Integrator/IM-PD1

User Guide

ARM

ARM DUI 0152E

Integrator/IM-PD1
User Guide

Release Information

Date	Issue	Change
June 2001	A	Initial issue
July 2001	B	Corrections to Table 3-3 on page 3-7.
July 2001	C	Corrections to Appendix A.
December 2001	D	Minor corrections to Chapter 3 Changes to Chapter 4
December 2003	E	Minor corrections to Chapter 3

Proprietary Notice

Confidentiality Status

This document is Open Access. This document has no restriction on distribution.

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

Conformance Notices

This section contains conformance notices.

Federal Communications Commission Notice

This device is test equipment and consequently is exempt from part 15 of the FCC Rules under section 15.103 (c).

CE Declaration of Conformity

CE

The system should be powered down when not in use.

The Integrator generates, uses, and can radiate radio frequency energy and may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment causes harmful interference to radio or television reception, which can be determined by turning the equipment off or on, you are encouraged to try to correct the interference by one or more of the following measures:

- ensure attached cables do not lie across the card
- reorient the receiving antenna
- increase the distance between the equipment and the receiver
- connect the equipment into an outlet on a circuit different from that to which the receiver is connected
- consult the dealer or an experienced radio/TV technician for help

Note

It is recommended that wherever possible Shielded interface cables be used.

Contents
Integrator/IM-PD1 User Guide

Preface

This preface introduces the Integrator/IM-PD1 interface module and its user documentation. It contains the following sections:

About this book

This book provides user information for the ARM® Integrator/IM-PD1 interface module. It describes the major and how to use the interface module with an Integrator development platform.

Intended audience

This book is written for all developers who are using the Integrator/IM-PD1 interface module with an Integrator/LM-XCV600E+ or LM-EP20K600E+ logic module to develop ARM-based devices. It assumes that you are an experienced developer, and that you are familiar with the ARM development tools.

Using this book

This book is organized into the following chapters:

Chapter 1 ***Introduction***

Read this chapter for an introduction to the Integrator/IM-PD1 interface module. This chapter describes the main features of the interface module and identifies the main components.

Chapter 2 ***Getting Started***

Read this chapter for information about preparing the interface module for use with a logic module.

Chapter 3 ***Hardware Reference***

Read this chapter for a description of the interface module hardware.

Chapter 4 ***Reference Design Example***

Read this chapter for a description of the example logic module configuration supplied that allows you to experiment with the interface module.

Appendix A ***Signal Descriptions***

Read this appendix for connector pinout information.

Appendix B ***Mechanical Specification***

Refer to this appendix for mechanical details of the Integrator/IM-PD1.

Typographical conventions

The following typographical conventions are used in this book:

italic	Highlights important notes, introduces special terminology, denotes internal cross-references, and citations.
bold	Highlights interface elements, such as menu names. Denotes ARM processor signal names. Also used for terms in descriptive lists, where appropriate.
`monospace`	Denotes text that can be entered at the keyboard, such as commands, file and program names, and source code.
`mono`space	Denotes a permitted abbreviation for a command or option. The underlined text can be entered instead of the full command or option name.
`monospace italic`	Denotes arguments to commands and functions where the argument is to be replaced by a specific value.
`monospace bold`	Denotes language keywords when used outside example code.

Further reading

This section lists publications from both ARM Limited and third parties that provide additional information on developing code for the ARM family of processors.

ARM periodically provides updates and corrections to its documentation. See `http://www.arm.com` for current errata sheets and addenda.

See also the ARM Frequently Asked Questions list on the ARM website.

ARM publications

The following documents provide information about related Integrator products:

- *ARM Integrator/AP User Guide* (ARM DUI 0098)
- *ARM Integrator/ CM920T-ETM User Guide* (ARM DUI 0149)
- *ARM Integrator/CM9x0T and CM7x0T User Guide* (ARM DUI 0157)
- *ARM Integrator/CM7TDMI User Guide* (ARM DUI 0126)
- *Integrator/CM946E-S Integrator/CM966E-S User Guide* (ARM DUI 0138).
- *ARM Integrator/LM-XCV600E+ LM-EP20K600E+ User Guide* (ARM DUI 0146)

The following publication provide information about ARM PrimeCell devices that can be used to control the interfaces described in this manual:

- *ARM PrimeCell UART (PL011) Technical Reference Manual* (ARM DDI 0183*)*
- *ARM PrimeCell Synchronous Serial Port Master and Slave (PL022) Technical Reference Manual* (ARM DDI 0171)
- *ARM PrimeCell Advanced Audio CODEC Interface (PL041) Technical Reference Manual* (ARM DDI 0173).
- *ARM PrimeCell GPIO (PL061) Technical Reference Manual* (ARM DDI 0187)
- *ARM PrimeCell Color LCD Controller (PL110) Technical Reference Manual* (ARM DDI 0161).
- *ARM PrimeCell Smartcard Interface (PL130) Technical Reference Manual* (ARM DDI 0148)
- *ARM PrimeCell Vectored Interrupt Controller (PL190) Technical Reference Manual* (ARM DDI 0181)
- *ARM PrimeCell Multimedia Card Interface (PL181) Technical Reference Manual* (ARM DDI 0205).

The following publications provide reference information about ARM architecture:

- *AMBA Specification* (ARM IHI 0011)
- *ARM Architectural Reference Manual* (ARM DDI 0100).

The following publications provide information about the ARM Developer Suite:

- *Getting Started* (ARM DUI 0064)
- *ADS Tools Guide* (ARM DUI 0067)
- *ADS Debuggers Guide* (ARM DUI 0066)
- *ADS Debug Target Guide* (ARM DUI 0058)
- *ADS Developer Guide* (ARM DUI 0056)
- *ADS CodeWarrior IDE Guid*e (ARM DUI 0065).

The following publication provides information about Multi-ICE:

- *Multi-ICE User Guide* (ARM DUI 0048).

Other publications

This section lists relevant documents published by third parties:

- *LM4549 AC '97 Rev 2.1 Codec with Sample Rate Conversion and National 3D Sound Data sheet* (DS101035) available at:

  `http://www.national.com/pf/LM/LM4549.html`

- *IRMS6400 and IRMT6400 4 Mb/s Infrared Data Transceiver Data sheet* (IRMS6400/IRMT6400) available at:

http://www.infineon.com/cmc_upload/0/000/019/200/IRMS_T6400.pdf

- *PDIUSBP11A Universal Serial Bus Transceiver Data sheet* (853-2008 21712) available at:

  http://www-us6.semiconductors.com/acrobat/data sheets/PDIUSBP11A_2.pdf

- *DAC-Controlled Boost/Inverter LCD Bias Supply with Internal Switch Data sheet* (19-1327) available at:

  http://pdfserv.maxim-ic.com/arpdf/MAX686.pdf

Feedback

ARM Limited welcomes feedback on both the Integrator/IM-PD1 and its documentation.

Feedback on this document

If you have any comments on this book, please send email to `errata@arm.com` giving:

- the document title
- the document number
- the page number(s) to which your comments apply
- a concise explanation of your comments.

General suggestions for additions and improvements are also welcome.

Feedback on the Integrator/IM-PD1

If you have any comments or suggestions about this product, please contact your supplier giving:

- the product name
- an explanation of your comments.

Chapter 1
Introduction

This chapter introduces the Integrator/IM-PD1. It contains the following sections:

1.1 About the Integrator/IM-PD1

The Integrator/IM-PD1 is an interface module that is designed to be used in conjunction with the Integrator/LM-XCV600E+ or LM-EP20K600E+ and future compatible logic modules. It provides standard interfaces to enable you to make external connections to PrimeCell™ or your own peripherals implemented in the logic module FPGA.

Figure 1-1 on page 1-3 shows the layout of the Integrator/IM-PD1.

Circuit diagrams of the Integrator/IM-PD1 and third party data sheets are available as pdf files after installation from the CDROM.

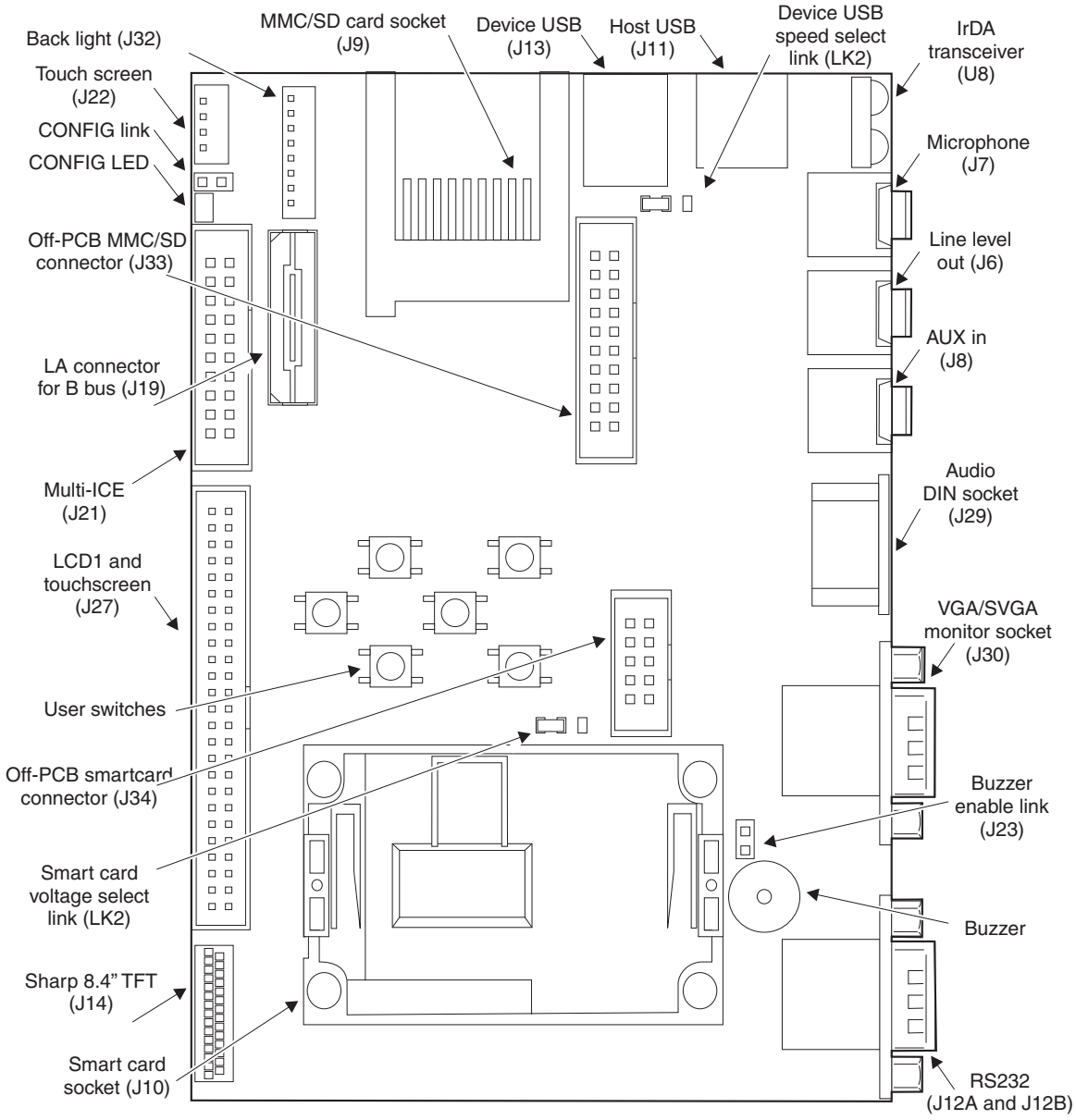


Figure 1-1 Integrator/IM-PD1 layout

1.2 Interface module features and architecture

This section describes the main features of the interface module and its architecture.

1.2.1 Features

The main features of the interface module are as follows:

- display support:
  - interface to 8.4 inch Sharp color full VGA LCD
  - generic interface to LCD with touch screen
  - video DAC to support the connection of a VGA or SVGA PC monitor.
- USB type A (host) and type B (device) interfaces
- audio CODEC
- combined *MultiMedia Card* (MMC) and SD card interface
- smartcard socket
- two serial RS232 transceivers
- IrDA transceiver
- Multi-ICE connector
- logic analyzer connector connected to the B bus
- six push buttons
- buzzer.

