Malta™ User's Manual

Document Number: MD00048
Revision 01.05
June 10, 2002

MIPS Technologies, Inc.
1225 Charleston Road
Mountain View, CA 94043-1353

Table of Contents

1 Introduction

This document is for the user of the Malta™ platform. It details how to use the board, set it up, and the software-relevant information.

1.1 Purpose of the Malta Platform

The Malta platform was designed in order to provide a standard platform for development work with MIPS32™ and MIPS64™ CPUs.

The design is composed of two parts, the Malta board, which is an ATX form factor and holds the CPU-independent parts of the circuitry, and a "Core" Board, which holds the CPU plus its System Controller and fast SDRAM memory. It is intended to be used in an ATX cabinet.

The four PCI connectors on the board give the user a high degree of flexibility that enables them to extend the functionality of the system according to requirements.

1.2 Vital Sections

To save you the trouble of looking in the Contents list, the most immediately interesting sections are probably:

- Block diagram (Section 3, "Design Overview / Block Diagram")
- Memory map (Section 4, "Memory Map")
- Hardware jumpers & configuration options (Section 5.1, "Connectors & Jumpers")

It is probably also well worth getting acquainted with the literature in the References section.

2 Getting Started

2.1 Required Hardware

In addition to the basic Malta motherboard, you will typically need:

- Suitable standard ATX cabinet with power supply. For a power supply with standby capabilities a minimum current of 720 mA is required (1A/1.5A peak recommended) for the 5V standby voltage.
- The Core Board that carries your choice of CPU.
- Serial cable for RS232 serial connection. The cable must be a Null Modem cable with 9 way ‘D’ female connector in both ends.

2.2 Optional Hardware

The following may also be useful depending on your application:

- Ethernet cable.
- USB cable.
- PS2 Keyboard / mouse.
- IDE disk drive and cabling.
- Type I or II CompactFlash module (supporting True IDE mode).
- Floppy disk drive and cabling.

If you are going in-depth and doing some real debugging, then the following:

- LA probe connectors (that match with the AMP Mictor headers) if you have a HP Logic Analyser.
- PCI probe board if you want to be able to monitor activity on the internal PCI bus (e.g. FuturePlus FS2000)
- Standard Parallel Port Download cable for extending the parallel male-male conversion cable.
- Standard 10 pin header to DB9 converter cable for tty2.

2.3 Wiring It Up

First, if this is not already fitted, connect the Core Board. It is fairly obvious how to do this, the connectors (J3 and J4) have the same numbers on both boards if you are in doubt about the orientation. One of the corner mounting pillars is also offset to prevent incorrect insertion.

When removing the Core Board - **be careful!** There will typically be BGA devices fitted which can take offence at having the board bent violently. Under each corner of the Core Board is a mounting pillar, with a gap where a screwdriver can be inserted to gently lever it up. Only apply the screwdriver to the PCB area around the mounting holes - you don’t want to cut any tracks by accident.

Before turning on the power, you will most likely want to also have the following set up:

- tty0. The supplied PROM monitor (YAMON) by default signs on via the tty0 port (J6), using 38.4 kbaud, 8 bits/char, RTS/CTS hardware handshaking and no parity. A 5-wire cable is sufficient. The implemented signals must be RXD, TXD, RTS, CTS and GND, see Section 6.7, "Serial Ports" for serial connector pinout.
- Ethernet. Twisted-pair ethernet cable will plug into the socket on the rear edge of the board, this will auto configure at 10 or 100Mbit/s, half/full-duplex.

- Check S5 switches are as they should be as specified in Section 5.2, "Switches".

2.4 Power-up Sequence

When you first connect the power supply and switch it on the board is powered up. Check that the green “ATX ON”, “3V3”, “5V” and “STBY” LEDs turn on to indicate good power.

NOTE: It has been seen with some ATX supplies that Malta draws so little current that the supply is not stable. This is technically a deviation from the ATX spec.

NOTE: The board is brought into “standby” mode by pressing the switch marked “ON/NMI” (S4) for more than four seconds. The “ATX ON” LED is lit when in “standby” mode. Press “ON/NMI” to bring up the board again.

The green “FPGA” LED should be on - this indicates that the board’s FPGA has booted.

The red “RST” LED should be off - if lit it indicates that something is holding the board in reset.

When the CPU initially boots, the YAMON monitor signs on using the tty0 serial port (the left one) with information about the board configuration details, i.e. board revision, SDRAM size etc.

Finally you should arrive at YAMON’s prompt line. Simultaneously, you should see the word “YAMON” on the ASCII LED display. If you do not see this, check with the YAMON User’s Manual for what the display messages mean.

The command “help” lists the available commands in the YAMON monitor, and “help <command name>” gives more detailed information about the specific command.

Note that you will be running in little-endian mode, and that you can change this with a switch setting. See the Malta User’s Manual for details.

3 Design Overview / Block Diagram

The diagram below gives an overview of the important features of the Malta Board.

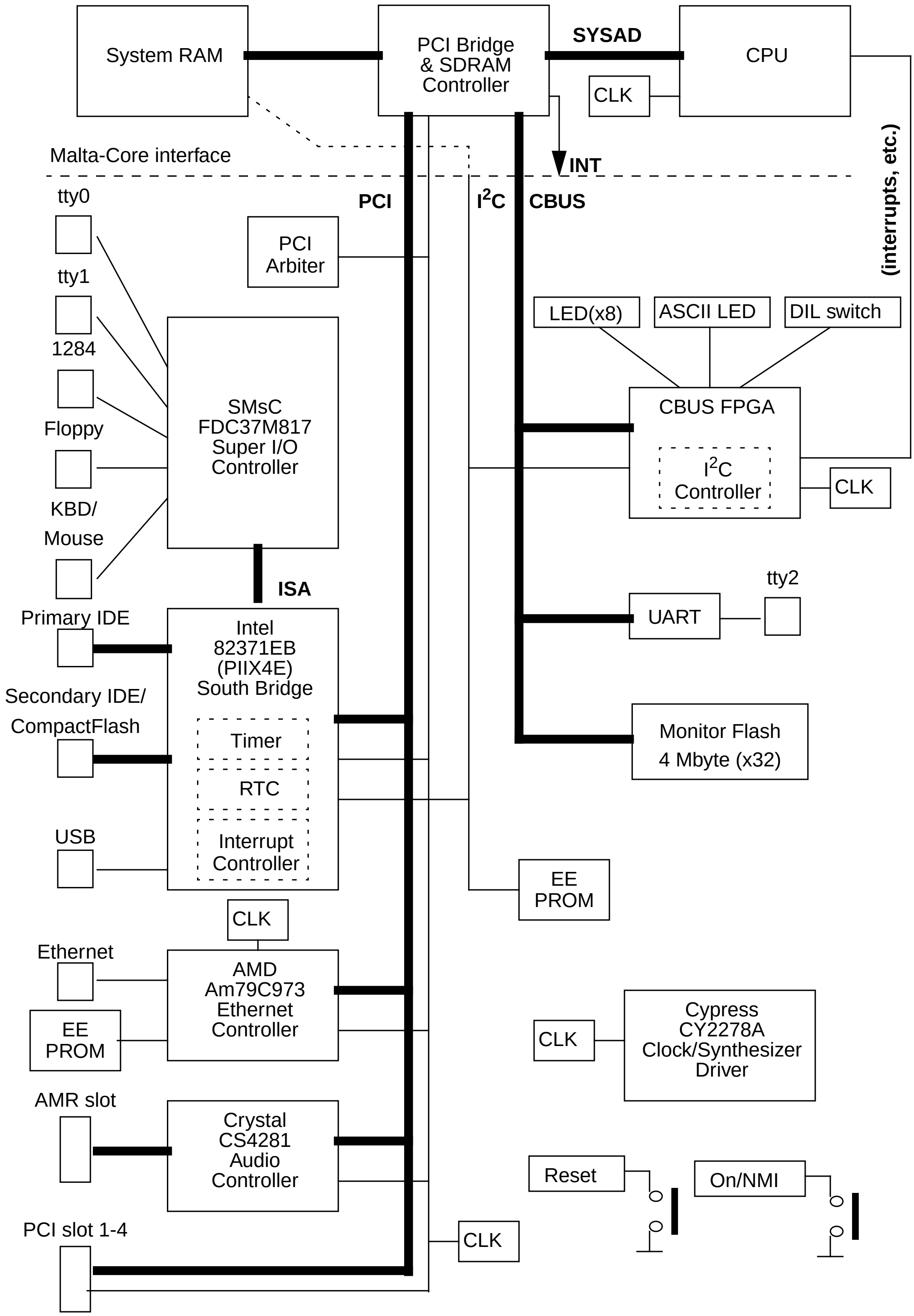


Figure 1 Block Diagram

The Core Board is not strictly a part of this document, and the design shown is only a typical implementation (in fact, pretty much every implementation will look like this until CPUs that have outlived the SysAD bus start arriving). However all Core Boards will conform to the same interface specification, see Ref[5].

Worth noting is that most Core Board generates its own clock indepently of the PCI clock. Nothing on Malta is synchronized to the Core Board clock, the CBUS protocol is asynchronous by nature.

The CBUS exists to allow the CPU to access peripherals which either have to be available before the PCI bus has been configured (for example the Flash memory it's booting from) or those that require simple, low-latency access, for example the debug LEDs and the ASCII display, the tty2 port and so on.

The Monitor Flash is used to boot the system.

The PCI bus is 32-bit, 33 MHz PCI standard version 2.2 compliant (Ref[1]), and allows devices on the bus DMA access to the DRAM on the Core Board. Four 5V PCI slots are provided on Malta to allow insertion of optional peripherals (for example a video controller), and also provides a way of monitoring traffic on this bus.

On the PCI bus are:

- Core Board connector for connection to the system controller on the Core Board.
- Intel PIIX4E South Bridge, 82371E (U9).
- AMD Ethernet controller, Am79C973 (U41).
- Crystal Audio controller, CS4281 (U23).
- Four 5V, 32 bit PCI connectors (J12-J15) that can be used for debug / trace purposes or for installation of a PCI board.

The Malta Board has a PC like structure with a South Bridge. An ISA bus is attached to the South Bridge for connection with the Super I/O (U11). On the Super I/O are:

- PS/2 Keyboard and Mouse (J7).
- 1284 parallel port (J6).
- Two serial ports tty0 and tty1 (J6).
- Floppy drive, only drive A supported (J21).

4 Memory Map

The memory map as seen from the CPU is partly dependent on the Core Board, and how the PCI bus is configured. However the Malta design places some requirements on how the CBUS is mapped, and therefore this area will always appear as shown below.

Table 1 Malta Physical Memory Map

Base address	Size	Function
0000.0000(*)	128Mbytes	Typically SDRAM (on Core Board)
0800.0000	256 Mbytes	Typically PCI
1800.0000	62 Mbytes	Typically PCI
1BE0.0000	2 Mbyte	Typically System controller's internal registers
1C00.0000	32 Mbytes	Typically not used
1E00.0000	4 Mbytes	Monitor Flash
1E40.0000	12 Mbytes	reserved
1F00.0000	12 Mbytes	Switches
		LEDs
		ASCII display
		Soft reset
		FPGA revision number
		CBUS UART (tty2)
		General purpose I/O
		I^2C controller
1F10.0000	11 Mbytes	Typically System Controller specific.
1FC0.0000	4Mbyte	Maps to Monitor Flash
1FD0.0000	3 Mbytes	Typically System Controller specific.

