Arm® MPS2 and MPS2+ FPGA Prototyping Boards

Technical Reference Manual

arm

100112_0200_09_en

Arm® MPS2 and MPS2+ FPGA Prototyping Boards

Technical Reference Manual

Release Information

Document History

Issue	Date	Confidentiality	Change
DDI0525A	20 December 2013	Non-Confidential	First issue of TRM
DDI0525B	1 October 2014	Non-Confidential	Second issue of TRM
DDI0525C	10 November 2014	Non-Confidential	Third issue of TRM
0100-03	2 April 2015	Non-Confidential	Fourth issue of TRM
0200-04	2 September 2015	Non-Confidential	Fifth issue of TRM
0200-05	11 January 2016	Non-Confidential	Sixth issue of TRM
0200-06	18 July 2016	Non-Confidential	Seventh issue of TRM
0200-07	27 April 2018	Non-Confidential	Eighth issue of TRM
0200-08	15 February 2019	Non-Confidential	Ninth issue of TRM
0200-09	07 January 2020	Non-Confidential	Tenth issue of TRM

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

LES-PRE-20349

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

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

Web Address

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 Conformity

The *Waste Electrical and Electronic Equipment* (WEEE) marking, that is, the crossed out wheelie-bin figure, indicates that this product must not be disposed of with general waste within the European Union. To prevent possible harm to the environment from uncontrolled waste disposal, the user is required to recycle the product responsibly to promote reuse of material resources. To comply with EU law, you must dispose of the product in one of the following ways:

- Return it to the distributer where it was purchased. The distributer is required to arrange free collection when requested.
- Recycle it using local WEEE recycling facilities. These facilities are now very common and might provide free collection.
- If purchased directly from Arm, Arm provides free collection. Please e-mail *weee@arm.com* for instructions.

The CE Declaration of Conformity for this product is available on request.

The system should be powered down when not in use.

It is recommended that ESD precautions be taken when handling this product.

The product generates, uses, and can radiate radio frequency energy and may cause harmful interference to radio communications. 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 any sensitive equipment.
- 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
Arm® MPS2 and MPS2+ FPGA Prototyping Boards Technical Reference Manual

Preface

About this book 8
Feedback 11

Chapter 1 Introduction

1.1 Precautions 1-13
1.2 About the MPS2 and MPS2+ FPGA Prototyping Boards 1-15
1.3 Location of components on the MPS2 FPGA Prototyping Board 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board 1-19

Chapter 2 Hardware Description

2.1 Overview of the MPS2 and MPS2+ hardware 2-22
2.2 Clocks 2-25
2.3 Powerup, powerdown, and resets 2-27
2.4 User expansion port 2-28
2.5 USB 2.0 Full Speed interface 2-29
2.6 SPI interface 2-30
2.7 UART interface 2-31
2.8 VGA and CLCD interfaces 2-32
2.9 Audio interface 2-33
2.10 Ethernet interface 2-34
2.11 User switches and user LEDs 2-35

2.12 External user memory 2-36
2.13 MCC FPGA serial interface 2-39
2.14 Power 2-42
2.15 Debug and trace 2-43
2.16 Minimum design settings for board operation 2-47

Chapter 3 Configuration
3.1 Overview of the configuration process 3-49
3.2 Remote USB operation 3-51
3.3 Configuration system 3-52
3.4 Powerup and configuration process 3-53
3.5 Reset push buttons 3-55
3.6 Configuration files 3-56

Chapter 4 Programmers Model
4.1 About this programmers model 4-60
4.2 Memory map 4-61
4.3 Register summary 4-62
4.4 SCC register descriptions 4-63
4.5 System configuration registers 4-70

Chapter 5 Signal Descriptions
5.1 Debug connectors 5-75
5.2 Expansion connectors 5-80
5.3 CLCD connector 5-82
5.4 USB 2.0 connector 5-83
5.5 UART connector 5-84
5.6 SPI connector 5-85
5.7 VGA connector 5-86
5.8 Audio connectors 5-87
5.9 Ethernet connector 5-88
5.10 12V power connector 5-89

Appendix A Specifications
A.1 Electrical specification Appx-A-91

Appendix B Revisions
B.1 Revisions Appx-B-93

Preface

This preface introduces the *Arm® MPS2 and MPS2+ FPGA Prototyping Boards Technical Reference Manual*.

It contains the following:

- *About this book* on page 8.
- *Feedback* on page 11.

About this book

This book describes the Arm® MPS2 and MPS2+ FPGA Prototyping Boards. Early issues of this book had document number DDI0525.

Intended audience

This book is written for experienced hardware and software engineers who are doing Arm M-Class processor evaluation and development using the Cortex®-M Prototyping System, on either the MPS2 or MPS2+ FPGA Prototyping Board.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction
This chapter provides an introduction to the MPS2 and MPS2+ FPGA Prototyping Boards.

Chapter 2 Hardware Description
This chapter describes the MPS2 and MPS2+ board hardware.

Chapter 3 Configuration
This chapter describes the powerup and configuration process of the MPS2 and MPS2+ FPGA Prototyping Boards.

Chapter 4 Programmers Model
This chapter describes the programmers model of the MPS2 and MPS2+ FPGA Prototyping Boards.

Chapter 5 Signal Descriptions
This chapter describes the signals present at the interface connectors of the MPS2 and MPS2+ FPGA Prototyping Boards.

Appendix A Specifications
This chapter contains the electrical specification of the MPS2 and MPS2+ FPGA Prototyping Boards and FPGAs.

Appendix B Revisions
This chapter describes the technical changes between released issues of this book.

Glossary

The Arm® Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the *Arm® Glossary* for more information.

Typographic conventions

italic
Introduces special terminology, denotes cross-references, and citations.

bold
Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

`monospace`
Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

`mono`space
Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

monospace italic
: Denotes arguments to monospace text where the argument is to be replaced by a specific value.

monospace bold
: Denotes language keywords when used outside example code.

<and>
: Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example:

```
MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>
```

SMALL CAPITALS
: Used in body text for a few terms that have specific technical meanings, that are defined in the *Arm® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Timing diagrams

The following figure explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are undefined, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.

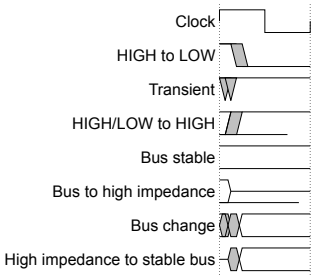

Figure 1 Key to timing diagram conventions

Signals

The signal conventions are:

Signal level
: The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:
- HIGH for active-HIGH signals.
- LOW for active-LOW signals.

Lowercase n
: At the start or end of a signal name, n denotes an active-LOW signal.

Additional reading

This book contains information that is specific to this product. See the following documents for other relevant information.

Arm publications

- *Cortex®-M0+ Technical Reference Manual* (Arm DDI 0484)
- *Cortex®-M0 Technical Reference Manual* (Arm DDI 0432)
- *Cortex®-M1 FPGA Development Kit Cortex-M1 User Guide: Altera Edition v1.1 (Arm DUI 0395)*
- *Cortex®-M1 FPGA Development Kit v1.1: Installation Guide* (Arm DSI 0048)
- *Cortex®-M1 Technical Reference Manual* (Arm DDI 0413)
- *Arm® Cortex®-M3 Technical Reference Manual* (Arm 100165)
- *Cortex®-M3 Devices Generic User Guide* (Arm DUI 0552)
- *Arm® Cortex®-M4 Processor Technical Reference Manual* (Arm 100166)
- *Cortex®-M4 Devices Generic User Guide* (Arm DUI 0553)
- *Arm® Cortex®-M System Design Kit Technical Reference Manual* (Arm DDI 0479)
- *Application Note AN382 Arm® Cortex®-M0 SMM on V2M_MPS2* (Arm DAI 0382)
- *Application Note AN383 Arm® Cortex®-M0+ SMM on V2M-MPS2* (Arm DAI 0383)
- *Application Note AN384 Arm® Cortex®-M1 SMM on V2M-MPS2* (Arm DAI 0384)
- *Application Note AN385 Arm® Cortex®-M3 SMM on V2M-MPS2* (Arm DAI 0385)
- *Application Note AN386 Arm® Cortex®-M4 SMM on V2M-MPS2* (Arm DAI 0386)
- *Application Note AN387 Arm® Cortex®-M0 Design Start SMM on V2M-MPS2* (Arm DAI 0387)
- *Application Note AN399 Arm® Cortex®-M7 SMM on V2M-MPS2* (Arm DAI 0399)
- *Application Note AN400 Arm® Cortex®-M7CS SMM on V2M-MPS2* (Arm DAI 0400)
- *Application Note AN502 Adapter for Arduino* (Arm DAI 0502)
- *Cortex®-M1 FPGA Development Kit Example System Tutorial: Altera Cyclone III Edition* (Arm DUI 0430)
- *Arm® DSTREAM System and Interface Design Reference* (Arm DUI 0499)
- *Arm® DSTREAM Setting up the Hardware* (Arm DUI 0481)
- *Arm® DSTREAM and RVI Using the Debug Hardware Configuration Utilities* (Arm DUI 0498)
- *Arm® CoreSight™ Components Technical Reference Manual* (Arm DDI 0314)

Other publications

- See the Altera website *http://altera.com* for information on the Altera Cyclone 5CEA7 FPGA and the Altera Cyclone 5CEA9 FPGA.

Feedback

Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

If you have comments on content then send an e-mail to *errata@arm.com*. Give:

- The title *Arm MPS2 and MPS2+ FPGA Prototyping Boards Technical Reference Manual*.
- The number 100112_0200_09_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

Note

Arm tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Chapter 1
Introduction

This chapter provides an introduction to the MPS2 and MPS2+ FPGA Prototyping Boards.

It contains the following sections:

- *1.1 Precautions* on page 1-13.
- *1.2 About the MPS2 and MPS2+ FPGA Prototyping Boards* on page 1-15.
- *1.3 Location of components on the MPS2 FPGA Prototyping Board* on page 1-17.
- *1.4 Location of components on the MPS2+ FPGA Prototyping Board* on page 1-19.

1.1 Precautions

You can take certain precautions to ensure safety and prevent damage to your MPS2 or MPS2+ FPGA Prototyping Board.

This section contains the following subsections:

- *1.1.1 Ensuring safety* on page 1-13.
- *1.1.2 Operating temperature* on page 1-13.
- *1.1.3 Preventing damage* on page 1-13.
- *1.1.4 Encryption key* on page 1-13.

1.1.1 Ensuring safety

An on-board connector supplies 12V DC to the board.

Warning

- Do not use the board near equipment that is sensitive to electromagnetic emissions, for example, medical equipment.
- Any external 12V DC +/- 10% power supply that is used must be a limited power source.

1.1.2 Operating temperature

The MPS2 and MPS2+ FPGA Prototyping Boards have been tested in the temperature range 15°C to 30°C.

1.1.3 Preventing damage

The board is intended for use within a laboratory or engineering development environment. The board is sensitive to electrostatic discharges and permits electromagnetic emissions.

Caution

To avoid damage to the board, observe the following precautions:

- You must connect the external power supply to the board before powerup to prevent damage.
- Never subject the board to high electrostatic potentials. Observe Electrostatic Discharge (ESD) precautions when handling any board.
- Always wear a grounding strap when handling the board.
- Only hold the board by the edges.
- Avoid touching the component pins or any other metallic element except the metal shielding for the connectors.
- Avoid contact with components on the board which might be hot or sharp.
- Ensure that the voltage on the pins of the FPGA and interface circuitry on the board is at the correct level.
- You must not configure as outputs any FPGA pins that connect directly to other outputs or an external signal source.
- Do not use the board near a transmitter of electromagnetic emissions.

1.1.4 Encryption key

Arm supplies the MPS2 and MPS2+ FPGA Prototyping Boards with encryption keys programmed into the FPGAs.

A battery supplies power to part of the FPGA that stores the AES decryption key. You use this key to enable loading of encrypted images.

Caution

The decryption key is lost when this part of the FPGA loses power. If this happens, you must return the board to Arm for reprogramming of the decryption key.

1.2 About the MPS2 and MPS2+ FPGA Prototyping Boards

The MPS2 and MPS2+ FPGA Prototyping Boards are development platforms for Arm Cortex-M evaluation and development.

The MPS2 and MPS2+ FPGA Prototyping Boards provide the following:

Altera Cyclone FPGA and board powerup and configuration

The MPS2 FPGA Prototyping Board provides an Altera Cyclone 5CEA7 FPGA, and the MPS2+ FPGA Prototyping Board provides an Altera Cyclone 5CEA9 FPGA, both speed grade C8. Both boards support Arm Cortex-M software evaluation and development. A *Motherboard Configuration Controller* (MCC) supports board powerup and configuration. An on-board EEPROM stores board and file identification information and a microSD card stores FPGA and software images and configuration files. You can access the microSD card to perform configuration file editing and to update FPGA and software images.

——— **Caution** ———

Images that are created for the Altera Cyclone 5CEA7 FPGA are not compatible with the Altera Cyclone 5CEA9 FPGA. Images that are created for the Altera Cyclone 5CEA9 FPGA are not compatible with the Altera Cyclone 5CEA7 FPGA.

———————————

External user memory

On-board external SSRAM and PSRAM connect to memory interfaces in the FPGA.

Access ports

The MPS2 and MPS2+ FPGA Prototyping Boards provide access through Ethernet, general-purpose UART, and SPI ports. A general-purpose user expansion port supports user expansion to extra signal or bus I/O.

Video and audio output

The MPS2 and MPS2+ FPGA Prototyping Boards provide video output through VGA and CLCD ports. The CLCD port drives an LCD module that is configured for SPI graphics and I^2C touch screen. Input and output audio ports connect to a stereo audio codec which connects to an I^2S digital audio interface on the FPGA.

User LEDs and user switches

The MPS2 and MPS2+ FPGA Prototyping Boards provide user LEDs, an 8-way dip switch and push buttons that connect to the FPGA and to the MCC. The meaning of these LEDs and push buttons depend on the image that you implement in the FPGA.

System LEDs

The MPS2 and MPS2+ FPGA Prototyping Boards provide LEDs which denote *Configuration Complete*, *MCC Powered*, *Ethernet Duplex Link Established*, *Ethernet Link Operating at 100Mbs*, *Ethernet Link Established*, and *microSD Card Read or Write Access*.

Debug

The MPS2 and MPS2+ FPGA Prototyping Boards support P-JTAG Processor debug, F-JTAG *Integrated Logic Analyzer* (ILA) FPGA debug, 4-bit trace and 16-bit trace debug, and CMSIS-DAP FPGA debug.

——— **Note** ———

The MPS2 and MPS2+ FPGA Prototyping Boards require MCC firmware version 2.0.1 or later to support CMSIS-DAP FPGA debug.

———————————

1.3 Location of components on the MPS2 FPGA Prototyping Board

The following figure shows the upper face of the MPS2 FPGA Prototyping Board.

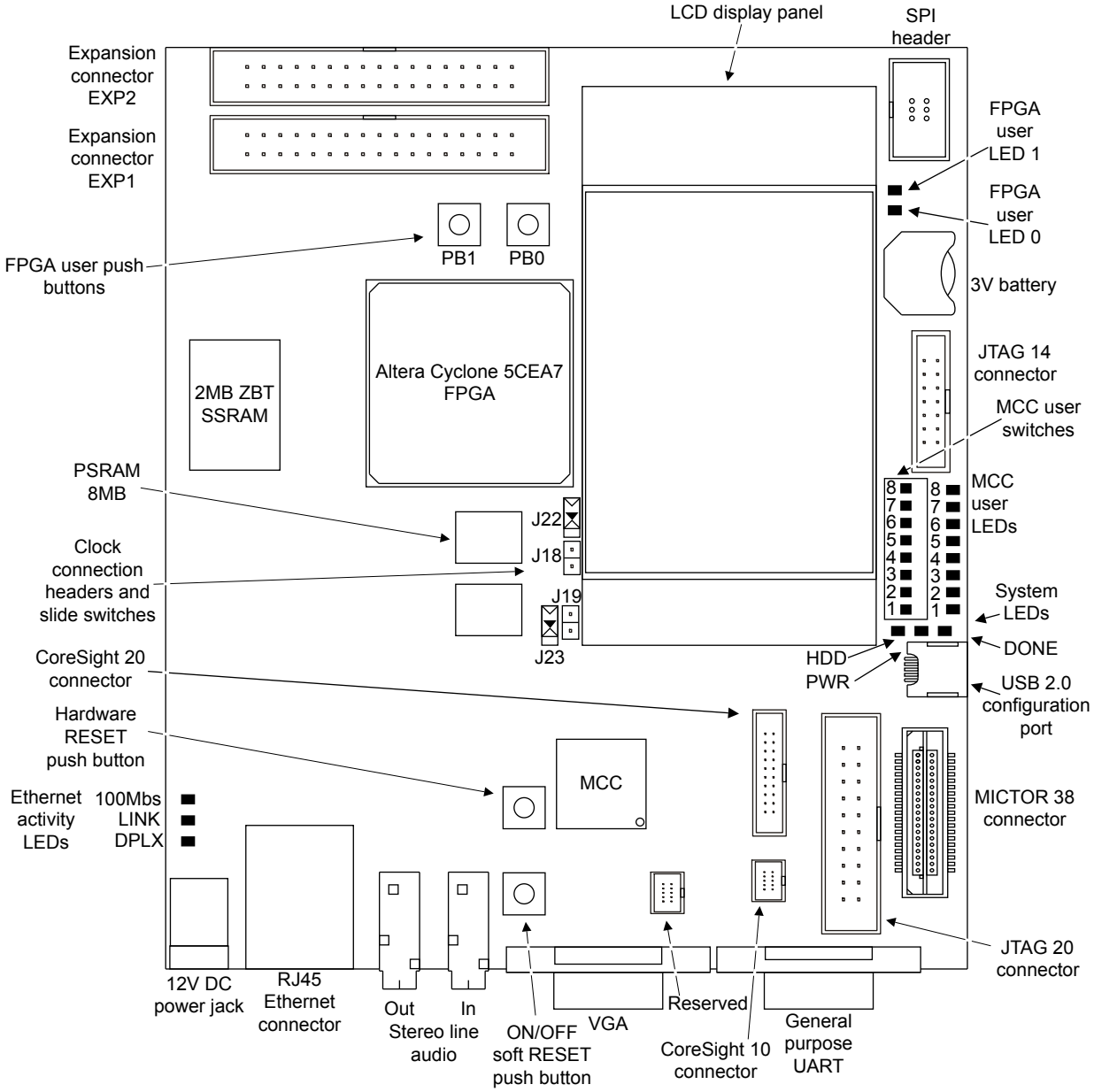

Figure 1-1 Upper face of the MPS2 FPGA Prototyping Board.