1.2.2 Architecture

Figure 1-2 on page 1-6 shows the architecture of the interface module. The routing of the various interface signals from the logic module is as follows:

- The peripheral input/output devices are connected to the FPGA on the logic module using the EXPIM socket that connects to EXPIM plug on the logic module.
- The display interfaces are connected to the FPGA on the logic module using the B bus pins on the EXPA socket and one F bus pin on the EXPB socket. The logic module FPGA supplies the pixel data and control signals for the display interface buffers. The B bus can be monitored with a logic analyzer connected to J19.

  Note

  If the logic module is mounted in the EXPA/EXPB position on an Integrator/AP, the pins marked F bus connect to the GPIO bus on the Integrator/AP. This bus is routed between the system controller FPGA on the motherboard and the FPGA on the logic module. These signals are available for your own applications.

  If the logic module is mounted in the HDRA/HDRB position on the motherboard, these pins connect to the F bus that is routed between any modules in the stack. there are no signals from the motherboard present on these pins.

- The Multi-ICE connector enables you to gain access to the JTAG signals on the modules in the stack on which the interface module is mounted.

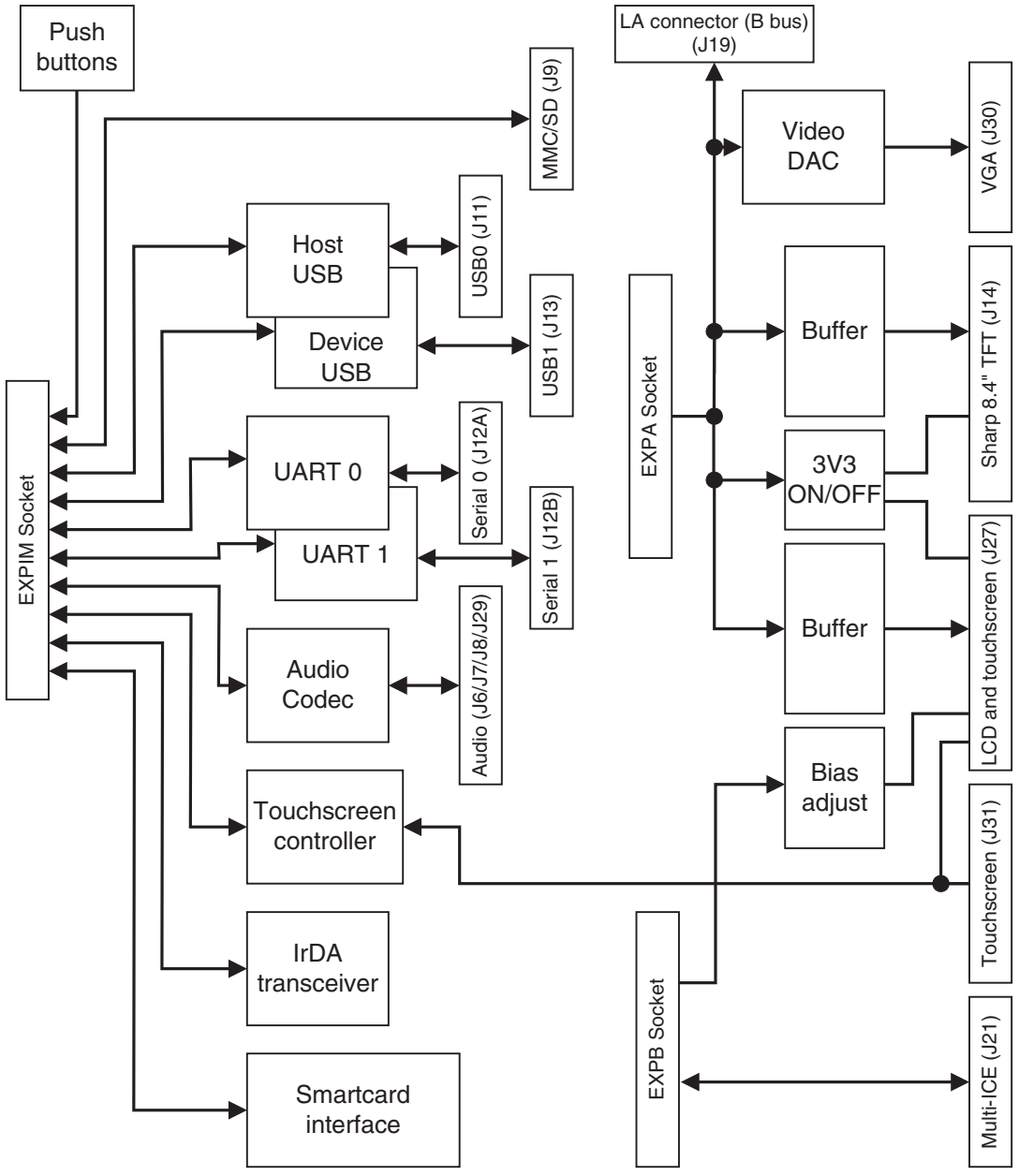


Figure 1-2 Integrator/IM-PD1 block diagram

1.3 Links

The interface module provides four links:

- *CONFIG link J22*
- *Buzzer enable link J23*
- *USB device port speed select link LK1*
- *Smartcard voltage select link LK2*.

1.3.1 CONFIG link J22

The CONFIG link is a jumper type link that is used to enable and disable config mode.

Fitting the CONFIG link places the modules in the stack, onto which the interface module is mounted, into CONFIG mode (there are no components on the interface module that use Multi-ICE). This mode enables you to reprogram the FPGA image in the configuration flash on the logic module(s) in the stack using Multi-ICE (see the user guide for the logic module).

The CONFIG LED lights to indicate that the stack is in CONFIG mode.

1.3.2 Buzzer enable link J23

The buzzer enable link is a jumper type link used to connect and disconnect the buzzer (see *Buzzer* on page 3-25).

1.3.3 USB device port speed select link LK1

The USB speed select link is a soldered link that is used to set the operating speed of the USB device port (see *USB interface* on page 3-10).

1.3.4 Smartcard voltage select link LK2

The smartcard voltage select link is a soldered link that is used to set the operating voltage of the smartcard interface (see *Smart card interface* on page 3-3).

1.4 Care of modules

This section contains advice about how to prevent damage to your Integrator modules.

Caution

To prevent damage to your Integrator system, observe the following precautions:

- When removing a core or logic module from a motherboard, or when separating modules, take care not to damage the connectors. Do not apply a twisting force to the ends of the connectors. Loosen each connector first before pulling on both ends of the module at the same time.
- Use the system in a clean environment and avoid debris fouling the connectors on the underside of the PCB. Blocked holes can cause damage to connectors on the motherboard or module below. Visually inspect the module to ensure that connector holes are clear before mounting it onto another board.
- Observe *ElectroStatic Discharge* (ESD) precautions when handling any Integrator board.

Chapter 2
Getting Started

This chapter describes how to set up and start using the logic module. It contains the following sections:

2.1 Setting up the logic module

Before the interface module can be used it is necessary to load the required peripheral controllers into the logic module FPGA to drive the interfaces. The interface module is supplied with an example configuration that provides PrimeCell peripherals for supported logic modules.

The logic module user guide describes how to download new FPGA configurations.

When the interface module is fitted to the logic module, there is no access to the manufacturer-specific FPGA programming tool connector. This means that the logic module FPGA must be configured from flash or directly using the Multi-ICE connector if the logic module supports direct Multi-ICE configuration.

2.2 Fitting the interface module

The interface module is designed to be mounted on top of a logic module and provides connectivity for peripherals in the logic module FPGA. The interface module can be installed at the top of a stack of up to four logic modules. However, it only provides interface connections for the logic module immediately beneath it.

Figure 2-1 shows an example system comprising a core module and logic module attached to an Integrator/AP (see the *Integrator/AP User Guide* for more details) with interface module installed on top of the logic module.

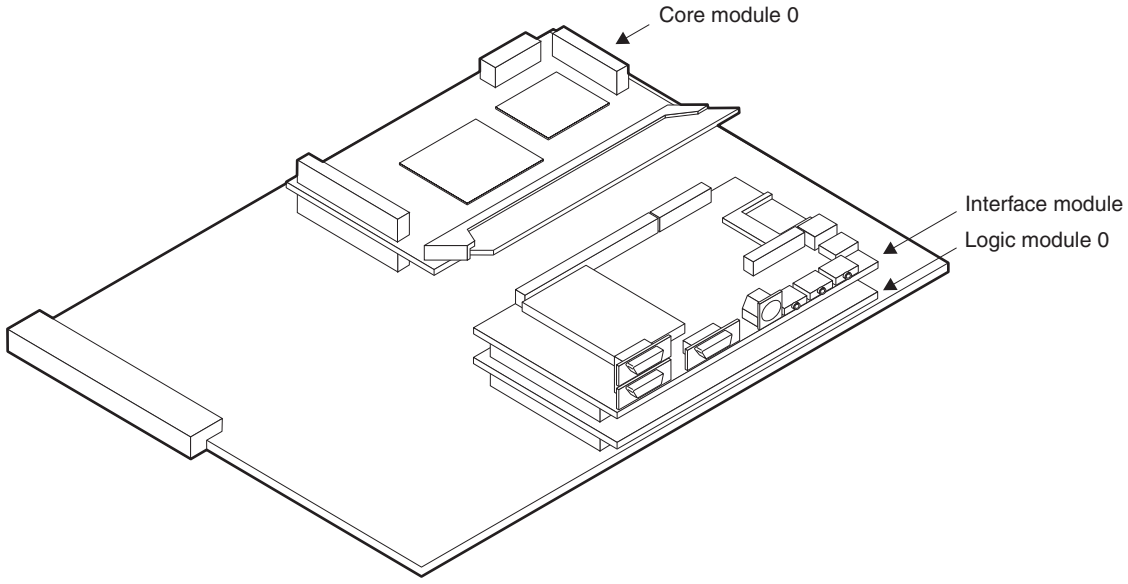


Figure 2-1 Assembled Integrator/AP development system

2.2.1 Using the Integrator/IM-PD1 without an Integrator/AP motherboard

This option uses a core module at the bottom of a stack of one or more other modules. One logic module must be included that provides the system control function (for example, a system bus arbiter) normally provided by the motherboard.

Note

Module stacking without a motherboard is supported by later core module types that have a link similar to LK3 on the logic module. At the time of publication supporting core modules are:

- Integrator/CM9x6E-S (rev C and later)
- Integrator/CM9x0T-ETM (rev C and later)
- Integrator/CM10200 (rev C and later).

For up to date information about core module support for this stacking option, refer to the ARM web site.

To use this option:

- on the core module at the bottom of the stack, set the link to the appropriate position (see the user guide for your core module).
- on any logic modules, set LK3 to the C-D position.
- on one logic module, program and enable the CLK2 clock generator (see *ARM Integrator/LM-XCV600E+ LM-EP20K600E+ User Guide*).

2.3 Connecting Multi-ICE or other JTAG equipment

JTAG equipment, such as Multi-ICE, is connected to the 20-way box header, as shown in Figure 2-2. Connect the JTAG equipment to the interface module at the top of the logic module stack. Refer to the logic module user guide for a description of the JTAG system.

