This may be cleared by writing a '1'.)						
5	MC4	0	No interrupt has occurred						
		1	An interrupt has occurred						
6	MC3	0	Case: MC3_INT_EN = 0 Indicates the status of the MC3 input: electrical: low = 0, high = 1 Case: MC3_INT_EN = 1 (default = 0)						
		1	The falling edge of the signal on MC3 pin sets this bit to '1' and generates the MCInt on the CPU provided it is enabled there (This may be cleared by writing a'1'.)						
7	MC5	0	Case: MC3_INT_EN = 0 Indicates the status of the MC3 input: electrical: low = 0, high = 1 Case: MC3_INT_EN = 1 (default = 0)						
		1	The falling edge of the signal on MC3 pin sets this bit to'1' and generates the MCInt on the CPU provided it is enabled there (This may be cleared by writing a'1'.)						

4.2.6 Interrupt Configuration Register

The interrupt configuration register acts as an interrupt enable register for the MC2, MC3 and MC5 signals.

Table 4-7: Interrupt Configuration Register

REGISTER NAME		INTERRUPT CONFIGURATION						ACCESS	
ADDRESS		0x4D80 0005						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		MC5_INT_EN	MC3_INT_EN	res.	MC2_INT_EN	res.	res.	res.	res.
DEFAULT		0	0	n/a	0	n/a	n/a	n/a	n/a
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	reserved	0							
		1							
1	reserved	0							
		1							
2	reserved	0							
		1							
3	reserved	0							
		1							
4	MC2_INT_EN	0	Disabled						
		1	Enabled						
5	reserved	0							
		1							
6	MC3_INT_EN	0	Disabled						
		1	Enabled						
7	MC5_INT_EN	0	Disabled						
		1	Enabled						

4.2.7 System Information Register

This register provides information regarding the system configuration which is required by the system logic and the NetBootLoader software during startup.

Table 4-8: System Information Register

REGISTER NAME		SYSTEM INFORMATION						ACCESS	
ADDRESS		0x4D80 0007						R	
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		res.	res.	res.	res.	CORE_ PLL1	CORE_ PLL0	res.	M66EN
DEFAULT		n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	M66EN	0	33 MHz PCI						
		1	66 MHz PCI						
1	reserved	0							
		1							
2	CORE_PLL0	0	CPU core PLL configuration: Bit 3 Bit 2 0 0 csb_dk x 1 0 1 csb_dk x 1.5 1 0 csb_dk x 2 1 1 csb_dk x 2.5						
		1							
3	CORE_PLL1	0							
		1							
4	reserved	0							
		1							
5	reserved	0							
		1							
6	reserved	0							
		1							
7	reserved	0							
		1							

4.2.8 Watchdog Control Register

The Watchdog Control register is the interface between applications and the operating system for controlling the functioning of the Watchdog timer. There are four possible modes of operation involving the Watchdog timer:

- Timer only
- Reset
- Interrupt
- Dual stage

At power on the Watchdog is not enabled. If not required, it is not necessary to enable it. If required, the bits of the Watchdog Control register must be set according to application requirements. To operate the Watchdog, the mode and time period required must first be set and then the Watchdog enabled. Once enabled, the Watchdog can only be disabled or the mode changed by powering down and then up again. To prevent a Watchdog timeout the Watchdog must be retriggered before timing out. This is done by writing a ’1’ to the WTR bit. In the event a Watchdog timeout does occur, the WTE bit is set to ’1’. What transpires after this depends on the mode selected. The four operational Watchdog timer modes are described as follows.

Timer only - In this mode the Watchdog is enabled using the required timeout period. Normally the Watchdog is retriggered by writing a ’1’ to the WTR bit. In the event a timeout occurs, the WTE bit is set to ’1’. This bit can then be polled by the application and handled accordingly. To continue using the Watchdog, write a ’1’ to the WTE bit, and then retrigger the Watchdog using WTR. The WTE bit retains its setting as long as no power down-up is done. Therefore, this bit may be used to verify the status of the Watchdog.

Reset mode - This mode is used to force a hard reset in the event of a Watchdog timeout. To be effective, the hard reset must not be masked or otherwise negated. In addition, the WTE bit is not reset by the hard reset which makes it available if necessary to determine the status of the Watchdog prior to the reset.

Interrupt mode - This mode causes the generation of an interrupt in the event of a Watchdog timeout. The interrupt handling is a function of the application. If required the WTE bit can be used to determine if a Watchdog timeout has occurred.

Dual stage mode - This a complex mode where in the event of a timeout two things occur: 1) an interrupt is generated, and 2) the Watchdog is retriggered automatically. In the event a second timeout occurs immediately following the first timeout, a hard reset will be generated. If the Watchdog is retriggered normally, operation continues. The interrupt generated at the first timeout is available to the application to handle the first timeout if required. As with all of the other modes the WTE bit is available for application use.

Refer to section 7 for further information regarding the Watchdog functional logic.

Table 4-9: Watchdog Control Register

REGISTER NAME		WATCHDOG CONTROL						ACCESS	
ADDRESS		0x4D80 0008						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		WTE	WDM1	WDM0	WEN/WTR	WTM3	WTM2	WTM1	WTM0
DEFAULT		0	0	0	0	0	0	0	0
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	WTM0	0	Watchdog timeout time: Settings: **WTM3 WTM2 WTM1 WTM0** 0 0 0 0 125 ms 0 0 0 1 250 ms 0 0 1 0 500 ms 0 0 1 1 1 s 0 1 0 0 2 s 0 1 0 1 4 s 0 1 1 0 8 s 0 1 1 1 16 s 1 0 0 0 32 s 1 0 0 1 64 s 1 0 1 0 128 s 1 0 1 1 256 s 1 1 0 0 reserved 1 1 0 1 reserved 1 1 1 0 reserved 1 1 1 1 reserved					
		1							
1	WTM1	0							
		1							
2	WTM2	0							
		1							
3	WTM3	0							
		1							
4	WEN/WTR	0	Indicates that the Watchdog timer has not been enabled. Prior to the Watchdog being enabled, this bit is known as WEN. After the Watchdog is enabled, it is known as WTR. Once the Watchdog timer has been enabled, this bit cannot be reset to 0. As long as the Watchdog timer is enabled it will indicate a 1.						
		1	Indicates that the Watchdog timer is enabled. Writing a '1' to this bit causes the Watchdog to be retriggered to the timer value indicated by bits WTM[3:0].						
5	WMD0	0	Watchdog mode: Settings: **WMD1 WMD0 Mode** 0 0 Timer only 0 1 Reset 1 0 Interrupt 1 1 Dual Stage						
		1							
6	WMD1	0							
		1							
7	WTE	0	Indicates that the Watchdog timer has not expired.						
		1	Indicates that the Watchdog timer has expired. Writing a '1' to this bit resets it to 0.						

4.2.9 Board Logic / Revision Register

The Board Revision Register provides the hardware (BRn) and logic (LRn) status of the board. 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 4-10:Board Logic / Revision Register

REGISTER NAME	BOARD LOGIC/REVISION						ACCESS	
ADDRESS	0x4D80 0009						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

4.2.10 Serial Interrupt Pending 1 Register

The Serial Interrupt Pending 1 Register in conjunction with the Serial Interrupt Pending 2 Register is used to identify the source of the pending interrupt request. All serial interrupts are coupled together to one CPU Interrupt (IRQ7). A logical 1 indicates that an interrupt has been asserted.

Table 4-11:Serial Interrupt Pending 1 Register

REGISTER NAME	SERIAL INTERRUPT PENDING 1						ACCESS	
ADDRESS	0x4D80 000C						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SIRQ7	SIRQ6	SIRQ5	SIRQ4	SIRQ3	SIRQ2	SIRQ1	SIRQ0
DEFAULT	0	0	0	0	0	0	0	0

4.2.11 Serial Interrupt Pending 2 Register

The Serial Interrupt Pending 2 Register in conjunction with the Serial Interrupt Pending 1 Register is used to identify the source of the pending interrupt request. All serial interrupts are coupled together to one CPU Interrupt (IRQ7). A logical 1 indicates that an interrupt has been asserted.

Table 4-12:Serial Interrupt Pending 2 Register

REGISTER NAME	SERIAL INTERRUPT PENDING 2						ACCESS	
ADDRESS	0x4D80 000D						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SIRQ15	SIRQ14	SIRQ13	SIRQ12	SIRQ11	SIRQ10	SIRQ9	SIRQ8
DEFAULT	0	0	0	0	0	0	0	0

4.2.12 Serial Interrupt Mask 1 Register

The Serial Interrupt Mask 1 and 2 Registers enable the generation of a CPU interrupt. Writing a '1' to the bit “SIRQ_ENx” enables the generation of a CPU interrupt and enables the corresponding bit in the Serial Interrupt Pending Registers 1 and 2.

Table 4-13:Serial Interrupt Mask 1 Register

REGISTER NAME	SERIAL INTERRUPT MASK 1						ACCESS	
ADDRESS	0x4D80 000E						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SIRQ_EN 7	SIRQ_EN 6	SIRQ_EN 5	SIRQ_EN 4	SIRQ_EN 3	SIRQ_EN 2	SIRQ_EN 1	SIRQ_EN 0
DEFAULT	0	0	0	0	0	0	0	0

4.2.13 Serial Interrupt Mask 2 Register

The Serial Interrupt Mask Registers 1 and 2 enable the generation of a CPU interrupt. Writing a '1' to the bit “SIRQ_ENx” enables the generation of a CPU interrupt and enables the corresponding bit in the Serial Interrupt Pending Registers 1 and 2.