The following figure shows the lower face of the MPS2 FPGA Prototyping Board.

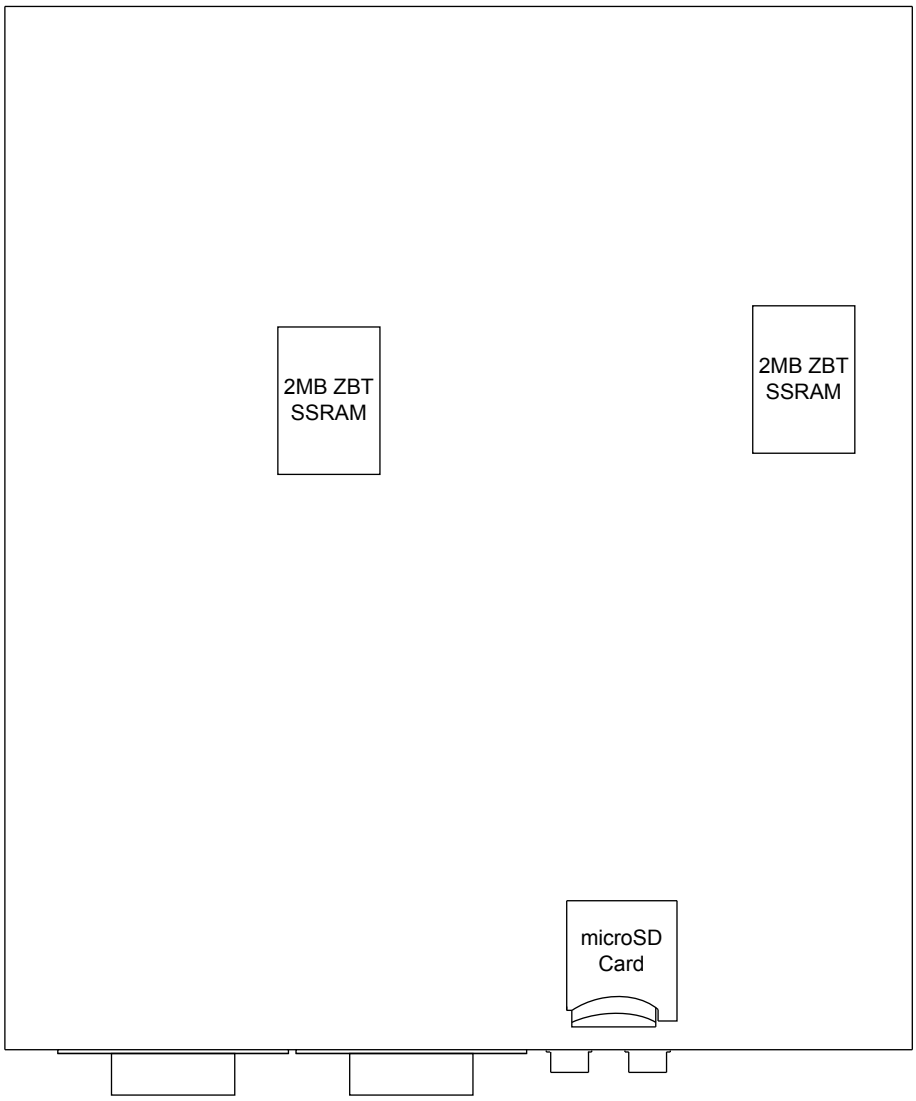

Figure 1-2 Lower face of the MPS2 FPGA Prototyping Board.

1.4 Location of components on the MPS2+ FPGA Prototyping Board

The following figure shows the upper face of the MPS2+ FPGA Prototyping Board.

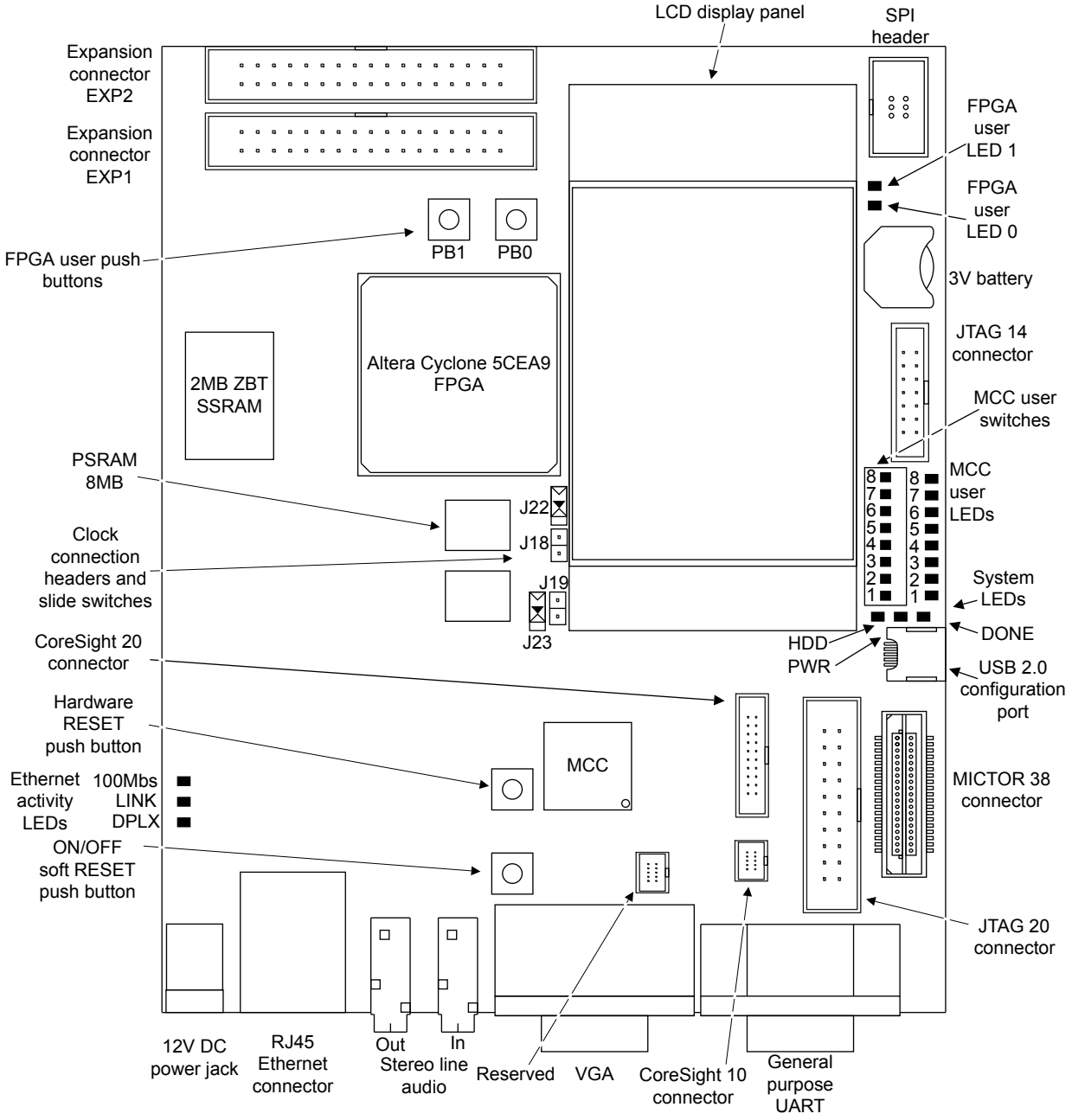

Figure 1-3 Upper face of the MPS2+ FPGA Prototyping Board

The following figure shows the lower face of the MPS2+ FPGA Prototyping Board.

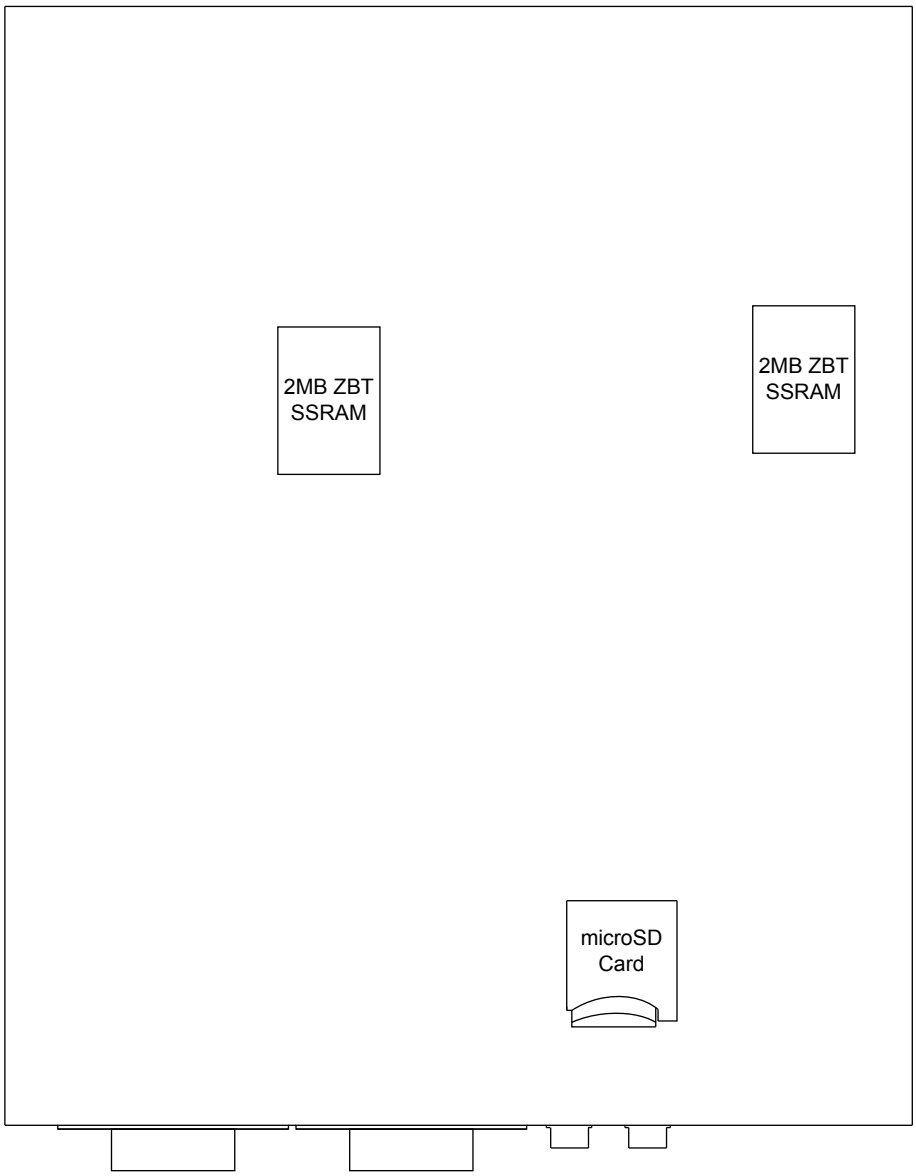

Figure 1-4 Lower face of the MPS2+ FPGA Prototyping Board.

Chapter 2
Hardware Description

This chapter describes the MPS2 and MPS2+ board hardware.

It contains the following sections:

- *2.1 Overview of the MPS2 and MPS2+ hardware* on page 2-22.
- *2.2 Clocks* on page 2-25.
- *2.3 Powerup, powerdown, and resets* on page 2-27.
- *2.4 User expansion port* on page 2-28.
- *2.5 USB 2.0 Full Speed interface* on page 2-29.
- *2.6 SPI interface* on page 2-30.
- *2.7 UART interface* on page 2-31.
- *2.8 VGA and CLCD interfaces* on page 2-32.
- *2.9 Audio interface* on page 2-33.
- *2.10 Ethernet interface* on page 2-34.
- *2.11 User switches and user LEDs* on page 2-35.
- *2.12 External user memory* on page 2-36.
- *2.13 MCC FPGA serial interface* on page 2-39.
- *2.14 Power* on page 2-42.
- *2.15 Debug and trace* on page 2-43.
- *2.16 Minimum design settings for board operation* on page 2-47.

2.1 Overview of the MPS2 and MPS2+ hardware

The hardware infrastructure supports Arm M-class processor evaluation and development, system expansion, and debug interfaces.

The following figure shows the high-level hardware infrastructure.

Note

The image that the configuration systems loads into the FPGA at powerup defines the functionality of the MPS2 and MPS2+ FPGA Prototyping Boards.

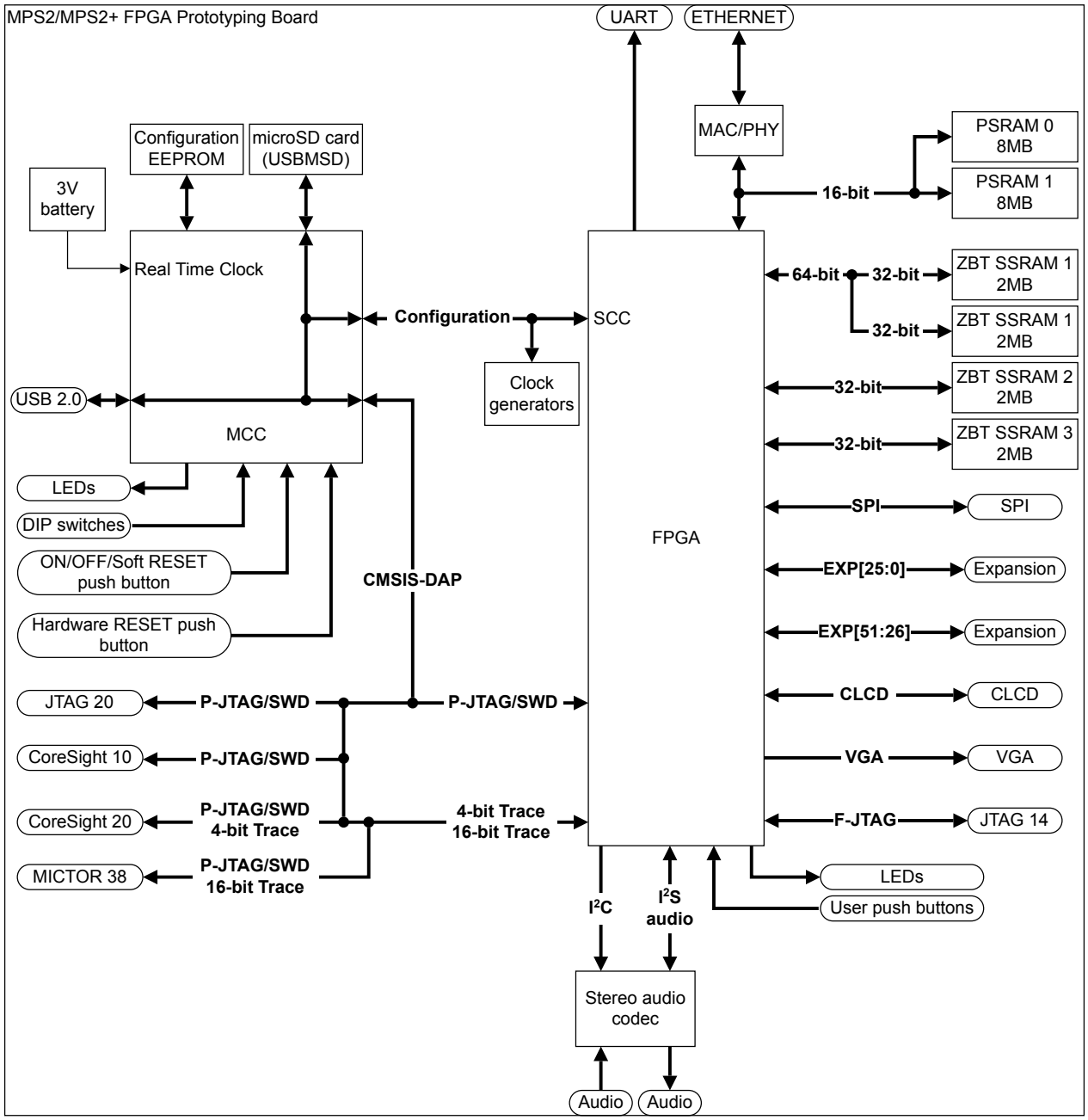

Figure 2-1 Board hardware infrastructure

The MPS2 and MPS2+ FPGA Prototyping Boards contain the following components and interfaces:

- One Altera Cyclone 5CEA7 FPGA on the MPS2 FPGA Prototyping Board:
  - Speed grade C8.
- One Altera Cyclone 5CEA9 FPGA on the MPS2+ FPGA Prototyping Board:
  - Speed grade C8.
- External user system memory for Cortex-M processors:
  - Two 32-bit 2MB ZBT SSRAM with independent 32-bit memory interfaces.
  - One 64-bit 4MB ZBT SSRAM:
    - Two 32-bit 2MB ZBT SSRAM connected as one 4MB 64-bit memory.
  - Two 16-bit 8MB PSRAM to supplement ZBT SSRAM.
- One MCC that supports board configuration at powerup or reset:
  - FPGA configuration.
  - Clock generator configuration.
  - Board configuration.
  - Pre-loading of SRAM images.
  - Loading of *Real Time Clock* (RTCC) registers.
  - CMSIS-DAP FPGA debug through the USB 2.0 port.
- One microSD card that stores the following:
  - FPGA images.
  - Software images.
  - Board configuration files.
- On-board clock generators:
  - One fixed 25MHz clock for Ethernet MAC/PHY.
  - Three programmable system clocks.
  - Two crystals for MCC.
- *Real Time Clock* (RTC) in MCC:
  - Powered by 3V lithium coin cell battery.
- I^2S digital audio output:
- IDC expansion ports:
- Ethernet port.
- UART.
- SPI interface.
- Video output:
  - VGA output.
  - CLCD output with SPI interface.
- USB 2.0 Full Speed port that supports:
  - USB memory access to the microSD card for *Drag-and-Drop* configuration file editing.
- User switches and user LEDs:
  - Two green LEDs and two push buttons that connect to the FPGA.
  - Eight green LEDs and one 8-way dip switch that connect to the MCC.
- System LEDs:
  - *DONE* green LED that denotes board powerup and configuration complete.
  - *PWR* green LED that denotes MCC powered up and active.
  - *HDD* green LED that flashes during access to microSD card.
  - *LINK* green LED that denotes Ethernet activity.
  - *DPLX* green LED that denotes Ethernet connection operating in duplex mode.
  - *100Mbs* green LED that denotes Ethernet connection operating at 10Mbs or 100Mbs.
- Debug and trace interfaces:
  - *JTAG 20* connector that supports P-JTAG Processor debug and SWD.
  - *CoreSight 10* connector that supports P-JTAG Processor debug and SWD.
  - *CoreSight 20* connector that supports P-JTAG Processor debug, SWD, and 4-bit trace.
  - *MICTOR 38* connector that supports P-JTAG Processor debug, SWD, and 16-bit trace.
  - *JTAG 14* connector that supports F-JTAG FPGA debug.
  - USB 2.0 Full Speed port that supports CMSIS-DAP FPGA debug.