Note: The shaded area of the table indicates memory areas the mapping of which depends on the implementation of the Core Board, and of software.

Note: The memory area 000F.0000-000F.FFFF (PC BIOS area) is only accessible from the CPU, not from the PCI bus (The South Bridge decodes this memory area).

Note: Address 1FC0.0010 is "special", in the sense that when the software read this address it is overridden and does NOT decode to an address in Flash, but rather to register address REVISION. This is done to ensure future compatibility - all MIPS Technologies boards will be able to identify their hardware environment and configure themselves accordingly. Reads from address 1E00.0010 will decode to an address in Flash.

RAM is typically mapped at the bottom of memory so that the exception vectors are located in fast memory.

Malta does not specify a mapping for addresses above 0x2000.0000 which cannot be addressed in kseg0/1.

The I^2C bus (called SMB bus in the Intel documentation) is controlled by the CBUS FPGA (it can also be controlled by the controller in the South Bridge), its slave address map is configured as follows:

Table 2 I^2C Slave Address Map

I^2C slave address	Size	Function
0x50	256 bytes	Core Board, PC-100 SDRAM
0x51	256 bytes	Core Board, optional PC-100 SDRAM
0x52	256 bytes	Core Board, optional PC-100 SDRAM
0x53	256 bytes	Core Board, optional PC-100 SDRAM
0x54-0x57	1024 bytes	Malta EEPROM - read-only Contains serial number

4.1 Devices

Devices in the memory map above are listed below, with their internal register maps and programming details. Note that all addresses listed below are physical addresses. It is recommended to use the macros in the header files to access all registers and fields, see Ref[3].

All registers shall be addressed as 32-bit words, and lie on 64-bit word boundaries.

This convention allows software to access all registers using the same word address in both little and in big endian mode. Those registers that contain a single value are not described in bit-field detail, these values are aligned to the LSB of the word.

4.1.1 Revision Info

Name: REVISION

Address: 0x1FC0.0010

Access: RO

Reset Value: n/a

Table 3 REVISION Register

Bits	Field name	Function	Initial value
31:24	reserved		0
23:16	FPGRV	8 bit binary number gives revision of CBUS FPGA.	n/a
15:10	CORID	6 bit Core Board ID	n/a
9:8	CORRV	2 bit Core Board revision	n/a
7:4	PROID	4 bit binary number gives product ID	0x2
3:0	PRORV	4 bit binary number gives product revision.	n/a

The contents of this register gives information about the revision of the Malta and Core Boards. This information is for MIPS internal use.

4.1.2 NMI Interrupts

There are two sources of NMI :

- ON/NMI push button
- South Bridge due to PCI SERR assertion (from PCI slot or Core card) or ISA IOCHK assertion.

Either of these interrupts can generate a NMI. When the ON/NMI push button is activated the signal is debounced and latched in the NMI interrupt controller. The South Bridge NMI is routed through the NMI controller as it is. These signals then generate an active state on MIPS core board NMIN pin.

Name:	NMISTATUS
Address:	0x1F00.0024
Access:	RO
Reset Value:	n/a

Table 4 NMISTATUS Register

Bits	Field name	Function	Initial value
31:1	reserved		n/a
1	SB	Pending NMI from the South Bridge	n/a
0	ONNMI	Pending NMI from the ON/NMI push button	n/a

4.1.3 NMI Acknowledge

The ON/NMI interrupt is by nature transient. Therefore it is debounced and latched and thereafter treated as ordinary level-based interrupt in the NMI interrupt controller. The NMI interrupt can be cleared by writing to NMIACK. When the bit in this register is written as “1”, the interrupt is cleared. Note that South Bridge NMI is acknowledged in the South Bridge.

Name:	NMIACK
Address:	0x1F00.0104
Access:	WO
Reset Value:	n/a

Table 5 NMIACK Register

Bits	Field name	Function	Initial value
31:1	reserved		n/a
0	ONNMI	Write 1 to acknowledge ON/NMI NMI	n/a

4.1.4 Switches / Status

The following registers allow software to monitor the state of various switches and jumpers on the Malta board. All DIP switches give a value of “1” for a switch in the “ON” position.

A switch is considered ON if any of the following are true:

- It is in the position marked “ON” on the switch body.

- It is in the position marked “CLOSED” or not in the “OPEN” position as marked on the switch body.

There is no debouncing on these registers, so if software wants to monitor a value while it changes, allowance for this must be made by waiting for the new value to become stable.

For the DIP switches S2 & S5, bit 0 is mark by a dot or by a “0” in the silkscreen or the switch is marked by a “1”.

Name: SWITCH

Address: 0x1F00.0200

Access: RO

Reset Value: n/a

Table 6 SWITCH Register

Bits	Field name	Function	Initial value
31:8	reserved		0
7:0	S2	8-bit value of the setting of DIP switch S2.	n/a

Name: STATUS

Address: 0x1F00.0208

Access: RO

Reset Value: n/a

Table 7 STATUS Register

Bits	Field name	Function	Initial value
31:5	reserved		0
4	MFWR	“1” indicates Monitor Flash lock bits are write protected (JP1 fitted).	n/a
3	S54	DIP switch S5-4. YAMON use this switch. “1” will set YAMON in factory default mode (communication on tty0 etc.).	n/a
2	S53	DIP switch S5-3.	n/a
1	BIGEND	“1” indicates big endian mode as controlled by switch S5-2.	n/a
0	reserved		0

Name: JMPRS

Address: 0x1F00.0210

Access: RO

Reset Value: n/a

Table 8 JMPRS Register

Bits	Field name	Function	Initial value
31:5	reserved		0
4:2	PCICLK	PCI clock frequency 10-37.5MHZ. See jumper Table 23 Bit 4 is Pins 5-6 Bit 3 is Pins 3-4 Bit 2 is Pins 1-2 “1” = jumper fitted.	n/a
1	EELOCK	State of JP2: Not fitted ~ “1” = I^2C EEPROM write protected.	n/a
0	reserved		0

4.1.5 Displays

There are 2 display devices on the board: An 8-LED array (D28, is a 10-LED but only 8 are used), and an 8-character ASCII display (U42). These are controlled through the following registers.

Table 9 Display Registers. BASE = 0x1F00.0400

Name	Offset Address	Access	Function
LEDBAR	0x0000.0008	R/W	8 bits each corresponding to 1 LED (1 = ON).
ASCIIWORD	0x0000.0010	WO	Writing a 32-bit word to this register will cause it to be displayed in hex on the ASCII character display.
ASCIIPOS0	0x0000.0018	WO	Writing an ASCII value to this register updates ASCII display position ‘0’, which is the left-most positioned character.
ASCIIPOS1	0x0000.0020	WO	Writing an ASCII value to this register updates ASCII display position ‘1’.
ASCIIPOS2	0x0000.0028	WO	Writing an ASCII value to this register updates ASCII display position ‘2’.
ASCIIPOS3	0x0000.0030	WO	Writing an ASCII value to this register updates ASCII display position ‘3’.
ASCIIPOS4	0x0000.0038	WO	Writing an ASCII value to this register updates ASCII display position ‘4’.
ASCIIPOS5	0x0000.0040	WO	Writing an ASCII value to this register updates ASCII display position ‘5’.

Table 9 Display Registers. BASE = 0x1F00.0400

Name	Offset Address	Access	Function
ASCIIPOS6	0x0000.0048	WO	Writing an ASCII value to this register updates ASCII display position ‘6’.
ASCIIPOS7	0x0000.0050	WO	Writing an ASCII value to this register updates ASCII display position ‘7’, which is the right-most positioned character.

Name: LEDBAR

Address: 0x1F00.0408

Access: R/W

Reset Value: 0x00

Table 10 LEDBAR Register

Bits	Field name	Function	Initial value
31:8	reserved		0
7:0	BAR	8 bits each corresponding to 1 LED (1 = ON).	0x00

Name: ASCIIWORD

Address: 0x1F00.0410

Access: WO

Reset Value: n/a

Table 11 ASCIIWORD Register

Bits	Field name	Function	Initial value
31:0	HEX	Writing a 32-bit word to this register will cause it to be displayed in hex on the ASCII character display.	n/a

Name: ASCIIPOS0

Address: 0x1F00.0418

Access: WO

Reset Value: n/a

Table 12 ASCIIPOS0 Register

Bits	Field name	Function	Initial value
31:8	reserved		n/a
7:0	ASCII	Writing an ASCII value to this register updates ASCII display position ‘0’. Position ‘0’ is the left-most positioned character.	n/a

Register descriptions for ASCIIPOS1 to ASCIIPOS7 are similar with the description above for ASCIIPOS0.

4.1.6 Reset Control

There are two different reset functionalities that are controlled by software. Writing a “magic” value to the SOFTRES register immediately triggers a reset of the whole board, and the BRKRES register controls how the “break” condition on the tty0 port is monitored. Both reset functions generate a board reset with the exact same effect as if you had pressed the reset button.

Name:	SOFTRES
Address:	0x1F00.0500
Access:	WO
Reset Value:	0x00

Table 13 SOFTRES Register

Bits	Field name	Function	Initial value
31:8	reserved		n/a
7:0	RESET	Writing the magic value GORESET (==0x42) to this field will initiate a board reset	0x00

Name:	BRKRES
Address:	0x1F00.0508
Access:	R/W
Reset Value:	0x0A

Table 14 BRKRES Register

Bits	Field name	Function	Initial value
31:8	reserved		0
7:0	WIDTH	Writing a value to this address indicates the number of milliseconds in length a “Break” must be on the tty0 port in order to trigger a reset. Valid values are from 0 to 255. A value of zero prevents this reset ever occurring.	0x0A (i.e. 10ms)

NOTE: The initial value for WIDTH of 10 ms will cause problems, if the baud rate of the tty0 port is less than 2400 Baud. If baud rates below 2400 Baud are used, this register must be programmed to a larger value.

4.1.7 CBUS UART, tty2

For details on programming the CBUS UART (a TI 16C550C), see the appropriate data sheet on their web site (Ref[10]). The clock frequency for baud rate calculations is 3.6864 MHz.

The registers of the UART, which have a native width of 8-bit, are all memory mapped on 64-bit aligned boundaries with the following layout:

Table 15 UART Registers. BASE = 0x1F00.0900

Name	Offset Address	Access	Function
RXTX	0x0000.0000	R/W	Receive / Transmit char register
INTEN	0x0000.0008	R/W	Interrupt enable register
IIFIFO	0x0000.0010	R/W	Read: Interrupt identification Write: FIFO control
LCTRL	0x0000.0018	R/W	Line control register[a]
MCTRL	0x0000.0020	R/W	Modem control register
LSTAT	0x0000.0028	R/W	Line status register
MSTAT	0x0000.0030	R/W	Modem status register
SCRATCH	0x0000.0038	R/W	Scratch register

a. The Divisor Latch Registers are accessible through RXTX and INTEN registers when bit 7 (Divisor Latch Access Bit) of the Line Control Register is set.

4.1.8 General Purpose I/O

Eight GP inputs and eight GP outputs connected to the Core Board are implemented on the Malta board. Read documentation on the individual Core Board for usage details.