Table 4-14:Serial Interrupt Mask 2 Register

REGISTER NAME	SERIAL INTERRUPT MASK 2						ACCESS	
ADDRESS	0x4D80 000F						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SIRQ_EN 15	SIRQ_EN 14	SIRQ_EN 13	SIRQ_EN 12	SIRQ_EN 11	SIRQ_EN 10	SIRQ_EN 9	SIRQ_EN 8
DEFAULT	0	0	0	0	0	0	0	0

4.2.14 Serial Interrupt Polarity 1 Register

The Serial Interrupt Polarity 1 Register bits define the polarity of their corresponding serial interrupt. A '1' written to the required bit position results in an active high sensitivity of the corresponding interrupt and vice versa.

Table 4-15:Serial Interrupt Polarity 1 Register

REGISTER NAME	SERIAL INTERRUPT POLARITY 1						ACCESS	
ADDRESS	0x4D80 0010						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SerInt_POL 7	SerInt_POL 6	SerInt_POL 5	SerInt_POL 4	SerInt_POL 3	SerInt_POL 2	SerInt_POL 1	SerInt_POL 0
DEFAULT	0	0	0	0	0	0	0	0

4.2.15 Serial Interrupt Polarity 2 Register

The Serial Interrupt Polarity 2 Register bits define the polarity of their corresponding serial interrupt. A '1' written to the required bit position results in an active high sensitivity of the corresponding interrupt and vice versa.

Table 4-16:Serial Interrupt Polarity 2 Register

REGISTER NAME	SERIAL INTERRUPT POLARITY 2						ACCESS	
ADDRESS	0x4D80 0011						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	SerInt_POL 15	SerInt_POL 14	SerInt_POL 13	SerInt_POL 12	SerInt_POL 11	SerInt_POL 10	SerInt_POL 9	SerInt_POL 8
DEFAULT	0	0	0	0	0	0	0	0

4.2.16 Device Disable Register

The register indicates the devices within the CPU which are required to be disabled by the NetBootLoader firmware. This must be done via the SCCR register in the MPC8347 CPU.

Table 4-17:Device Disable Register

REGISTER NAME	DEVICE DISABLE REGISTER						ACCESS	
ADDRESS	0x4D80 0012						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	DIS_ USBDR	DIS_ USBMPH	DIS_ TSEC2	DIS_ TSEC1	res.	DIS_ENCR	res.	DIS_PCI
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

BIT	CONTENT	STATE	DESCRIPTION
0	DIS_PCI	0	Disabled
		1	Enabled
1	reserved	0	
		1	
2	DIS_ENCR	0	Encryption engine disabled
		1	Encryption engine enabled
3	reserved	0	
		1	
4	DIS_ TSEC1	0	Disabled
		1	Enabled
5	DIS_ TSEC2	0	Disabled
		1	Enabled

Table 4-17:Device Disable Register (Continued)

BIT	CONTENT	STATE	DESCRIPTION
6	DIS_ USBMPH	0	Disabled
		1	Enabled
7	DIS_ USBDR	0	Disabled
		1	Enabled

4.2.17 Delay Timer Control/Status Register

The delay timer provides the capability to realize short, reliable delay times. The delay timer control/status register provides three functions for operating the delay timer. The first function is to indicate the timing intervals available, the second function is to trigger/reset the timer, and the third is to indicate the time elapsed since the initial triggering or the last timer reset.

Writing a 0x00 and then reading the register will provide the timer intervals that are available for application usage. In the case of the EB8347, all of the possible timer intervals are available.

Writing anything other than 0x00 to the register sets the timer to zero and restarts it, and sets all of this register's bits to 0. As time elapses, interval bits are set accordingly and remain set until the timer is retriggered.

For example, the timer is started, after 1 µs bit 0 is set to 1, after 5 µs bit 1 is set 1. This process continues until all bits are set or the timer is retriggered. Once a bit is set it remains set until the timer is again retriggered.

Table 4-18:Delay Timer Control/Status Register

REGISTER NAME	DELAY TIMER CONTROL/STATUS REGISTER						ACCESS	
ADDRESS	0x4D80 0013						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	DTC7	DTC6	DTC5	DTC4	DTC3	DTC2	DTC1	DTC0
DEFAULT	0	0	0	0	0	0	0	0

BIT	CONTENT	STATE	DESCRIPTION
0:7	DTC[0:7]	0 1	The hardware delay timer is operated via one simple 8-bit control/status register. The following table indicates 1) the possible timing intervals provided, and 2) when read, the time elapsed since the last trigger/reset of the timer. **DTC[0:7] Value Accuracy** Bit 0: 1 µs < + 40% Bit 1: 5 µs < + 8% Bit 2: 10 µs < + 4% Bit 3: 50 µs < + 0.8% Bit 4: 100 µs < + 0.4% Bit 5: 250 µs < + 0.16% Bit 6: 0.5 ms < + 0.08% Bit 7: 1 ms < + 0.04%

4.2.18 GPIO Configuration Register 1

This register is used to configure IO signals either as general purpose digital IO or as local bus address lines.

Table 4-19:GPIO Configuration Register 1

REGISTER NAME		GPIO CONFIGURATION REGISTER 1						ACCESS	
ADDRESS		0x4D80 0014						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		GPIO_CFG7	GPIO_CFG6	GPIO_CFG5	GPIO_CFG4	GPIO_CFG3	GPIO_CFG2	GPI1_CFG7	GPIO_CFG0
DEFAULT		0	0	0	0	0	0	0	0
BIT	**CONTENT**	**STATE**	**DESCRIPTION**						
0	GPIO_CFG0	0	GPIO0						
		1	Local bus address line: XA0						
1	GPIO_CFG1	0	GPIO1						
		1	Local bus address line: XA1						
2	GPIO_CFG2	0	GPIO2						
		1	Local bus address line: XA2						
3	GPIO_CFG3	0	GPIO3						
		1	Local bus address line: XA3						
4	GPIO_CFG4	0	GPIO4						
		1	Local bus address line: XA4						
5	GPIO_CFG5	0	GPIO5						
		1	Local bus address line: XA5						
6	GPIO_CFG6	0	GPIO6						
		1	Local bus address line: XA6						
7	GPIO_CFG7	0	GPIO7						
		1	Local bus address line: XA7						

4.2.19 GPIO Configuration Register 2

The first four bits of this register are used to configure the IO signals either as general purpose digital IO or as local bus address lines. The last three bits define the usage of the three highest IO nibbles on the Communications Interface Extension (CIE) connector

Table 4-20:GPIO Configuration Register 2

REGISTER NAME	GPIO CONFIGURATION REGISTER 2						ACCESS	
ADDRESS	0x4D80 0015						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	GPIO_ NIB23-20	GPIO_ NIB19-16	GPIO_ NIB15-12	res.	GPIO_ CFG11	GPIO_ CFG10	GPI1_ CFG9	GPIO_ CFG8
DEFAULT	0	0	0	n/a	0	0	0	0

BIT	CONTENT	STATE	DESCRIPTION
0	GPIO_CFG8	0	GPIO8
		1	Local bus address line: XA8
1	GPIO_CFG9	0	GPIO9
		1	Local bus address line: XA9
2	GPIO_ CFG10	0	GPIO10
		1	Local bus address line: XA10
3	GPIO_ CFG11	0	GPIO11
		1	Local bus address line: XA11
4	Reserved	0	
		1	
5	GPIO_ NIB15-12	0	GPIO[15:12] configured as GPIO (source MPC8347)
		1	XA[15:12] configured as local bus address lines
6	GPIO_ NIB19-16	0	GPIO[19:16] configured as GPIO (source MPC8347)
		1	XA[19:16] configured as local bus address lines
7	GPIO_ NIB23-20	0	GPIO[23:20] configured as GPIO (source MPC8347)
		1	XA[23:20] configured as local bus address lines

4.2.20 GPIO Direction Register 1

This register is used to configure individual GPIO signals either as an input or as an output

Table 4-21:GPIO Direction Register 1

REGISTER NAME	GPIO DIRECTION REGISTER 1						ACCESS	
ADDRESS	0x4D80 0016						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	GPIO_ DIR7	GPIO_ DIR6	GPIO_ DIR5	GPIO_ DIR4	GPIO_ DIR3	GPIO_ DIR2	GPI1_ DIR7	GPIO_ DIR0
DEFAULT	0	0	0	0	0	0	0	0

BIT	CONTENT	STATE	DESCRIPTION
0	GPIO_DIR0	0	Input signal: GPIO0
		1	Output signal: GPIO0
1	GPIO_DIR1	0	Input signal: GPIO1
		1	Output signal: GPIO1
2	GPIO_DIR2	0	Input signal: GPIO2
		1	Output signal: GPIO2
3	GPIO_DIR3	0	Input signal: GPIO3
		1	Output signal: GPIO3
4	GPIO_DIR4	0	Input signal: GPIO4
		1	Output signal: GPIO4
5	GPIO_DIR5	0	Input signal: GPIO5
		1	Output signal: GPIO5
6	GPIO_DIR6	0	Input signal: GPIO6
		1	Output signal: GPIO6
7	GPIO_DIR7	0	Input signal: GPIO7
		1	Output signal: GPIO7

4.2.21 GPIO Direction Register 2

This register is used to configure individual GPIO signals either as an input or as an output

Table 4-22:GPIO Direction Register 2

REGISTER NAME	GPIO DIRECTION REGISTER 2						ACCESS	
ADDRESS	0x4D80 0017						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	GPIO_ DIR11	GPIO_ DIR10	GPI1_ DIR9	GPIO_ DIR8
DEFAULT	n/a	n/a	n/a	n/a	0	0	0	0

BIT	CONTENT	STATE	DESCRIPTION
0	GPIO_DIR0	0	Input signal: GPIO8
		1	Output signal: GPIO8
1	GPIO_DIR1	0	Input signal: GPIO9
		1	Output signal: GPIO9
2	GPIO_DIR2	0	Input signal: GPIO10
		1	Output signal: GPIO10
3	GPIO_DIR3	0	Input signal: GPIO11
		1	Output signal: GPIO11
4	Reserved	0	
		1	
5	Reserved	0	
		1	
6	Reserved	0	
		1	
7	Reserved	0	
		1	

4.2.22 GPIO RxData Register 1

This register indicates the input status of the individual GPIO signals of the BPCC: GPIO[7:0].