Note

— The availability of P-JTAG, SWD, 4-bit trace or 16-bit trace, F-JTAG FPGA debug or CMSIS-DAP FPGA debug depends on the design which you implement in the FPGA.
— CMSIS-DAP debug accesses the FPGA on the same bus as P-JTAG/SWD.
— The MPS2 and MPS2+ FPGA Prototyping Boards require *Motherboard Configuration Controller* (MCC) firmware version 2.0.1 or later to support CMSIS-DAP.

2.2 Clocks

The MPS2 and MPS2+ FPGA Prototyping Boards each provide three programmable on-board clock generators.

Overview of clocks

Three clock generators on the MPS2 and MPS2+ FPGA Prototyping Boards and other clock generators inside the FPGA generate the clocks that the board uses. The MCC configures the programmable clock generators during powerup sequencing using values that the configuration files define.

You configure the frequencies of the on-board clock generators by editing the *OSCLKS* section of the application note `.txt` file in the microSD card.

You can bypass the on-board clock generators and import external clocks to the MCC board using the connection headers that the MPS2 and MPS2+ FPGA Prototyping Boards provide.

The following figure shows the clocks on the MPS2 and MPS2+ FPGA Prototyping Boards. The figure shows two of the clock generators that connect to PLLs inside the FPGA with external loopback to logic inside the FPGA. The third clock generator connects directly to logic inside the FPGA.

Note

- The slide switches and connection headers are for general-purpose use. You can use them for any purpose whatsoever according to the needs of the design which you implement in the FPGA.
- The availability of clock generators and logic in the FPGA depend on the design which you implement in the FPGA.

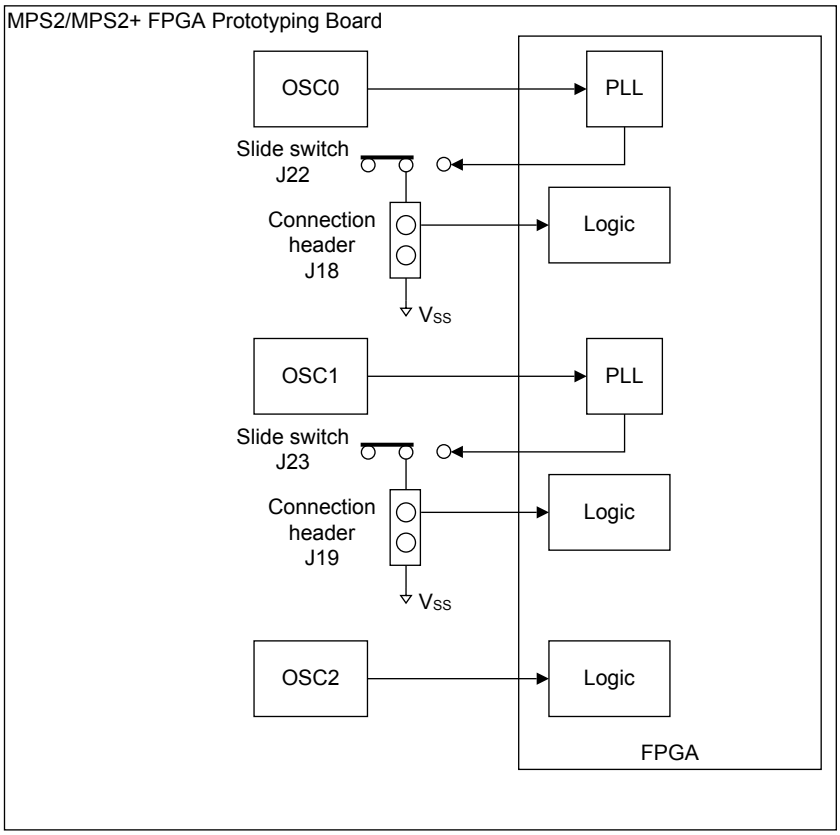

Figure 2-2 Board clocks

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

3.6.2 `config.txt` generic board configuration file on page 3-57

2.3 Powerup, powerdown, and resets

The two reset push buttons generate the reset request signals on the MPS2 and MPS2+ FPGA Prototyping Boards.

When you press one of the reset push buttons, *Hardware RESET* button or *ON/OFF/Soft RESET* button, the MCC generates appropriate signals to reset the system.

The MCC asserts **nRST** after a reset request from the *ON/OFF/Soft RESET* button. Optionally, when you set `ASSERTNPOR` to TRUE in the `config.txt` file, the MCC also asserts **nPOR**.

The following figure shows the powerup *ON/OFF/Soft RESET* and *Hardware RESET* push-button signals in the system.

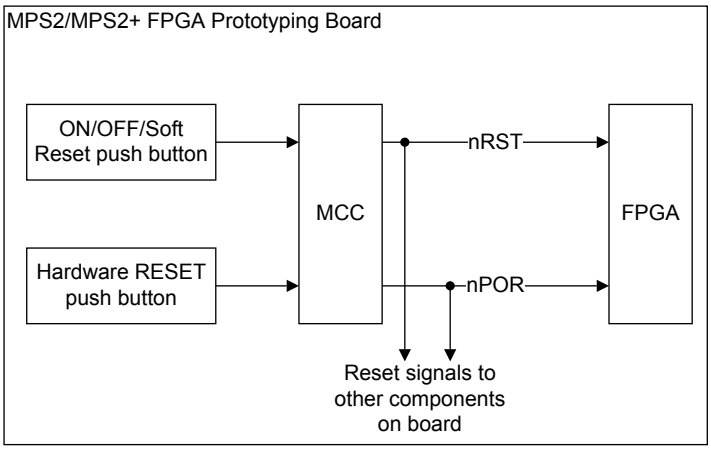

Figure 2-3 Board push-button resets

Note

- The board labels the *ON/OFF/Soft RESET* push button as *nPBON*.
- The board labels the *Hardware RESET* push button as *nCFGRST*.

The MPS2 and MPS2+ FPGA Prototyping Boards perform a full system configuration at powerup.

Related information

3.4 Powerup and configuration process on page 3-53
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
3.6.2 `config.txt` generic board configuration file on page 3-57

2.4 User expansion port

The MPS2 and MPS2+ FPGA Prototyping Boards support expansion to add user functionality.

Overview of the user expansion port

Two IDC headers support general-purpose user expansion.

The general-purpose headers provide:

- Six 3V volt power pins:
  - The 3V supply can supply up to 300mA total through the six power pins.
- 52 general-purpose I/O signals.

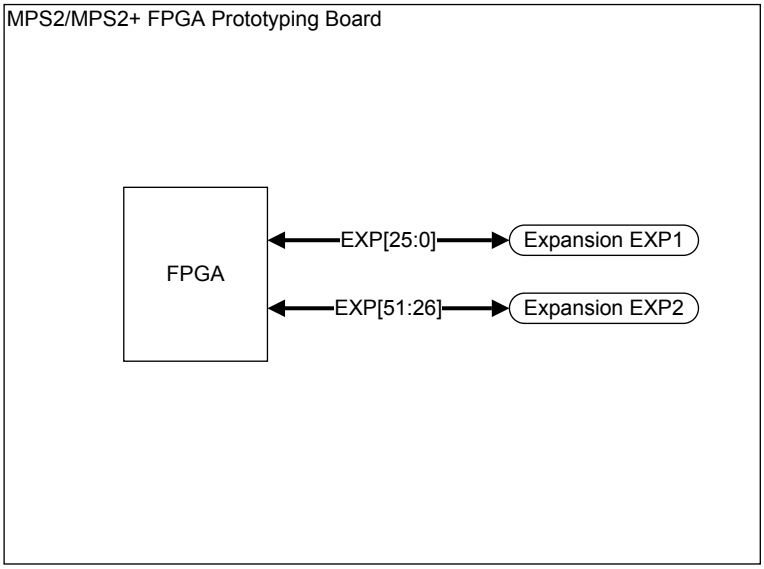

Figure 2-4 Board user expansion port

The Arduino Adapter Board, available from Arm, supports connectivity to up to two Arduino form factor shield boards. See *Application Note AN502 Adapter for Arduino* for information on the Arduino Adapter Board.

Related information

5.2 Expansion connectors on page 5-80

2.5 USB 2.0 Full Speed interface

The MPS2 and MPS2+ Boards provide one USB 2.0 Full Speed interface that connects to the MCC.

In the standby state or during runtime, the USB 2.0 Full Speed interface supports memory access to the microSD card for *Drag-and-Drop* configuration file editing. The board then uses the updated files during the configuration process.

During runtime, the USB 2.0 Full Speed interface supports:

- Virtual UART access to the FPGA through a connection between the MCC and the FPGA.
- Remote USB access to the MCC to enable remote remote reboot, remote reset, and remote shutdown.
- CMSIS-DAP FPGA debug through a connection between the MCC and the FPGA.

The following figure shows the USB 2.0 Full Speed interface.

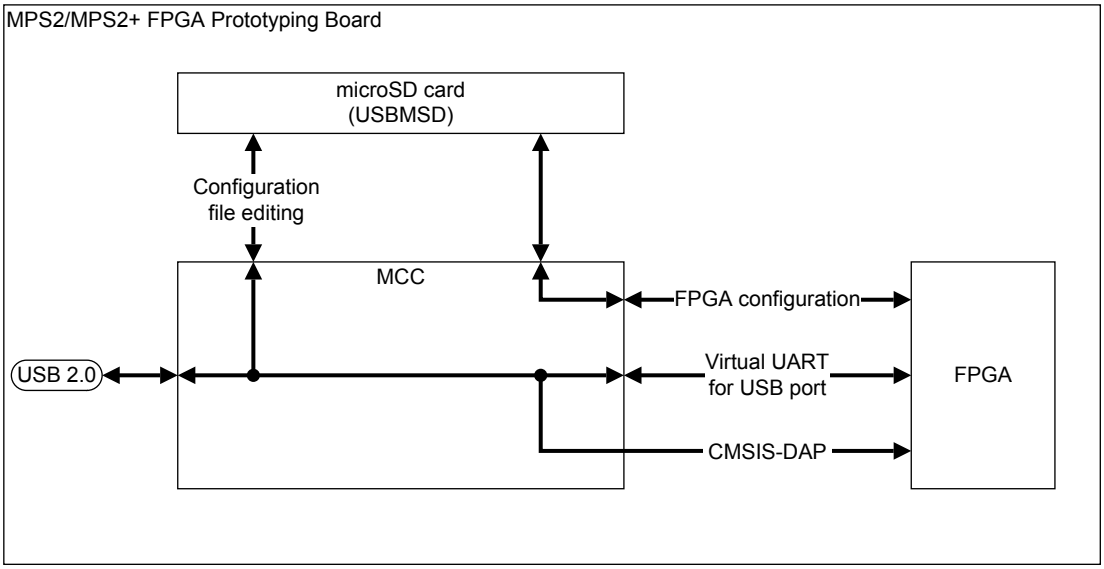

Figure 2-5 Board USB 2.0 Full Speed interface

Note

The designs that Arm supplies do not implement virtual UART access to the FPGA. If you implement the virtual UART, you must make the following pin assignments in the FPGA to enable the virtual UART to operate.

```
assign CLCD_PDH[10]    =  uart_txd;
assign uart_rxd_mcu_i  =  CLCD_PDH[11];
```

Note

After configuration, the board uses the FPGA configuration interface as the SCC and SPI interface.

Related information

3.1 Overview of the configuration process on page 3-49
3.3 Configuration system on page 3-52
5.4 USB 2.0 connector on page 5-83
3.6.1 Overview of configuration files and microSD card directory structure on page 3-56
2.15.7 CMSIS-DAP FPGA debug on page 2-46
3.2 Remote USB operation on page 3-51

2.6 SPI interface

The MPS2 and MPS2+ FPGA Prototyping Boards provides access to a general-purpose SPI interface from the FPGA.

The following figure shows the general-purpose SPI interface.

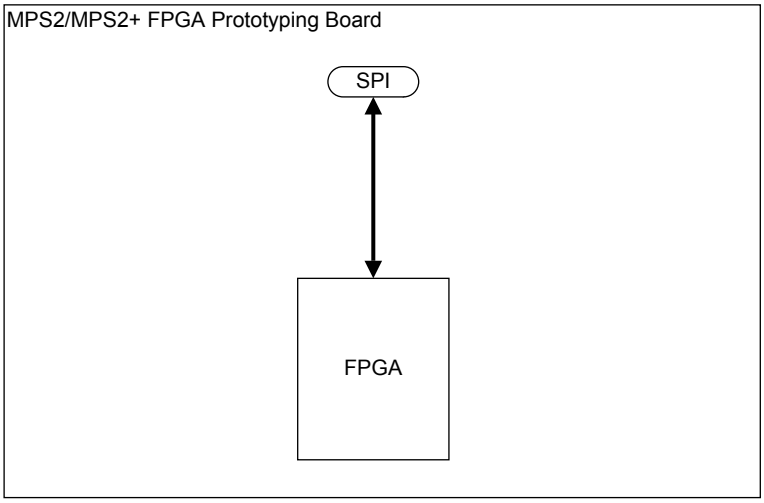

Figure 2-6 MPS2 and MPS2+ FPGA Prototyping Boards general-purpose SPI interface

Related information

5.6 SPI connector on page 5-85

2.7 UART interface

The MPS2 and MPS2+ FPGA Prototyping Boards provide access to a general-purpose UART interface from the FPGA in Data Communications Equipment (DCE) configuration.

The following figure shows the general-purpose UART interface.

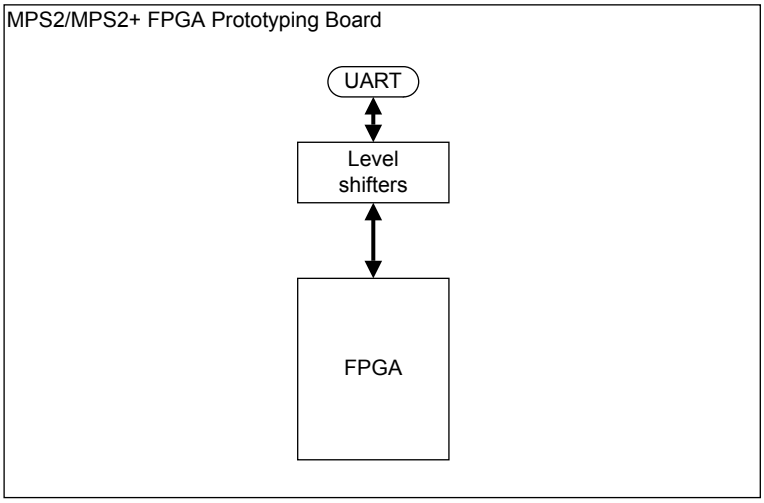

Figure 2-7 General-purpose UART interface

Related information

5.5 UART connector on page 5-84

2.8 VGA and CLCD interfaces

The MPS2 and MPS2+ FPGA Prototyping Boards support VGA and CLCD video output.

Overview of the VGA interface

The VGA interface consists of a VGA controller in the FPGA, a passive resistor network, and an output VGA port. The VGA output has 4 bits per channel color depth.

Overview of the CLCD interface

The CLCD interface consists of an SPI bus that is connected between the FPGA and a CLCD connector and an LCD touch screen.

The LCD *Thin Film Technology* (TFT) display panel is fitted to the CLCD connector and has 320x240 pixel resolution. The panel includes a 4-wire resistive touch screen and an STEMPE811 touch screen controller that is connected to the FPGA over an I^2C bus.

The following figure shows the board VGA and CLCD interfaces.