Name: GPOUT
Address: 0x1F00.0A00
Access: R/W
Reset Value: n/a

Table 16 GPOUT Register

Bits	Field name	Function	Initial value
31:8	reserved		n/a
7:0	OUTVAL	Writing to this address sets the 8 GP output pins. Reading gives the actual setting of the GP output pins. Functionality is Core Board dependent.	0x00

Name: GPINP
Address: 0x1F00.0A08
Access: RO
Reset Value: n/a

Table 17 GPINP Register

Bits	Field name	Function	Initial value
31:8	reserved		0
7:0	INPVAL	Reading gives the actual state of the GP input pins. Functionality is Core Board dependent.	n/a

4.1.9 I^{2}C

I^{2}C bus access registers. As both lines have an open-drain output no conflicts can occur during direction shift on the bi-directional lines. There are three registers for I^{2}C control:

- I2CINP: Reads input values.
- I2COE: Controls output enables
- I2COUT: Controls output values.

And one register for selecting I^{2}C controller:

- I2CSEL: Selects between the FPGA I^{2}C controller and the South bridge I^{2}C controller (the two I^{2}C controllers cannot co-exist).

Name: I2CINP
Address: 0x1F00.0B00
Access: RO
Reset Value: n/a

Table 18 I2CINP Register

Bits	Field name	Function	Initial value
31:2	reserved		n/a
1	I2CSCL	Reading gives the actual value of the I^{2}C SCL pin.	1
0	I2CSDA	Reading gives the actual value of the I^{2}C SDA pin.	1

Name: I2COE
Address: 0x1F00.0B08
Access: R/W
Reset Value: n/a

Table 19 I2COE Register

Bits	Field name	Function	Initial value
31:2	reserved		n/a
1	I2CSCL	“1” means that the I2CSCL bit of the I2COUT register will be driven to the I^{2}C SCL pin. “0” means that the I^{2}C SCL pin is tri-stated.	0

Table 19 I2COE Register

Bits	Field name	Function	Initial value
0	I2CSDA	“1” means that the I2CSDA bit of the I2COUT register will be driven to the I^2C SDA pin. “0” means that the I^2C SDA pin is tri-stated.	0

Name:	I2COUT
Address:	0x1F00.0B10
Access:	R/W
Reset Value:	n/a

Table 20 I2COUT Register

Bits	Field name	Function	Initial value
31:2	reserved		n/a
1	I2CSCL	The value of this bit will be driven to the I^2C SCL pin when the I2CSCL bit of the I2COE register is “1”.	1
0	I2CSDA	The value of this bit will be driven to the I^2C SDA pin when the I2CSDA bit of the I2COE register is “1”.	1

Name:	I2CSEL
Address:	0x1F00.0B18
Access:	R/W
Reset Value:	0x01

Table 21 I2CSEL Register

Bits	Field name	Function	Initial value
31:1	reserved		n/a
0	I2CFPGA	“1” means that the I^2C controller in the FPGA is enabled and the I^2C controller in the South Bridge is disconnected from the I^2C bus. “0” means that the I^2C controller in the FPGA is disabled and the I^2C controller in the South Bridge is connected to the I^2C bus.	1

5 Board Layout

The diagram in Figure 2 shows the basic layout of Malta.

The board has an ATX (305 mm x 244mm) form factor and is in accordance with the ATX specification (Ref[12]) with regards to board size, mounting hole placement, connector placement and height constraints.

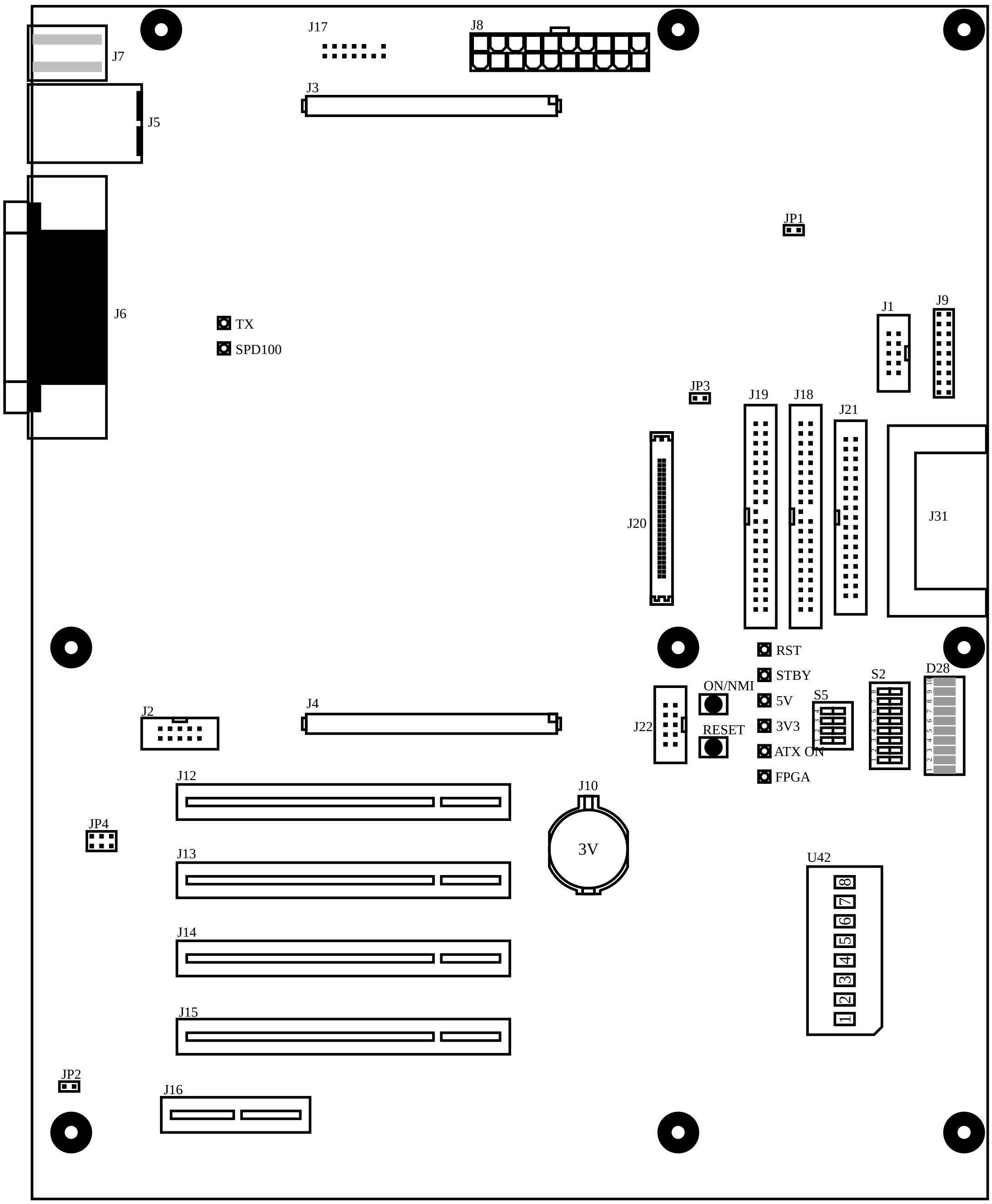


Figure 2 Malta Board Layout

Figure 3 shows the rear panel connector layout.

The connector layout on the rear panel i.e. the shield design as described in Design Guide for Intel ATX Motherboard I/O Implementations Version 1.1 (Ref[13]) is a subset of Intel Core design #1. This enables the Malta Board to be installed in an ordinary ATX chassis.

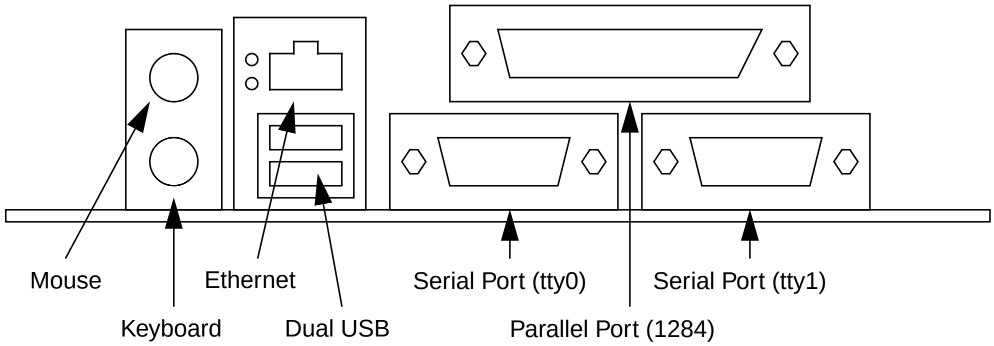


Figure 3 Rear Panel Connector Layout

5.1 Connectors & Jumpers

All jumpers & connectors are listed below. All jumpers are standard 0.1” pitch. It is worth getting some spares in case you lose any. A dot by a pin indicates which pin is pin 1. On all jumpers, pin numbering goes crosswise, i.e. the end pins are 1 & 2, this is not always the case on other connectors.

Table 22 Interface Connectors

Ref	Type	Description
J1	10 pin 0.1” header	Download connector for CBUS FPGA EEPROM.
J2	10 pin 0.1” header	Download connector for Arbiter EPLD.
J3	200way header	Connects the Core Board. Carries amongst other things the CBUS.
J4	200way header	Connects the Core Board. Carries amongst other things the primary PCI bus.
J5	RJ45	Ethernet.
	Dual USB A	Two USB host ports.
J6	25 pin DSUB	IEEE1284 / Flash programming port. This can either be used as a parallel port for communications, and is also used to program Flash devices during production.
	Dual 9 pin DSUB	tty0 (left) and tty1 (right).
J7	Dual 6 pin miniDIN	Connection to a PC keyboard (lower) and mouse (upper).
J8	ATX PSU	This connects the power.
J9	18 pin 0.1” header	Front Panel connector. Power LED, HD LED, Power Switch, Reset Switch. Pin 14 is removed from the connector.
J10	Battery Connector	CR2032 Battery Connector used by the RTC (South Bridge).
J12-15	PCI slot	Allows insertion of probe board, or PCI board with additional functionality.
J16	AMR slot	Allows insertion of audio modem riser (AMR) board.

Table 22 Interface Connectors

Ref	Type	Description
J17	14 pin 0.1" header	EJTAG connector. As per EJTAG specification, see Ref[4]. Pin 12 is removed from the connector.
J18	40 pin 0.1" header	Primary IDE interface. Pin 20 is removed from the connector.
J19	40 pin 0.1" header	Secondary IDE interface. Pin 20 is removed from the connector.
J20	Compact Flash connector	Compact Flash interface at Secondary IDE interface. Type I or II module.
J21	34 pin 0.1" header	Floppy Disk interface. Connects to FD via IDC ribbon cable (7 wires twisted, PC-style). Drive A support only.
J22	10 pin 0.1" header	tty2 (CBUS UART).
J23	AMP 38 pin Mictor	HP Logic Analyzer connector (CBUS data).
J24	AMP 38 pin Mictor	HP Logic Analyzer connector (CBUS control + address).
J25	AMP 38 pin Mictor	HP Logic Analyzer connector (CBUS FPGA).
J26	AMP 38 pin Mictor	HP Logic Analyzer connector (Reset + Interrupts).
J27	AMP 38 pin Mictor	HP Logic Analyzer connector (JTAG + PCI Arbiter).
J30	3 pin 0.1" header	Power LED connector
J31	Compact Flash connector	Compact Flash interface at Secondary IDE interface. Type I or II module. Alternative fit for J20

Table 23 Jumpers

Ref	Silkscreen	Pins	Options	Default	Description	
JP1	MFWR	2	fit-notfit	notfit	When fitted	Enables writing to the Monitor Flash lock bits from software. It also allows writing to the Monitor Flash itself, regardless of the state of the Lock bits.
					When not fitted	Disables writing to the Monitor Flash lock bits from software.
JP2	EEWR	2	fit - notfit	notfit	When fitted	Enables writing to the I^2C EEPROM (U14). Do NOT fit this - it is reserved for production use.
					When not fitted	Disables writing to the I^2C EEPROM (U14).
JP3	CF MASTER	2	fit - notfit	notfit	When fitted	Sets Compact Flash module as Master IDE drive on the secondary IDE bus.
					When not fitted	Sets Compact Flash module as Slave IDE drive on the secondary IDE bus.