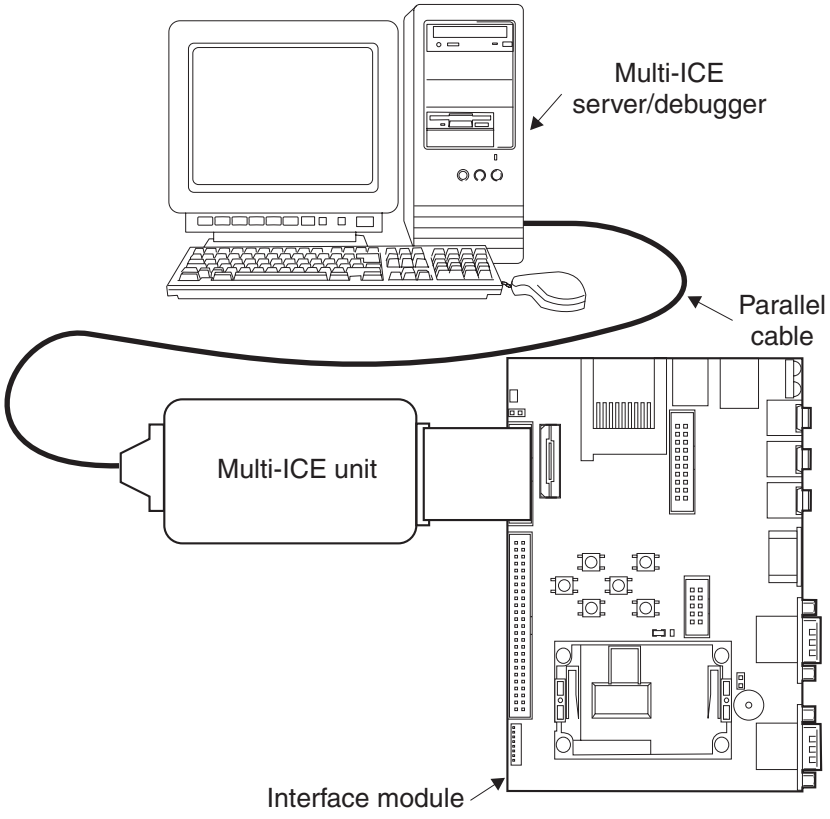


Figure 2-2 Connecting Multi-ICE

Note

There are no components on the interface module that use the JTAG signals. The connector provides you with access to the JTAG signals on the modules below.

Chapter 3
Hardware Reference

This chapter describes the hardware on the interface module. The descriptions assume that PrimeCell peripherals are being used to control these interfaces. This chapter contains the following sections:

3.1 Differences in signal naming between supported logic modules

The Integrator/LM-XCV600E+ and Integrator/LM-EP20K600E+ logic module types route the signals between the FPGA and the interface module differently as follows:

- the LM-XCV600E+ is fitted with a Xilinx FPGA and routes the interface module **ABANK[12:0]** signals to bank 0 on the FPGA and the **BBANK[57:0]** signals to bank 1 on the FPGA.
- the LM-EP20K600E+ is fitted with an Altera FPGA and routes the interface module **ABANK[12:0]** signals to bank 5 on the FPGA and the **BBANK[57:0]** signals to bank 6 on the FPGA.

The logic module output voltage on these banks is adjustable. Ensure that the logic module selection link is set to the 3V3 position.

Note

These pin assignments are contained in the example pin constraints file on the CD that accompanies the interface module.

3.2 Smart card interface

Figure 3-1 shows the architecture of the *Smart Card Interface* (SCI). This provides a suitable interface for the PrimeCell SCI (PL130) or similar peripheral. The diagram shows the tristate buffers that are used to provide the interface between the SCI and the card itself. An additional 10-way box header, J34, is provided to enable you to monitor the signals or to connect an off-board smartcard connector.

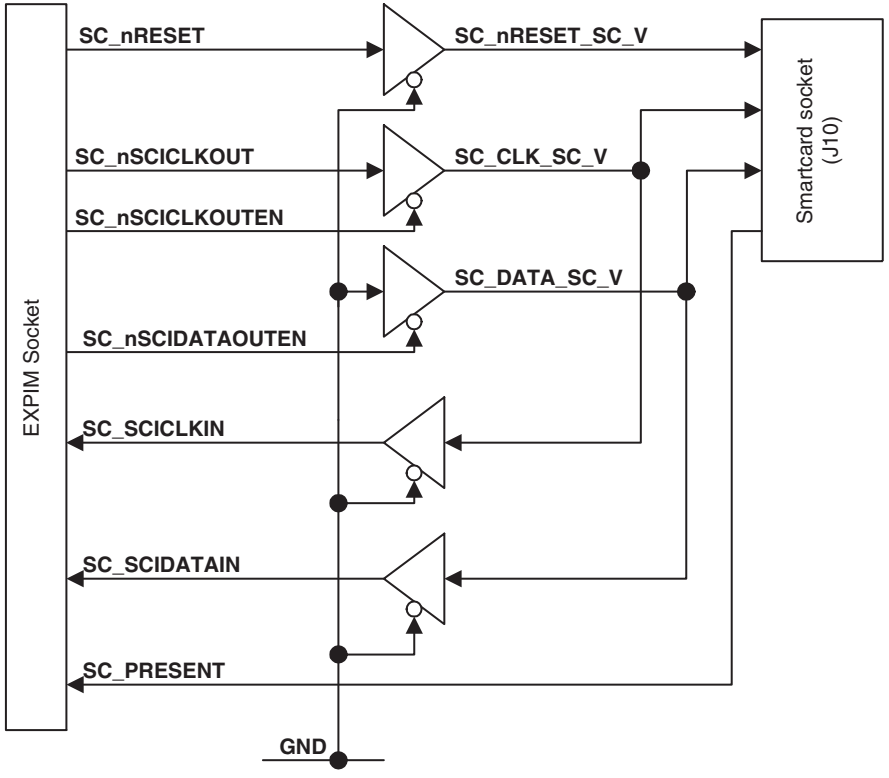


Figure 3-1 Smart card interface

You can set the SCI to operate at 3.3V or at 5V by setting the solder link LK2. The default setting is 5V.

The signals associated with the SCI are assigned to the EXPIM socket pins as shown in Table 3-1.

Table 3-1 Smartcard signal assignment

Signal name	EXPIM connector	Description
SC_SCICLKIN	IM_BBANK34	Clock input to controller
SC_SCIDATAIN	IM_BBANK35	Serial data input to controller
SC_nSCIDATAOUTEN	IM_BBANK36	Data output enable
SC_nSCICLKOUTEN	IM_BBANK37	Clock buffer output control
SC_SCICLKOUT	IM_BBANK38	Clock output from controller
SC_nRESET	IM_BBANK39	Reset to card
SC_PRESENT	IM_BBANK40	Card detect signal

Figure 3-2 shows the signal assignment of a smartcard. Pins 4 and 8 are not connected and are omitted on some cards.

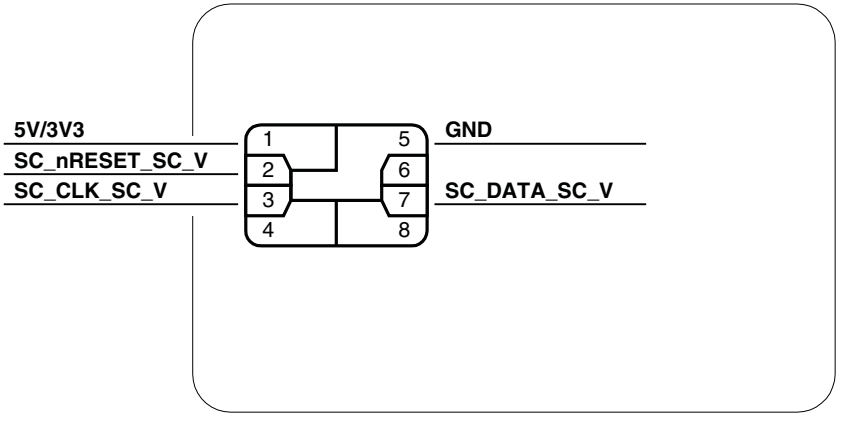


Figure 3-2 Smartcard contacts assignment

The smart card is inserted into the smartcard socket with the contacts face down.

Figure 3-3 on page 3-5 shows the pinout of the connector J34. This can be used to connect to an off-PCB smart card device.

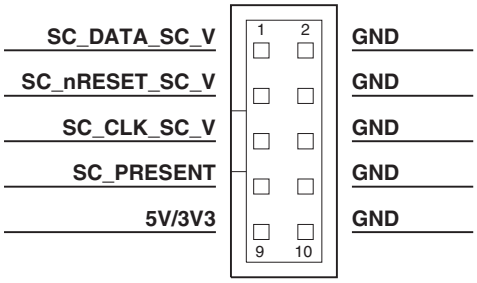


Figure 3-3 J34 pinout

3.3 IrDA interface

Figure 3-4 shows the infrared interface suitable for use with a PrimeCell UART (PL011) or similar peripheral.

The example IRMS6400 is an IrDA compatible transceiver produced by the Infineon Technologies Corp.

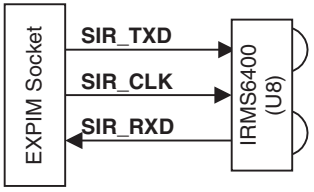


Figure 3-4 IrDA interface

The signals associated with the infrared interface are assigned on the EXPIM socket pins as shown in Table 3-1 on page 3-4.

Table 3-2 IrDA interface signal assignment

Signal name	EXPIM connector	Description
SIR_SCLK	IM_BBANK55	Serial clock
SIR_TXD	IM_BBANK56	Transmit data
SIR_RXD	IM_BBANK57	Receive data

3.4 UART interface

The interface module provides two serial transceivers suitable for use with the PrimeCell UART (PL011) or other similar peripherals. Figure 3-5 shows the architecture of one UART interface channel.

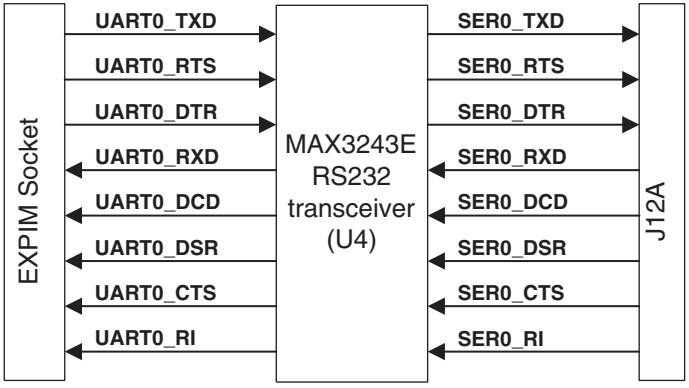


Figure 3-5 Serial interface (one channel)

The signals associated with the UART interface are assigned to the EXPIM socket pins as shown in Table 3-3.

Table 3-3 Serial interface signal assignment

Signal name	EXPIM connector	Description
UART0_TXD	IM_BBANK47	Transmit data
UART0_RTS	IM_BBANK48	Ready to send
UART0_DTR	IM_BBANK49	Data terminal ready
UART0_CTS	IM_BBANK50	Clear to send
UART0_DSR	IM_BBANK51	Data set ready
UART0_DCD	IM_BBANK52	Data carrier detect
UART0_RXD	IM_BBANK53	Receive data
UART0_RI	IM_BBANK54	Ring indicator
UART1_TXD	IM_ABANK0	Transmit data

Table 3-3 Serial interface signal assignment (continued)

Signal name	EXPIM connector	Description
UART1_RTS	IM_ABANK1	Ready to send
UART1_DTR	IM_ABANK2	Data terminal ready
UART1_CTS	IM_ABANK3	Clear to send
UART1_DSR	IM_ABANK4	Data set ready
UART1_DCD	IM_ABANK5	Data carrier detect
UART1_RXD	IM_ABANK6	Receive data
UART1_RI	IM_ABANK7	Ring indicator

Figure 3-6 shows the assignment of the two serial interfaces to the 9-pin D-type male connector at J12.

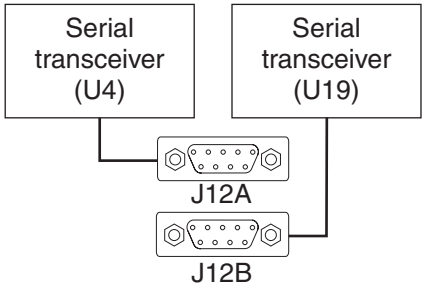


Figure 3-6 Serial interface connector assignment

Figure 3-7 on page 3-9 shows the pin numbering for a 9-pin D-type male connector and Table 3-4 on page 3-9 shows the signal assignment for the two connectors.