Table 4-23:GPIO RxData Register 1

REGISTER NAME		GPIO RXDATA REGISTER 1						ACCESS	
ADDRESS		0x4D80 0018						R	
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		GPIO_ RxDat7	GPIO_ RxDat6	GPIO_ RxDat5	GPIO_ RxDat4	GPIO_ RxDat3	GPIO_ RxDat2	GPIO_ RxDat1	GPIO_ RxDat0
DEFAULT		n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BIT	CONTENT	STATE	DESCRIPTION						
0	GPIO_ RxDat0	0	Input signal low: GPIO0						
		1	Input signal high: GPIO0						
1	GPIO_ RxDat1	0	Input signal low: GPIO1						
		1	Input signal high: GPIO1						
2	GPIO_ RxDat2	0	Input signal low: GPIO2						
		1	Input signal high: GPIO2						
3	GPIO_ RxDat4	0	Input signal low: GPIO3						
		1	Input signal high: GPIO3						
4	GPIO_ RxDat4	0	Input signal low: GPIO4						
		1	Input signal high: GPIO4						
5	GPIO_ RxDat5	0	Input signal low: GPIO5						
		1	Input signal high: GPIO5						
6	GPIO_ RxDat6	0	Input signal low: GPIO6						
		1	Input signal high: GPIO6						
7	GPIO_ RxDat7	0	Input signal low: GPIO7						
		1	Input signal high: GPIO7						

4.2.23 GPIO RxData Register 2

This register indicates the input status of the individual GPIO signals of the BPCC: GPIO[11:8].

Table 4-24:GPIO RxData Register 2

REGISTER NAME	GPIO RXDATA REGISTER 2						ACCESS	
ADDRESS	0x4D80 0019						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	GPIO_ RxDat11	GPIO_ RxDat10	GPIO_ RxDat9	GPIO_ RxDat8
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

BIT	CONTENT	STATE	DESCRIPTION
0	GPIO_ RxDat8	0	Input signal low: GPIO8
		1	Input signal high: GPIO8
1	GPIO_ RxDat9	0	Input signal low: GPIO9
		1	Input signal high: GPIO9
2	GPIO_ RxDat10	0	Input signal low: GPIO10
		1	Input signal high: GPIO10
3	GPIO_ RxDat11	0	Input signal low: GPIO11
		1	Input signal high: GPIO11
4	Reserved	0	
		1	
5	Reserved	0	
		1	
6	Reserved	0	
		1	
7	Reserved	0	
		1	

4.2.24 GPIO TxData Register 1

This register is used to either set (write) the signal level of the individual GPIO signals of the BPCC: GPIO[7:0] or to obtain their output status via the read function.

Table 4-25:GPIO TxData Register 1

REGISTER NAME		GPIO TXDATA REGISTER 1						ACCESS	
ADDRESS		0x4D80 001A						R	W
BIT POSITION		MSB 7	6	5	4	3	2	1	0 LSB
CONTENT		GPIO_ TxDat7	GPIO_ TxDat6	GPIO_ TxDat5	GPIO_ TxDat4	GPIO_ TxDat3	GPIO_ TxDat2	GPIO_ TxDat1	GPIO_ TxDat0
DEFAULT		n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BIT	CONTENT	STATE	DESCRIPTION						
0	GPIO_ TxDat0	0	Output signal low: GPIO0						
		1	Output signal high: GPIO0						
1	GPIO_ TxDat1	0	Output signal low: GPIO1						
		1	Output signal high: GPIO1						
2	GPIO_ TxDat2	0	Output signal low: GPIO2						
		1	Output signal high: GPIO2						
3	GPIO_ TxDat4	0	Output signal low: GPIO3						
		1	Output signal high: GPIO3						
4	GPIO_ TxDat4	0	Output signal low: GPIO4						
		1	Output signal high: GPIO4						
5	GPIO_ TxDat5	0	Output signal low: GPIO5						
		1	Output signal high: GPIO5						
6	GPIO_ TxDat6	0	Output signal low: GPIO6						
		1	Output signal high: GPIO6						
7	GPIO_ TxDat7	0	Output signal low: GPIO7						
		1	Output signal high: GPIO7						

4.2.25 GPIO TxData Register 2

This register is used to either set (write) the signal level of the individual GPIO signals of the BPCC: GPIO[11:8] or to obtain their output status via the read function.

Table 4-26:GPIO TxData Register 2

REGISTER NAME	GPIO TXDATA REGISTER 2						ACCESS	
ADDRESS	0x4D80 001B						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	res.	res.	res.	res.	GPIO_ TxDat11	GPIO_ TxDat10	GPIO_ TxDat9	GPIO_ TxDat8
DEFAULT	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

BIT	CONTENT	STATE	DESCRIPTION
0	GPIO_ TxDat8	0	Output signal low: GPIO8
		1	Output signal high: GPIO8
1	GPIO_ TxDat9	0	Output signal low: GPIO9
		1	Output signal high: GPIO9
2	GPIO_ TxDat10	0	Output signal low: GPIO10
		1	Output signal high: GPIO10
3	GPIO_ TxDat11	0	Output signal low: GPIO11
		1	Output signal high: GPIO11
4	Reserved	0	
		1	
5	Reserved	0	
		1	
6	Reserved	0	
		1	
7	Reserved	0	
		1	

4.2.26 Device Interrupt Pending Register

The Device Interrupt Pending Register is used to identify the source of the pending interrupt request from the following onboard devices:

- Temperature sensor (TEMP)
- Alarm from the RTC

A logical ’1’ indicates that an interrupt has been asserted.

Table 4-27:Device Interrupt Pending Register

REGISTER NAME	DEVICE INTERRUPT PENDING						ACCESS	
ADDRESS	0x4D80 001C						R	
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	ALARM	TEMP	res.	res.	res.	res.	res.	res.
DEFAULT	0	0	n/a	n/a	n/a	n/a	n/a	n/a

4.2.27 Device Interrupt Mask Register

The Device Interrupt Mask Register is used to mask an interrupt request from the following onboard devices:

- Temperature sensor (TEMP)
- Alarm from the RTC

A logical ’1’ indicates that a mask has been enabled.

Table 4-28:Device Interrupt Mask Register

REGISTER NAME	DEVICE INTERRUPT MASK						ACCESS	
ADDRESS	0x4D80 001D						R	W
BIT POSITION	MSB 7	6	5	4	3	2	1	0 LSB
CONTENT	ALARM_EN	TEMP_EN	res.	res.	res.	res.	res.	res.
DEFAULT	0	0	n/a	n/a	n/a	n/a	n/a	n/a

4.3 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 M41T81 and the Motorola MPC8347(E) (I2C bus).

Table 4-29:Register Map RTC M41T81

ADR (HEX)	ADDRESS BITS								FUNCTION RANGE IN BCD FORMAT
	D7	D6	D5	D4	D3	D2	D1	D0	
00	0.1 Seconds				0.01 Seconds				Seconds: 00 - 99
01	ST	10 Seconds			Seconds				Seconds: 00 - 59
02	0	10 Minutes			Minutes				Minutes: 00 - 59
03	CEB	CB	10 Hours		Hours				Century: 0 - 1 Hours: 00 - 23
04	0	0	0	0	0	Day			Day: 00 - 07
05	0	0	10 Date		Date				Date: 01 - 31
06	0	0	0	10M.	Month				Month: 01 - 12
07	10 Years				Year				Year: 00 - 99
08	OUT	FT	S	Calibration					Control:
09	0	BMB4	BMB3	BMB2	BMB1	BMB0	RB1	RB0	Watchdog:
0A	AFE	SQWE	ABE	Al 10M	Alarm Month				Alarm Month: 01 - 12
0B	RPT4	RPT5	Al 10 Date		Alarm Date				Alarm Date: 01 - 31
0C	RPT3	HT	Al 10 Hour		Alarm Hour				Al Hour: 00 - 23
0D	RPT2	Alarm 10 Minutes			Alarm Minutes				Al Min: 00 - 59
0E	RPT1	Alarm 10 Seconds			Alarm Seconds				Al Sec: 00 - 59
0F	WDF	AF	0	0	0	0	0	0	Flags:
10	0	0	0	0	0	0	0	0	Reserved:
11	0	0	0	0	0	0	0	0	Reserved:
12	0	0	0	0	0	0	0	0	Reserved:
13	RS3	RS2	RS1	RS0	0	0	0	0	SQW:

Legend for Table 64

0	=	Must set to ’0’
ABE	=	Alarm in battery back-up mode enable bit
AF	=	Alarm flag (read only)
AFE	=	Alarm flag enable flag
BMBn	=	Watchdog multiplier bit(s)
CB	=	Century bit
CEB	=	Century enable bit
FT	=	Frequency test bit
HT	=	Halt update bit
OUT	=	Output level
RBn	=	Watchdog resolution bit(s)
RPTn	=	Alarm repeat mode bit(s)
RSn	=	SQW frequency
S	=	Sign bit
SQWE	=	Square wave enable
ST	=	Stop bit
WDF	=	Watchdog flag (read only)

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.