Table 23 Jumpers

Ref	Silkscreen	Pins	Options	Default	Description			
JP4	PCI CLK	6	10 - 37.5	33.33(2)	Sets PCI clocking frequency between 10MHz - 37.5MHz. “X” = fitted.			
					MHz/Pins	1-2	3-4	5-6
					10(1)	X	X	X
					12.5(1)			X
					16.67(1)	X		X
					20(1)	X		
					25(1)		X	
					30(1)	X	X	
					33.33			
					37.5		X	X

Note 1: Only 10BASE-T is supported (100BASE-TX is not).

Note 2: Some Core Boards cannot run with PCI clock frequency of 33.33 MHz. See the respective Core Board User’s Manuals for maximum clock frequency.

5.2 Switches

Switches are listed below, together with their functions, and default settings. This is how you should receive the Malta board. For those switches that are software-readable, a switch in position “ON” or “CLOSED” (not in the “OPEN” position) will give a “1” in the appropriate register.

Table 24 Switches

Ref	Type		Default	Description
S2	8 way DIP		All OFF	This switch provides a value which can be read from the SWITCH register.
S3	Push-button		n/a	Reset button.
S4	Push-button		n/a	NMI/Power ON button. This button will bring the ATX power supply out of standby. It also generates an NMI to the CPU, for example to shut down the PSU again. This button will also make a hardware shutdown if pressed for more than four seconds at a PCI clock at 33MHz. For PCI clocks below 33.33 MHz the button has to be pressed for a longer period of time (up to 12 seconds).

Table 24 Switches

Ref	Type		Default	Description
S5	4-way DIP	S5-1	OFF	When ON, enables Flash programming via 1284 parallel port. This swithch enables writing to the Monitor Flash lock bits. It overrides Jumper JP1.
		S5-2	OFF	When ON, set operation mode to big endian. If the endianess is changed, Malta must be reset again in order for the new endian mode to take effect. If the board is not reset unpredictable operation can occur.
		S5-3	OFF	No default function - can be read from the STATUS register.
		S5-4	OFF	When ON at power-on or at reset, sets YAMON in factory default mode eg. communication on tty0 port is forced to 38.4 kbaud, 8 bits/char, RTS/CTS hardware handshaking and no parity.

DIP switches S2, S5-2, S5-3 and S5-4 are readable by software.

For the DIP switches S2 & S5, switch referred to as “1” is marked by a dot or by a “0” in the silkscreen or the switch is marked by a “1”.

5.3 Displays / LEDs

There are two displays (see Section 4.1.5, "Displays" on how to control these) and various individual status LEDs placed on the Malta board.

Table 25 LEDs on the Malta Board

Ref	Silkscreen	Type	Description
D28	n/a	8-way bar	Controlled by software.
U42	n/a	8 char ASCII display	Used by YAMON to display status. Can be used for any user purpose.
D7	ATX ON	Green SMD	Indicates that power is applied to the ATX power-supply. Also lit when board is in standby mode.
D2	STBY	Green SMD	Indicates that power is applied to 3V3STBY (+/-5%) and 5VSTBY (+/-5%). Not led when board is in standby mode.
D6	5V	Green SMD	Indicates that power is applied to 5V (+/-5%).
D4	3.3V	Green SMD	Indicates that power is applied to 3.3V (+/-5%).
D5	FPGA	Green SMD	Indicates that CBUS FPGA programming completed OK.
D9	RST	Red SMD	Indicates that RSTN is active.
D1	TX	Yellow SMD	Ethernet LED3: Blinks on TX Ethernet packets (Programmable).
D3	SPD100	Green SMD	Ethernet LED2: ndicates that 100 Mbit speed is selected (Programmable).

The two LEDs which are built-in to the RJ45 connector J5 give the status as follows:

Table 26 Ethernet Connector LED Functionality

LED	Function
Green	Ethernet LED0: Link up (Programmable)
Yellow	Ethernet LED1: Activity (Programmable)

All four ethernet LEDs are programmed/controlled by the ethernet controller. LED0-LED3 are linked to the four LEDs.

6 Hardware Description

This section details the board design details at a reasonable user level. See also the schematics.

6.1 PCI Bus

The PCI bus is implemented as a 5V, 32bit and 33 MHz PCI standard version 2.2 compliant bus (Ref[1]) that connects the main components on the Malta Board. The devices on the PCI bus are:

- Core Board connector for connection to the system controller on the Core Board.
- Intel PIIX4E South Bridge, 82371E (U9).
- AMD Ethernet controller, Am79C973 (U41).
- Crystal Audio controller, CS4281 (U23).
- Four 5V, 32 bit PCI connectors (J12-J15) that can be used for debug / trace purposes and/or for installation of PCI boards.

For configuration purposes the IDSEL and INT# signals to the PCI devices are connected as shown below:

Table 27 IDSEL and INT# for PCI Devices

Device	IDSEL PCI address line	PCI Interrupts mapping			
		PCI_INTAN	PCI_INTBN	PCI_INTCN	PCI_INTDN
South Bridge	PCI_ADP20				USB_IRQ#
Ethernet controller	PCI_ADP21		ETHER_IRQ#		
Audio controller	PCI_ADP22			AUDIO_IRQ#	
Core Card	PCI_ADP27				
PCI Connector 1	PCI_ADP28	INTA#	INTB#	INTC#	INTD#
PCI Connector 2	PCI_ADP29	INTD#	INTA#	INTB#	INTC#
PCI Connector 3	PCI_ADP30	INTC#	INTD#	INTA#	INTB#
PCI Connector 4	PCI_ADP31	INTB#	INTC#	INTD#	INTA#

6.1.1 PCI Arbiter

The PCI arbiter controls the request and grant scheduling to the eight PCI components and is implemented in an Altera MAX3064 EPLD.

The PCI arbiter implements a round-robin scheme where each of the eight components have equal priority.

Figure 4 shows the signals used during PCI arbitration.

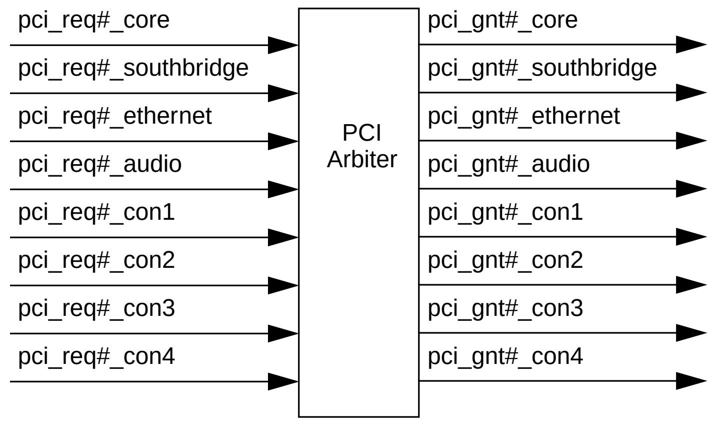


Figure 4 PCI Arbiter Connections

6.2 I^2C Bus

There are two I^2C controllers on the board:

- A simple one in the FPGA, used for getting SDRAM information, for debug purposes and similar operations when the PCI bus is unconfigured.
- A more advanced one in the South Bridge used for normal operation.

The active I^2C controller is set in the I2CSEL register in FPGA.

6.3 Power

The board can run using 3.3V and 5V supplied from a standard PC ATX power supply connected to J8. This should comfortably be able to supply enough current for the board and conceivable Core Board options. The 12V and -12V is only connected to the Core Board (only 12V), the AMR and PCI connectors. See Ref[11] for details of a suitable supply.

Power On/Off is controlled by the South Bridge. It is functioning the same way as a PC.

The board also supports Power Management Events, e.g. Wake On LAN events, for powering up when in standby mode.

6.4 Reset

A push-button switch (S3) is provided to reset the board. Alternative sources of reset are:

- The CBUS FPGA when a software-reset register is written.
- The EJTAG probe system reset signal (EJRSTN).
- An incoming break on the tty0 port (J6). This may be disabled by software.

All resets are the same - there is no distinguishing between a “warm” or a “cold” reset. All hardware, incl. hardware driven by standby voltages, are reset at reset.

6.5 Clocks

The PCI clock normally runs at 33 MHz, generated from a 14.31818MHz crystal using a clock synthesizer/driver (U13). The PCI clock can be configured via JP4, see Table 23.

This will not affect the clock frequency of a CPU mounted on its Core Board. The Core Board generates its own clock.

The different clocks on the board are:

- RTC (32.768 KHz).
- CBUS UART (tty2) (3.6864 MHz).
- ISA Environment (14.31818 MHz).
- Ethernet (25 MHz).
- PCI clock (33 MHz - configurable to 10, 12.5, 16.67, 20, 25, 30, 33.33 and 37.5 MHz).
- CBUS FPGA (40 MHz).
- USB (48 MHz).

6.6 Interrupt

The interrupt controller is located in the South Bridge device. A NMI interrupt controller (for South bridge NMI and ON/NMI button) is located in the FPGA (see Section 4.1.2, "NMI Interrupts").

Interrupts routed to the South Bridge are triggered by the following devices :

- South Bridge internal devices (timer, real time clock, USB).
- Super IO devices (keyboard, 2 UARTs, floppy disk, parallel port, mouse).
- Ethernet controller.
- Audio controller.
- Primary and Secondary IDE devices.
- PCI slots 1..4.
- SERR (PCI bus) and IOCHK (ISA bus) signals may trigger the South Bridge NMI interrupt.
- Various power management related events in the South Bridge may trigger the South Bridge SMI interrupt.
- I^2C bus controller in the South Bridge may trigger either the South bridge SMI or IRQ9 interrupt.

Interrupts routed directly to the Core Card are triggered by the following devices :

- Core card (COREHI, CORELO signals).
- Discrete 16550 UART device (CBUS UART (tty2)).

Figure 5 shows the interrupt wiring. The figure does not include the connections of SERR from the PCI slots and the Core Card to the South Bridge.

IRQ 0..15 from devices located in the SuperIO device are routed to the South Bridge using a serial connection.

PCI A..D interrupts including the South Bridge USB controller (using PCI D) are mapped onto IRQ 0..15, which are further multiplexed to South Bridge INTR.

Based on the interrupt sources, the South Bridge generates 3 interrupts : INTR, SMI and NMI.