The pinout shown in Figure 3-7 on page 3-9 is configured as a *Data Terminal Equipment* (DTE) device.

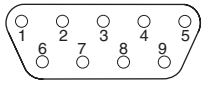


Figure 3-7 Serial connector pinout

Table 3-4 Serial plug signal assignment

Pin	J12A	J12B
1	**SER0_DCD**	**SER1_DCD**
2	**SER0_RX**	**SER1_RX**
3	**SER0_TX**	**SER1_TX**
4	**SER0_DTR**	**SER1_DTR**
5	**SER0_GND**	**SER1_GND**
6	**SER0_DSR**	**SER1_DSR**
7	**SER0_RTS**	**SER1_RTS**
8	**SER0_CTS**	**SER1_CTS**
9	**SER0_RI**	**SER1_RI**

3.5 USB interface

Figure 3-8 shows one of the two USB interfaces.

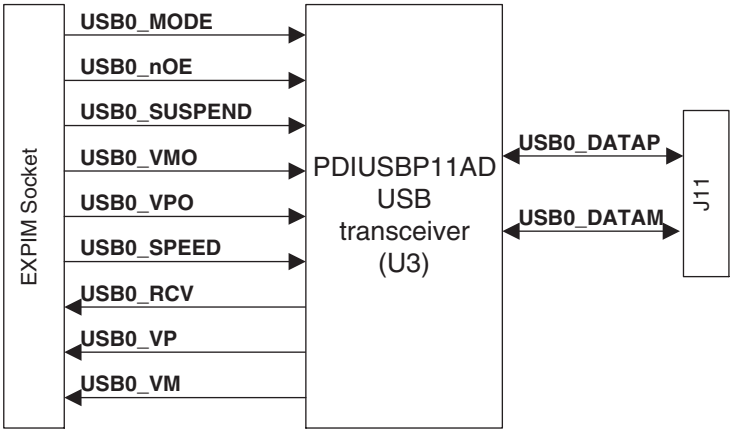


Figure 3-8 USB interface (one channel)

USB0 provides a USB host interface and connects through the type A connector J11. USB1 provides a USB device interface and connects through the type B connector J13.

The signals associated with the USB interfaces are assigned to the EXPIM socket pins as shown in Table 3-5.

Table 3-5 Serial interface signal assignment

Signal name	EXPIM connector	Description
USB0_VM	IM_BBANK16	Gated version of D–
USB0_VP	IM_BBANK17	Gated version of D+
USB0_RCV	IM_BBANK18	Receive data
USB0_SUSPEND	IM_BBANK19	Suspend for power save
USB0_nOE	IM_BBANK20	Output enable
USB0_VMO	IM_BBANK21	Differential input –
USB0_VPO	IM_BBANK22	Differential input +
USB0_MODE	IM_BBANK23	Mode

Table 3-5 Serial interface signal assignment (continued)

Signal name	EXPIM connector	Description
USB0_SPEED	IM_BBANK24	Edge rate control
USB1_VM	IM_BBANK25	Gated version of D–
USB1_VP	IM_BBANK26	Gated version of D+
USB1_RCV	IM_BBANK27	Receive data
USB1_SUSPEND	IM_BBANK28	Suspend for power save
USB1_nOE	IM_BBANK29	Output enable
USB1_VMO	IM_BBANK30	Differential input –
USB1_VPO	IM_BBANK31	Differential input +
USB1_MODE	IM_BBANK32	Mode
USB1_SPEED	IM_BBANK33	Edge rate control

Configure the device USB interface to operate at slow or full speed by moving the soldered link LK1. The two settings are:

Full speed Fit link in position A-B (default).

Slow speed Fit link in position B-C.

Note

For a full description of the USB signals refer to the datasheet for the Philips PDIUSBP11AD transceiver.

The two USB interfaces provide different types of USB connector manufactured by Berg. Figure 3-9 identifies the connectors for the host and device interfaces and shows how the pins are numbered.

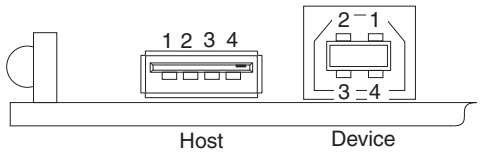


Figure 3-9 Identifying the USB connectors

3.6 Audio CODEC

The interface module provides a National Semiconductors LM4549 audio CODEC. The audio CODEC is compatible with AC'97 Rev 2.1, is PC98 compliant, and features sample rate conversion and analog 3D sound. The CODEC can be driven with a PrimeCell AACI (PL041) or similar peripheral. Figure 3-10 shows the audio CODEC.

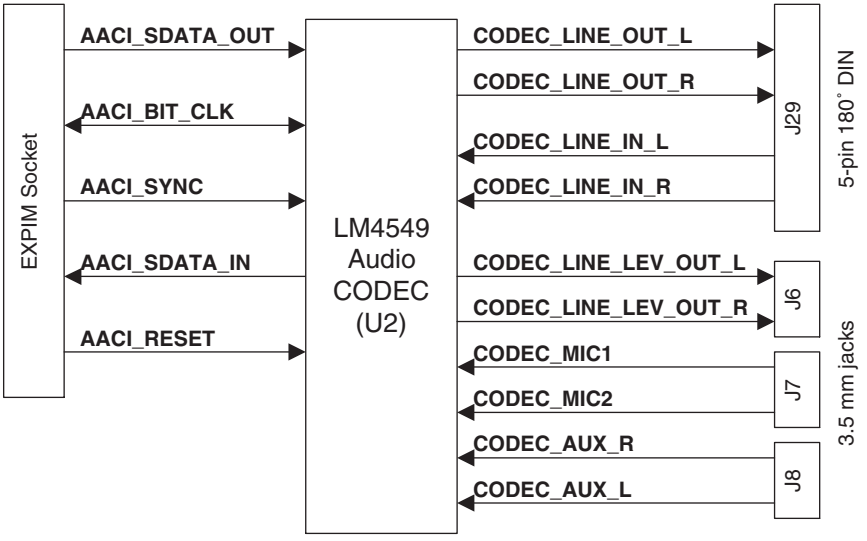


Figure 3-10 Audio interface

The signals associated with the audio CODEC interface are assigned to the EXPIM socket pins as shown in Table 3-6.

Table 3-6 Audio CODEC signal assignment

Signal name	EXPIM connector	Description
AACI_SDATA_OUT	IM_ABANK8	Serial data from AACI to the CODEC
AACI_BIT_CLK	IM_ABANK9	Clock from the CODEC
AAC_SYNC	IM_ABANK10	Frame synchronization signal from the AACI
AACI_SDATA_IN	IM_ABANK11	Serial data from the CODEC to the AACI
AACI_RESET	IM_ABANK12	Reset signal from the PrimeCell AACI

Note

For a description of the audio CODEC signals, refer to the LM4549 datasheet available from National Semiconductors.

The interface module provides three jack plugs that allow you to connect to the microphone and auxiliary inputs, and line level output on the CODEC. Stereo inputs and outputs are also provided by a 5-pin 180° DIN socket. Figure 3-11 shows the pinout of the DIN socket.

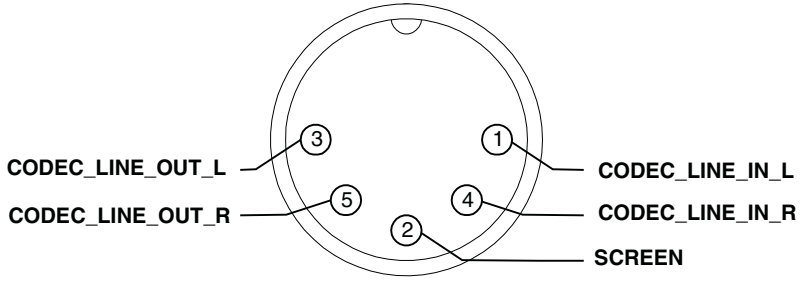


Figure 3-11 Audio DIN connector pinout

A cable is supplied with the interface module to enable you to connect audio devices with phono sockets to the DIN socket. This cable is shown in Figure 3-11.

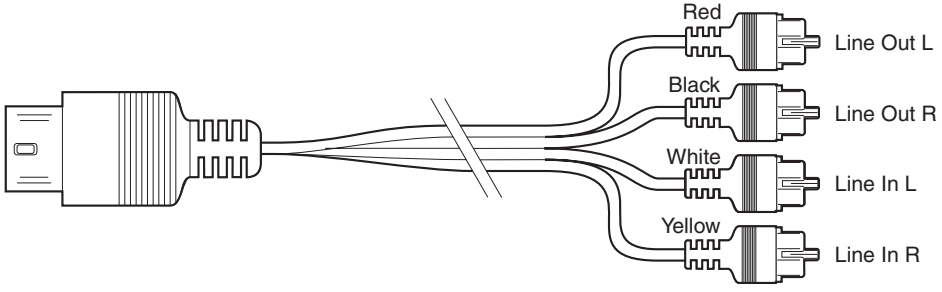


Figure 3-12 Supplied audio cable

Note

For correct operation of the CODEC interface, you must mute the PC Beep input by setting bit 15 in the PC Beep register within the CODEC (see the LM4549 datasheet available from National Semiconductors).

3.7 MMC and SD flash card interface

Figure 3-13 shows the MMC and SD flash card interface that can be driven as both an MMC or SD interface. A suitable MMC interface is the PrimeCell MMCI (PL181).

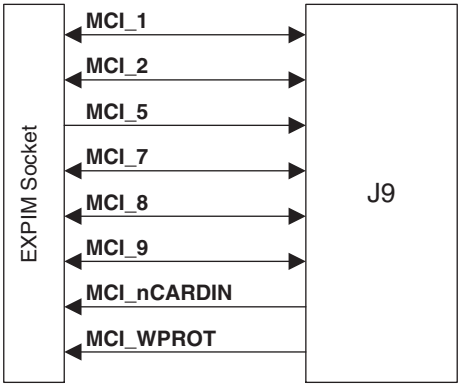


Figure 3-13 MMC/SD

The function of the interface signals depend on whether an MMC or SD card is fitted. Both card types default to MMC but the SD card has an additional operating mode called widebus mode. Table 3-7 shows the use of the signals for both modes of operation.

Table 3-7 MMC/SD interface signals

Signal name	EXPIM connector	Widebus mode	MMC mode
nMCI_ON	IM_BBANK7	Controls card power: • LOW = power ON • HIGH = power OFF	Controls card power: • LOW = power ON • HIGH = power OFF
MCI_1	IM_BBANK8	Card detect/Data(3)	Chip select (active LOW)
MCI_2	IM_BBANK9	Command/Response	Command/Response
MCI_5	IM_BBANK10	CLK	CLK
MCI_7	IM_BBANK11	Data(0)	Data
MCI_8	IM_BBANK12	Data(1)	not used

Table 3-7 MMC/SD interface signals (continued)

Signal name	EXPIM connector	Widebus mode	MMC mode
MCI_9	IM_BBANK13	Data(2)	not used
MCI_nCARDIN	IM_BBANK14	Card presence detect (active LOW)	Card presence detect (active LOW)
MCI_WPROT	IM_BBANK15	Card write-protection detect	Card write-protection detect

The MMC/SD card socket (J9) provides nine pins that connect to a card when it is inserted into the socket. Figure 3-14 shows the pin numbering and signal assignment. In addition the socket contains switches that operated are by card insertion and provide signaling on the **MCI_nCARDIN** and **MCI_WPROT** signals.

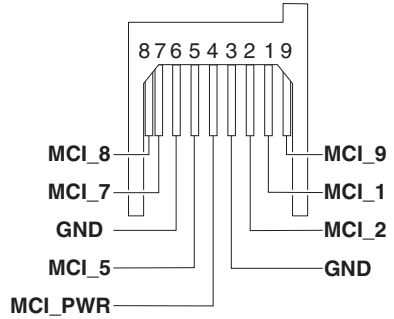


Figure 3-14 MMC/SD card socket pin numbering

The MMC card uses seven pins, and the SD card uses all nine pins. The additional pins are located as shown in Figure 3-14 with pin 9 next to pin 1 and pins 7 and 8 spaced more closely together than the other pins. Figure 3-15 shows an MMC card, with the contacts face up.