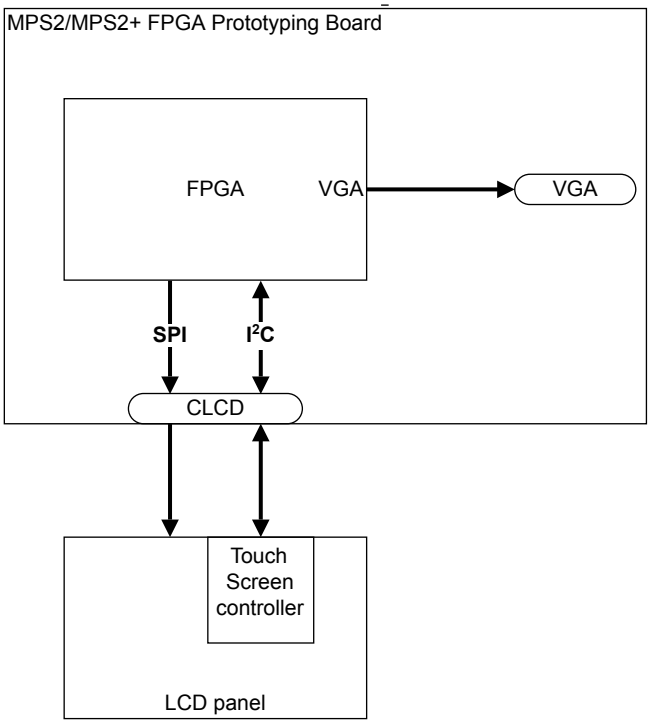

Figure 2-8 VGA and CLCD interfaces.

LCD panel

The LCD panel is configured for SPI graphics and I^2C touch-screen.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.3 CLCD connector on page 5-82

5.7 VGA connector on page 5-86

2.9 Audio interface

The MPS2 and MPS2+ FPGA Prototyping Boards provide a stereo line-level input and a stereo line-level output.

The audio interface consists of an audio controller in the FPGA, a low-power audio codec, and input and output stereo audio ports. The audio controller configures the codec over an I^2C bus. The FPGA and codec send audio data to each other over an I^2S bus.

The following diagram shows the audio interface.

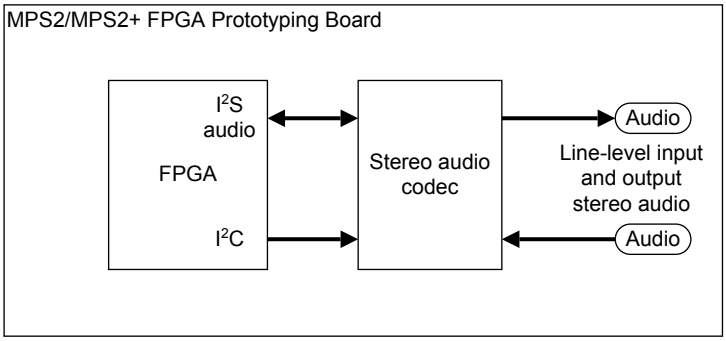

Figure 2-9 Audio interface

Related information

5.8 Audio connectors on page 5-87
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

2.10 Ethernet interface

The MPS2 and MPS2+ FPGA Prototyping Boards provide an Ethernet MAC interface.

Overview of the Ethernet interface.

The Ethernet interface consists of a 16-bit interface on the FPGA, a combined Ethernet MAC and PHY on the board, and Ethernet connector for external connection.

Note

The MAC/PHY connects to the same 16-bit interface as the 16MB PSRAM external memory.

The MPS2 and MPS2+ FPGA Prototyping Boards contain three LEDs that denote Ethernet activity:

- *LINK LED*:
  - On denotes Ethernet connection is established.
  - Off denotes Ethernet connection is not established.
- *DPLX LED*:
  - On denotes connection operating in duplex mode.
  - Off denotes Ethernet connection operating in simplex mode.
- *100Mbs LED*:
  - On denotes Ethernet connection operating at 100Mbs.
  - Off denotes Ethernet connection operating at 10Mbs.

The following diagram shows the Ethernet interface.

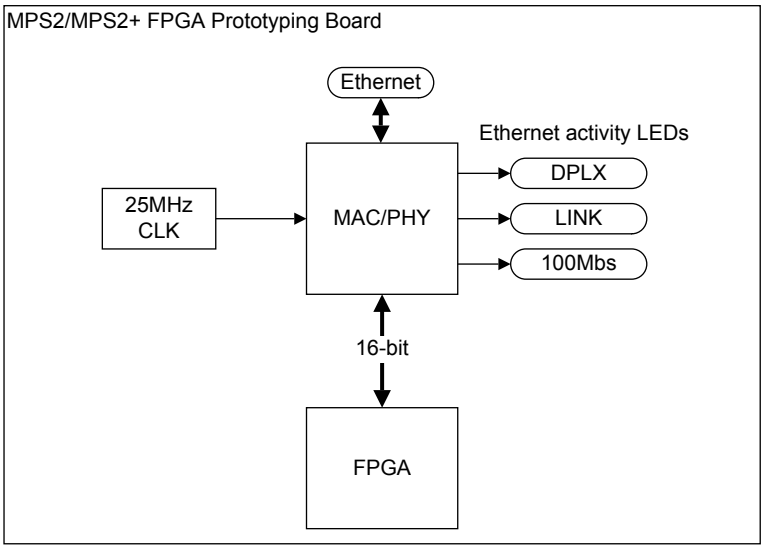

Figure 2-10 Ethernet interface

Related information

5.9 Ethernet connector on page 5-88
2.12 External user memory on page 2-36

2.11 User switches and user LEDs

The MPS2 and MPS2+ FPGA Prototyping Boards provide a user interface that consists of user LEDs and user switches that connect to the FPGA and MCC. The FPGA image and application software define their meaning.

The board provides the following LEDs and switches:

- Two LEDs and two push buttons that connect directly to the FPGA:
  - The board labels the LEDs as *USERLED0* and *USERLED1*.
  - The board labels the push buttons as *PB0* and *PB1*.
- Eight LEDs and an 8-way DIP switch that connect to the MCC.

You can write the values from these LEDs and switches to system registers inside the FPGA using a serial SCC interface.

The following diagram shows the user switches and LEDs that form the FPGA and MCC user interface.

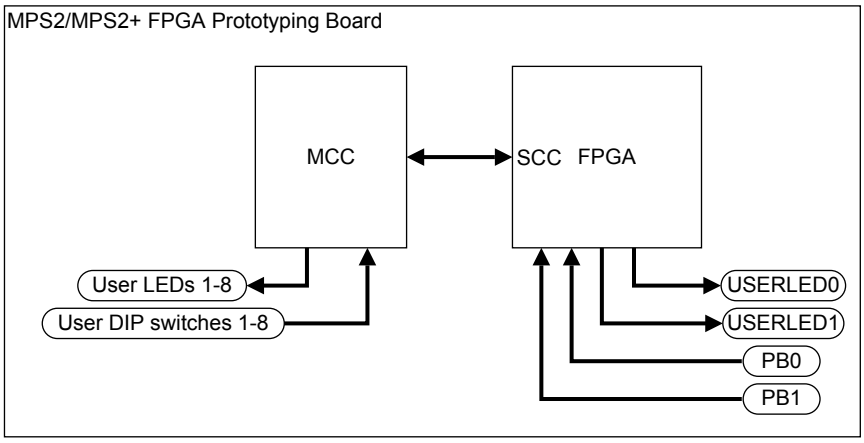

Figure 2-11 FPGA and MCC user interface

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
4.4.3 SCC_CFG1 Register on page 4-64
4.4.4 SCC_CFG3 Register on page 4-64
2.13 MCC FPGA serial interface on page 2-39

2.12 External user memory

The MPS2 and MPS2+ FPGA Prototyping Boards provide external user ZBT SSRAM and PSRAM memory.

The external user memory connects to the FPGA and consists of the following:

- Two 2MB 32-bit fast single-cycle ZBT SSRAM memories:
  - Connect to two independent 32-bit memory interfaces on the FPGA.
- One 4MB 32-bit fast single-cycle ZBT SSRAM memory:
  - Consists of two 2MB 32-bit memories which the board configures as one 64-bit memory. They connect to one 64-bit memory interface on the FPGA.
- Two 16-bit 8MB PSRAM memory:
  - Supplement the ZBT SSRAM.

Note

The 16MB PSRAM connects to the same 16-bit interface on the FPGA as the Ethernet MAC/PHY.

The MCC can pre-load the external user memory using the `images.txt` file before releasing reset, enabling the MPS2 and MPS2+ to use the external memory as boot RAM.

The following diagram shows the external ZBT SSRAM user memory interface.

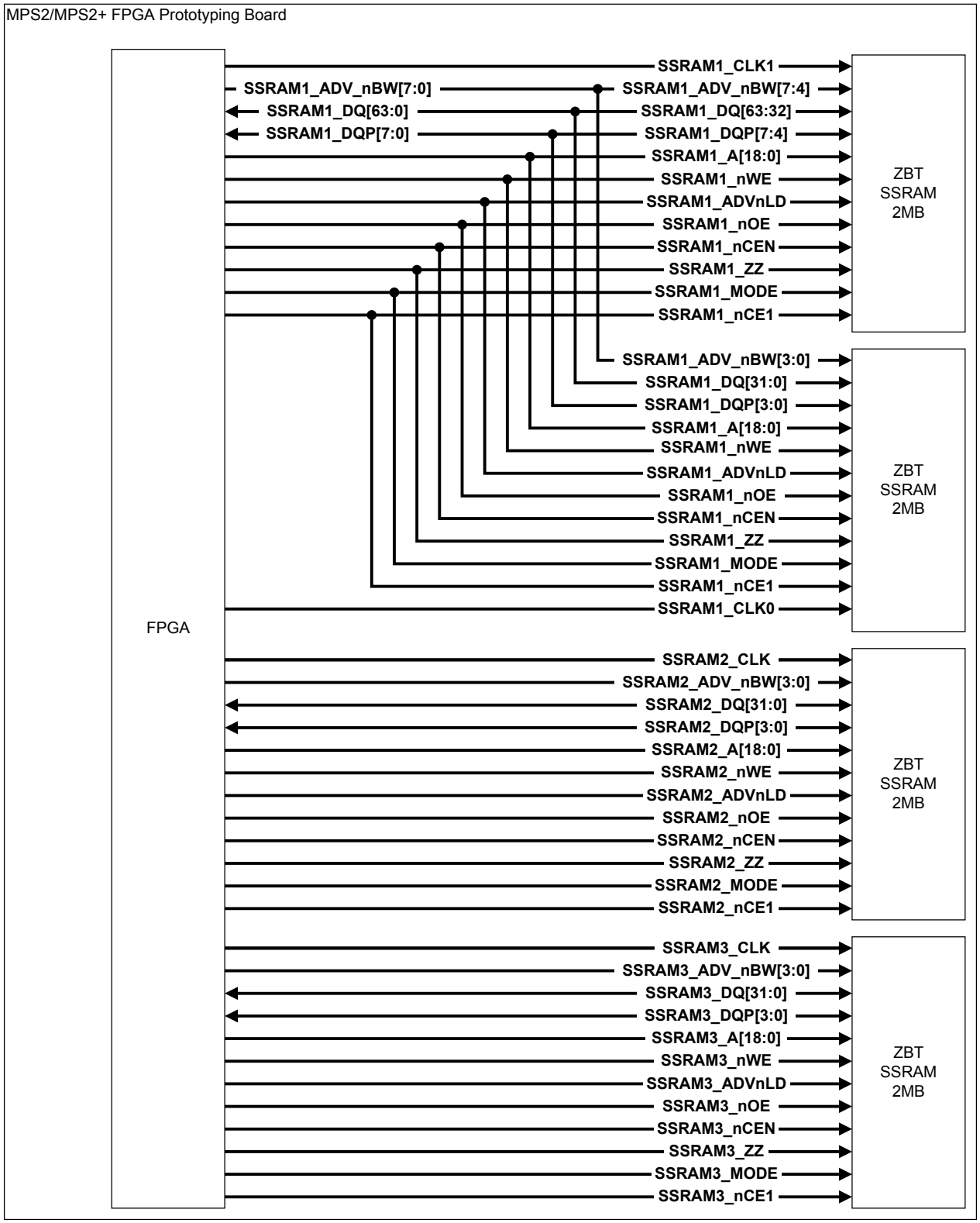

Figure 2-12 ZBT SSRAM external user memory interface

The following diagram shows the external PSRAM user memory interface.

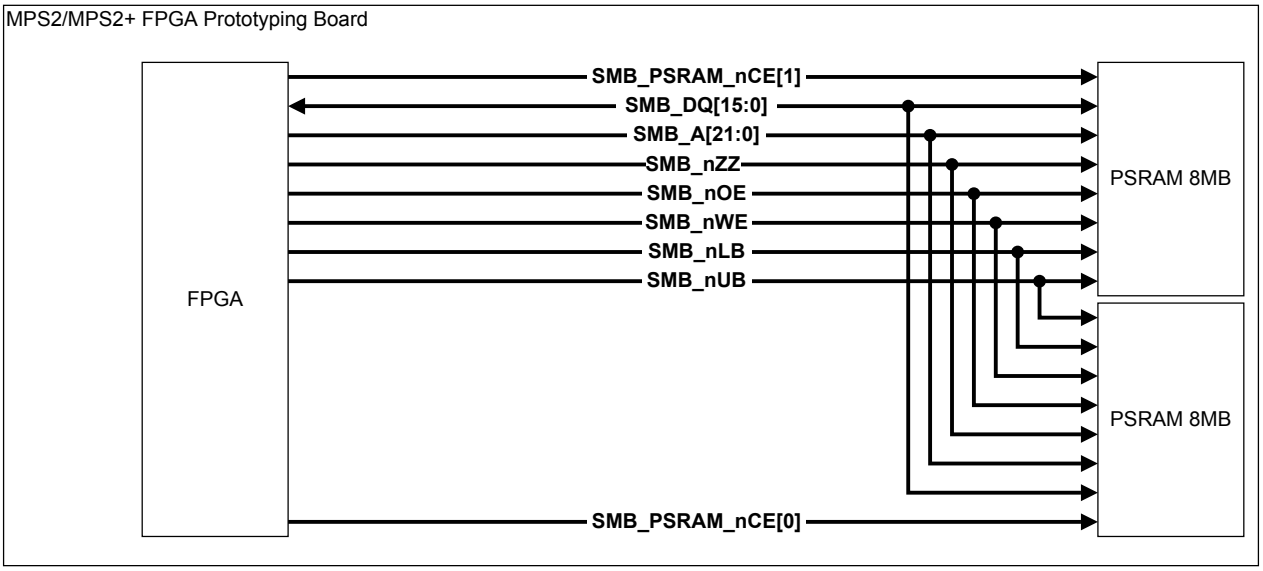

Figure 2-13 PSRAM external user memory interface

Related information

2.10 Ethernet interface on page 2-34

2.13 MCC FPGA serial interface

This section describes the Serial Configuration Controller (SCC) between the MCC and the FPGA.

This section contains the following subsections:

- *2.13.1 Serial Configuration Controller (SCC)* on page 2-39.
- *2.13.2 SCC READ and WRITE operations* on page 2-40.
- *2.13.3 SCC READ and WRITE timings* on page 2-40.

2.13.1 Serial Configuration Controller (SCC)

The MCC uses a serial communication channel to receive from and transmit information to the FPGA on the board. You must implement a Serial Configuration Controller (SCC) in the FPGA.

Overview of MCC-SCC interface

The SCC interface operates at 0.5MHz. The serial interface is similar to a memory-mapped peripheral because it has an address and data phase.

The **nCFGRST** output from the MCC loads the default configuration settings into the FPGA. **CFGLOAD** determines when WRITE DATA is complete, or when the system expects READ DATA to be ready. The MCC provides the SCC clock. **CFGWnR** changes depending on the access type. The SCC operates a 12-bit address and 32-bit data phase.

The following figure shows the MCC-SCC interface.

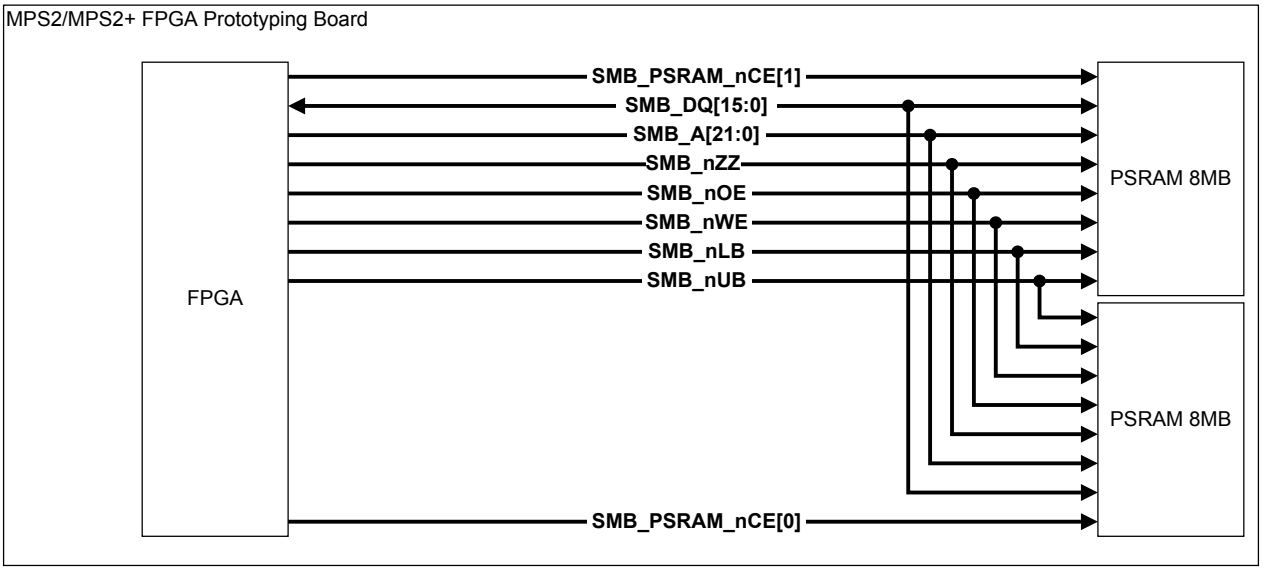

Figure 2-14 MCC-FPGA serial interface

FPGA pin assignments to implement an MCC-SCC interface

If you implement an SCC interface, you must make the following pin assignments in the FPGA:

```
assign CFGCLK         =   CLCD_PDH[13];
assign nCFGRST        =   CLCD_PDH[14];
assign CFGLOAD        =   CLCD_PDH[15];
assign CFGWnR         =   CLCD_PDH[16];
assign CFGDATAIN      =   CLCD_PHD[17];
assign CLCD_PDH[12]   =   CFGDATAOUT;
```

Note

See *Application Note AN387 Arm® Cortex®-M0 Design Start SMM on V2M-MPS2* for an example scc.v file that implements these registers.