4.4 IRQ Routing

The Embedded Programmable Interrupt Controller of the MPC8347(E) (CPU) supports 8 external IRQs. The IRQ routing is listed below.

Table 4-30:IRQ Routing

IRQ NAME	SOURCE
IRQ0	CompactFlash
IRQ1	MC2 or MC3 or MC5 IRQ, Alarm, Temperature, Watchdog Timer
IRQ2	PCI_INTA
IRQ3	PCI_INTB
IRQ4	PCI_INTC
IRQ5	PCI_INTD
IRQ6	SM501 Interrupt (two UARTs)
IRQ7	SerIRQ (LPC)

4.5 CompactFlash

Write access to the CompactFlash address area is only possible using word-wide (16-bit) write commands.

Table 4-31:CompactFlash Register

REGISTER	READ/WRITE	ADDRESS
Data Register	R/W	0x4200 0000
Error Register	R	0x4200 8003
Feature Register	W	0x4200 8003
Sector Count Register	R/W	0x4200 8005
Sector Number Register	R/W	0x4200 8007
Cylinder Low Register	R/W	0x4200 8009
Cylinder High Register	R/W	0x4200 800B
Drive/Head Register	R/W	0x4200 800D
Status Register	R	0x4200 800F
Device Control Register	W	0x4200 801D
Alternate Status Register	R	0x4200 801D
Card Drive Address Register	R	0x4200 801F

4.6 EEPROM

Access to the EEPROM is effected via the I2C bus controller of the MPC8347(E) and not the CPM I2C channel. The EEPROM uses the I2C address 0xA0. Write protection is achieved by installing a 0 ohm resistor (R5). Default is unprotected.

For more detailed information please refer to the manuals for the MICROCHIP 24LC64 or Catalyst 24WC64, and the Freescale MPC8347.

4.7 Digital Temperature Sensor, LM75

Access to the onboard digital temperature sensor (DTS) is effected via the I2C bus controller of the MPC8347(E). The DTS uses the I2C address 0x90.

For more detailed information please refer to the manuals for the National Semiconductor LM75 and the Freescale MPC8347(E).

Chapter 5

NetBootLoader

This page has been intentionally left blank.

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 EB8347 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 sections 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 ETH1. 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 ETH1, the port must be configured. This is done either via the TERM port or via a dhcp/bootp server.

5.2 NetBootLoader Interfaces

There are four possibilities to interface with the NetBootLoader:

- Via the MC1 (Abort) signal
- Via the TERM (terminal) serial interface
- Via the SER0 (console) serial interface
- Via the Ethernet port ETH1 interface

Gaining access to the NetBootLoader is a function of the contents of the Flash memory and the “BootWaitTime” setting. If there is not a 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 port ETH1. If the operator does not intervene, the boot operation is continued after the boot wait time has been exceeded.

5.2.1 MC1 (Abort) Signal

The MC1 (Abort) signal is routed to the EB8347 from the carrier board via the System Interface (J1 connector) and, if made available from 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 normally deactivated and the boot process is continued. If the NetBootLoader does not find a valid boot image, the boot process is discontinued and system control is returned to the operator via 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 Port ETH1 Interface

The Ethernet port ETH1 provides the capability of remotely interfacing with the NetBootLoader. Prior to using this interface it is necessary to configure the Ethernet port settings. This is accomplished via the TERM interface or a dhcp/bootp server. Once the port settings have been configured, 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 performing a telnet login it via the Ethernet port ETH1. This causes the boot operation to be terminated and gives control to the remote operator.

In addition to the operator interface via this interface, the NetBootLoader can also use it for tftp and ftp server accessing.

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
- network accessing
- 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

5.3.1 NetBootLoader Control

The NetBootLoader provides various functions for controlling the operation of the NetBootLoader 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	
CBL	Change Bootline	Set or display a bootline	Applies to a specific kernel
DHCP	-	Dynamically set Ethernet port ETH1 parameters	Requires that a dhcp or bootp server be available in the same network as the EB8347
HELP or ?	-	Display online HELP pages	
LOGOUT	-	Terminate telnet session	
NET	-	Manually set Ethernet port ETH1 parameters	Must be set before attempting to use the Ethernet port; see also the DHCP command
PASSWD	Password	Set telnet password	Must be set before attempting telnet login
PF	Port Format	Set serial port parameters	Used for the TERM and SER0 ports
RS	Reset	Resets system	
SCRIPT	-	Command scripting	Contents are executed only during boot up
SQ	Boot Sequence	Set or display boot sequence	Defines selection order of image booting

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 available system status monitoring functions.

Table 5-2: System Status Monitoring Commands

CMD TITLE	ALIAS	FUNCTION	REMARKS
CHECK	-	Application validation; displays information for each image	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 NetBoot-Loader	

5.3.3 Network Accessing

To support application development and operational requirements for various boot strategies, the NetBootLoader provides several functions for gaining access to network services. These functions include: access to dhcp/bootp servers, accessing tftp servers, and accessing ftp servers.

At initial startup of the EB8347, only the NetBootLoader is installed in onboard FLASH. To support application development or remote boot capability, the NetBootLoader can provide networking interfacing via the Ethernet port ETH1.

To achieve this, certain network parameters must first be configured. This can be done manually via a terminal or dynamically via the network. The command DHCP makes it possible to download such parameters and to configure the Ethernet port ETH1 for network operation. Once the Ethernet port is configured, the commands TFTP and FTP are available to download bootable images or other files as required.

5.3.3.1 dhcp/bootp Server Access

Use of this access method requires the availability of either a dhcp or bootp server in the same network as the EB8347. The DHCP command causes the NetBootLoader to first attempt to establish contact with a DHCP server. If contact is not achieved, it then tries to contact a BootP server. When contact is established, parameters required by the EB8347 are provided accordingly and the Ethernet port is configured and then made available for normal operation.

In the event the EB8347 is reset or cold started the configuration parameters set by the above method are lost. Only if the parameters have been set by the NET command are they still available.

Prior to using the "dhcp" command, the IP address of the EB8347 must be set to 255.255.255.255 using the "net" command.

5.3.3.2 tftp/ftp Server Access

The NetBootLoader provides various functions for interfacing with either a tftp or ftp server.

The tftp server access is a simple method of acquiring a userimage from a remote source. Its primary use is to download a single executable userimage from a given source. For example, once an application has been programmed it would be possible to store it at a remote location where it then would be available for remote booting of an EB8347 via the Ethernet port ETH1.

The ftp server commands provide various functions consistent with interfacing with such a server.

The following table provides an overview of available tftp/ftp server functions.

Table 5-3: tftp/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
TFTP	-	Download a file from tftp server	

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

5.4 Operating the NetBootLoader

5.4.1 Initial Setup

The EB8347 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 the Ethernet port ETH1 interfaces. If there is not a 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 port parameters must be set and this can only be accomplished initially via the TERM interface.

The operator must either manually set the parameters using the "net" command or dynamically via the "dhcp" command. Prior to using the "dhcp" command, the IP address of the EB8347 must be set to 255.255.255.255 using the "net" command.

Once valid Ethernet port 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 the Ethernet port ETH1, 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.

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)
- CBL (Change Bootline)
- DHCP
- NET
- PASSWD
- PF (Port Format)
- SCRIPT
- SQ (Boot Sequence)

Default settings are available for all the above commands except for "dhcp, net, and script”.

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 CBL

This command is used to set or display the bootline associated with a particular kernel image or which is common to all images.

5.4.3.3 DHCP

This command is used to obtain automatically networking parameters from either a dhcp or bootp server for the Ethernet port ETH1. Its use requires the availability of one or the other of these servers to function.

5.4.3.4 NET

This command is used to set or display the parameters for the configuration of the Fast Ethernet interface of the CPU board. The Fast 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.

5.4.3.5 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.

5.4.3.6 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.

5.4.3.7 SCRIPT

This command permits the automatic invoking of NetBootLoader commands during boot up. The operator issues this command with appropriate options and then restarts the system. During the boot operation at boot wait time expiration, the "script" commands will be executed.

Use of this command permits, for example, remote booting from an tftp server.

5.4.3.8 SQ

This command is used to set or display the order in which application images are to be booted.

The NetBootLoader provides the capability to program up to four application images into the FLASH. With this command the operator can define the order in which images may be used when the system is booted. This provides operational flexibility as well as the possibility for the system to compensate for a defective image.

For example, in the event the first image specified is defect, the NetBootLoader will attempt to load the next image specified. This is continued until either a valid image is loaded or no further image is available.

If no valid image is found, the NetBootLoader invokes its command interpreter and remains available for inputs.

5.4.4 telnet Login

A telnet login to the NetBootLoader is only possible during the boot wait time and only after the Ethernet port ETH1 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 may be programmed to FLASH. As mentioned before, the NetBootLoader supports the programming of up to four application images to FLASH whereby each image is assigned its own image number. 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.