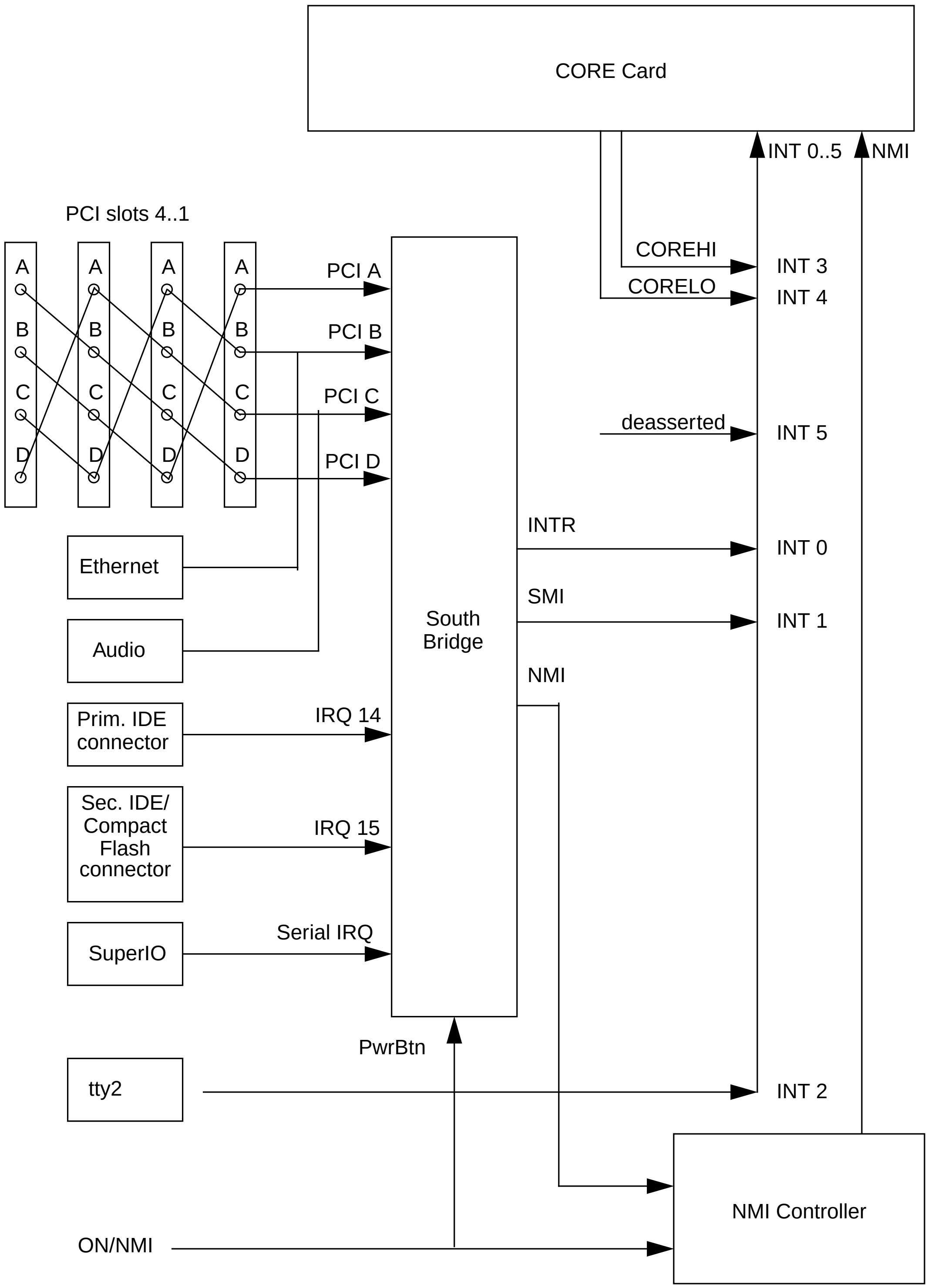


Figure 5 Interrupt Wiring

Most sources of interrupt are handled in an interrupt controller located in the South Bridge. A few are handled in the CBUS FPGA (COREHI, CORELO and CBUS UART (tty2) interrupt), this means that CPU and CBUS interrupt sources can be handled before the PCI bus has been configured. Please use the macros in the header file to access all registers and fields of the interrupt controller, see Ref[3].

IRQ 0..15 are prioritized in the sequence : 0, 1, 8..15, 3..7. IRQ 2 is reserved for cascading the two 82C59 devices that together constitute the South Bridge Interrupt Controller.

The mapping of IRQ 0..15, as used by YAMON, is shown in Table 28.

Table 28 IRQ 0..15 Mapping

IRQ #	Source(s)	Device(s)
0	Timer	South Bridge
1	Keyboard	SuperIO
2		Reserved by South Bridge (for cascading)
3	UART (tty1)	SuperIO
4	UART (tty0)	SuperIO
5		Not used
6	Floppy Disk	SuperIO
7	Parallel port (1284)	SuperIO
8	Real Time Clock	South Bridge
9	I^2C bus	South Bridge
10	PCI A, PCI B (including Ethernet)	PCI slot 1..4, Ethernet
11	PCI C (including audio), PCI D (including USB)	PCI slot 1..4, Audio, USB (South Bridge)
12	Mouse	SuperIO
13		Reserved by South Bridge
14	Primary IDE	Primary IDE slot
15	Secondary IDE	Secondary IDE slot/Compact flash connector

The mapping of CPU INT0..5 and CPU NMI is shown in Table 29.

Table 29 CPU INT0..5 and CPU NMI Mapping

CPU INT/NMI	Source(s)	Device(s)
NMI	South Bridge NMI or NMI button	South Bridge or On/NMI Buttom
0	South Bridge INTR	South Bridge
1	South Bridge SMI	South Bridge
2	CBUS UART (tty2)	Discrete 16550
3	COREHI	Core Card
4	CORELO	Core Card
5	Not used, driven inactive	Typically used for CPU internal timer interrupt

6.7 Serial Ports

There are 2 serial ports (tty0 and tty1) on Malta which are available on the rear edge via standard, male DB9 connectors (J6). These ports are provided by the Super I/O. A third serial port (tty2) is available via a 10-pin header (J22). This port is provided by the discrete 16550. The pinout of these connectors is shown below:

Each port is electrically identical, with the pinout shown in the table below allowing full hardware handshaking.

Table 30 Serial Port Pinouts

PIN NO (DB9)	PIN NO (10-pin)	Name	Direction
1	1	CD	Input
2	3	RXD	Input
3	5	TXD	Output
4	7	DTR	Output
5	9	GND	
6	2	DSR	Input
7	4	RTS	Output
8	6	CTS	Input
9	8	RI	Input
	10	No Connect	

The following 5-wire symmetric wired cable must be used to guarantee the correct operation of the hardware flow control which is used by YAMON. Pin connections between the two 9-pole male-connectors (for standard PC serial port):

- 2 to 3 (RXD to TXD)
- 3 to 2 (TXD to RXD)
- 5 to 5 (GND to GND)
- 7 to 8 (RTS to CTS)
- 8 to 7 (CTS to RTS)

6.7.1 File Download

A recommended data terminal program for PC's running Windows OS is Procomm Plus32 from DATASTORM TECHNOLOGIES. The setup sequence to allow simple Motorola S-record file download is:

Communication settings in the menu area:

- Options->SystemOptions->ModemConnection->System

From here, select the 'com' port and click on the 'Modem/ConnectionProperties' to set the 'baudrate'(=38400), 'parity'(=none), data bits(=8), stop bits(=1) and important select 'use hardware flow control'.

Download protocol setting in the menu area:

- Options->SystemOptions->ModemConnection->Data:

From here, set ‘current transfer protocol’ to ‘ASCII’, set all delays to ‘0’ and set the CR/LF options to ‘don’t translate CR/LF’. The file to be downloaded (to Malta) is selected via the path:

- Data->SendFile

6.7.2 Serial Port Reset

The tty0 port (J6) can be used to reset the Malta board. By default a “Break” condition on the tty0 port for more than 10 ms will reset the board exactly as if the reset button had been pressed. This functionality can be disabled or the time can be changed to a different value by programming the BRKRES register in the CBUS FPGA, see Section 4.1.6, "Reset Control".

6.8 Ethernet

The Ethernet controller (U41, see Ref[8]) supports both 10base-T and 100base-TX standard on a twisted pair connection via the rear panel connector J5. The device has an integrated PHY section and is capable of auto-negotiating the line speed/duplex with the far end. Its MAC address is stored in the locally-connected EEPROM (U39) and must not be altered.

Note: Only 10BASE-T is supported (100BASE-TX is not) for PCI clocking frequencies below 33.33 MHz, see Table 23

See Section 5.3, "Displays / LEDs" for details of the Ethernet LEDs which are built in to the RJ45 connector and the on board LEDs.

The ethernet controller supports Wake On LAN for remote wake-up, see Ref[8] for further details.

6.9 USB

Two host USB ports are available on a double connector (J5) on the rear edge of the board. These are controlled through the South Bridge (see Ref[6]).

6.10 Keyboard / Mouse / IEEE1284 Parallel Port / Floppy Disk

All these functions are provided by the Super I/O chip (U11, see Ref[9]). The PC keyboard and mouse are on a double mini-DIN connector (J7), and the parallel (1284) port on a 25-pin DSUB connector (J6) both on the rear board edge. The floppy disk connector is header J21.

The parallel port also allows the user to reprogram the Flash memory, although this will typically be done during production or software upgrades. See Section 6.13, "Flash Memory".

The Super I/O only supports one floppy disk. It is connected via a 34 wires ribbon cable (7 wires twisted, PC-style).

6.11 RTC

The South Bridge (U9) contains the Real Time Clock for the board (see Ref[6]). This has an external battery backup (CR2032 coin cell) (J10), which has an expected life time of 5-10 years.

6.12 IDE/CompactFlash (True IDE mode)

The South Bridge (U9) provides both primary and secondary IDE busses (see Ref[6]). The primary bus is taken to connector J18, and may have both master and slave devices attached. The secondary bus connects both to the Compact

Flash connector J20 (Type I or II connector) and to the secondary IDE connector J19. Jumper JP3 selects whether the Compact Flash module behaves as (if fitted) a master or (if not fitted) as a slave device. If a Compact Flash module is used any device plugged into the secondary IDE connector must have the opposite setting.

For further information of Compact Flash modules see Ref[15].

6.13 Flash Memory

Malta is fitted with 4 Mbytes of Flash memory (refered to as Monitor Flash), which the system boots from. See Section 4, "Memory Map" for details of the Malta memory map.

The Monitor Flash can be programmed via a download cable, see Section 7, "1284 Flash Download Format".

The Monitor Flash can also be reprogrammed by software. There is one protection mechanism:

- Jumper JP1 (See Section 5.1, "Connectors & Jumpers") must be installed for any writing to the Monitor Flash Lock Bits from software.

Note that while the Monitor Flash is being reprogrammed by software, the code that performs the programming will have to be copied into RAM and executed there, as the Flash will be inaccessible during this process. All the Flash fitted are Intel 16Mbit FlashFile devices. See Intel's web site for the documentation or see YAMON documentation (Ref[3]) for an easy to use software interface.