Note

If the FPGA does not implement an SCC interface, Arm recommends that you tie off the **CFGDATAOUT** and **nRSTREQ** signals as follows:

- Tie the **CFGDATAOUT** signal from the FPGA LOW.
- Tie the **nRSTREQ** signal from the FPGA HIGH.

Related information

2.16 Minimum design settings for board operation on page 2-47

2.13.2 SCC READ and WRITE operations

During WRITE operations, after clocking in the address and data, **CFGLOAD** loads the address and data into the SCC. The MCC clocks the WRITE DATA Most Significant Bit (MSB) first.

During READ operations, after clocking in the address, **CFGLOAD** loads the read address into the SCC. The SCC clocks the READ DATA Least Significant Bit (LSB) first.

2.13.3 SCC READ and WRITE timings

The following figure shows the timings for the WRITE operation.

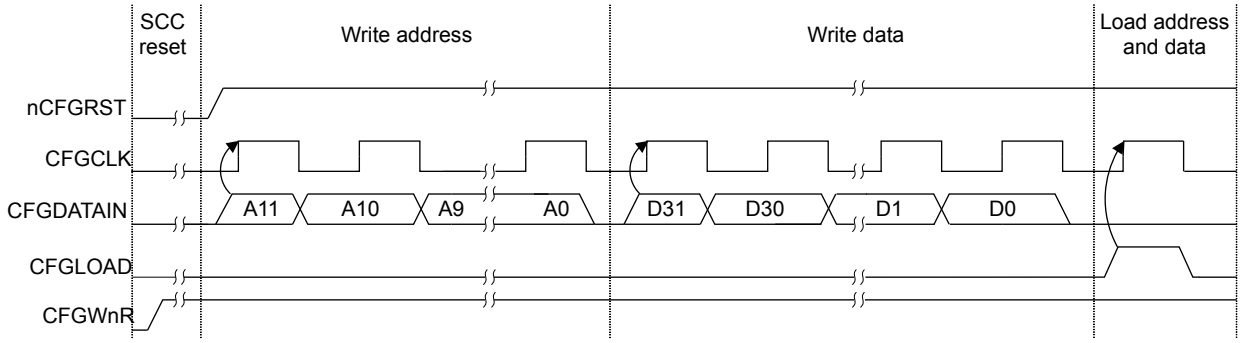

Figure 2-15 MCC write to SCC

The following figure shows the timings for the READ operation.

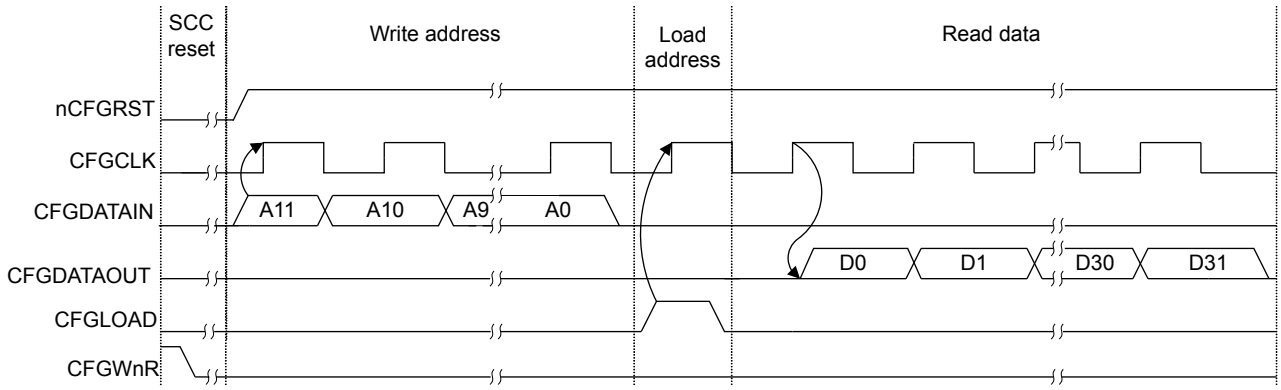

Figure 2-16 MCC read from SCC

The following table shows the MCC AC timing requirements.

Table 2-1 MCC AC timing requirements for SCC interface

Variable	Time
MCC Clock to output valid time: Tov	-500ns
MCC Clock to output invalid time: Toh	500ns
MCC input setup time: Tis	500ns
MCC input hold time: Tih	-500ns

Related information

2.11 User switches and user LEDs on page 2-35
4.3 Register summary on page 4-62
4.4 SCC register descriptions on page 4-63

2.14 Power

You supply power to the board from the mains supply using the on-board connector and the connector cable that Arm supplies with the board.

Arm supplies an external power supply unit which converts mains power to 12V DC which connects to the 12V connector on the board. The unit accepts mains power in the range 110V AC to 240V AC.

——— Warning ———

If you supply your own external mains adaptor, it must be a Low-Power Source. Some such are marked *LPS* on their rating label. Otherwise it might be necessary to consult you adaptor supplier to ensure that it meets this criterion.

———————————

Alternatively, you can connect an external 12V DC supply, +/- 10%, directly to the 12V connector.

——— Warning ———

Any external 12V DC +/- 10% power supply that is used must be a limited power source.

———————————

On-board regulators supply power to the board power domains and to the FPGA power domains.

——— Note ———

The *PWR* LED illuminates when the MCC is powered up and active.

———————————

Related information

5.10 12V power connector on page 5-89
A.1.1 FPGA current requirements on page Appx-A-91

2.15 Debug and trace

This section describes the debug and trace systems on the MPS2 and MPS2+ FPGA Prototyping Boards.

This section contains the following subsections:

- *2.15.1 Overview of FPGA debug and trace systems* on page 2-43.
- *2.15.2 F-JTAG* on page 2-44.
- *2.15.3 P-JTAG* on page 2-45.
- *2.15.4 4-bit Trace* on page 2-45.
- *2.15.5 16-bit Trace* on page 2-45.
- *2.15.6 Serial Wire Debug* on page 2-45.
- *2.15.7 CMSIS-DAP FPGA debug* on page 2-46.

2.15.1 Overview of FPGA debug and trace systems

The MPS2 and MPS2+ FPGA Prototyping Boards provide several debug and trace interfaces:

- P-JTAG processor debug.
- F-JTAG FPGA debug.
- 16-bit parallel Trace.
- 4-bit parallel Trace.
- *Serial Wire Debug* (SWD).
- CMSIS-DAP FPGA debug.

Note

- The availability of system debug depends on the design that you implement in the FPGA.
- The MPS2 and MPS2+ FPGA Prototyping Boards require *Motherboard Configuration Controller* (MCC) firmware version 2.0.1 or later to support CMSIS-DAP.

The following diagram shows a simplified view of the F-JTAG, P-JTAG, Trace, SWD, and CMSIS-DAP connections.

Caution

The total current limit for the four P-JTAG connectors, JTAG 20, MICTOR 38, CoreSight 20, and CoreSight 10 is 50mA.

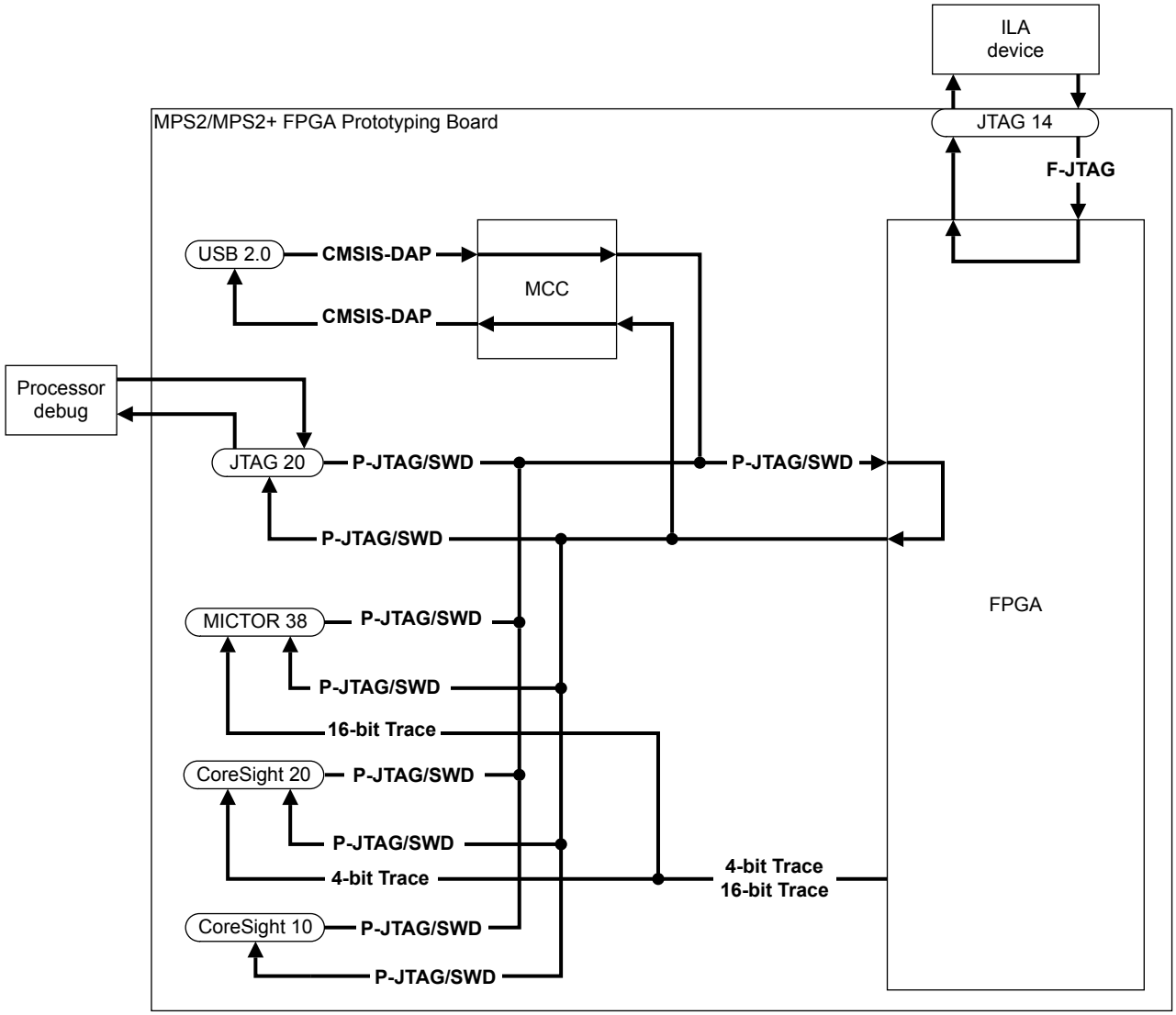

Figure 2-17 Overview of debug and trace

Note

- The ILA device can be any compatible device, for example, SignalTap II.
- The processor debug device can be any compatible debugger, for example SignalTap II.

2.15.2 F-JTAG

The JTAG 14 connector supports F-JTAG, FPGA debug.

The F-JTAG chain connects to the hard TAP controllers in the FPGA. The availability of F-JTAG depends on the design which you implement in the FPGA.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.1.1 JTAG 14 connector on page 5-75

2.15.3 P-JTAG

The MPS2 and MPS2+ FPGA Prototyping Boards provide several connectors that support P-JTAG, processor debug.

- JTAG 20 connector.
- CoreSight 10 connector.
- CoreSight 20 connector.
- MICTOR 38 connector.

The availability of P-JTAG depends on the design which you implement in the FPGA.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
5.1.2 JTAG 20 connector on page 5-76
5.1.3 CoreSight 10 connector on page 5-76
5.1.4 CoreSight 20 connector on page 5-77
5.1.5 MICTOR 38 connector on page 5-78

2.15.4 4-bit Trace

The CoreSight 20 connector supports 4-bit Trace on the MPS2 and MPS2+ FPGA Prototyping Boards.

The availability of 4-bit Trace depends on the design which you implement in the FPGA.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
5.1.4 CoreSight 20 connector on page 5-77

2.15.5 16-bit Trace

The MICTOR 38 connector supports 16-bit Trace on the MPS2 and MPS2+ FPGA Prototyping Boards.

The availability of 16-bit Trace depends on the design which you implement in the FPGA.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
5.1.5 MICTOR 38 connector on page 5-78

2.15.6 Serial Wire Debug

The MPS2 and MPS2+ FPGA Prototyping Boards provide several connectors that support Serial Wire Debug (SWD).

- JTAG 20 connector.
- CoreSight 10 connector.
- CoreSight 20 connector.
- MICTOR 38 connector.

The availability of SWD depends upon the design which you implement in the FPGA.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
5.1.2 JTAG 20 connector on page 5-76
5.1.3 CoreSight 10 connector on page 5-76
5.1.4 CoreSight 20 connector on page 5-77
5.1.5 MICTOR 38 connector on page 5-78

2.15.7 CMSIS-DAP FPGA debug

The USB 2.0 Full Speed ports on the MPS2 and MPS2+ FPGA Prototyping Boards support CMSIS-DAP debug.

A workstation running a debugger connects to the CMSIS-DAP port which connects to the MCC. The MCC implements CMSIS-DAP which runs over JTAG connection to the debug access port in the FPGA.

The availability of CMSIS-DAP depends on the design that you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards require *Motherboard Configuration Controller* (MCC) software version 2.0.1 or later to support CMSIS-DAP.

Related information

2.5 USB 2.0 Full Speed interface on page 2-29

2.16 Minimum design settings for board operation

You must implement a minimum amount of RTL in the FPGA for theMPS2 or MPS2+ FPGA Prototyping Board to operate correctly.

You must tie off the following FPGA signals to generate the minimum RTL in the FPGA for correct operation:

1. Set the following signals to the inactive HIGH state:
   - **SMB_PSRAM_nce[1:0]**.
   - **SMB_ETH_nCS**.
   - **SSRAM1_nCE1**.
   - **SSRAM2_nCE1**
   - **SSRAM3_nCE1**
2. Set the SMB chip select to the inactive HIGH state by tying the chip selects **SMB_nCS** to `0b11111111`.
3. Set the **CFGDATAOUT** signal to the inactive LOW state by tying **NAND_D[5]** to `0b0`.

   Note

   This informs the MCC that the board does not implement any of its features.

4. Set the **nRSTREQ** to the inactive HIGH state by tying **NAND_D[7]** to `0b1`.

   Note

   This prevents **nRSTREQ** from generating a reset. **nRSTREQ** is usually a system-wide master soft reset signal that is both generated and observed by the JTAG debug box.

Note

Arm recommends that you tie all unused pins to their inactive states.

Related information

2.13.1 Serial Configuration Controller (SCC) on page 2-39

Chapter 3
Configuration

This chapter describes the powerup and configuration process of the MPS2 and MPS2+ FPGA Prototyping Boards.

It contains the following sections:

- *3.1 Overview of the configuration process* on page 3-49.
- *3.2 Remote USB operation* on page 3-51.
- *3.3 Configuration system* on page 3-52.
- *3.4 Powerup and configuration process* on page 3-53.
- *3.5 Reset push buttons* on page 3-55.
- *3.6 Configuration files* on page 3-56.

3.1 Overview of the configuration process

The MCC, in association with the microSD card, configures the MPS2 and MPS2+ FPGA Prototyping Boards during powerup or reset. When the configuration process starts after application of power or a press of one of the RESET buttons, the configuration process completes without further intervention from the user.

The microSD card stores the board configuration files, including the `board.txt` and `config.txt` files. You can access the microSD card as a *Universal Serial Bus Mass Storage Device* (USBMSD)

The MCC:

- Reads the FPGA image from the microSD card and loads it into the FPGA.
- Sets the board oscillator frequencies using values from the `board.txt` file.
- Configures the FPGA SCC registers using values from the `board.txt` file
- Pre-loads the SRAM with the boot image that the `images.txt` file defines.

Note

You must make the following pin assignments in the FPGA to enable the MCC to pre-load the boot image into the SRAM.

```
assign config_spiclk    =    CLCD_PDL[6];
assign config_spidi     =    CLCD_PDL[8];
assign CLCD_PDL[7]      =    config_spido;
```

At the start of the configuration process, the MCC reads the contents of the configuration EEPROM. The EEPROM contains the following information:

- Board HBI number.
- Board revision.
- Board variant.
- Number of FPGAs.
- The names of the current images in 8.3 format and the file creation dates.

Note

The HBI number is a unique code that identifies the board. The root directories of the EEPROM and the microSD card contain subdirectories in the form *HBIBoardNumberBoardrevision*, for example HBI0263B.