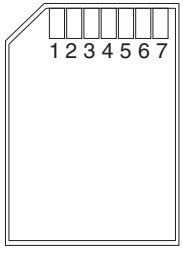


Figure 3-15 MMC card

Insert the card into the socket with the contacts face down. Cards are normally labelled on the top surface and provide an arrow to indicate the correct way to insert them.

Remove the card by gently pressing it into the socket. It springs back and can be removed. This ensures that the card detection switches within the socket operate correctly.

The connector J33 enables you to access the signals for debugging or to an off-PCB card socket. The pinout of J33 is shown in Figure 3-16.

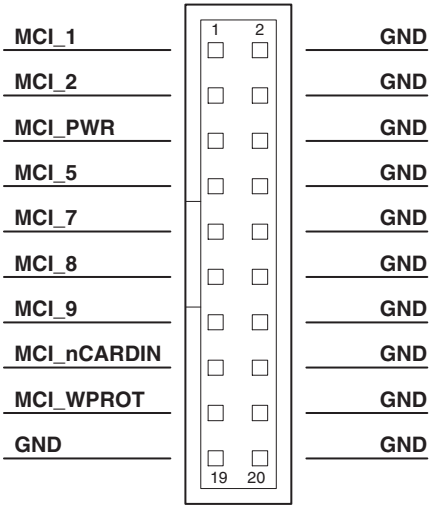


Figure 3-16 J33 pinout

3.8 Display interface

The interface module provides a display interface with outputs for a:

- VGA or SVGA monitor connected to J30
- Sharp LQ084V1DG21 8.4 inch TFT VGA LCD panel connected to J14
- Generic LCD and touchscreen connector.

A suitable peripheral for driving these types of display is the PrimeCell CLCD controller (PL110).

Figure 3-17 shows the architecture of the display interface. The diagram shows the signals used to provide pixel data and for buffer control. B27 is used to enable the signals on J27 and B28 is used to enable the buffers for the Sharp display signals on J14.

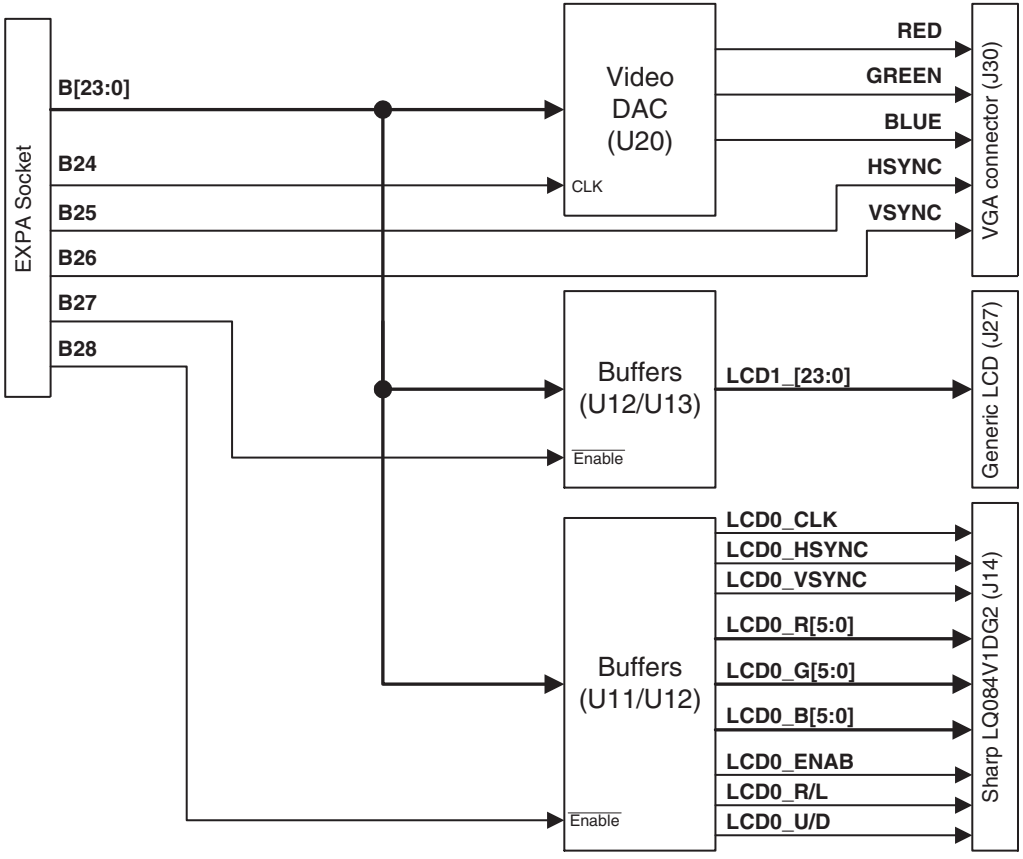


Figure 3-17 Display interfaces

The interface module provides two power outputs for the display interfaces as shown in Figure 3-18. These are:

- **LCD1_BIAS**
- **LCD_3V3.**

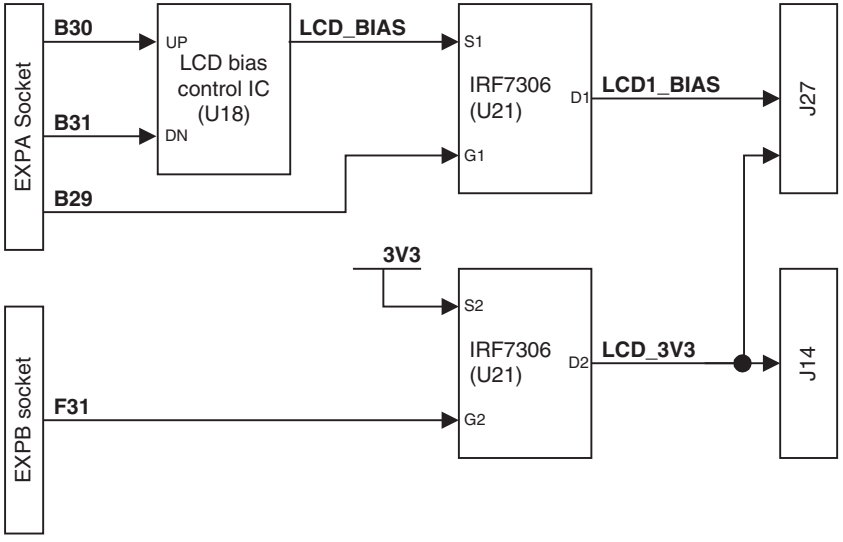


Figure 3-18 LCD1 power supply control

LCD1_BIAS is a variable supply controlled by the MAX686 DAC IC (U18) and switched ON and OFF by the MOSFET switch (U21). **LCD1_BIAS** is varied between 11.5V and 24V in 64 steps using the inputs on pins B30 and B31 on the EXPA socket. These are edge-triggered inputs. The MAX686 is reset to the midpoint by a power-on reset. The MOSFET switch is controlled by the input from B29.

LCD_3V3 is fixed level power output that is controlled by a second MOSFET switch within U21. The switch is controlled by the input on the pin F31 on the EXPB connector.

The interface module provides two connectors. One (J14) is a dedicated connector for a 8.4 inch Sharp LCD display and the second (J27) provides a generic interface. Figure 3-19 on page 3-20 shows the pinout of connector J27.

Note

If the logic module is mounted in the EXPA/EXPB position on an Integrator/AP, the pins marked F bus connect to the GPIO bus on the Integrator/AP. This bus is routed between the system controller FPGA on the motherboard and the FPGA on the logic module. These signals are available for your own applications.

If the logic module is mounted in the HDRA/HDRB position on the motherboard, these pins connect to the F bus that is routed between any modules in the stack. there are no signals from the motherboard present on these pins.

The release note provided with the shipped example provides information on how the signals shown in Figure 3-19 on page 3-20 are connected to the PrimeCell.

If you intend to use anything other than the shipped example then signal allocation is a user decision.

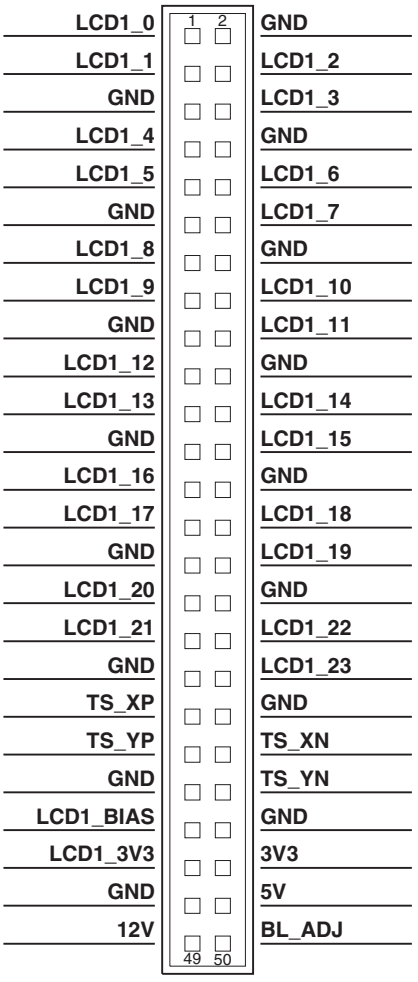


Figure 3-19 J27 pinout

Note

The **LCD1[23:0]** signals are 3V3 buffered versions of **B[23:0]**, see the schematic diagram for more details.

3.9 Touchscreen controller

The touchscreen interface is designed to connect to a 4-wire resistive touchscreen. It can be driven by the PrimeCell SSP (PL021), PrimeCell SSP (PL022), or similar peripheral.

Figure 3-20 shows the touch-screen interface. The signals to the touchscreen are routed to the 50-pin connector J27 and also to J31.

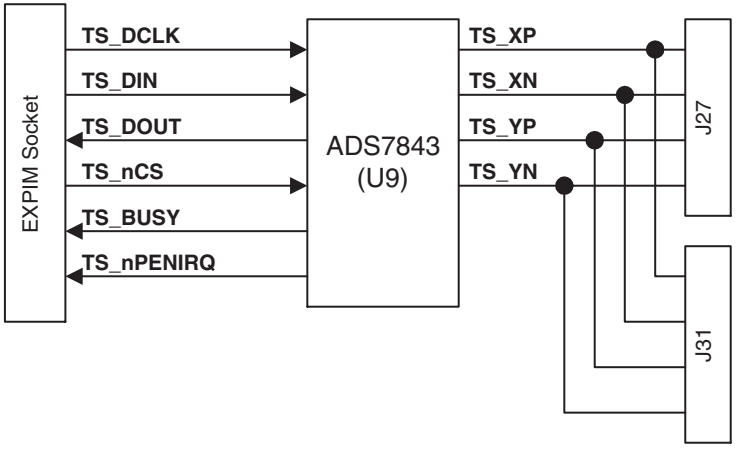


Figure 3-20 Touch screen interface

The touch-screen interface uses an Analog Devices ADS7843 (U9) controller to provide an interface with a 4-wire resistive touch screen. It communicates with the host using a serial interface. The host interface signals are shown in Table 3-8.

Table 3-8 Touch screen host interface signal assignment

Signal name	EXPIM connector	Description
TS_DIN	IM_BBANK41	Serial data input to controller
TS_nCS	IM_BBANK42	Controller chip select
TS_DCLK	IM_BBANK43	Clock input to controller
TS_DOUT	IM_BBANK44	Data output from controller
TS_BUSY	IM_BBANK45	Busy indicator from controller
TS_nPENIRQ	IM_BBANK46	Interrupt from controller

Figure 3-21 shows the pinout of the connector J31.

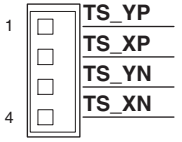


Figure 3-21 J31 pinout

3.10 Backlight control

The interface module provides an adjustable LCD backlight voltage controlled by the 2kΩ potentiometer R153. This enables you to adjust the **BL_ADJ** output between 0V and 2.5V. This is available from the connector J32.

Figure 3-22 shows the pinout of J32.