From a hardware viewpoint, the Flash appears as a 32-bit wide block, with no individual write control capability to allow writing to just one 16-bit halfword. However, this function can be achieved through software by running a Read-Modify-Write cycle.

6.14 EEPROM

The I^2C-connected EEPROM (U14) contains, on manufacture, the board serial number. The remain of the EEPROM is not available for application use.

A second EEPROM (U39) is directly connected to the Ethernet controller, and is used to store the board's MAC address.

6.15 AMR (Audio Modem Riser)

Connector J16 is an AMR connector (see Ref[14]), allowing an audio/modem interface card to be plugged into the motherboard. This is controlled via U23, a PCI Audio controller, (see Ref[7]). The AMR connector is a dual AC'97 audio codec interface.

6.16 Front Panel Connector

The Front Panel connector (J9) contains all signals to/from the front panel of a normal PC-chassis. An additional connector (J30) is available for a three pin Power LED connector. See Figure 6.

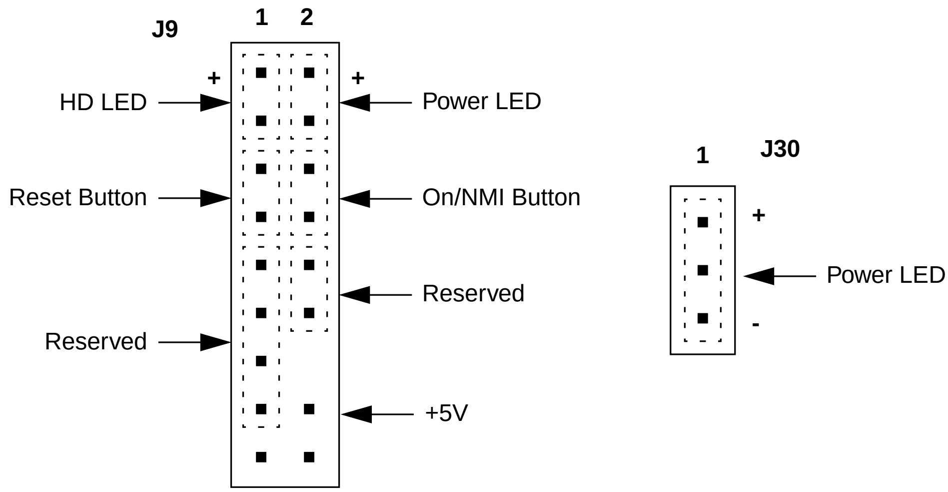


Figure 6 Front Panel Connector

6.17 Debug Access

6.17.1 Software Debug

The EJTAG connector (J17) allows connection of a suitable EJTAG debugger probe directly to the CPU. This allows access to the internal hardware debug functionality of the CPU core. See Ref[4] for details.

6.17.2 Hardware Debug

You have access to most if not all interesting signals found on the Malta board via testpoints and HP Logic Analyser high-density connectors. Refer to the tables below and the schematics for details of these.

Table 31 Testpoints

Ref	Silkscreen	Colour	Function
TP1	**D3V3SB**	**Red**	Digital 3.3V Standby
TP2	**D5VSB**	**Red**	Digital 5V Standby rail
TP3	**D3V3**	**Red**	Digital 3.3V rail
TP4	**D5V**	**Red**	Digital 5V rail
TP5	**D12V**	**Red**	Digital 12V rail
TP6	**D12VN**	**Red**	Digital -12V rail
TP7-12	**GND**	**Black**	Ground

Table 32 Logic Analyser Connectors

Ref	Function	Bits	Signals	Description
J23 Even	CBUS DATA	Clock	CLK_40MHZ	40 MHz
		15:0	CBUS_D[31:16]	CBUS data

Table 32 Logic Analyser Connectors (Continued)

Ref	Function	Bits	Signals	Description
J23 Odd	CBUS DATA	Clock	PCI_CLK	33.33 MHz (default)
		15:0	CBUS_D[15:0]	CBUS data
J24 Even	CBUS	Clock	CLK_1KHZ	1 KHz
		15:13	FPGA_RSTN, RSTN, BIGEND	CBUS FPGA signals
		12:10	CBUS_WRN, CBUS_RDN, CBUS_CSN	CBUS control
		9:0	CBUS_A[25:16]	CBUS address
J24 Odd	CBUS ADDR	Clock	CLK_32KHZ	32 KHz
		15:2	CA[15:2]	CBUS address
		1:0	GND	Used as CA[1:0]
J25 Even	CBUS FPGA	Clock	CLK_40MHZ	40 MHz
		15:12	FRSTN, PAR_ENAB, FPGA_RSTN, MFSTS	
		11:4	UCSN, REVCSN, CGPIOCSN, CGPIOWR, MFCSN, ASWCSN, ADCSN, ABRLEDCS	Chip selects
		3:0	Not used	
J25 Odd	IIC + ASCII Display	Clock	PCI_CLK	33.33 MHz (default)
		15:14	IIC_SCL, IIC_SDA	IIC signals
		13	Not Used	
		12:8	ADA[4:0]	ASCII display address
		7:0	ADD[7:0]	ASCII display data
J26 Even	RESET	Clock	CLK_1KHZ	1 KHz
		15	CLK_32KHZ	32 KHz
		14:13	DVSB_OK, SB_RSTN	Standby Reset control
		12:5	FRSTN, EJRSTN, ATX_OK, CBUS_FPGA_OK, D5V_OK, D3V3_OK, POWER_OK, CORE_OK	Reset control
		4:2	FPGA_RSTN, RSTN, RST	RESET outputs
		1	PAR_ENAB	Parallel download
		0	CPRESN	Core card present

Table 32 Logic Analyser Connectors (Continued)

Ref	Function	Bits	Signals	Description
J26 Odd	CBUS FPGA	Clock	PCI_CLK	33.33 MHz (default)
		15:2	CPU_INTN5, CPU_INTN4, CPU_INTN3, CPU_INTN2, CPU_INTN1, CPU_INTN0, CPU_NMIN, SOUTHBRIDGE_INT, SOUTHBRIDGE_NMI, SOUTHBRIDGE_SMIN, UINT, CINTHIN, CINTLON	Interrupts
		2:1	Not used	
		0	ISA_SERIRQ	Super I/O Interrupts
J27 Even	JTAG + PCI	Clock	CLK_40MHZ	40 MHz
		15:10	EJTRSTN, EJTDO, EJTDI, EJTMS, EJTCK, EJDINT	JTAG signals
		9:6	PCI_INTAN, PCI_INTBN, PCI_INTCN, PCI_INTDN	PCI Interrupts
		5:0	PCI_FRAMEN, PCI_IRDYN, PCI_TRDY, PCI_DEVSELN, PCI_STOPN, PCI_LOCKN	PCI signals
J27 Odd	ARBITER	Clock	PCI_CLK	33.33 MHz (default)
		15:8	PCI_REQN_CORE, PCI_REQN_SOUTHBRIDGE, PCI_REQN_ETHERNET, PCI_REQN_AUDIO, PCI_REQN_CON1, PCI_REQN_CON2, PCI_REQN_CON3, PCI_REQN_CON4	Requests
		7:0	PCI_GNTN_CORE, PCI_GNTN_SOUTHBRIDGE, PCI_GNTN_ETHERNET, PCI_GNTN_AUDIO, PCI_GNTN_CON1, PCI_GNTN_CON2, PCI_GNTN_CON3, PCI_GNTN_CON4	Grants

7 1284 Flash Download Format

Onboard Flash memory can be programmed and reprogrammed via a download cable that connects directly to a PC parallel port. The CBUS FPGA can read data from this port, and execute the appropriate erase/write cycles in the Flash. The PC must be configured so that its printer port is set to “Generic - text only”, to avoid unpredictable escape sequences being sent. The file format is a sequence of ASCII encoded hex bytes as described below.

How to download:

- Switch OFF both the Malta, and the PC or workstation that will be used to download.
- Connect the parallel cable between the PCs parallel port, and J6 on the Malta.
- Switch S5-1 on the Malta to ON or CLOSED.
- Switch both PC and Malta ON.
- Run the download script to dump the file to the parallel port on the PC or workstation.
- Switch S5-1 to OFF, and reset Malta.
- Disconnect the parallel cable.

According to the memory map in Section 4, "Memory Map" the Monitor Flash is programmed on base address 0x1E00.0000 or 0x1FC0.0000. If any address outside the Flash is addressed the attempt will be ignored.

Note: When programming the address 1FC0.0010 it is the Monitor Flash that is being programmed, but when reading the address it is overridden and does NOT decode to an address in Flash, but rather to register address REVISION. The only way to read the programmed value back is to read the “non-boot” Monitor Flash address, ie. address 0x1E00.0010. See Section 4, "Memory Map".

The Flash devices are organised in sectors of 64 Kbyte. “Erase” and “Set Lock Bit” commands operate on exactly one sector, this being the sector currently addressed. After the last block of 16 words in a sector are written into flash, the address counter has advanced to the next sector. This implies that a Set Address (@) to the sector has to be executed before a Set Lock Bit command (!S) can be issued.

The file to be loaded into the Flash via the 1284 port has the following format:

Type:	ASCII hex (both lower- and uppercase letters are accepted)..
White space:	Any characters below or equal to 20h are ignored (character 20h (space) is allowed in the Print command. After the character 1Bh (start of a printer initialisation command) it ignores any character until next Reset Command.
Word width:	32 bits (data has to be in blocks of 16 words and has to be placed on 16 word boundaries).

A number of codes are used to control code download and Flash memory handling:

Table 33 Download Codes

Code	Meaning
@	Sets current writing/erasing address (in physical memory map format)
!R	Reset download system
!E	Erase the current Flash sector (64 KB)
!C	Clear all Flash lock-bits
!S	Set current Flash sector lock-bit

Table 33 Download Codes (Continued)

Code	Meaning
#	Comment (rest of line)
>	Print command (shows next 8 characters in ASCII display, the command needs exactly 8 non-white space characters). Any character except for "!" and "#" may be printed - use of these in the print command is reserved.
data	data has to be in blocks of 16 words, without interruption of any Comments (#) and Print Commands (>)

Example of code download format:

```
#Example
!R
@1E000000
!E
12345678 23456789 3456789A 456789AB
56789ABC 6789ABCD 789ABCDE 89ABCDEF
9ABCDEF0 ABCDEF01 BCDEF012 CDEF0123
DEF01234 EF012345 F0123456 01234567
# always 16 words in a block
```

After a Reset it will start at 1E00.0000 (i.e. base of Monitor Flash), erase the base sector, and then write the 16 words into offset 0.

If an error should occur during Flash download the ASCII display will show an error message:

Table 34 Flash Download Error Messages

Message	Meaning
Ill cmd	Illegal command received, eg. “R” is received (not !R)
Ill !cmd	Illegal “!” command received, eg. “!A” is received
Ill hex	Illegal hex received (in data or addr), eg. “ABCDEFGH” both “G” and “H” is illegal characters
Hex exp	Hex expected (always data blocks of 16 words), eg. a comment (#) is received in the middle of a block of 16 words.
Era susp	Block erase suspended
Err era	Error in block erasure or clear lock-bits
Err prog	Error in programming or set block lock-bits
Low volt	Low programming voltage detected
Lock det	Master lock-bit, Block lock-bit or RP# lock detected

8 Core Card design

This section details the external specification that all Core cards must comply with.

8.1 Interfaces

The following features shall be present on the interface to the Core card.