There are two stages in programming and configuring the images into the FPGA:

1. The MCC compares the file names and dates in the EEPROM to the information in the configuration files in the microSD card. The microSD card contains the FPGA images and information about their creation dates.

   If the microSD card contains an updated image, the MCC validates it and updates the information in the EEPROM.

   Note

   - The USB 2.0 Full Speed connection to the MCC supports MSD class enabling *Drag-and-Drop* for transferring new images to the microSD card. The microSD card appears in the file system as a device with removable storage.
   - The *HDD* LED illuminates during read or write accesses to the microSD card.

2. The MCC loads the FPGA image into the FPGA.
3. The MCC illuminates the *DONE* LED to indicate that configuration is complete.

Related information

2.5 USB 2.0 Full Speed interface on page 2-29

3.2 Remote USB operation

You can control the *Motherboard Configuration Controller* (MCC) through the USB 2.0 configuration port.

To enable remote USB operation, you must set the USB_REMOTE parameter in the `config.txt` file to:

- TRUE to enable the remote USB feature.
- FALSE to disable the feature.

You initiate a command by putting a file into the USBMSD root directory through the USB port. You can reboot the system, reset the system, or shut down the system by using one of the following filenames:

- `reboot.txt`.
- `reset.txt`.
- `shutdown.txt`.

The MCC detects the presence of the files, performs the requested command, and deletes the file.

Note

In versions 1 and 2 of the firmware, the contents of the files have no effect. They can be empty files.

In version 3 of the firmware, the contents of the files have an effect. The software performs reboot, shutdown, and reset by checking for a particular string in the text file on the USB drive. The string must be written at the start of the text file:

- `reboot.txt` must contain "hsyxhj".
- `reset.txt` must contain "jkmcgx".
- `shutdown.txt` must contain "bmqjfe".

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19
2.5 USB 2.0 Full Speed interface on page 2-29
5.4 USB 2.0 connector on page 5-83

3.3 Configuration system

The MPS2 and MPS2+ FPGA Prototyping Boards provide hardware infrastructure to enable board configuration during powerup or reset.

The following diagram shows the board configuration system.

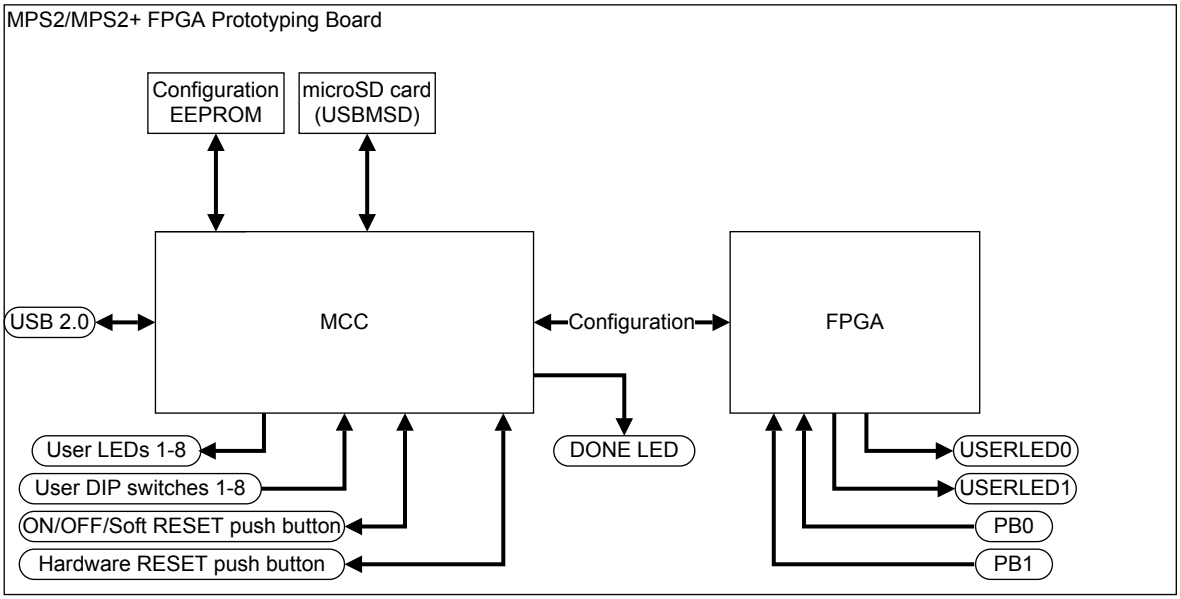

Figure 3-1 Configuration system

Configuration port connected to an external workstation

If you connect an external workstation to the MCC USB 2.0 Full Speed port, you can access the configuration memory in the microSD card. You can then edit and copy configuration files and software images to the SD card.

Related information

2.5 USB 2.0 Full Speed interface on page 2-29
3.6.2 `config.txt` generic board configuration file on page 3-57
3.6.3 Contents of the MB directory on page 3-57
3.6.4 Contents of the SOFTWARE directory on page 3-58
3.6.1 Overview of configuration files and microSD card directory structure on page 3-56

3.4 Powerup and configuration process

The power push buttons and configuration files control the sequence of events of the MPS2 and MPS2+ board powerup and configuration process.

The following figure shows the powerup and configuration process.

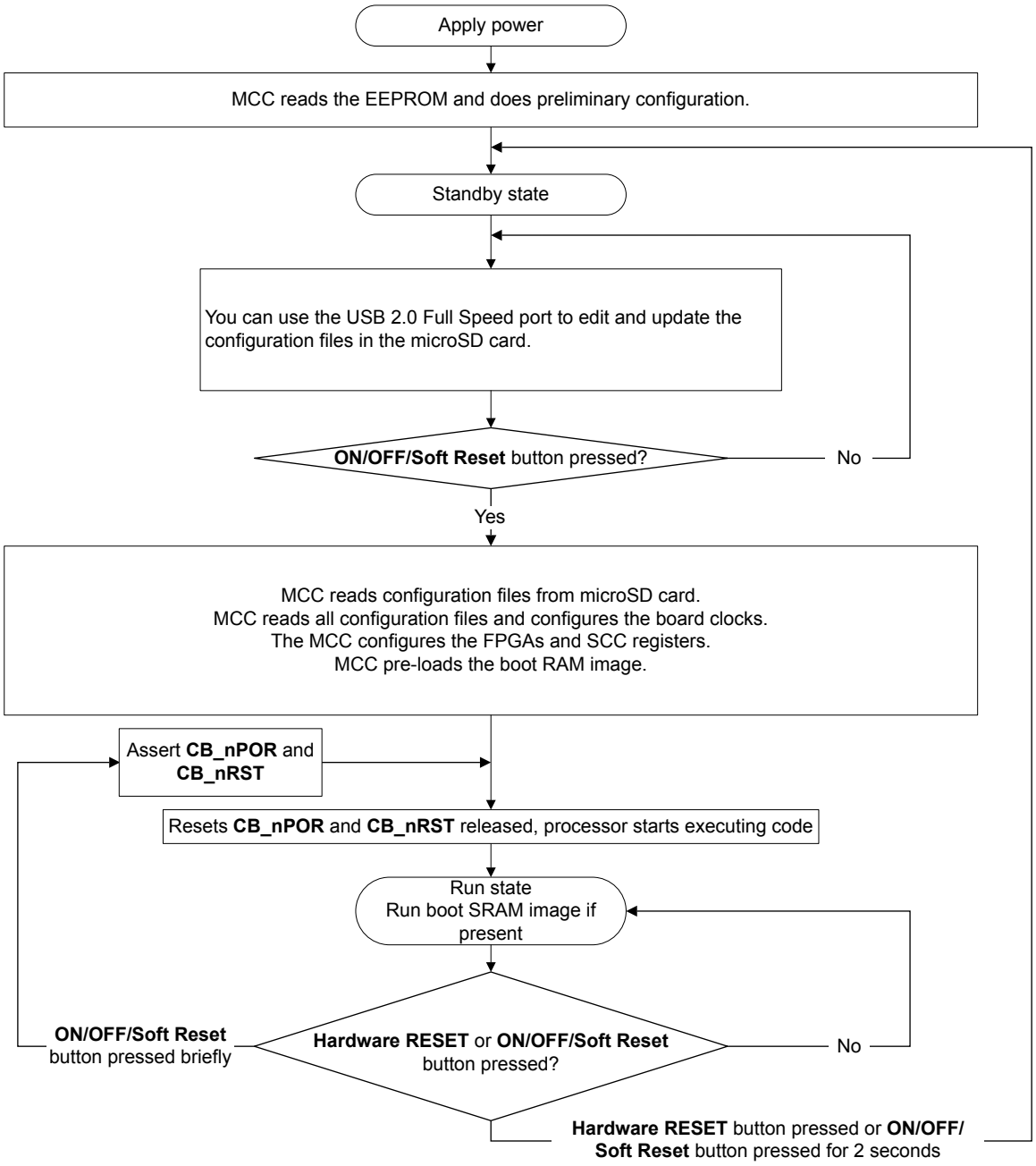

Figure 3-2 Powerup and configuration sequence

The board powerup and configuration sequence is:

1. The board applies power to the system.
2. The MCC powers the EEPROM and reads it to determine the HBI identification code for the board.
3. The system enters standby state.
4. The system enables the microSD memory card and you can connect a workstation to the MCC USB 2.0 Full Speed port to edit existing configuration files or *Drag-and-Drop* new configuration files.
5. The system stays in standby state until you press the *ON/OFF/Soft RESET* push button.

6. The system loads the board configuration file:
   - The MCC reads the generic `config.txt` file.
   - The MCC searches the microSD card MB directory for the HBI0263x subdirectory that matches the board HBI code from the board EEPROM.
7. The next steps depend on the configuration files:
   - If the MCC finds configuration subdirectories that match the HBI code of the board, configuration continues and the MCC reads the `board.txt` file.
   - If the MCC does not find the correct configuration files, it records the failure to a log file on the microSD card. Configuration stops and the system re-enters standby state.
8. The MCC measures the board power supplies.
9. The MCC configures the board clocks and FPGA SCC registers.
10. If the MCC finds new software images, it loads them into the SRAM through the FPGA.
11. The MCC releases the system resets nPOR and nRST, the system enters run state.
12. Normal operation continues until a new event occurs:

Related information

3.6.2 `config.txt` generic board configuration file on page 3-57
3.6.3 Contents of the MB directory on page 3-57
3.6.4 Contents of the SOFTWARE directory on page 3-58
3.6.1 Overview of configuration files and microSD card directory structure on page 3-56

3.5 Reset push buttons

The MPS2 and MPS2+ FPGA Prototyping Boards provide two push buttons which initiate reset and configuration. The two reset push buttons are the ON/OFF/Soft RESET and the Hardware RESET buttons. This section describes the use and functions of these push buttons.

This section contains the following subsections:

- *3.5.1 ON/OFF/Soft RESET button* on page 3-55.
- *3.5.2 Hardware RESET button* on page 3-55.

3.5.1 ON/OFF/Soft RESET button

The MPS2 and MPS2+ FPGA Prototyping Boards provide an ON/OFF/Soft RESET push button that enables you to perform a software reset of the system.

You initiate a software reset of the system by briefly pressing the *ON/OFF/Soft RESET* button during runtime. The MCC performs a software reset of the FPGA and resets the devices on the board. It does not perform a full reconfiguration of the FPGA.

The software reset sequence is as follows:

1. You briefly press the ON/OFF/Soft RESET button.

   Caution

   If you press and hold the *ON/OFF/Soft RESET* button for more than two seconds, the system enters the standby state in the same way as pressing the *Hardware RESET* button.

2. The MCC asserts the **nRST** reset signal. It might also assert **nPOR** but this depends on the setting of the variable `ASSERTNPOR` in the configuration file `config.txt`.
3. The MCC releases **nPOR** if it is active depending on the setting of the variable `ASSERTNPOR` in the configuration file `config.txt`.
4. The MCC releases **nRST**.
5. The board enters the run state.

Note

The MCC does not read the configuration files or perform a board reconfiguration as a result of a software reset.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

3.5.2 Hardware RESET button

The MPS2 and MPS2+ FPGA Prototyping Boards provide a Hardware RESET push button that enables you to perform a hardware reset of the system.

You can change the operation of the board from ON to standby by briefly pressing this button. This switches off the power to the board and resets the system to the default values.

If you then press the ON/OFF/Soft RESET push button, the system performs a full configuration and enters the run state.

Related information

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

3.6 Configuration files

This section describes the MPS2 and MPS2+ board configuration files in the microSD card which control the board powerup and configuration process.

This section contains the following subsections:

- *3.6.1 Overview of configuration files and microSD card directory structure* on page 3-56.
- *3.6.2 `config.txt` generic board configuration file* on page 3-57.
- *3.6.3 Contents of the MB directory* on page 3-57.
- *3.6.4 Contents of the SOFTWARE directory* on page 3-58.

3.6.1 Overview of configuration files and microSD card directory structure

Because the board microSD card is non-volatile memory, it is only necessary to load new configuration files if you change the system configuration. The microSD card contains default configuration files.

If you connect a workstation to the MCC USB 2.0 Full Speed port, the configuration memory device, the SD card, appears as a *USB Mass Storage Device* (USBMSD). You can then add, edit, or delete files.

You can use a standard text editor that produces DOS line endings to read and edit the board configuration files.

The following figure shows a typical example of the directory structure in the microSD card memory.

Caution

Files names and directory names are in 8.3 format:

- File names that you generate must be in lowercase.
- Directory names must be in uppercase.
- All configuration files must end in DOS line endings (`0x0D`/`0x0A`).

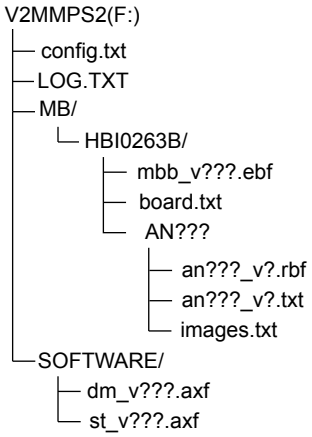

Figure 3-3 Example USBMSD directory structure

The directory structure and file name format ensure that each image is matched to the correct target device defined in the board configuration EEPROM.

- `config.txt` is the generic configuration file for all boards. This file applies to all Versatile™ Express motherboards as well as the MPS2 and MPS2+ FPGA Prototyping Boards.
- The MB directory contains subdirectories for any board variants that might be present in the system. The subdirectory names match the HBI codes for the specific board variants. The files in this directory contain clock, register, and other settings for the board.
- The `SOFTWARE` directory contains application files that the MCC can load into the ZBT SSRAM or PSRAM on the board. The `images.txt` file defines the file that the MCC loads.

Related information

2.5 USB 2.0 Full Speed interface on page 2-29

3.6.2 `config.txt` generic board configuration file on page 3-57
3.6.3 Contents of the MB directory on page 3-57
3.6.4 Contents of the SOFTWARE directory on page 3-58

3.6.2 `config.txt` generic board configuration file

You can use the MPS2 or MPS2+ FPGA Prototyping Board USB 2.0 Full Speed port to update the generic configuration file `config.txt` from your workstation to the root directory of the microSD card.

The following example shows a configuration file that you can load into the configuration flash memory.

Note

- Colons (:) indicate the end of commands and must be separated by a space character (`0x20`) from the value fields.
- Semicolons (;) indicate comments.

```
TITLE: Arm MPS2 FPGA prototyping board configuration file

[CONFIGURATION]
AUTORUN: FALSE                      ;Auto Run from power on
AUTORUNDELAY: 3                     ;Delay in seconds to wait for key press to stop bootup
RTC: FALSE                          ;TRUE = Enable RTC, FALSE = Disable RTC

ASSERTNPOR: TRUE                    ;External resets assert nPOR
WDTRESET: NONE                      ;Watchdog reset options NONE/RESETMB/RESETDB

USB_REMOTE: TRUE                    ;Selects remote command via USB

MACADDRESS: 0xFFFFFFFFFFFF          ;MAC Address
```

3.6.3 Contents of the MB directory

The MPS2 and MPS2+ board MB directories contain a configuration HBI subdirectory that matches the HBI code of the board.

The HBI subdirectory contains:

- A file of the form *mbb_vxxx.ebf*. This is an MCC BIOS image.
- A `board.txt` file that defines the MCC bios image.
- An application-specific subdirectory that contains the following board configuration files:
  - Image files for the FPGA and MCC on the board. These have `.rbf` extensions.
  - An `images.txt` file that defines the `.axf` files that the MCC loads into external memory during configuration.
  - An application note `.txt` file that defines the number of FPGAs on the board, the number of oscillators and their frequencies, FPGA and MCC image files, and details of the SCC registers.