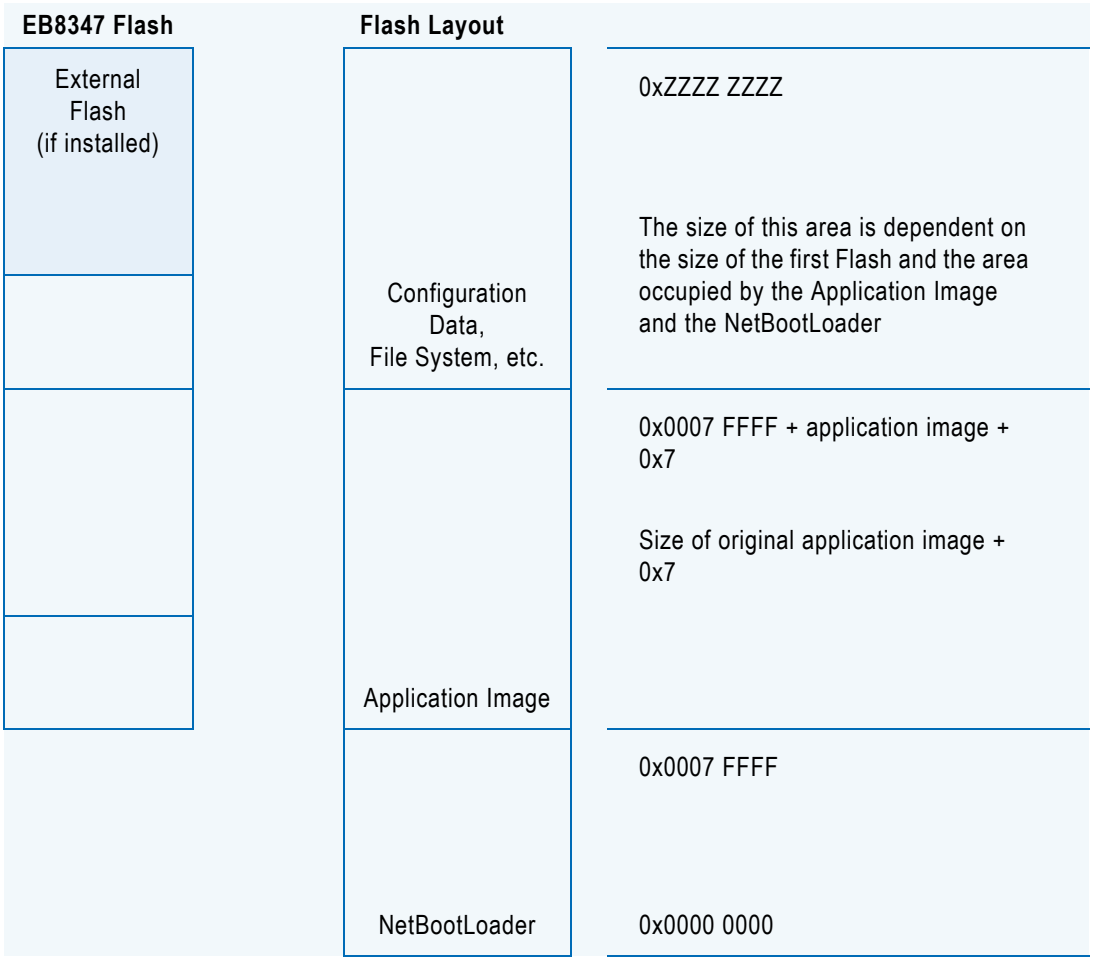
5.4.5.1 FLASH Offsets

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 first bank of the FLASH area and for this reason this area cannot be used for application image programming. Figure 12 on the next page illustrates this concept. 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.

Figure 2-1: Flash Addressing Scheme - EB8347



To obtain exact information concerning the current Flash usage, the INFO command is provided by the NetBootLoader. Please refer to this command.

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 entry point of the executable PPC code must be at offset 0x100.

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

- tftp server
- 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 a tftp server, 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 tftp server, the “tftp” command is used. 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 Accessing tftp and ftp Servers

To gain access to an application image file stored on a tftp or ftp server the Ethernet port ETH1 is used. Images are downloaded to the data buffer using the ftp protocol. To use this interface the Ethernet port parameters must first be set and the operator must have control of the NetBootLoader.

To download an application image from a tftp server, the command "tftp" is used. The tftp server IP and file name of the application must be known and provided to the "tftp" command or be provided by the dhcp server via the "dhcp" command.

To perform a download from an ftp server, the operator must first login to the ftp server. 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 "tftp", “get”, and “ls” use the same data buffer. Therefore if an “ls” command is issued after a "tftp" or “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.

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.

5.4.7 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.

5.4.8 Uploading a FLASH Area

The NetBootLoader also has the possibility to upload certain areas of the FLASH to a host using the Ethernet port ETH1. To use this interface this Ethernet port parameters must first be set and then 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.

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.

Downloaded images are never executed from the FLASH. 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.

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 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 0, 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**
	Set boot wait time to 0 seconds COMMAND / RESPONSE (none): **bw 0** **Choosing a waittime of 0s will make a network login impossible**

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”.

CBL

FUNCTION:	Set or display the parameters of the bootline function
SYNTAX:	**`cbl <num> [<bootline>]`** where: cbl — command <num> — parameter: value: string “0, 1, 2, 3, c” ID number of the image to be associated with the bootline or bootline which is common to all images <bootline> — parameter: value: string (max. of 256 characters) defines the bootline to be used with the kernel indicated by <num> or the common bootline
DESCRIPTION:	When an application image is programmed to FLASH, it is assigned an ID number (0, 1, 2, or 3). This number is used to identify which image is to be addressed by the command “cbl”. In addition, a bootline common to all images may also be defined using the “c” parameter. If the command “cbl” is invoked with only the <num> parameter, it returns the bootline for the image specified or the common bootline. Invoking the command “cbl” with the <bootline> parameter overwrites any previous bootline for the image specified.
USAGE:	Display the bootline for image 2 COMMAND / RESPONSE: **`cbl 2`** `<contents of the bootline of image 2>`

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 provides status information for the current application images in FLASH.
USAGE:	Verify valid application is stored in FLASH COMMAND / RESPONSE: **check** **Checking Image: 0** **check image crc: ok** **length in flash: 0x0053d004** **sectors used : 84**

CHECK

USAGE:

```
Checking Image: 1
  check image crc: ok
  length in flash: 0x0001029c
  sectors used   :     2

Checking Image: 2
  check image crc: fail
  length in flash: -
  sectors used   : -

Checking Image: 3
  check image crc: fail
  length in flash: -
  sectors used   : -
```

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.

CLONE

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.

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>
```

DHCP

FUNCTION:	Interface to a dhcp or bootp server; exchange network configuration parameters
SYNTAX:	**dhcp [<timeout>]** where: dhcp command <timeout> parameter: value: numerical string set timeout, in seconds
DESCRIPTION:	This command is used to set the network parameters for operation of the Ethernet port ETH1 via either a dhcp or bootp server. Initially the CPU board does not have a valid Ethernet interface configuration, and, therefore, this interface is inoperable. The initial configuration must be done either manually from the TERM interface using the command "net", or, if a dhcp or bootp server is available, it can be done automatically by the "dhcp" command. Manually configured parameters are permanently stored. Parameters configured using the "dhcp" command are temporary and will be lost if the system is reset or cold started. Prior to using the "dhcp" command, the IP address must be set to 255.255.255.255 with the "net" command.
USAGE:	Program NetBootLoader (normal operation) COMMAND / RESPONSE: **NetBtLd> dhcp** **Sending request... reply from BOOTP/DHCP server.** **Network eth0 initializated ok.** **Server address is 192.168.112.2, our IP address is 192.168.112.14.** **Filename :** **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.

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. Displayed are the following: CPU type, the board type, the size of the installed RAM and FLASH, and the FLASH areas occupied by the NetBootLoader and the operating system images. This information is displayed in hexadecimal offsets. Images programmed using the “-o” option of the command “lf” are not shown.
USAGE:	Display system information COMMAND / RESPONSE: **info** **CPU : PowerPC, MPC8347** **Board : EB8347** **Ram : 8000000** **Flash :** **Name : AMD 29LV128** **Bank : 0** **Bytelane : 0** **BankPortsize : 16** **ChipPortsize : 16** **Offset : 0x0** **Size : 0x1000000** **NetBootLoader used FLASH: 0x0 - 0x80000** **Sector usage map:** **0x0000: nnnnnnnn 00000000** **0x0010: 00000000 00000000** **0x0020: 00000000 00000000** **0x0030: 00000000 00000000** **0x0040: 00000000 00000000** **0x0050: 00000000 0011....** **0x0060: ....** **0x0070: ....** Where: n = NetBootLoader; 0 = image 0; 1 = image 1; . = usage unknown

LF

FUNCTION: Load Flash

SYNTAX:

```
lf [<num> ][-o[=]<offset> [-k ]]
… [-m[=]<adr> -l[=]<len>]
```

where:

lf	command
<num>	parameter: value: numeric “0, 1, 2, 3”
	ID number assigned to this image
-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:

If <num> is not specified, 0 is assumed.

Without options, 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 added along with the 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” options are specified, the image is programmed from the absolute address specified.

If the “-o” option 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.

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 (Image ID = 0 is assumed) COMMAND / RESPONSE (none): **`lf`**
	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`**

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.

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.

NET

FUNCTION:

Set or display the parameters for the Fast 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 Fast Ethernet interface the command “net” is used.

Initially the CPU board does not have a valid Fast 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 Fast 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 Fast Ethernet interface does not properly initialize, may it be necessary to re-enter the parameters using the “-f” option.

The <ip-addr> parameter must be set to 255.255.255.255 if the "dhcp" command is to be used for the configuration of the Ethernet port.

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 Fast 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 -f** **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 where: term = SP4 and ser0 = SP6 <baud> — parameter: value: numeric: “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

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 SP4 and the SP6 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 SP4 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 SP6 to 7 COMMAND / RESPONSE (none): **`pf ser0 //7`**
	Set the stop bits parameter of SP6 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
```

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`**
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.