8.1.1 Power

Power supplies at 3.3V, 5V and 12V are present on the interface. These are positioned such that if a Core card is placed on the motherboard 180 degrees incorrect, all rails are shorted out by a number of pins. This should place the PSU in shutdown mode.

It is not guaranteed that the 5V rail will be present before the 3.3V rail.

8.1.2 PCI Bus

The interface to the core card includes a PCI bus. All core cards shall be 5V tolerant on inputs but drive 3.3V on all outputs on the PCI bus.

8.1.3 Clock

The PCI bus clock is driven to the Core card from the motherboard. The Core card shall be able to run with this clock at any frequency from 0-33MHz.

8.1.4 Revision number.

The Core card drives a processor-readable revision number, via pins CREV[7:0], down to the motherboard where the CPU will be able to read them via the CBUS.

This revision number could for example be set via 8 fit/not-fit resistors.

8.1.5 I2C bus

An I2C bus is present on the interface. This will typically be used for interrogating SDRAM DIMMs. See User Manual for the address map.

8.1.6 Interrupts

Six interrupt signals to the MIPS CPU on the Core card are present, as is a single NMI interrupt signal, triggered by a front-panel pushbutton. Core cards must route all of these interrupts to the CPU - if the CPU chip has fewer external interrupt pins, then they should be ORed together.

8.1.7 Endian

The endian control signal BIGEND, driven by the motherboard according to setting of S5-2.

8.1.8 CBUS

The CBUS is designed to interface to simple devices on the the motherboard which it is necessary that the CPU can access either before the PCI is up and running, or with a low latency (e.g. interrupt controller and Flash).

All core cards shall decode CPU addresses from 0x1C00.0000 to 0x1FFF.FFFF. These address shall then translate into addresses 0x0000.0000 to 0x03FF.FFFF on the CBUS.

Bus signals are:

- CA[25:2] Address bus.
- CD[31:0] Data bus
- CCSN Chip select.
- CWRN Write strobe.
- CRDN Read strobe.

All CBUS signals must use 3.3 volt signalling levels. Read and write cycles are illustrated in the following figures:

Figure 7 CBUS Read cycle

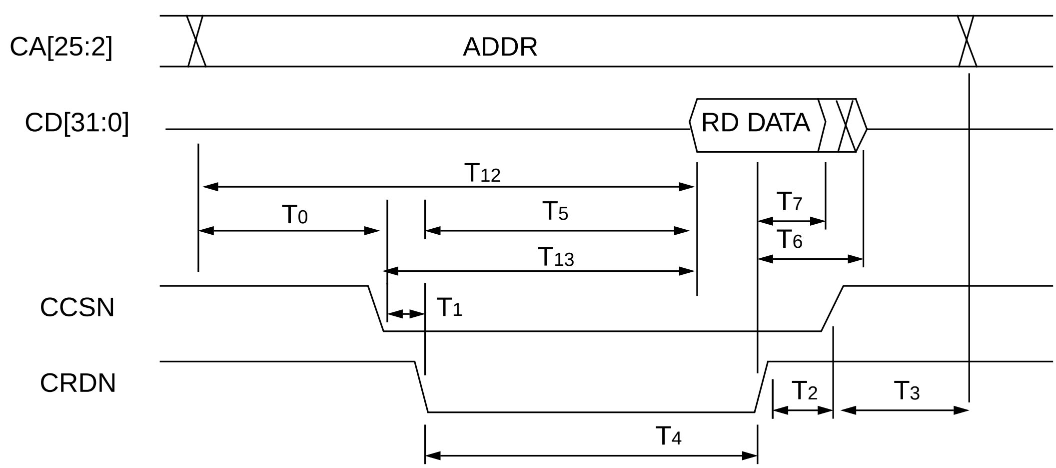


Figure 8 CBUS Write cycle

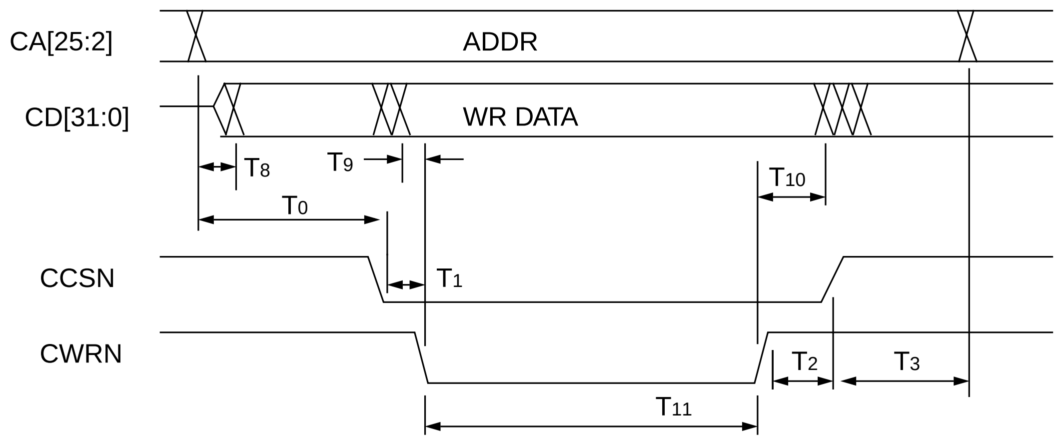


Table 35 CBUS AC timing parameters

Ref	Description	Tmin(ns)	Tmax(ns)
T0	Address valid to CCSN active	5	-
T1	CCSN valid to strobe (CRWN or CRDN) valid	10	-
T2	strobe (CRWN or CRDN) inactive to CCSN inactive	10	-
T3	Address hold from CCSN inactive	20	-
T4	CRDN width	120	-
T5	CRDN active to read data valid	-	120
T6	CRDN inactive to data bus tristated	-	20
T7	Read data hold time after CRDN inactive	0	-
T8	Address valid to data bus driven	0	-
T9	Write data setup to CRWN active	0	-
T10	Write data hold time after CWRN inactive	10	-
T11	CWRN pulse width	75	-
T12	Address valid to read data valid	-	150
T13	CCSN active to read data valid	-	150

8.1.9 EJTAG

The EJTAG signals from the “basic” EJTAG connector are taken to the interface from the front panel connector.

8.1.10 Misc.

Various debug, reserved and presence detect functions - see below.

8.2 Signals.

These signals are carried on J3 and J4, which are 200 pin (4-row) 1.27mm pitch connectors, SAMTEC type MOLC-150-31-S-Q (male, fitted to Core card). The interface signal list is as follows:

Table 36 Core card interface signals

Signal	Direction (from Core card)	Pull (on Core)	Pull (on motherboard)	Description
INTN[5:0]	Input	up	-	Interrupt signals to CPU.
NMIN	Input	up	-	NMI signal to CPU
CD[31:0]	I/O	-	-	CBUS data bus
CCSN	Output	-	-	CBUS chip select
CA[25:2]	Output	-	-	CBUS Address
CRDN	Output	-	-	CBUS read strobe
CWRN	Output	-	-	CBUS write strobe
SCK	I/O	-	-	IIC bus clock
SDA	I/O	-	-	IIC bus data
CINTHIN	Output	-	up	Core card interrupt signal down to the motherboard (high priority).
CINTLON	Output	-	up	Core card interrupt signal down to the motherboard (low priority).
CREV[7:0]	Output	-	-	Indicates Core card revision as a 6.2 bit binary number
BIGEND	Input	-	-	Sets CPU Endianness.
EJTCK	Input	-	down	
EJTMS	Input	-	up	
EJTRSTN	Input	-	down	
EJTDI	Input	-	up	
EJTDO	Output	-	-	
EJDINT	Input	-	down	
JTGCPU	Input	-	down	Sets Core card's JTAG output to come direct from the CPU rather than also via other circuits.
CGPI[7:0]	Output	-	-	General purpose output from Core which can be read from the motherboard.
CGPO[7:0]	Input	-	-	General purpose input to Core which the motherboard can drive.
CUU[15:0]	-	-	-	Unused pins, connected to wire-wrap area on both Core and the motherboard.
CPRESN	Output	strong down	up	Wired to zero, to indicate presence of Core card.
APRESN	Input	up	strong down	Wired to zero on the motherboard, to indicate attachment.
D12V	Input	-	-	Twelve volt power for possible fan
D5V	Input	-	-	5 Volt power
D3V3	Input	-	-	3.3 Volt power
CPWR_OK	Input	-	-	Indicates that power on both 3V3 and 5V rails is up.
CORE_OK	Output	-	up	Indicates that Core is ready to come out of reset.
RSTN	Input	-	-	Global reset signal.
PCI_AD[31:0]	I/O	-	-	PCI bus

Signal	Direction (from Core card)	Pull (on Core)	Pull (on motherboard)	Description
PCI_DEVSELN	I/O	-	-	PCI bus
PCI_CBEN[3:0]	I/O	-	-	PCI bus
PCI_REQN	Output	-	-	PCI bus
PCI_GNTN	Input	-	-	PCI bus
PCI_SERRN	I/O	-	-	PCI bus
PCI_FRAMEN	I/O	-	-	PCI bus
PCI_IRDYN	I/O	-	-	PCI bus
PCI_IDSEL	I/O	-	-	PCI bus
PCI_PAR	I/O	-	-	PCI bus
PCI_STOPN	I/O	-	-	PCI bus
PCI_CLK	Input	-	-	PCI bus
PCI_TRDYN	I/O	-	-	PCI bus
PCI_LOCKN	I/O	-	-	PCI bus
PCI_PERRN	I/O	-	-	PCI bus

8.2.1 J3 Connector

The following pin layout is used for J3.