The following example shows a typical MPS2 or MPS2+ FPGA Prototyping Board configuration `board.txt` file.

```
BOARD: HBI0263
TITLE: Motherboard configuration file

[MCCS]
MBBIOS mbb_v201.ebf                 ;MB BIOS IMAGE

[APPLICATION NOTE]                  ;Please select the required processor
;APPFILE: AN382\an382_v2.txt        ; - Cortex-M0
;APPFILE: AN383\an383_v2.txt        ; - Cortex-M0+
;APPFILE: AN384\an384_v2.txt        ; - Cortex-M1
APPFILE: AN385\an385_v2.txt         ; - Cortex-M3
;APPFILE: AN386\an386_v2.txt        ; - Cortex-M4
;APPFILE: AN387\an387_v2.txt        ; - Cortex-M0 Design Start
;APPFILE: AN399\an399_v2.txt        ; - Cortex-M7
;APPFILE: AN400\an400_v2.txt        ; - Cortex-M7 with CoreSight
```

The following example shows a typical MPS2 or MPS2+ FPGA Prototyping Board application note `.txt` file.

```
BOARD: HBI0263
TITLE: Application Note File


[FPGAS]
TOTALFPGAS: 1                          ;Total Number of FPGAS (Max : 1)
F0FILE: an385_v2.rbf                   ;FPGA0 Filename
F0MODE: FPGA                           ;FPGA0 Programming Mode

[OSCCLKS]
TOTALOSCCLKS: 3
OSC0: 50.0                             ;OSC0 System clock in MHz
OSC1: 24.576                           ;OSC1 AACI clock in MHz
OSC2: 25.0                             ;OSC2 MISC clock in MHz

[SCC REGISTERS]
TOTALSYSCONS: 1                        ;Total Number of SYSCON registers defined
SYSCON: 0x000 0x00000001               ;SCC enable ZBT remap
```

The following example shows a typical MPS2 or MPS2+ FPGA Prototyping Board `images.txt` file.

```
TITLE: Arm MPS2 FPGA prototyping board Images Configuration File

[IMAGES]
TOTALIMAGES: 1                          ;Number of Images (Max : 32)

IMAGE0ADDRESS:0x00000000                ;Please select the required executable program
IMAGE0FILE: \SOFTWARE\dm_v111.axf          ; - demo
;IMAGE0FILE: \SOFTWARE\st_v111.axf         ; - selftest
```

The following example shows the use of a binary file as an alternative to generating a `.axf` file.

```
TITLE: Arm MPS2 FPGA prototyping board Images Configuration File

[IMAGES]
TOTALIMAGES: 1                          ;Number of Images (Max : 32)

IMAGE0ADDRESS:0x00000000                ;Please select the required executable program
IMAGE0FILE: \SOFTWARE\test.bin          ;
```

`.axf` and `.elf` files are treated as elf files. All other files are treated as binary.

3.6.4 Contents of the SOFTWARE directory

The SOFTWARE directory contains applications that you can load into the ZBT SSRAM or PSRAM memory.

You can create applications and load them into the RAM on the board. Application images are typically boot images or demo programs and have a `.axf` extension.

Typical applications in this directory are:

- `dm_v102.axf` board demonstration software.
- `st_v103.axf` board test software.
- `user.axf` user test software.

Chapter 4
Programmers Model

This chapter describes the programmers model of the MPS2 and MPS2+ FPGA Prototyping Boards.

It contains the following sections:

- *4.1 About this programmers model* on page 4-60.
- *4.2 Memory map* on page 4-61.
- *4.3 Register summary* on page 4-62.
- *4.4 SCC register descriptions* on page 4-63.
- *4.5 System configuration registers* on page 4-70.

4.1 About this programmers model

The following information applies to the SCC registers and to the system configuration registers or SYS_CFG registers:

- The base address is not fixed, and can be different for any particular system implementation. The offset of each register from the base address is fixed.
- Do not attempt to access reserved or unused address locations. Attempting to access these locations can result in UNPREDICTABLE behavior.
- Unless otherwise stated in the accompanying text:
  - Do not modify undefined register bits.
  - Ignore undefined register bits on reads.
  - All register bits are reset to a logic 0 by a system or powerup reset.
  - *Table 4-1 MPS2 and MPS2+ SCC register summary* on page 4-62 describes register access type as follows:

    RW: Read and write.

    RO: Read only.

    WO: Write only.

4.2 Memory map

The image that you load into the FPGA on the MPS2 or MPS2+ FPGA Prototyping Board determines the board memory map. See the relevant SMM Application Note for an example memory map for the board.

4.3 Register summary

This section summarizes the MPS2 and MPS2+ SCC registers and system configuration registers characteristics.

The following table shows the registers in offset order from the base memory address.

Table 4-1 MPS2 and MPS2+ SCC register summary

Offset	Name	Type	Reset	Width	Comment
0x000	SCC_CFG0	RW	0x00000000	32	Remaps block RAM to ZBT.
0x004	SCC_CFG1	RW	0x00000000	32	Controls the MCC user LEDs.
0x008	-	-	-	-	Reserved. Do not write to or read from this register.
0x00C	SCC_CFG3	RO	0x00000000	32	Denotes the state of the MCC user switches.
0x010	SCC_CFG4	RO	0x00000000	32	Denotes the board revision.
0x014 - 0x09C	-	-	-	-	Reserved. Do not write to or read from these registers
0x0A0	SYS_CFGDATA_RTN	RO	0x00000000	32	This is one oUser data return register for reads from board peripherals.
0x0A4	SYS_CFGDATA_OUT	RW	0x00000000	32	User data output register for writes to board peripherals.
0x0A8	SYS_CTRL	RW	0x00100000	32	Control register. Controls read and write operations.
0x0AC	SYS_CFGSTAT	RO	0x00000000	32	Status register. Shows status of read and write operations.
0x0AD - 0x0FC	-	-	-	-	Reserved. Do not write to or read from these registers.
0x100	SCC_DLL	RO	0xFFFC0000	32	DLL Lock Register
0x104 - 0xFF7	-	-	-	-	Reserved. Do not write to or read from these registers
0xFF8	SCC_AID	RO	0x03100708	32	Contains the FPGA build number, the target board of the FPGA build, and information about the SCC register implementation.
0xFFC	SCC_ID	RO	0x4104XXX1	32	Contains information about the FPGA image.

Related information

2.13 MCC FPGA serial interface on page 2-39
4.4.2 SCC_CFG0 Register on page 4-63
4.4.3 SCC_CFG1 Register on page 4-64
4.4.4 SCC_CFG3 Register on page 4-64
4.4.5 SCC_CFG4 Register on page 4-65
4.5.2 SYS_CFGDATA_RTN Register on page 4-70
4.5.3 SYS_CFGDATA_OUT Register on page 4-71
4.5.4 SYS_CFGCTRL Register on page 4-72
4.5.5 SYS_CFGSTAT Register on page 4-73
4.4.6 SCC_DLL Register on page 4-66
4.4.7 SCC_AID Register on page 4-67
4.4.8 SCC_ID Register on page 4-68

4.4 SCC register descriptions

This section describes the SCC registers in the FPGAs on the MPS2 and MPS2+ FPGA Prototyping Boards.

This section contains the following subsections:

- *4.4.1 Overview of SCC registers* on page 4-63.
- *4.4.2 SCC_CFG0 Register* on page 4-63.
- *4.4.3 SCC_CFG1 Register* on page 4-64.
- *4.4.4 SCC_CFG3 Register* on page 4-64.
- *4.4.5 SCC_CFG4 Register* on page 4-65.
- *4.4.6 SCC_DLL Register* on page 4-66.
- *4.4.7 SCC_AID Register* on page 4-67.
- *4.4.8 SCC_ID Register* on page 4-68.

4.4.1 Overview of SCC registers

The MCC on the MPS2 or MPS2+ FPGA Prototyping Board writes to the SCC registers in the FPGA at powerup with the values that the configuration board file defines.

During runtime, the MCC polls the LED and switch values to ensure that they match the SCC_CFG1 and SCC_CFG2 register values. The MCC updates the LEDs to match the SCC_CFG1 register values and updates the SCC_CFG2 register to match the user switches.

The SCC_CFG3 register contains board revision information.

The MCC reads the SYS_DLL, SCC_AID and SCC_ID registers to determine whether the PLLs in the FPGA are locked and to determine the loaded FPGA image.

4.4.2 SCC_CFG0 Register

The SCC_CFG0 Register characteristics are:

Purpose

Remaps block RAM to ZBT.

Usage constraints

There are no usage constraints.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

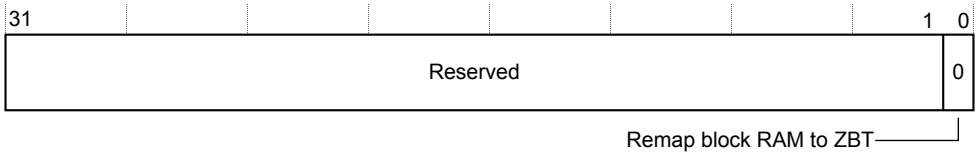

Figure 4-1 SCC_CFG0 Register bit assignments

The following table shows the bit assignments.

Table 4-2 SCC_CFG0 Register bit assignments

Bits	Name	Function
[31:1]	-	Reserved. Do not write to these bits.
[0]	REMAP_BLOCK_RAM_TO_ZBT	This bit remaps block RAM to ZBT: • `0b0` Does not remap block RAM to ZBT. • `0b1` Remaps block RAM to ZBT.

Related information

4.3 Register summary on page 4-62

4.4.3 SCC_CFG1 Register

The SCC_CFG1 Register characteristics are:

Purpose

Controls the USER LEDS on the MPS2 and MPS2+ FPGA Prototyping Boards. The MCC polls this SCC register in the FPGA and updates the appropriate LEDs.

Usage constraints

There are no usage constraints.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

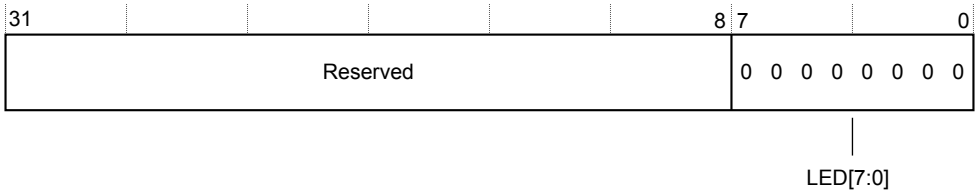

Figure 4-2 SCC_CFG1 Register bit assignments

The following table shows the bit assignments.

Table 4-3 SCC_CFG1 Register bit assignments

Bits	Name	Function
[31:8]	-	Reserved. Do not write to these bits.
[7:0]	LED[7:0]	These bits control the individual USER LEDs: • `0b0` Off • `0b1` On

Related information

2.11 User switches and user LEDs on page 2-35
4.3 Register summary on page 4-62

4.4.4 SCC_CFG3 Register

The SCC_CFG3 Register characteristics are:

Purpose

Determines the state of the eight user switches on the MPS2 and MPS2+ FPGA Prototyping Boards. The MCC polls the switches and updates this SCC register in the FPGA.

Usage constraints

The SCC_CFG3 Register is read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

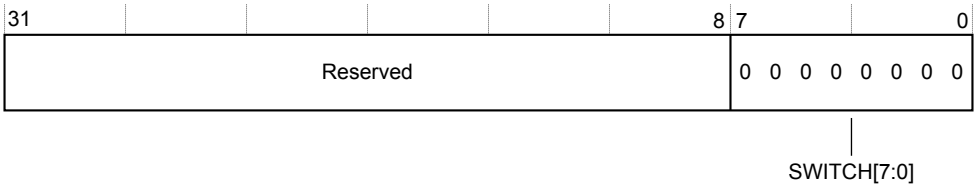

Figure 4-3 SCC_CFG3 Register bit assignments

The following table shows the bit assignments.

Table 4-4 SCC_CFG3 Register bit assignments

Bits	Name	Function
[31:8]	-	Reserved. Do not write to these bits.
[7:0]	SW[7:0]	These bits indicate the state of the user switches: • `0b0` Off • `0b1` On

Related information

2.11 User switches and user LEDs on page 2-35

4.3 Register summary on page 4-62

4.4.5 SCC_CFG4 Register

The SCC_CFG4 Register characteristics are:

Purpose

Contains MPS2 and MPS2+ board revision information.

Usage constraints

The SCC_CFG4 Register is read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

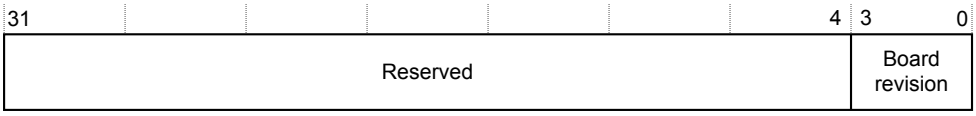

Figure 4-4 SCC_CFG4 Register bit assignments

The following table shows the bit assignments.

Table 4-5 SCC_CFG4 Register bit assignments

Bits	Name	Function
[31:4]	-	Reserved. Do not write to these bits.
[3:0]	Board revision	• 0000 = A • 0001 = B • 0010 = C

Related information

4.3 Register summary on page 4-62

4.4.6 SCC_DLL Register

The SCC_DLL Register characteristics are:

Purpose

Digital Locked Loop (DLL) lock status bits from the FPGA. You read from it to determine the lock status of the DLLs. You can ignore the lock status of some of the DLLs by writing to the DLL_LOCK_MASK bits. The DLLs you ignore depends upon the values of these bits.

Usage constraints

The SCC_DLL Register is read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

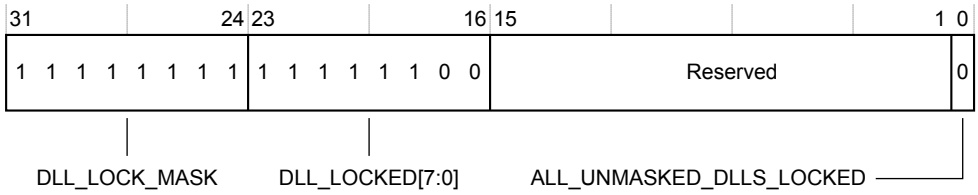

Figure 4-5 SCC_DLL Register bit assignments

The following table shows the bit assignments.

Table 4-6 SCC_DLL Register bit assignments

Bits	Name	Function
[31:24]	DLL_LOCK_MASK[7:0]	These bits mask bits[23:16], that is, you ignore the corresponding DLL_LOCKED bits. • `0b0`: — Mask DLL lock status. Corresponding DLL_LOCKED bit, that is the lock status of the corresponding DLL, does not contribute to the value of bit[0]. • `0b1`: — Do not mask DLL lock status. Corresponding DLL_LOCKED bit, that is the lock status of the corresponding DLL, contributes to the value of bit[0].
BIT[23:16]	DLL_LOCKED[7:0]	These bits indicate the lock statuses of the individual DLLs: • `0b0` DLL unlocked. • `0b1` DLL locked.

Table 4-6 SCC_DLL Register bit assignments (continued)

Bits	Name	Function
[15:1]	-	Reserved. Do not write to these bits.
[0]	ALL_UNMASKED_DLLS_LOCKED	This bit indicates whether or not all unmasked DLLs are locked: • `0b0` One or more unmasked DLLs are not locked. • `0b1` All unmasked DLLs are locked.

Related information

4.3 Register summary on page 4-62

4.4.7 SCC_AID Register

The SCC_AID Register characteristics are:

Purpose

The SCC_AID Register contains the FPGA build number, the target board of the FPGA build, and information about the SCC register implementation.

Usage constraints

The SCC_AID Register is read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

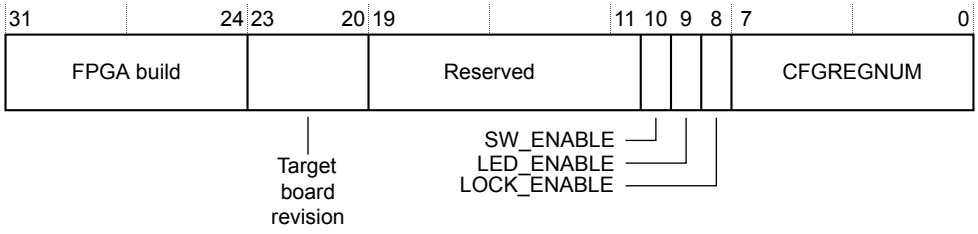

Figure 4-6 SCC_AID Register bit assignments

The following table shows the bit assignments.

Table 4-7 SCC_AID Register bit assignments

Bits	Name	Function
[31:24]	Build	FPGA build number.
[23:20]	Target board revision.	These bits indicate the target board revision of the FPGA build. • 0000 = A • 0001 = B • 0010 = C
[19:11]	-	Reserved. Do not write to these bits.
[10]	SW_ENABLE	This bit indicates whether the system supports the SCC_SW_READ command.
[9]	LED_ENABLE	This bit indicates whether the system supports the SCC_LED_READ command.
[8]	LOCK_ENABLE	This bit indicates whether the system supports the SCC_LOCK_READ command.
[7:0]	CFGREGNUM	These bits indicate the number of user registers you can define. The maximum number is eight.

Related information

4.3 Register summary on page 4-62

4.4.8 SCC_ID Register

The SCC_ID Register characteristics are:

Purpose

The MCC reads this register and uses the information to determine information about the design in the FPGA that you can read through the MPS2 or MPS2+ FPGA Prototyping Board SYS_CFG interface.

Usage constraints

The SCC_ID Register is Read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

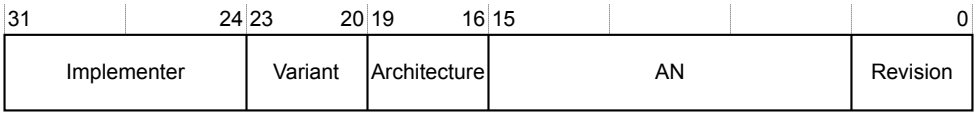

Figure 4-7 SCC_ID Register bit assignments

The following table shows the bit assignments.

Table 4-8 SCC_ID Register bit assignments

Bits	Name	Function
[31:24]	Implementer	Implementer ID. `0x41` = Arm.
[23:20]	Variant	Product variant or major revision number: **Note** This refers to the core or cluster image in the FPGA and is the x in *rxpy*
[19:16	IP Architecture	Architecture. `0x04` = AHB
[15:4]	AN	Application Note number. • 382 AN382: V2M-M0 Application Note • 383 AN383: V2M-M0+ Application Note • 384 AN384: V2M-M1 Application Note • 385 AN385: V2M-M3 Application Note • 386 AN386: V2M-M4 Application Note • 387 AN387: V2M-M0 Design Start Application Note • 399 AN399: V2M-M7 Application Note • 400 AN400: V2M-M7 Application Note with CoreSight
[3:0]	Revision	Product revision or minor revision number. **Note** This refers to the core or cluster image in the FPGA and is the y in *rxpy*

Related information

4.3 Register summary on page 4-62

Related information

2.13 MCC FPGA serial interface on page 2-39

4.5 System configuration registers

This section describes system configuration registers, that is, the SYS_CFG registers.