SCRIPT

FUNCTION:	Provides very basic scripting capability
SYNTAX:	**`script [<newscript>]`** where: script — command <newscript> — parameter: string string may only include simple commands; flow control constructs are not permitted; commands must be separated by semi-colons
DESCRIPTION:	With the "script" command, it is possible to control the boot process. During booting, if a valid script is available, the NetBootLoader will process it once the boot wait time is expired. If this command is issued without any parameters, the currently active script contents are displayed.
USAGE:	Download a boot image from a tftp server and run the boot image. COMMAND / RESPONSE (none): **`script dhcp; tftp; run`** Upon the next reset or cold start, after the boot wait time has expired the commands "dhcp", "tftp", and "run" will be executed in that order. This command sequence configures the Ethernet port ETH1, downloads the specified bootable image from an tftp server, and then starts this image.

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: source = SER0 instead of TERM
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 otherwise the TERM interface is used.
USAGE:	Download S-Records to data buffer and reduce each record’s address by 0x10000. COMMAND / RESPONSE (none): **`sl -o=10000`**

SQ

FUNCTION:	Set or display the boot sequence
SYNTAX:	**sq [<num1> <num2> <num3> <num4>]** where: sq command <num1> parameter: value: numeric: “0, 1, 2, 3” ID number of image to be booted <num2> parameter: value: numeric: “0, 1, 2, 3” ID number of image to be booted <num3> parameter: value: numeric: “0, 1, 2, 3” ID number of image to be booted <num4> parameter: value: numeric: “0, 1, 2, 3” ID number of image to be booted
DESCRIPTION:	Up to four bootable images may be programmed into the boot FLASH. The boot sequence defines to the NetBootLoader the order in which images are to be accessed when booting. The NetBootLoader starts with num1 and continues until a valid image is found. In the case no valid image is found, the NetBootLoader stops searching and waits for operator intervention. All four number parameters must be defined even if there is not an image in the FLASH with that ID number. Any given ID number may only be used once: e.g. a sequence of 0120 is not permitted. The default sequence is 0123 if the boot sequence has not been programmed.
USAGE:	Display the current boot sequence setting. COMMAND / RESPONSE: **sq <cr>** **0312**
	Set the boot sequence to 3201. COMMAND / RESPONSE(none): **sq 3201**

TFTP

FUNCTION:	Download file from a tftp server
SYNTAX:	**`tftp [<serverip>] [<filename>]`** where: tftp — command <serverip> — parameter: value: numerical string IP address of the tftp server <filename> — parameter: value: string name of image file to be downloaded, or path and name of image file to be downloaded
DESCRIPTION:	The "tftp" command makes it possible to download a file from a tftp server via the Ethernet channel ETH1. If used with the "dhcp" command, it is possible to use the IP address and file information returned by the "dhcp" command. If this command is issued without any parameters, it will use the previously stored information returned with the "dhcp" command.
USAGE:	Download a file from a tftp server. COMMAND / RESPONSE (none): **`tftp 195.178.125.55 image2`** This command downloads the file "image2" from the specified tftp server.

VER

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

This page has been intentionally left blank.

Chapter 6

System Considerations

This page has been intentionally left blank.

6. System Considerations

Successful implementation of the EB8347 E²Brain™ module in any given application is a function of a wide variety of factors. To assist system integrators in achieving optimum solutions, additional technical information concerning:

- Thermal design
- Module stacking heights
- Power requirements

is provided by this section.

6.1 Thermal Design Considerations

The EB8347 is designed to be operated in a wide range of thermal environments. Basic thermal management mechanisms have been incorporated, and, depending on the application requirements, there are configurations available which should satisfy most situations. Still it is necessary for system integrators to be aware of certain concepts and capabilities of the EB8347 when projecting overall system thermal management. The following provides more detailed information concerning aspects that must be considered before the EB8347 is integrated in an application system.

6.1.1 Thermodynamics Terminology

In order to facilitate understanding the factors involved in the thermal considerations being discussed here, the following definitions provided.

Table 6-1: Thermodynamics Terminology as Relating to the EB8347

TERM	DEFINITION
Heat	The thermal energy of a body. The primary source of heat in EB8347 is the CPU which requires cooling.
Cooling	Reduction of the temperature of a body. The transfer of heat from a body. Cooling is a somewhat relative term when applied to systems which generate heat due to the continuous application of external energy. In the case of the EB8347 continuous operation without “cooling” would lead to the ultimate destruction of the CPU and other components. With cooling, however, board components are allowed to operate within a range of higher temperatures than would otherwise be possible. Cooling in this case means maintaining a working temperature that does not exceed the maximum level.
Conduction	The cooling of a body by means of direct contact with another body whose temperature is lower. The EB8347 uses conduction to transfer heat from the CPU to a heat sink. The heat sink is the primary thermal interface to the EB8347 and is an integrated part of the EB8347. Secondary cooling of the EB8347 may be achieved either by conduction or convection.

Table 6-1: Thermodynamics Terminology as Relating to the EB8347

TERM	DEFINITION
Convection (Free and Forced)	The cooling of a body by means of the use of a fluid (liquid or gas) whose temperature is lower. Convectional cooling always requires that some fluid is moved over a body and that the fluid temperature is lower than that of the body to be cooled. Free convection relies on the buoyancy of the fluid caused by the transfer of heat from the body to be cooled to the fluid. This results in a very minimal amount of fluid flow which, depending on the temperature of the fluid and the amount of fluid available, may be sufficient to cool the body. In the case of the EB8347, the fluid would be ambient air. Forced convection relies on the forceful movement of a fluid across a body caused by the application of external energy. Again in the case of the EB8347, the fluid would be ambient air which is forced over the heat sink of the EB8347.
Radiation	The transfer of heat from a body by means of thermal radiation. The EB8347 is not designed for radiation cooling.
Ambient Air	The air occupying the immediate space surrounding a body to be cooled. For cooling to take place using ambient air, the ambient air must pass over the body to be cooled and the ambient air temperature must be lower than that of the body to be cooled.
Ambient Air Temperature	The temperature of the ambient air surrounding the body to be cooled. When applied to convectional cooling, it is the temperature of the ambient air directly prior to its flowing over the body to be cooled.
Air Flow	Movement of ambient air across a body to be cooled. The flow of ambient air is critical to convectional thermal management in particular if the ambient air is recycled (reflowed over the body to be cooled). Continuous recycling without cooling of the ambient air will quickly lead to overheating. In addition, the cooling effect of air flow in itself is limited. It requires a minimum velocity in order to have any cooling effect at all, and above a given velocity no further increase in real cooling effect is achieved.
Heat Spreader	A mechanism to achieve a rapid transfer of heat away from one body to another.
Heat Sink	A mechanism to achieve a (rapid) transfer of heat away from a body and to act as a transport medium to another body or fluid.
Heat Pipe	A special form of heat spreader which employes both conduction and convection in a discrete body to achieve a rapid transfer of heat from one body to another or to a fluid.

6.1.2 System Environment

The EB8347 system environment plays the major role in determining the thermal management concept to be applied to the EB8347. Generally speaking, there are two basic types of system environments: open and closed.

Open environments use ambient air convectional cooling as their main instrument of system thermal management. Closed environments rely heavily on conductional cooling as their first line of system thermal management. Closed systems may also be of a hybrid nature where convectional cooling is used on top of conductional cooling.

The initial means of cooling the EB8347 is conductional. There are two basic configurations available. The standard configuration uses a flat surface heat sink which also functions as a heat spreader. This configuration may either be used for conductional or convectional cooling. The second configuration which employs a special ribbed heat sink (heat spreader) is designed for convectional cooling only.

From the above it can be readily seen that the EB8347 supports both open and closed system environments. The key point of issue regardless of the type of environment, however, is the requirement of the EB8347 for additional cooling. In particular, if convectional cooling is planned it is imperative that a minimum air flow be ensured. The minimum air flow requirement is dependent on the configuration employed and the temperature of the ambient air.

If conductional cooling is to be implemented, a comprehensive empirical thermal design analysis and verification must be performed prior to implementation to ensure safe and proper operation of the EB8347.

6.2 Module Stacking Heights

The EB8347 is standardly equipped with four Hirose header connectors of the type: FX8C-nnnP-SV. The corresponding receptacle connector for the carrier board is type: FX8C-nnnS-SVn.

This combination of connectors makes it possible to achieve a wide variety of module stacking heights. The following table lists the available connectors and the resulting stacking heights.

Table 6-2: Module Stacking Heights

	RECEPTACLE (CARRIER)	
HEADER (EB8347)	FX8C-nnnS-SV	FX8C-nnnS-SV5
FX8C-nnnP-SV	5 mm	10 mm
FX8C-nnnP-SV1	6 mm	11 mm
FX8C-nnnP-SV2	7 mm	12 mm
FX8C-nnnP-SV4	9 mm	14 mm
FX8C-nnnP-SV6	11 mm	16 mm

For stacking heights other than the standard 5 mm or 10 mm, the stacking height must be specified when ordering the EB8347.

6.3 Power Considerations

6.3.1 System Power

Increasing the performance of a CPU in addition to higher integration results in higher power consumption. This results in special requirements for the power supply and the E²Brain™ carrier. The considerations presented in the ensuing sections must be taken into account by system integrators when specifying the EB8347 system environment.