Table 37 J3 pin listing

1	CPWR_OK	51	GND	101	D3V3	151	SCK
2	JTGCPU	52	D5V	102	GND	152	CGPI4
3	CINTHIN	53	GND	103	GND	153	SDA
4	CORE_OK	54	CINTLON	104	GND	154	CGPO7
5	CGPI7	55	GND	105	D3V3	155	CGPO6
6	CD31	56	D5V	106	GND	156	CGPO5
7	CD30	57	GND	107	GND	157	CGPO4
8	CD29	58	GND	108	GND	158	INTN5
9	CD28	59	GND	109	D3V3	159	INTN4
10	CD27	60	D5V	110	GND	160	INTN3
11	CD26	61	GND	111	CGPI0	161	INTN2
12	CD25	62	GND	112	GND	162	INTN1
13	CD24	63	GND	113	D3V3	163	INTN0
14	CD23	64	D5V	114	CA6	164	NMIN
15	CD22	65	GND	115	GND	165	CA10
16	CD21	66	GND	116	GND	166	CA9
17	CD20	67	GND	117	CA5	167	CA8
18	CD19	68	D5V	118	GND	168	CA7
19	CD18	69	GND	119	GND	169	CGPI3
20	CD17	70	GND	120	CA4	170	D12V
21	CD16	71	GND	121	D3V3	171	D12V

22	CD15	72	D5V	122	GND	172	CGPI2
23	CD14	73	GND	123	CA3	173	CPRESN
24	CD13	74	GND	124	GND	174	APRESN
25	CD12	75	GND	125	D3V3	175	BIGEND
26	CD11	76	D5V	126	CA2	176	CGPO3
27	CD10	77	GND	127	GND	177	CGPO2
28	CD9	78	GND	128	GND	178	CGPO1
29	CD8	79	GND	129	CA24	179	CGPO0
30	CD7	80	D5V	130	GND	180	CGPI1
31	CD6	81	GND	131	CA25	181	EJTCK
32	CD5	82	D5V	132	GND	182	GND
33	CD4	83	GND	133	D3V3	183	CREV0
34	CD3	84	GND	134	GND	184	EJTMS
35	CD2	85	GND	135	GND	185	CREV1
36	CD1	86	D5V	136	GND	186	EJTDI
37	CD0	87	GND	137	D3V3	187	CREV2
38	CA23	88	GND	138	GND	188	EJTDO
39	CWRN	89	GND	139	GND	189	CREV3
40	CRDN	90	D5V	140	GND	190	EJTRSTN
41	CCSN	91	GND	141	D3V3	191	CREV4
42	CGPI6	92	GND	142	GND	192	EJDINT
43	CGPI5	93	GND	143	GND	193	CREV5
44	CA22	94	D5V	144	GND	194	CREV6
45	CA21	95	GND	145	D3V3	195	CREV7
46	CA20	96	GND	146	GND	196	CA15
47	CA19	97	GND	147	GND	197	CA14
48	CA18	98	D5V	148	GND	198	CA13
49	CA17	99	GND	149	D3V3	199	CA12
50	CA16	100	CUU4	150	GND	200	GA11

8.2.2 J4 Connector

The following pin layout is used for J4

Table 38 J4 pin listing

1	RSTN	51	GND	101	D5V	151	GND
2	PCI_AD31	52	GND	102	GND	152	NC
3	PCI_AD30	53	GND	103	D3V3	153	GND
4	PCI_AD29	54	GND	104	D3V3	154	NC
5	PCI_AD28	55	GND	105	GND	155	GND

6	GND	56	GND	106	D5V	156	NC
7	GND	57	GND	107	GND	157	GND
8	PCI_CLK	58	GND	108	D3V3	158	NC
9	GND	59	GND	109	D3V3	159	GND
10	GND	60	GND	110	GND	160	NC
11	PCI_PAR	61	GND	111	D5V	161	GND
12	PCI_FRAMEN	62	GND	112	GND	162	NC
13	PCI_IRDYN	63	GND	113	D3V3	163	GND
14	PCI_AD27	64	GND	114	D3V3	164	NC
15	PCI_AD26	65	GND	115	GND	165	GND
16	PCI_AD25	66	GND	116	D5V	166	NC
17	PCI_AD24	67	PCI_AD4	117	GND	167	GND
18	PCI_CBEN3	68	GND	118	D3V3	168	NC
19	PCI_CBEN2	69	GND	119	D3V3	169	GND
20	PCI_CBEN1	70	GND	120	GND	170	NC
21	PCI_CBEN0	71	GND	121	D5V	171	GND
22	PCI_AD23	72	PCI_AD3	122	GND	172	NC
23	PCI_AD22	73	GND	123	D3V3	173	GND
24	PCI_AD21	74	GND	124	D3V3	174	NC
25	PCI_AD20	75	GND	125	GND	175	GND
26	PCI_AD19	76	GND	126	D5V	176	NC
27	PCI_AD18	77	PCI_AD2	127	GND	177	GND
28	PCI_AD17	78	GND	128	D3V3	178	NC
29	PCI_AD16	79	GND	129	D3V3	179	GND
30	PCI_TRDYN	80	GND	130	GND	180	NC
31	PCI_STOPN	81	GND	131	D5V	181	GND
32	PCI_LOCKN	82	PCI_AD1	132	GND	182	NC
33	PCI_IDSEL	83	GND	133	D3V3	183	GND
34	PCI_DEVSELN	84	GND	134	D3V3	184	NC
35	PCI_AD15	85	GND	135	GND	185	GND
36	PCI_AD14	86	GND	136	D5V	186	NC
37	PCI_AD13	87	PCI_AD0	137	GND	187	GND
38	PCI_AD12	88	GND	138	D3V3	188	NC
39	PCI_AD11	89	GND	139	D3V3	189	GND
40	PCI_REQN	90	GND	140	GND	190	NC
41	PCI_GNTN	91	GND	141	D5V	191	GND
42	PCI_PERRN	92	GND	142	GND	192	NC
43	PCI_SERRN	93	GND	143	D3V3	193	GND
44	NC	94	GND	144	D3V3	194	NC
45	PCI_AD10	95	GND	145	GND	195	GND
46	PCI_AD9	96	GND	146	D5V	196	NC
47	PCI_AD8	97	GND	147	GND	197	GND
48	PCI_AD7	98	GND	148	D3V3	198	NC

49	PCI_AD6	99	GND	149	D3V3	199	GND
50	PCI_AD5	100	TMS	150	TDO	200	NC

8.3 Physical design

The core card is 170mm x 100mm, and is mounted by pillars at each corner, plus 2 x 200-way (4 row x 25 pin) 1.27mm pitch connectors of type Samtec MOLC-150-31-x-Q. See Figure 9.

Pin numbering on these connectors is as follows:

Figure 9 J3 and J4 alignment

Note that one corner pillar (top left in diagram) is placed offset from a symmetrical position, which is to guarantee the board cannot be inserted the wrong way around.

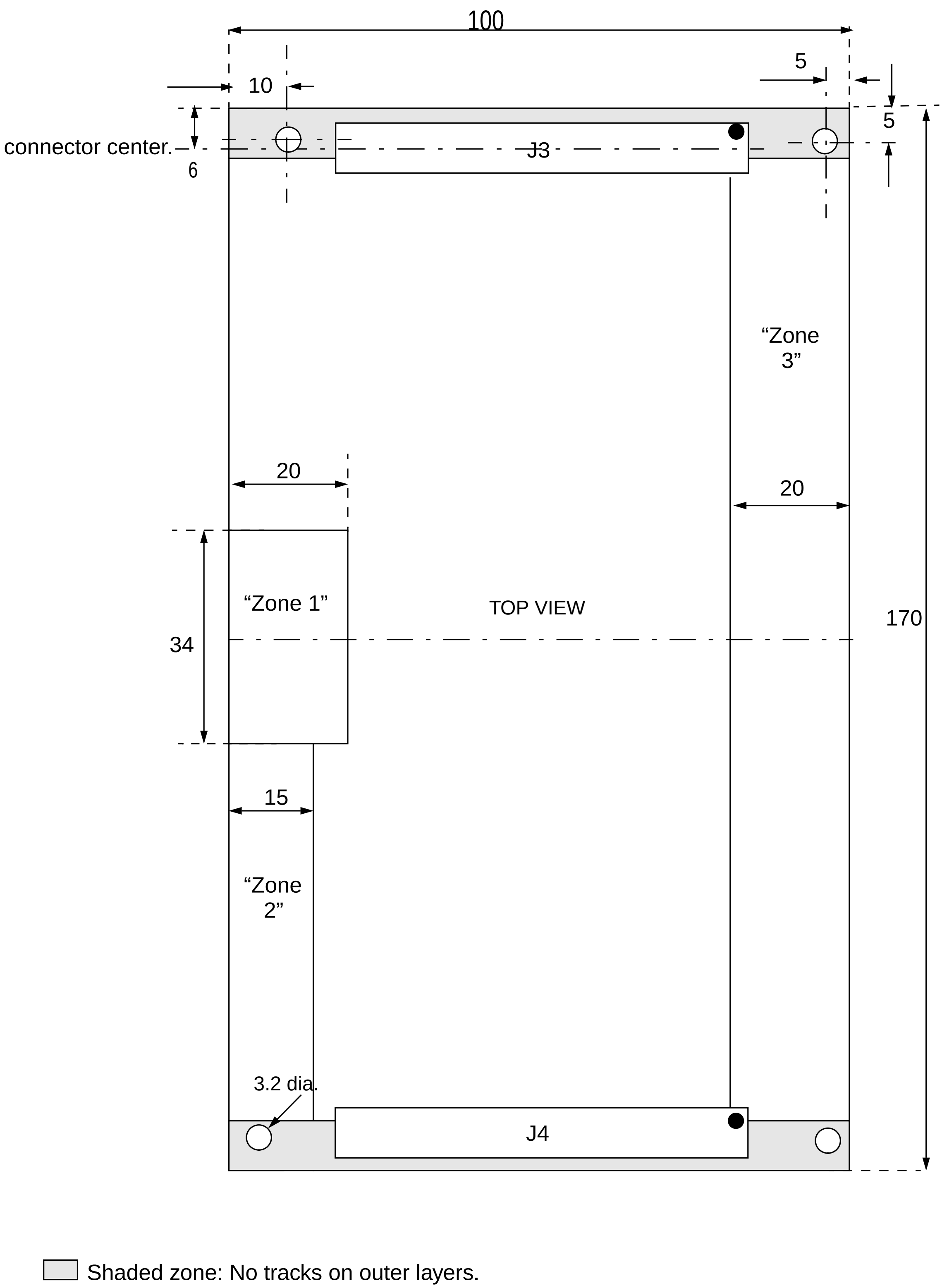
The connectors chosen are low-insertion force ones, however the intention is that the user should be able to “lever” the card up by placing a screwdriver between the mounting pillars and the card. Therefore tracking is forbidden in the areas at the ends of the card.

The Core card is mounted at a height of 11mm over the motherboard when using the connectors given above. However, the existing placement of high components on the Atlas and Malta motherboards gives the following restrictions when placing components on the Core board:

Table 39 Core card component height restrictions

Location	Height restrictions on underside
Whole card (default)	No underside SMDs thicker than 6.5mm
Zone 1	No leaded components at all. No underside SMDs thicker than 1.2mm. (Note: this requirement applies only to Core cards that must be compatible with Atlas. Malta does not have this restriction.)
Zone 2	No underside SMDs thicker than 4.4mm
Zone 3	No underside SMDs thicker than 4.4mm

Figure 10 Core card template layout



Appendices

A References

[1] PCI Local Bus Specification.
Revision 2.2. December 18, 1998

[2] MIPS YAMON™ User's Manual
MD00008

[3] MIPS YAMON™ Reference Manual
MD00009

[4] MIPS EJTAG Specification.
MD00047

[5] MIPS Atlas User's Manual
MD00005

[6] Intel 82371EB (PIIX4E) South Bridge chipset datasheets.

[7] Crystal CS4281 Audio controller Datasheet

[8] AMD 79C973 Ethernet Controller Datasheet

[9] SM$_S$C FDC37M817 Super I/O Controller datasheet.

[10] Texas Instruments TI16C550C UART datasheet.

[11] Intel ATX Power Supply Design Guide. Version 0.9

[12] ATX Specification 2.03

[13] Design Guide for Intel ATX Motherboard I/O Implementations. Version 1.1

[14] Audio/Modem Riser Specification. Revision 1.01

[15] CF+ and CompactFlash Specification. Revision 1.4

B Revision History

Revision	Date	Description
01.00	2000/08/29	Initial release.
01.01	2000/11/27	Updated Compact Flash connector information. Added defintion of connectors J30-31
01.02	2001/01/25	Layout updated
01.03	2001/06/13	Added Section 8 - Core card specification.
01.04	2001/07/31	Added 3.3V specification to CBUS spec.
01.05	2002/06/10	Fixed Figure 3 with reference to keyboard and mouse connectors swapped.