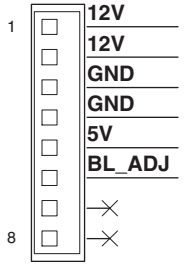


Figure 3-22 Backlight connector pinout

3.11 Push buttons

The interface module is fitted with six push buttons. These can be monitored using a PrimeCell GPIO (PL061) or other suitable peripheral. Each switch is a normally open type and is connected to the EXPIM connector through inverters so that pressing the button drives the associated input to the GPIO HIGH.

Table 3-9 shows the assignment of the push button inputs to the EXPIM connector.

Table 3-9 Push button interface signal assignment

Signal name	EXPIM connector	Description
PB0	IM_BBANK1	Input from S1
PB1	IM_BBANK2	Input from S2
PB2	IM_BBANK3	Input from S3
PB3	IM_BBANK4	Input from S4
PB4	IM_BBANK5	Input from S5
PB5	IM_BBANK6	Input from S6

3.12 Buzzer

The buzzer on the interface module can be driven with a PrimeCell GPIO (PL061) or other suitable peripheral. The buzzer is connected to the drain of a MOSFET device within U17 and the gate is controlled by an output from the GPIO. Enable the buzzer inserting fitting the jumper J23. Disable the buzzer by removing jumper J23.

The signal assignment is shown in Table 3-10.

Table 3-10 Buzzer interface signal assignment

Signal name	EXPIM connector	Description
nBUZZER	IM_BBANK0	Controls power to the buzzer: • LOW = power ON • HIGH = power OFF.

Chapter 4
Reference Design Example

This chapter describes how to set up and start using the logic module. It contains the following sections:

4.1 About the design example

This chapter describes the reference design example supplied with the interface module. The interface module is not fitted with any programmable devices because it is intended to provide interfaces for peripherals instantiated into a logic module FPGA.

A VHDL example is supplied for the Integrator/LM-XCV2000E and LM-EP20K1000E logic modules with PrimeCell peripherals instantiated into the FPGA design. This example is designed to operate with the logic module mounted on a suitable motherboard, for example the Integrator/AP. If you are using your own peripheral designs, the example will help you become familiar with using the interface module.

4.1.1 About PrimeCells

The interface module is supplied with executable software that demonstrates the functionality of the PrimeCells included in the design example.

The ARM PrimeCells are a range of synthesizable peripherals that are ideally suited for use in ARM-based designs. The interface module is supplied with an FPGA image containing PrimeCell peripherals for each supported interface on the board and the accompanying CD contains documentation for them.

HDL and device source code for the PrimeCell peripherals are not supplied with the interface module. You must license the PrimeCell peripherals to obtain the source code.

4.2 Design example

The interface module design example is supplied in VHDL. The example is an AHB design with an APB subsystem. The PrimeCell peripherals are instantiated in the top-level VHDL file.

4.2.1 Example architecture

The architecture of the example is shown in Figure 4-1. The interface module is provided with an example and release notes that define the clock sources, interrupt assignments, memory map, and peripherals.

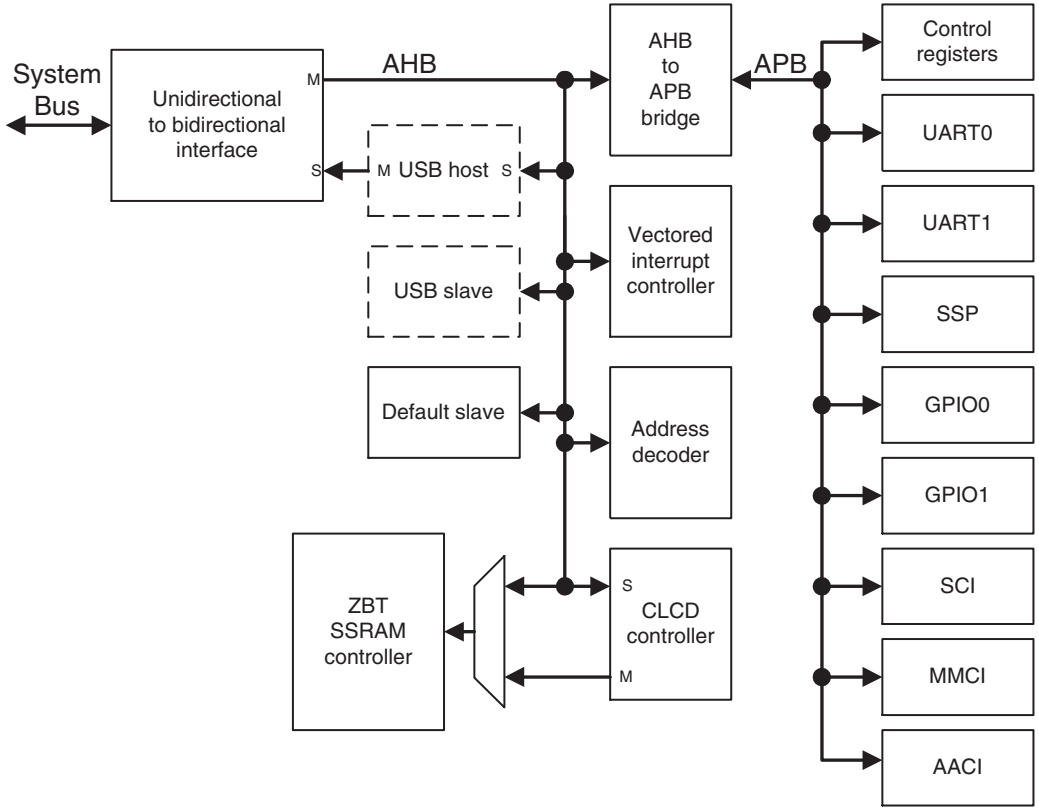


Figure 4-1 Design example

Note

The example FPGA image shipped with the board does not contain all of the peripherals shown in Figure 4-1 on page 4-3.

Universal serial bus

Connectors and circuitry are provided for the two USB boxes shown in Figure 4-1 on page 4-3. However, there are no USB PrimeCells currently available from ARM Limited. These blocks can be licensed from other IP providers.

Vectored interrupt controller

The PrimeCell PL190 *Vectored Interrupt Controller* (VIC) provides a software interface to the interrupt system. In an ARM system, two levels of interrupt are available:

- *Fast Interrupt Request* (FIQ) for fast, low latency interrupt handling
- *Interrupt ReQuest* (IRQ) for more general interrupts.

Only a single FIQ source at a time is generally used in a system, to provide a true low-latency interrupt. This has the following benefits:

- You can execute the interrupt service routine directly without determining the source of the interrupt.
- Interrupt latency is reduced. You can use the banked registers available for FIQ interrupts more efficiently, because a context save is not required.

There are 32 interrupt lines. The VIC uses a bit position for each different interrupt source. The software can control each request line to generate software interrupts.

4.2.2 Supplied VHDL files

Table 4-1 provides a summary description of the supplied VHDL (`.vhd`) files. A more detailed description of each block is included within the files in the form of comments.

Table 4-1 VHDL file descriptions

File	Description
`AHBTop.vhd`	This file is the top-level VHDL that instantiates all of the PrimeCells for the example. The VHDL for the PrimeCells themselves are not supplied but are available from ARM as separate products.
`AHBDecoder.vhd`	The decoder block provides the high-speed peripherals with select lines. These are generated from the address lines and the module ID (position in stack) signals from the motherboard. The Integrator family of boards uses a distributed address decoding system (see *Address assignment of logic modules* on page 4-7).
`AHBDefaultSlave.vhd`	This block provides a default slave response when the logic module address space is addressed but the address does not correspond to any of the instantiated peripherals.
`AHBMux8S1M.vhd`	This is the AHB multiplexor that connects the read data buses and the **HRESP** and **HREADY** signals from all of the slaves to the AHB master.
`AHBZBTRAM.vhd`	High-speed peripherals require SSRAM controller block to support word, halfword, and byte operations to the SSRAM on the logic module.
`AHB2APB.vhd`	This is the bridge block required to connect APB peripherals to the AMBA AHB bus. It produces the peripheral select signals for each of the APB peripherals.
`APBRegs.vhd`	The APB register peripheral provides memory-mapped registers that you can use to: • configure the two clock generators • write to the user LEDs on the logic module • read the user switch inputs on the logic module. It also latches the pressing of the push button to generate an expansion interrupt.
`BuildOptions.vhd`	This file defines generation of the PrimeCells in the example and allows control over the synthesis so that PrimeCells can be included or excluded. It also specifies the base address of all the peripherals.

Note

The VHDL files provided are only for information and will not build without additional components that can be licensed separately.

4.2.3 Loading the FPGA image

Multi-ICE configuration files, the progcards utility, and the FPGA images for the supported logic modules are available after installation from the CDROM. Refer to the Logic Module User Guide for programming information.

4.2.4 Example memory map

The supplied examples set up the memory map for the logic module as shown in Figure 4-2. This shows the locations to which logic modules are assigned by the main address decoder on the motherboard. The diagram also shows how the example decodes the address space for the logic module when it is LM0 (bottom of the stack). Refer to the release notes shipped with the example for more information.

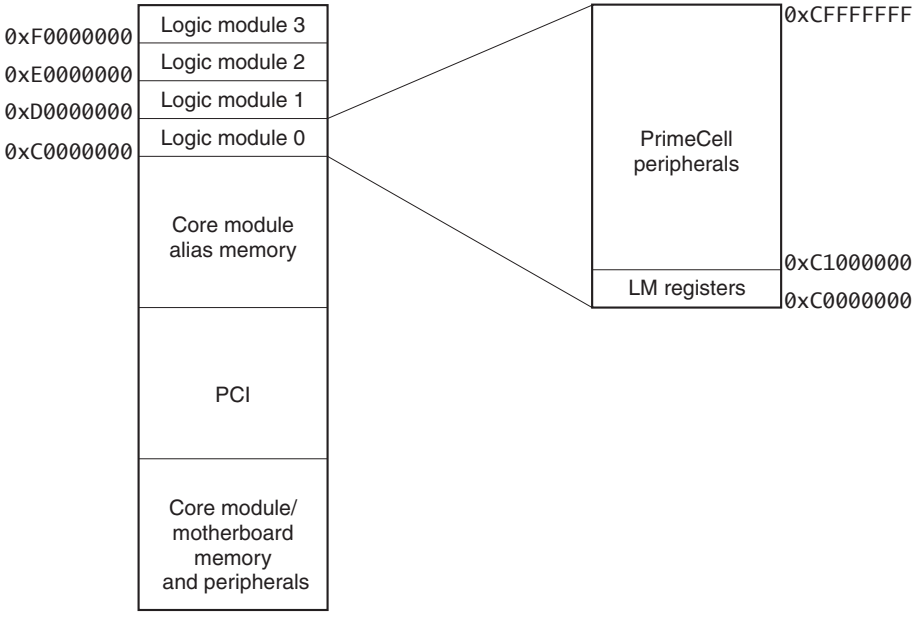


Figure 4-2 Integrator system memory map

Note

The Integrator system implements a distributed address decoding scheme in which each core or logic module is responsible for decoding its own address space. It is important when implementing a logic module design, to ensure that the module responds to all memory accesses in the appropriate memory region (see the user guide for your motherboard).

4.2.5 Address assignment of logic modules

You can mount up to four logic modules on an Integrator/AP motherboard. The base address of each logic module depends on its position in the stack. This defines the value of bits [31:28] of the address of devices on the logic module as shown in Table 4-2.

Table 4-2 Logic module addresses

Position	Bits 31:28
0 (bottom)	0xC
1	0xD
2	0xE
3 (top)	0xF

4.2.6 Example memory map

The addresses in Table 4-3 assume that there is one logic module present in position 0 (bottom) of the expansion stack on the Integrator/AP. If the logic module is present at any other position, then the top four bits of the address decoding will change.