6.3.2 EB8347 Voltage Ranges

The EB8347 board itself has been designed for optimal power input and distribution. Still it is necessary to observe certain criteria essential for application stability and reliability.

The table below indicates the absolute maximum input voltage rating that must not be exceeded. Power supplies to be used with the EB8347 should be carefully tested to ensure compliance with these ratings.

Table 6-3: Absolute Maximum Rating

SUPPLY VOLTAGE	ABSOLUTE MAXIMUM RATINGS
+3.3 V	+3.6 V

WARNING!

The maximum permitted voltage indicated in the table above must not be exceeded. Failure to comply with the above may result in damage to the board.

The following table specifies the range for the input power voltage within which the board is functional. The EB8347 is not guaranteed to function if the board is not operated within the prescribed limits.

Table 6-4: DC Operational Input Voltage Range

INPUT SUPPLY VOLTAGE	ABSOLUTE RANGE	RECOMMENDED RANGE	REMARKS
+3.3 V	3.2 V min. to 3.47 V max.	3.3 V min. to 3.47 V max.	Main voltage

6.3.3 E²Brain™ Carrier Requirements

Carriers to be used with the EB8347 must be adequately specified. The carrier must provide optimal power distribution for the +3.3 V power input. It is recommended to use only carriers which have two power planes for the 3.3 V and GND.

Input power connections to the carrier itself should be carefully specified to ensure a minimum of power loss and to guarantee operational stability. Long input lines, under dimensioned cabling or bridges, high resistance connections, etc. must be avoided.

6.3.4 Power Supply Units

Power supplies for the EB8347 must be specified with enough reserve for the remaining system consumption. In order to guarantee stable system functioning, it is recommended to provide more power than the system requires. An industrial power supply unit should be able to provide at least twice as much power as the entire system requires.

As the design of the EB8347 has been optimized for minimal power consumption, the power supply unit should be stable even under no load conditions.

Where possible, power supplies which support voltage sensing should be used. Depending on the system configuration this may require an appropriate carrier. The power supply should be sufficient to allow for die resistance variations.

Note ...

Some PSUs require a greater minimum load than a single EB8347 is capable of creating. When a PSU of this type is used, it will not power up correctly and the EB8347 may hang-up. The solution is to use an industrial PSU or to add more load to the system.

If DC/DC power supplies are used, please ensure that the external main supply provides sufficient power in order to start-up the system properly. The external main supply should provide at least as much power as the system power supply is able to provide taking into consideration the inrush current.

WARNING!

An under dimensioned power supply may cause damage to system components.

6.3.4.1 Voltage Ramp

Power supplies must comply with the following guidelines, in order to be used with the EB8347.

- Beginning at 10% of the nominal output voltage, the voltage must rise within > 0.1 ms to < 20 ms to the specified regulation range of the voltage. Typically: > 5 ms to < 15 ms.
- There must be a smooth and continuous ramp of each DC output voltage from 10% to 90% of the regulation band.
- The slope of the turn-on waveform shall be a positive, almost linear voltage increase and have a value from 0 V to nominal Vout.

6.3.4.2 Recommended Operating Conditions

The output voltage overshoot generated during the application (load changes) or during the removal of the input voltage must be less than 5% of the nominal value. No voltage of reverse polarity may be present on any output during turn-on or turn-off.

The following table provides information regarding the required characteristics for the input voltage.

Table 6-5: Input Voltage Characteristics

VOLTAGE	NOMINAL VALUE	TOLERANCE	MAX. RIPPLE (p-p)
3.3 V	+3.3 VDC	+5%/-3%	50 mV
GND	Ground, not directly connected to protective earth (PE)		

6.3.4.3 Supply Voltage Regulation

The power supply shall be unconditionally stable under line, load, unload and transient load conditions including capacitive loads. The operation of the power supply must be consistent even without the minimum load on all output lines.

Note ...

If the main power input is switched off, the supply voltages will not go to 0V instantly. It will take a couple of seconds until capacitors are discharged. If the voltage rises again before it has gone below a certain level, the circuits may enter a latch-up state where even a hard RESET will not help any more. The system must be switched off for at least 3 seconds before it may be switched on again. If problems still occur, turn off the main power for 30 seconds before turning it on again.

6.3.5 Power Consumption of the EB8347

The goal of this description is to provide a method to calculate the power consumption for the EB8347 board and for additional configurations. The MPC8347(E) processor and the DDR SDRAM dissipate the majority of the thermal power.

The power consumption tables below list the voltage and power specifications for the EB8347. The software used for this test was the ECOS based NetBootLoader without any power management features being enabled during the measurement. All measurements were conducted at a temperature of 25°C.

The following table provides power consumption information based on one memory size.

Table 6-6: Power Consumption with 256 MB Memory

CORE FREQ (MHZ)	MEMORY FREQ (MHZ)	WATTS
528	264	5.0
660	264	<tbd>

Note ...

If less power consumption is required, it is possible to disable features of the CPU such as TSEC controllers, PCI, Security Engine, etc. which are not being used. For further assistance, contact technical Support at Kontron Modular Computers.

6.4 Start-Up Current of the EB8347

During the startup process, clamping and leakage effects result in a much higher “inrush current” than the current required for normal operation. For this reason power supplies used with the EB8347 must be able to provide 10 A during the first 10 ms of operation.

Chapter 7

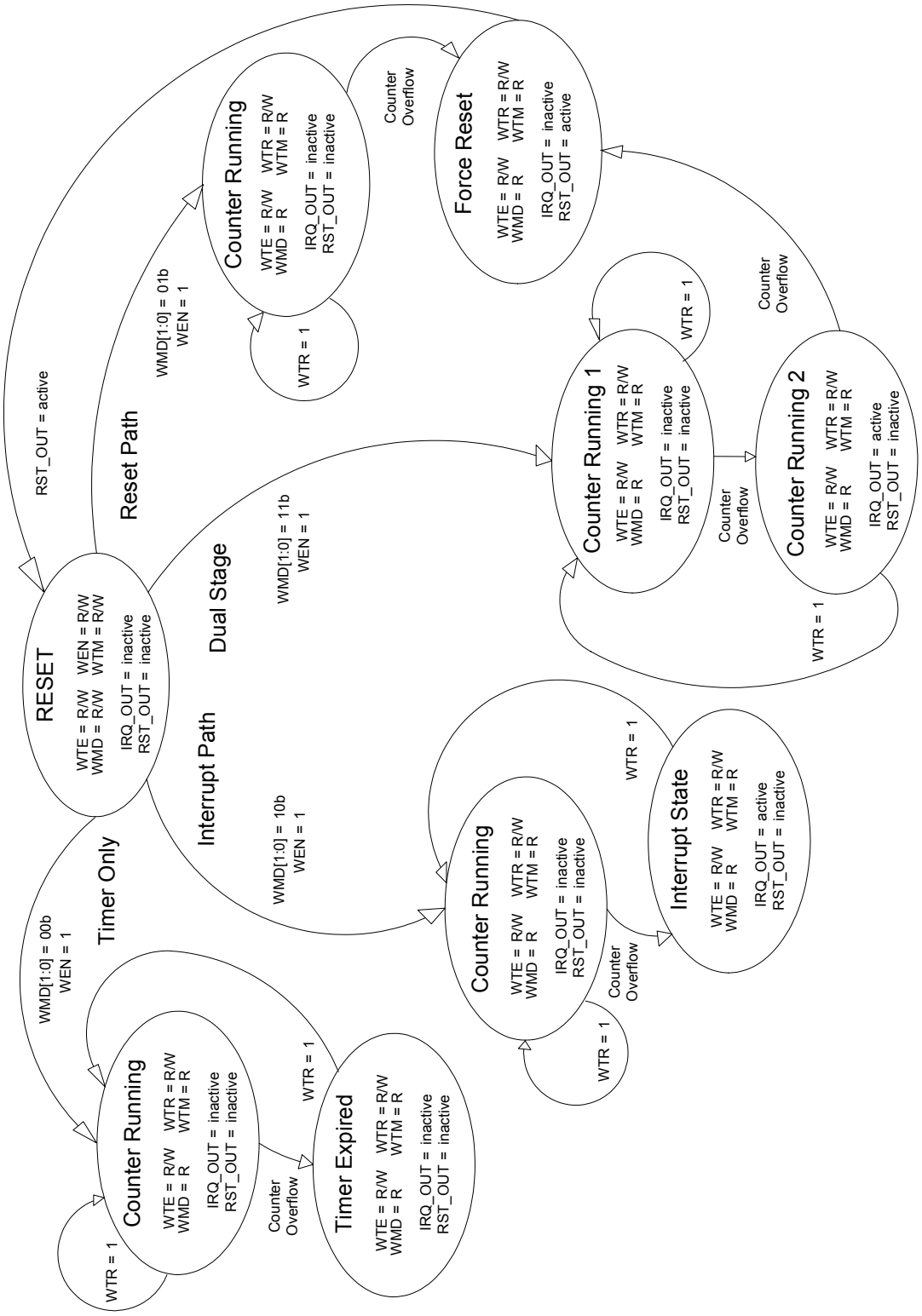
Watchdog Functionality

This page has been intentionally left blank.

7. Watchdog Functionality

To assist in understanding the functionality of the EB8347 Watchdog (WDOG), the following functional state diagram is provided.

Figure 7-1: WDOG Functional States



This page has been intentionally left blank.