Table 4-3 Address map

Peripheral	Clock source	VIC Interrupt	Base address	Reference
Control registers	n/a	0	0xC0000000	*Example APB register peripheral* on page 4-8
UART 0	1/4 CLK2	1	0xC0100000	*ARM PrimeCell UART (PL011) Technical Reference Manual*
UART 1	1/4 CLK2	2	0xC0200000	*ARM PrimeCell UART (PL011) Technical Reference Manual*
SSP	1/64 CLK2	3	0xC0300000	*ARM PrimeCell Synchronous Serial Port Master and Slave (PL022) Technical Reference Manual*
GPIO 0	n/a	4	0xC0400000	*ARM PrimeCell GPIO (PL061) Technical Reference Manual*

Table 4-3 Address map

Peripheral	Clock source	VIC Interrupt	Base address	Reference
GPIO 1	n/a	5	0xC0500000	*ARM PrimeCell GPIO (PL061) Technical Reference Manual*
SCI	1/4 CLK2	6	0xC0600000	*ARM PrimeCell Smartcard Interface (PL130) Technical Reference Manual*
MMCI	n/a	7 and 8	0xC0700000	*ARM PrimeCell Multimedia Card Interface (PL181) Technical Reference Manual*
AACI	AACIBITCLK	9	0xC0800000	*ARM PrimeCell Advanced Audio CODEC Interface (PL041) Technical Reference Manual*
CLCD	CLK1	11	0xC1000000	*ARM PrimeCell Color LCD Controller (PL110) Technical Reference Manual*
SSRAM	n/a	n/a	0xC2000000	*SSRAM* on page 4-12
VIC	n/a	n/a	0xC3000000	*ARM PrimeCell Vectored Interrupt Controller (PL190) Technical Reference Manual*
PIB	n/a	n/a	0xCFFFFF00	*Peripheral Information Block, PIB* on page 4-12

4.2.7 Example APB register peripheral

Table 4-4 shows the mapping of the logic module control registers. The addresses shown are offsets from the base addresses shown in Figure 4-2 on page 4-6.

Table 4-4 Logic module registers

Offset address	Name	Type	Function
0x0000000	LM_OSC1	Read/write	Oscillator 1 divisor register
0x0000004	LM_OSC2	Read/write	Oscillator 2 divisor register
0x0000008	LM_LOCK	Read/write	Oscillator lock register
0x000000C	LM_LEDS	Read/write	User LEDs control register

Table 4-4 Logic module registers

Offset address	Name	Type	Function
0x0000010	LM_INT	Read/write	Push button interrupt register
0x0000014	LM_SW	Read	Switches register
0x0000018	LM_CONTROL	Read/write	Control register

Oscillator divisor registers

The oscillator registers control the frequency of the clocks generated by the two clock generators on the logic module.

Before writing to the oscillator registers, you must unlock them by writing the value 0x0000A05F to the LM_LOCK register. After writing the oscillator register, relock them by writing any value other than 0x0000A05F to the LM_LOCK register.

Table 4-5 describes the oscillator register bits.

Table 4-5 LM_OSCx registers

Bits	Name	Access	Function
18:16	OD	Read/write	Output divider: • 000 = divide by 10 • 001 = divide by 2 • 010 = divide by 8 • 011 = divide by 4 • 100 = divide by 5 • 101 = divide by 7 • 110 = divide by 9 • 111 = divide by 6.
15:9	RDW	Read/write	Reference divider word. Defines the binary value of the R[6:0] pins of the clock generator.
8:0	VDW	Read/write	VCO divider word. Defines the binary value of the V[8:0] pins of the clock generator.

Note

The default values set **CLK1** to 25MHz and **CLK2** to 48MHz.

The reference divider (R[6:0]) and VCO divider (V[8:0]) are used to calculate the output frequency as follows:

$$\text{Frequency} = 48\text{MHz} \cdot \frac{(V[8:0] + 8)}{(R[6:0] + 2) \cdot OD}$$

You must also observe the operating range limits:

$$10\text{MHz} < 48\text{MHz} \cdot \frac{(V[8:0] + 8)}{(R[6:0] + 2)}$$

$$R[6:0] < 118$$

Note

You can calculate values for the clock control signals using the ICS525 calculator on the Integrated Circuit Systems website at:

`http://www.icst.com/products/ics525inputForm.html`.

Oscillator lock register

The lock register is used to control access to the oscillator registers, allowing them to be locked and unlocked. This mechanism prevents the oscillator registers from being overwritten accidently. Table 4-6 describes the lock register bits.

Table 4-6 LM_LOCK register

Bits	Name	Access	Function
16	LOCKED	Read	This bit indicates if the oscillator registers are locked or unlocked: • 0 = unlocked • 1 = locked.
15:0	LOCKVAL	Read/write	Write the value `0x0000A05F` to this register to enable write accesses to the oscillator registers. Write any other value to this register to lock the oscillator registers.

User LEDs control register

The LEDs register is used to control the user LEDs on the logic module. Writing a 0 to a bit lights the associated LED.

Push button interrupt register

The push button interrupt register contains 1 bit. It is a latched indication that the push button on the logic module has been pressed. The output from this register is used to drive an input to the interrupt controller. Table 4-7 describes the operation of this register.

Table 4-7 LM_INT register

Bits	Name	Access	Function
0	LM_INT	Read	This bit when SET is a latched indication that the push button has been pressed.
		Write	Write 0 to this register to CLEAR the latched indication. Writing 1 to this register has the same effect as pressing the push button.

Switches register

This register is used to read the setting of the 8-way DIP switch on the logic module. A 0 indicates that the associated switch element is CLOSED (ON).

Control register

This register controls the multiplexors that are used to select the display type.

Table 4-8 describes the operation of this register.

Table 4-8 LM_CONTROL register

Bits	Name	Access	Function
7:3	RESERVED	-	-
2	DISPLAY ENABLE	Read/write	This bit enables and disables the selected display: • 0 = DISABLED • 1 = ENABLED (default).
1:0	DISPLAY SELECT	Read/write	These bits control the display outputs: • 00 = Sharp 8.4 inch display (default) • 01= VGA/SVGA monitor 16bpp • 10 = LCD1 connector • 11 = VGA/SVGA monitor 24bpp.

4.2.8 SSRAM

The example design instantiates a 1MB block of SRAM at `0xC2000000`. This memory is physically located on the logic module.

4.2.9 Peripheral Information Block, PIB

The FPGA image contains a ROM block that provides information on the peripherals present. PrimeCell number, base address and PrimeCell revision number is given. The information stored in the PIB will become more advanced in future releases.

The ROM block is situated at the top of the memory map allocated to the logic module. There are up to 32 words of information stored. For example, a logic module at the bottom of the stack will have the PIB present at `0xCFFFFF00` to `0xCFFFFFFF`.

Format:

```
0x00000000 - no entry, ignore. Check next address.
```

else

```
0xaabbbbcc
```

where:

aa bits 31:28 depend on the position in the stack

bits 27:20 of the PrimeCell base address

bits 19:0 are zero

bbbb PrimeCell number, for example 0041 = AACI (PL041)

If a peripheral is not a PrimeCell, then a value `0xFF`*`bb`* will be assigned, where *bb* is just a unique look-up value. The value `0xFFFF` is used as a special case to indicate FPGA build number.

cc PrimeCell revision, for example 11 = v1.1

For example, `0x48004111` would be an AACI rev 1.1 with base address `0xC4800000`, assuming the logic module is in bottom position.

The last word in the ROM is used to hold the FPGA build number in the format `0xFFFFFF`*`ab`*, where the build revision is *a.b*

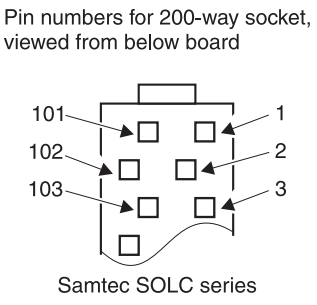
A utility `read_pib.axf` is provided that reads the PIB and lists the peripherals present. This also displays the FPGA build number.

Appendix A
Signal Descriptions

This appendix describes the Integrator/IM-PD1 interface connectors and signal connections. It contains the following sections:

A.1 EXPA

Figure A-1 shows the how the pins on the EXPA socket on the on underside of the interface module are numbered.



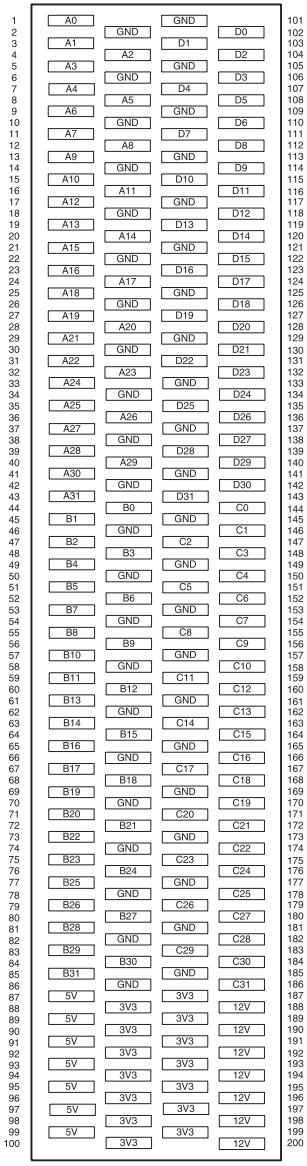


Figure A-1 EXPA socket pin numbering

The signals present on the EXPA connector are described in Table A-1.

Table A-1 AHB signal assignment

Pin label	Signal	Description
A[31:0]	Not used	-
B[31:0]	**B[31:0]**	These signals connect to the FPGA on the logic module. They are used to carry display interface signals (see *Display interface* on page 3-17)
C[31:0]	Not used	-
D[31:0]	Not used	-

A.2 EXPB

Figure A-2 shows the pin numbers of the EXPB socket on the underside of the interface module.

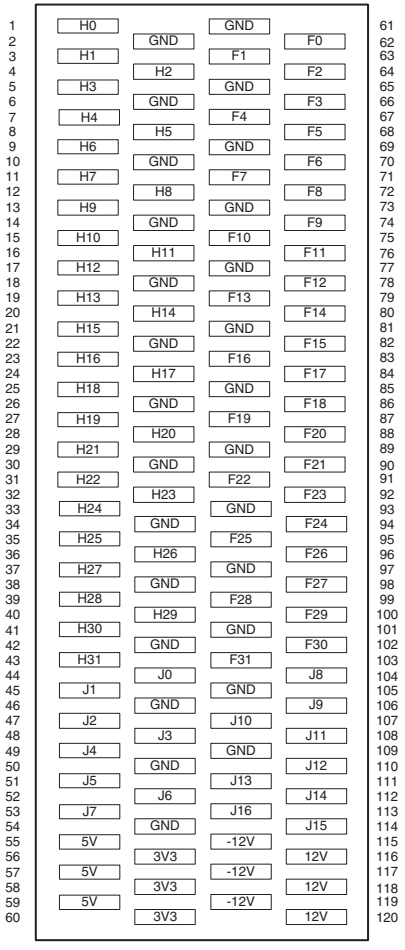


Figure A-2 EXPB socket pin numbering

Table A-2 describes the signals on the pins labeled H[31:0], J[16:0].

Table A-2 EXPB signal assignment

Pin label	Name	Description
H[31:29]	Not used	-
H28	**SYSCLK**	System clock from the logic module.
H[27:0]	Not used	-
J[16:14]	Not used	-
J13	**nCFGEN**	Sets motherboard into configuration mode.
J12	**nSRST**	Multi-ICE reset (open collector).
J11	Not used	-
J10	**RTCK**	Returned JTAG test clock.
J9	Not used	-
J8	**nTRST**	JTAG reset.
J7	**TDO**	JTAG test data out.
J6	**TDI**	JTAG test data in.
J5	**TMS**	JTAG test mode select.
J4	**TCK**	JTAG test clock.
J[3:0]	Not used	-
F31	**F31**	Used used by the interface module to control the 3.3V power output for LCD1 from J27 and J14 (see *Display interface* on page 3-17). See note below.
F[30:0]	Not used	See note below.

Note

If the logic module is mounted in the EXPA/EXPB position on an Integrator/AP, the pins marked F bus connect to the GPIO bus on the Integrator/AP. This bus is routed between the system controller FPGA on the motherboard and the FPGA on the logic module. These signals are available for your own applications.