Chapter 8

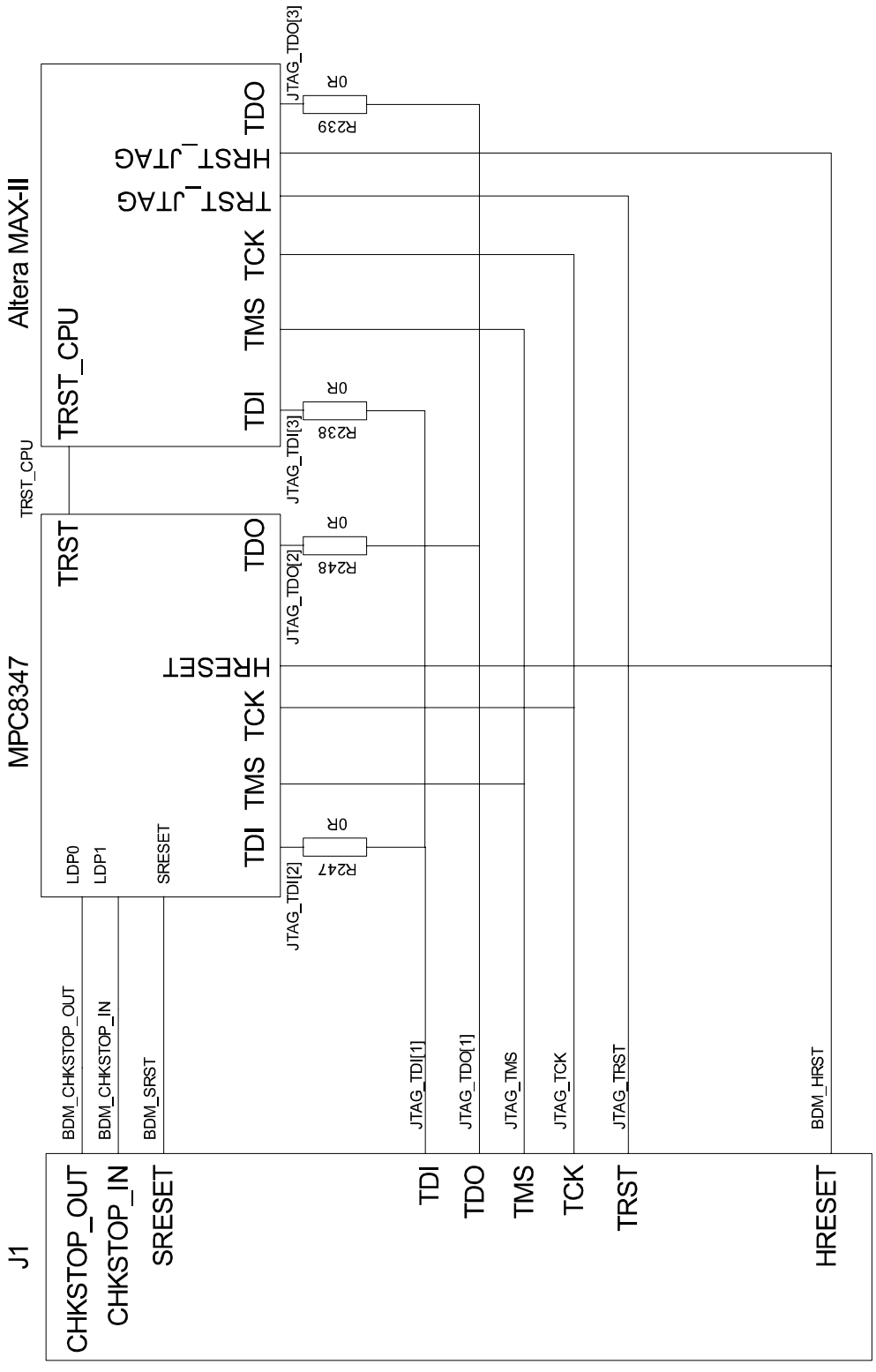
JTAG Chain

This page has been intentionally left blank.

8. JTAG Chain

The following figure indicates the configuration of the JTAG chain and associated resistors.

Figure 8-1: JTAG Chain Interface



This page has been intentionally left blank.

Chapter 9

GPIO/LB Functionality

This page has been intentionally left blank.

9. GPIO/Local Bus Functionality

The EB8347 provides a very flexible set of general purpose IOs as well as local bus data transfer and addressing capability.

The GPIO functionality is realized via the CPU (up to 12 DIOs) and the BPCC (up to 12 DIOs). These signals are all provided on the Communications Interface Extension connector.

The local bus provides a 16-bit wide data bus to the System Interface Extension connector. Addressing is accomplished via a multiplexing scheme whereby the BPCC provides the address lines 12 to 23 to a separate multiplexer which shares the CPU GPIO lines. The address lines 0 to 11 are multiplexed internally by the BPCC along with twelve GPIOs. All local bus address lines are provided on the Communications Interface Extension connector.

The following figure demonstrates the GPIO and local bus data/addressing interfacing and the table below provides information regarding the possible configurations for GPIOs and local bus data/addressing.

Figure 9-1: GPIO and Local Bus Data/Addressing Block Diagram

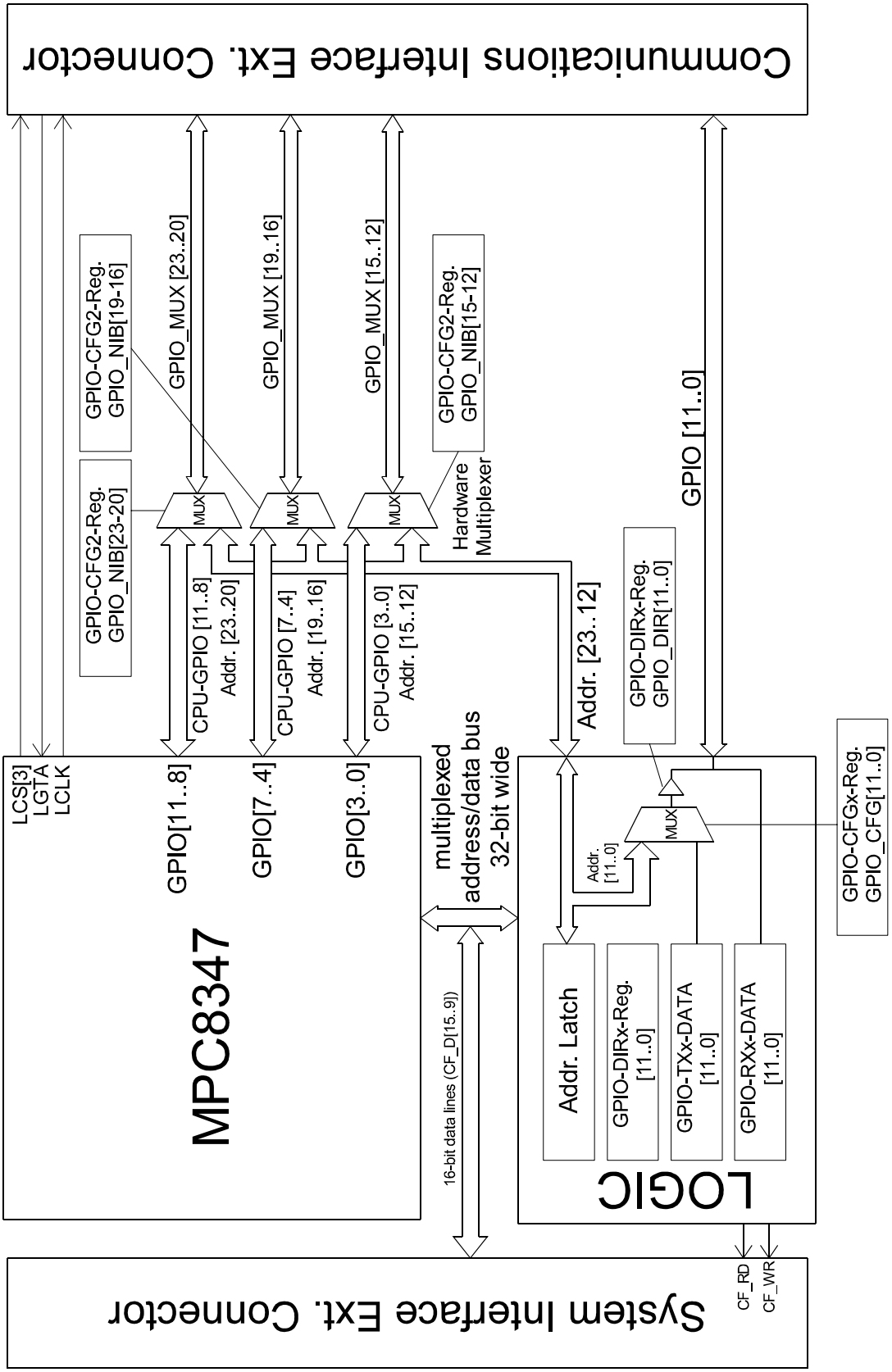


Table 9-1: GPIO/DATA/ADDRESS Configurations

DATA BUS WIDTH	LOCAL BUS ADDRESS LINES	CPU GPIO	BPCC GPIO
16-BIT	24	-	-
16-BIT	23	1	-
16-BIT	22	2	-
16-BIT	21	3	-
16-BIT	20	4	-
16-BIT	19	5	-
16-BIT	18	6	-
16-BIT	17	7	-
16-BIT	16	8	-
16-BIT	15	9	-
16-BIT	14	10	-
16-BIT	13	11	-
16-BIT	12	12	-
16-BIT	8	12	4
16-BIT	4	12	8
-	-	12	12

This page has been intentionally left blank.