If the logic module is mounted in the HDRA/HDRB position on the motherboard, these pins connect to the F bus that is routed between any modules in the stack. There are no signals from the motherboard present on these pins.

A.3 EXPIM

This connector is the same type of as that used for EXPA. Figure A-3 shows the pin numbers for the EXPIM socket on the interface module.

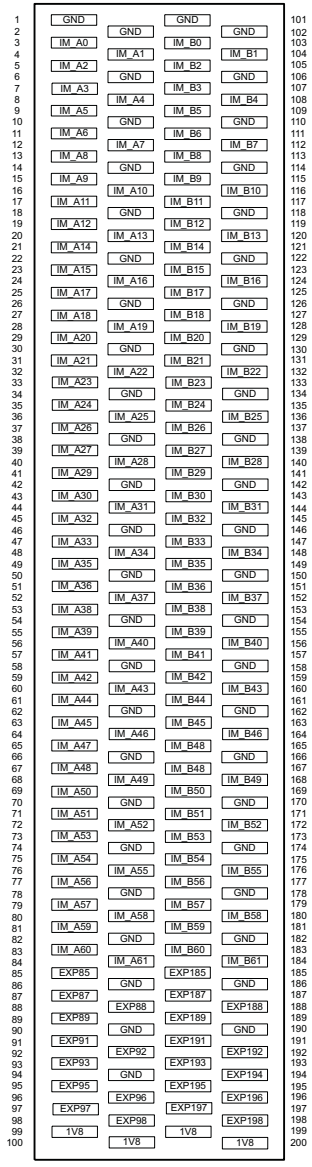


Figure A-3 EXPIM socket pin numbering

Table A-3 shows the signals for the interface module for Integrator/LM-XCV600E+ or LM-EP20K600E+ logic module types.

Table A-3 EXPIM signal assignment

Label	LM-XCV600E+	LM-EP20K600E+	Description
IM_ABANK[12:0]	**IM_0BANK[12:0]**	**IM_5BANK[12:0]**	FPGA input/output pins.
IM_BBANK[57:0]	**IM_1BANK[57:0]**	**IM_6BANK[57:0]**	FPGA input/output pins.
EXP[96:85]	Not used	Not used	-
EXP97	**VCCO_0**	**VCCO_5**	Configurable voltage power supply rail. Not used (socket).
EXP98	**VCCO_0**	**VCCO_5**	Configurable voltage power supply rail. Not used (socket).
EXP185	Not used	Not used	-
EXP[189:187]	Not used	Not used	-
EXP191	**CLK1_1**	**CLK1_1**	Clock signal from the **CLK1** buffer on the logic module.
EXP[196:192]	Not used	Not used	-
EXP197	**VCCO_1**	**VCCO_6**	Configurable voltage power supply rail. Not used (socket).
EXP198	**VCCO_1**	**VCCO_6**	Configurable voltage power supply rail Not used (socket).

Caution

For correct operation of the interface module, VCCO_A and VCCO_B must be set to 3.3V. Ensure that the VCCO links are set correctly on the logic module.

A.4 Logic analyzer connector

Figure A-4 shows the pin numbers of this type of connector.

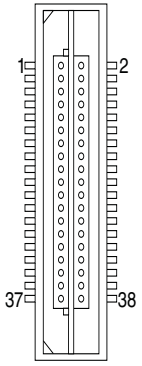


Figure A-4 J19 pin locations

Table A-4 shows the pinout of the logic analyzer connector.

Table A-4 J19 connector pinout

Signal	Pin	Pin	Signal
No connect	1	2	No connect
GND	3	4	No connect
SYSCLK	5	6	**CLK1**
B31	7	8	**B15**
B30	9	10	**B14**
B29	11	12	**B13**
B28	13	14	**B12**
B27	15	16	**B11**
B26	17	18	**B10**
B25	19	20	**B9**
B24	21	22	**B8**
B23	23	24	**B7**

Table A-4 J19 connector pinout (continued)

Signal	Pin	Pin	Signal
B22	25	26	**B6**
B21	27	28	**B5**
B20	29	30	**B4**
B19	31	32	**B3**
B18	33	34	**B2**
B17	35	36	**B1**
B16	37	38	**B0**

A.4.1 Multi-ICE (JTAG)

Figure A-5 shows the pinout of the Multi-ICE connector J21. For a description of the JTAG signals, see the user guide for your logic module.

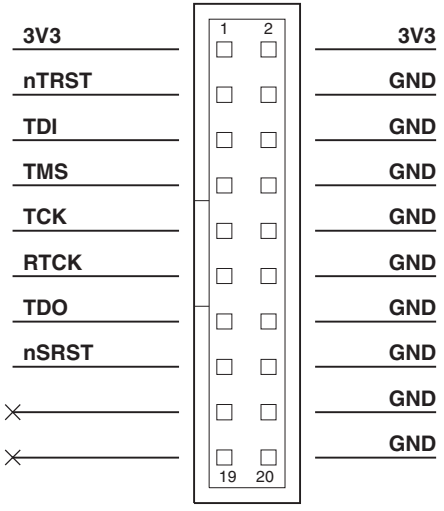


Figure A-5 Multi-ICE connector pinout

Appendix B
Mechanical Specification

This appendix contains the mechanical specification for Integrator/IM-PD1. It contains the following section:

B.1 Mechanical information

The Integrator/IM-PD1 is designed to be stackable. Figure B-1 on page B-2 shows the mechanical outline of a board on which you can mount an Integrator/IM-PD1 board. It shows the location of pin 1 of the Samtec connectors. (Dimensions are in millimeters.)

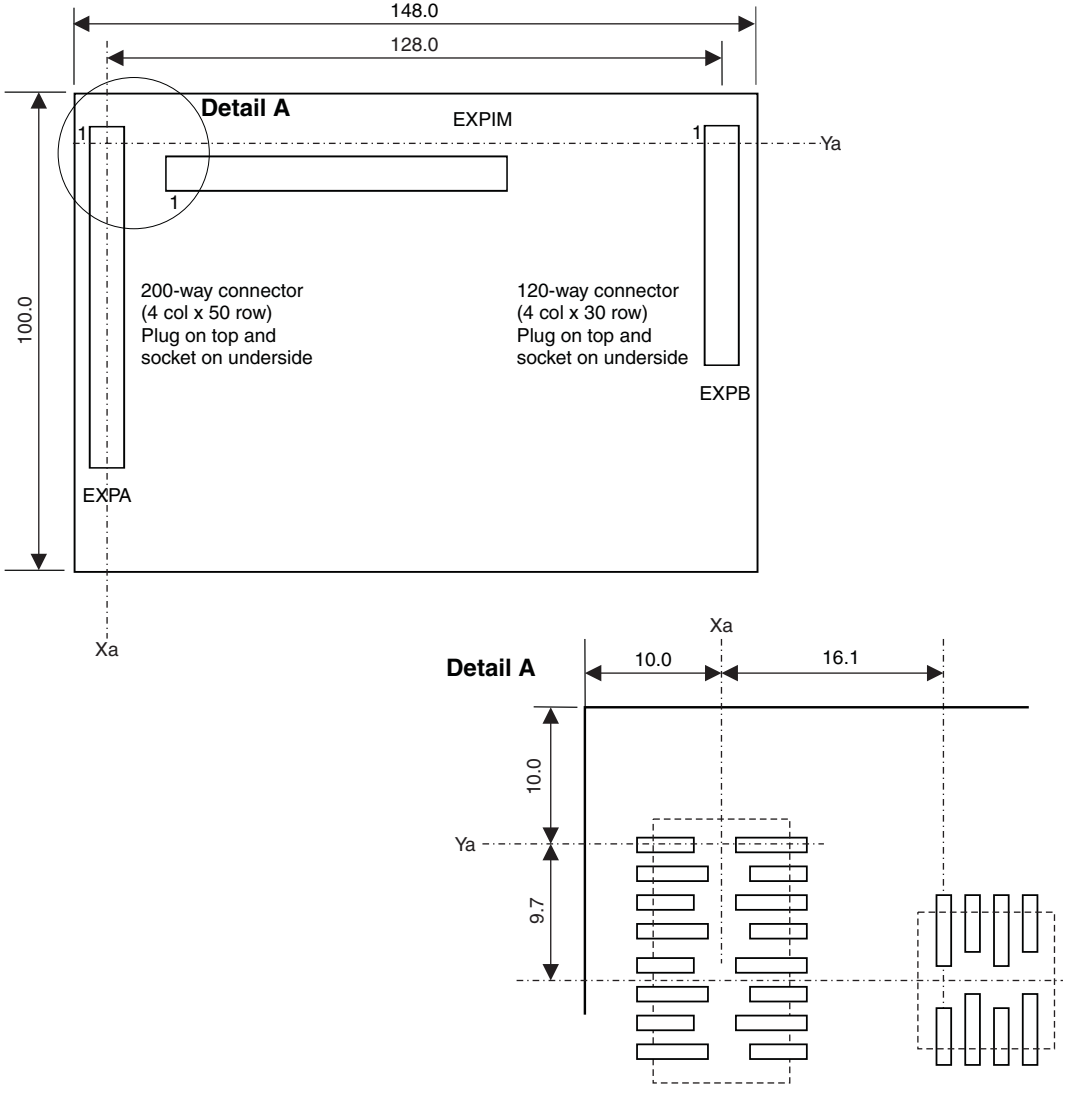


Figure B-1 Board dimensions

Note

In Figure B-1 on page B-2, the 148.0 and 100.0 dimensions show the size of a standard module produced by ARM Limited.

B.1.1 Connector part numbers

The Samtec connector part numbers are listed in Table B-1.

Table B-1 Samtec connector part numbers

Type	Part number
200 way connector	TOLC-150-32-F-Q-P-A
120 way connector	TOLC-130-32-F-Q-P-A

Glossary

This glossary lists all the abbreviations used in the Integrator/IM-PD1 User Guide.

AACI Advanced Audio CODEC Interface.

AMBA High-performance Bus The ARM open standard for on-chip buses. AHB conforms to this standard.

AMBA Peripheral Bus The ARM open standard for peripheral buses. APB conforms to this standard.

CODEC COder-DECoder. Hardware or software that converts analog sound, speech or video to digital code (analog to digital) and vice versa (digital to analog). Hardware codecs are built into devices such as digital telephones and videoconferencing stations. Software codecs are used to record and play audio and video over the web utilizing the CPU for processing. Although hardware codecs are faster than software routines, as desktop machines become more powerful, they can more adequately handle the processing load required for the conversion.

FPGA Field Programmable Gate Array.

GPIO General purpose input/output.

JTAG Joint Test Action Group. The committee which defined the IEEE test access port and boundary-scan standard.

Multi-ICE Multi-ICE is a system for debugging embedded processor cores using a JTAG interface.

MMC MultiMedia Card. A type of removable memory device that consists of a ROM or flash memory within a compact package. The Integrator/IM-PD1 provides a socket into which an MMC or SD can be placed for reading or writing.

MMCI MultiMedia Card Interface.

SCI Smartcard Interface.

SD A type of removable memory device that consists of a ROM or flash memory within a compact package. The Integrator/IM-PD1 provides a socket into which an MMC or SD can be placed for reading or writing.

Smartcard A card of similar size to a credit card that, typically, contains a microcontroller and memory that can be used to store secure data.

SSP Synchronous Serial Port.

UART Universal Asynchronous Receiver/Transmitter.

USB Universal Serial Bus.

VIC Vectored Interrupt Controller. The PrimeCell VIC provides an interface to the interrupt system, and improves interrupt latency in two ways:

- moves the interrupt controller to the AMBA AHB
- provides vectored interrupt support for high-priority interrupt sources.

Video DAC Video Digital to Analog Converter. A device that converts digital data into analog signals for a display monitor. The Integrator/IM-PD1 provides a video DAC that converts 24-bit parallel data into red, green, and blue signals for a display and generates horizontal and vertical synchronization signals from a clock input.

ZBT SSRAM Zero Bus Turnaround Synchronous Static Random Access Memory.

Index

The items in this index are listed in alphabetical order, with symbols and numerics appearing at the end. The references given are to page numbers.