This section contains the following subsections:

- *4.5.1 Overview of system configuration registers* on page 4-70.
- *4.5.2 SYS_CFGDATA_RTN Register* on page 4-70.
- *4.5.3 SYS_CFGDATA_OUT Register* on page 4-71.
- *4.5.4 SYS_CFGCTRL Register* on page 4-72.
- *4.5.5 SYS_CFGSTAT Register* on page 4-73.

4.5.1 Overview of system configuration registers

The system configuration registers enable communication between the MCC and the FPGA.

The configuration, or SYS_CFG, registers are: SYS_CFGDATA_RTN, SYS_CFGDATA_OUT, SYS_CFGCTRL and SYS_CFGSTAT. These registers implement and control write and read operations when the application software inside the FPGA writes and reads configuration information to and from peripherals on the MPS2 or MPS2+ FPGA Prototyping Board.

The following block of example pseudocode shows a write operation to write 24MHz to clock generator 1.

```
       //Write frequency value to the user data out register.
       1: Set SYS_CFGDATAT_OUT = 24000000
       //Write to control register. Initiate write process and select clock generator 1 and
oscillator function.
       //This clears Configuration error bit and Configuration complete bit of SYS_CFGSTAT
Register.
       2: Set SYS_CFGCTRL    Start = 1; nREAD_Write_access = 1; Function = 1; Device = 1;

       //Read status register to determine status of of write operation.
       3: Read    SYS_CFGSTAT[Configuration error bit, Configuration complete bit]

          If ( Configuration error bit = 1 )
          return TRANSACTION FAIL;

          If ( Configuration error bit = 0 AND Configuration complete bit = 1)
          return TRANSACTION OK;

          If ( Configuration error bit = 0 AND Configuration complete bit = 0)
          GOTO 1;
```

The following block of example pseudocode show a read operation to read the frequency of clock generator 2.

```
       //Write to control register. Initiate read process and select clock generator 2 and
oscillator function.
       //This clears Configuration error bit and Configuration complete bit of SYS_CFGSTAT
Register.
       1: Set SYS_CFGCTRL    Start = 1; nREAD_Write_access = 0; Function = 1; Device = 2;

       //Read status register to determine status of of read operation.
       2: Read    SYS_CFGSTAT[Configuration error bit, Configuration complete bit]

          If ( Configuration error bit = 1 )
          return TRANSACTION FAIL;

          If ( Configuration error bit = 0 AND Configuration complete bit = 1)
          return TRANSACTION OK;

          If ( Configuration error bit = 0 AND Configuration complete bit = 0)
          GOTO 1;
```

4.5.2 SYS_CFGDATA_RTN Register

The SYS_CFGDATA_RTN Register characteristics are:

Purpose

The MCC writes data to the SYS_CFGDATA_RTN Register during a read operation. This data represents the value that the addressed MPS2 or MPS2+ FPGA Prototyping Board component returns as a result of the read operation, for example, a clock generator frequency.

Usage constraints

The SYS_CFGDATA_RTN register is read-only to the application software in the FPGA.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

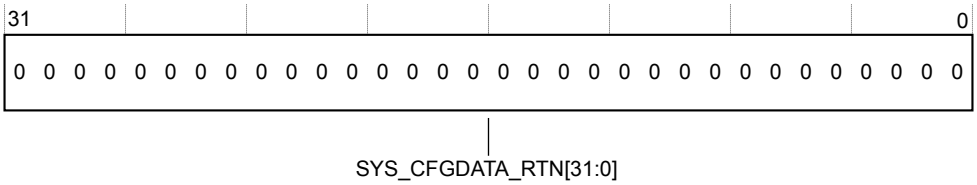

Figure 4-8 SYS_CFGDATA_RTN Register bit assignments

The following table shows the bit assignments.

Table 4-9 SYS_CFGDATA_RTN Register bit assignments

Bits	Name	Function
[31:0]	SYS_CFGDATA_RTN	User data register return bits.

Related information

4.3 Register summary on page 4-62

4.5.3 SYS_CFGDATA_OUT Register

The SYS_CFGDATA_OUT Register characteristics are:

Purpose

The application software in the FPGA writes data to the SYS_CFGDATA_OUT Register during a write operation. This data represents a value or function that the write operation sends to the addressed MPS2 or MPS2+ FPGA Prototyping Board component, for example, a frequency value to a clock generator.

Usage constraints

There are no usage constraints.

Configurations

Available in all MPS2 and MPS2+ Boards configurations.

The following figure shows the bit assignments.

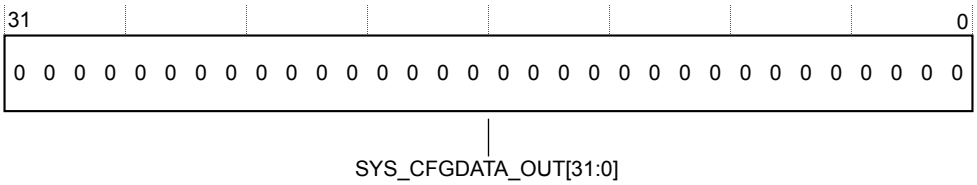

Figure 4-9 SYS_CFGDATA_OUT Register bit assignments

The following table shows the bit assignments.

Table 4-10 SYS_CFGDATA_OUT Register bit assignments

Bits	Name	Function
[31:0]	SYS_CFGDATA_OUT	User data register output bits.

Related information

4.3 Register summary on page 4-62

4.5.4 SYS_CFGCTRL Register

The SYS_CFGCTRL Register characteristics are:

Purpose

The SYS_CFGCTRL Register controls write and read data transfer between the MCC and the SCC interface in the FPGA.

Usage constraints

There are no usage constraints.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

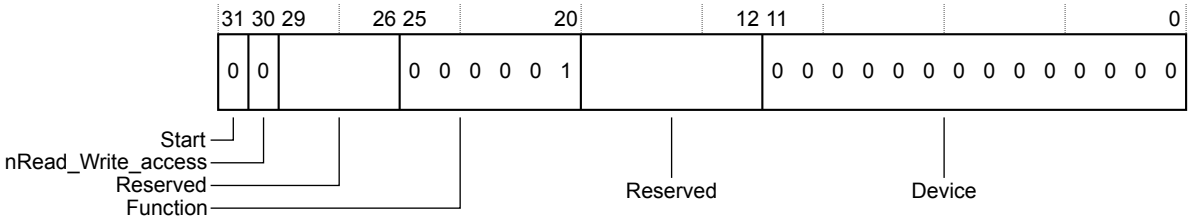

Figure 4-10 SYS_CFGCTRL Register bit assignments

The following table shows the bit assignments.

Table 4-11 SYSCFG_CTRL Register bit assignments

Bits	Name	Function
[31]	Start	Writing to this bit generates an interrupt.
[30]	nRead_Write_access	• `0b0` Read access • `0b1` Write access
[29:26]	-	Reserved. Do not write to these bits.
[25:20]	Function	These bits support the following function: • `0b000001` Clock generator
[19:12]	-	Reserved. Do not write to these bits.
[11:0]	Device	Selects device to write to or read from: • `0x000` Clock generator 0 • `0x001` Clock generator 1 • `0x002` Clock generator 2

Related information

4.3 Register summary on page 4-62

4.5.5 SYS_CFGSTAT Register

The SYS_CFGSTAT Register characteristics are:

Purpose

The SYS_CFGSTAT Register contains system configuration status information about read and write operations between the application software in the FPGA and the MPS2 or MPS2+ Board board peripherals.

Usage constraints

The SYS_CFGSTAT Register is read-only.

Configurations

Available in all MPS2 and MPS2+ configurations.

The following figure shows the bit assignments.

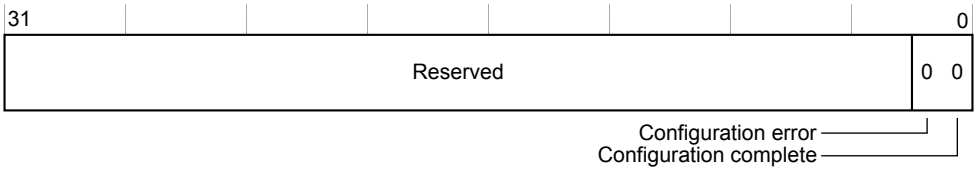

Figure 4-11 SYS_CFGSTAT Register bit assignments

The following table shows the bit assignments.

Table 4-12 SYS_CFGSTAT Register bit assignments

Bits	Name	Function
[31:2]	-	Reserved. Do not write to these bits.
[1]	Configuration error	A write to SYS_CFGCTRL clears this bit. • `0b0` Configuration successful • `0b1` Configuration failed
[0]	Configuration complete	A write to SYS_CFGCTRL clears this bit. • `0b0` Configuration not complete • `0b1` Configuration complete

Related information

4.3 Register summary on page 4-62

Chapter 5
Signal Descriptions

This chapter describes the signals present at the interface connectors of the MPS2 and MPS2+ FPGA Prototyping Boards.

It contains the following sections:

- *5.1 Debug connectors* on page 5-75.
- *5.2 Expansion connectors* on page 5-80.
- *5.3 CLCD connector* on page 5-82.
- *5.4 USB 2.0 connector* on page 5-83.
- *5.5 UART connector* on page 5-84.
- *5.6 SPI connector* on page 5-85.
- *5.7 VGA connector* on page 5-86.
- *5.8 Audio connectors* on page 5-87.
- *5.9 Ethernet connector* on page 5-88.
- *5.10 12V power connector* on page 5-89.

5.1 Debug connectors

This section describes the MPS2 and MPS2+ board connectors that support P-JTAG processor debug, F-JTAG FPGA debug, 16-bit Trace, 4-bit Trace, and SWD. This section describes the connectors and lists their signals.

This section contains the following subsections:

- *5.1.1 JTAG 14 connector* on page 5-75.
- *5.1.2 JTAG 20 connector* on page 5-76.
- *5.1.3 CoreSight 10 connector* on page 5-76.
- *5.1.4 CoreSight 20 connector* on page 5-77.
- *5.1.5 MICTOR 38 connector* on page 5-78.

5.1.1 JTAG 14 connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one F-JTAG, ILA, connector that supports FPGA debug. It enables you to connect an ILA device, such as SignalTap II, to a hard FPGA JTAG chain in the FPGA and debug your design.

The JTAG 14 connector connects to general-purpose pins on the FPGA. The availability of F-JTAG depends on the design which you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards label this connector as *FJTAG*.

The following figure shows the JTAG 14 connector.

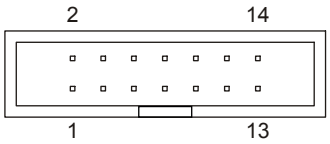

Figure 5-1 JTAG 14 connector

The following table shows the F-JTAG connector pin mapping for each ILA signal on the JTAG 14 connector J24.

Table 5-1 JTAG 14 connector, J24, signal list

Pin	Signal	Pin	Signal
1	**GND**	2	**+3V_EXT**
3	**GND**	4	**FPGA_TMS**
5	**GND**	6	**FPGA_TCK**
7	**GND**	8	**FPGA_TDO**
9	**GND**	10	**FPGA_TDI**

Note

- Pins 4, 8 and 10 have pullup resistors to **3V**.
- Pin 6 has a pulldown resistor to **GND**.

Related information

2.15.2 F-JTAG on page 2-44

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.1.2 JTAG 20 connector

The MPS2 and MPS2+ FPGA Prototyping Boards each provide one 3V 20-pin IDC connector that supports P-JTAG processor debug to enable connection of DSTREAM or a compatible third-party debugger. The connector also supports SWD.

The JTAG 20 connector connects to general-purpose pins on the FPGA. The availability of P-JTAG or SWD depends on the design which you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards label this connector as *PJTAG*.

The following figure shows the JTAG 20 connector.

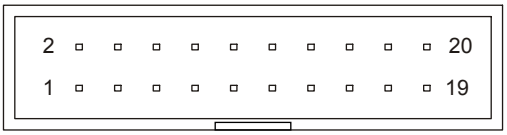

Figure 5-2 JTAG 20 connector

The following table shows the pin mapping for each SWD and P-JTAG signal on the JTAG 20 connector J4.

Table 5-2 JTAG 20 connector, J4, signal list

Pin	Signal	Pin	Signal
1	**3V0_OUT**	2	**3V0_OUT** -optional
3	**nTRST**	4	**GND**
5	**TDI**	6	**GND**
7	**SWDIO/TMS**	8	**GND**
9	**SWDCLK/TCK**	10	**GND**
11	**GND/RTCK**	12	**GND**
13	**SWO/TDO**	14	**GND**
15	**nSRST**	16	**GND**
17	NC/**DBGRQ**	18	**GNDDETECT**
19	NC/**DBGACK**	20	**GND**

Note

- The MPS2 and MPS2+ FPGA Prototyping Boards do not support adaptive clocking. The boards tie the **RTCK** signal low.
- Pins 3, 5, 7, 13, 15 and 18 have pullup resistors to 3V.
- Pin 9 has a pulldown resistor to **GND**.

Related information

2.15.3 P-JTAG on page 2-45
2.15.6 Serial Wire Debug on page 2-45
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.1.3 CoreSight 10 connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one 3V 10-pin IDC connector. This connector supports P-JTAG processor debug to enable connection of DSTREAM or a compatible third-party debugger. This connector also supports SWD.

The CoreSight 10 connector connects to general-purpose pins on the FPGA. The availability of P-JTAG or SWD depends on the design which you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards label this connector as *PJTAG*.

The following figure shows the CoreSight 10 connector.

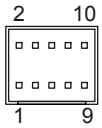

Figure 5-3 CoreSight 10 connector

The following table shows the pin mapping for each SWD and P-JTAG signal on the CoreSight 10 connector J11.

Table 5-3 CoreSight 10 connector, J11, signal list

Pin	Signal	Pin	Signal
1	**3V0_OUT**	2	**SWDIO/TMS**
3	**GND**	4	**SWDCLK/TCK**
5	**GND**	6	**SWO/TDO**
7	**KEY**	8	NC/**TDI**
9	**GNDDETECT**	10	**nSRST**

Note

- Pins 2, 6, 8 and 10 have pullup resistors to **3V**.
- Pins 4 on both connectors have pulldown resistors to **GND**.

Related information

2.15.3 P-JTAG on page 2-45

2.15.6 Serial Wire Debug on page 2-45

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.1.4 CoreSight 20 connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one 3V 20-pin Cortex debug and ETM connector. The connector supports P-JTAG processor debug to enable connection of DSTREAM or a compatible third-party debugger. The connector also supports SWD and 4-bit Trace.

The CoreSight 20 connector connects to general-purpose pins on the FPGA. The availability of P-JTAG, SWD, or 4-bit Trace depends on the design which you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards label this connector as *PJTAG*.

The following figure shows the CoreSight 20 connector.

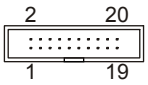

Figure 5-4 CoreSight 20 connector

The following table shows the pin mapping for each SWD, P-JTAG, and 4-bit Trace signal on the CoreSight 20 connector J1.

Table 5-4 CoreSight 20 connector, J1, signal list

Pin	Signal	Pin	Signal
1	**3V0_OUT**	2	**SWDIO/TMS**
3	**GND**	4	**SWDCLK/TCK**
5	**GND**	6	**SWO/TDO/EXTa**
7	**Key1**	8	NC/**TDI/EXTb**
9	**GNDDETECT**	10	**nSRST**
11	**3V0_OUT**	12	**TRACECLK**
13	**3V0_OUT**	14	**TRACEDATA[0]**
15	**GND**	16	**TRACEDATA[1]**
17	**GND**	18	**TRACEDATA[2]**
19	**GND**	20	**TRACEDATA[3]**

Note

- Pins 2, 6, 8, 9 and 10 have pullup resistors to **3V**.
- Pin 4 has a pulldown resistor to **GND**.

Related information

2.15.3 P-JTAG on page 2-45
2.15.4 4-bit Trace on page 2-45
2.15.6 Serial Wire Debug on page 2-45
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.1.5 MICTOR 38 connector

The MPS2 and MPS2+ FPGA Prototyping Boards provides one 3V 38-pin MICTOR connector that supports P-JTAG processor debug to enable connection of DSTREAM or a compatible third-party debugger. The connector also supports 16-bit Trace and SWD.

The MICTOR 38 connector connects to general-purpose pins on the FPGA. The availability of P-JTAG, SWD, and 16-bit Trace depends on the design which you implement in the FPGA. The MPS2 and MPS2+ FPGA Prototyping Boards label this connector as *PJTAG/TRACE*.

The following figure shows the MICTOR 38 connector.

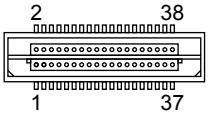

Figure 5-5 MICTOR 38 connector

The following table shows the pin mapping for each P-JTAG, SWD, and 16-bit Trace signal on the MICTOR 38 connector J6.

Table 5-5 MICTOR 38 connector, J6, signal list

Pin	Signal	Pin	Signal
1	NC	2	NC
3	NC	4	NC
5	**GND**	6	**TRACECLK**
7	**GND**	8	**GND**
9	NC/**nSRST**	10	**GND**
11	**TDO/SWO**	12	**3V** reference
13	**RTCK**	14	**3V0_OUT**
15	**TCK/SWCLK**	16	**TRACEDATA[7]**
17	**TMS/SWDIO**	18	**TRACEDATA[6]**
19	**TDI**	20	**TRACEDATA[5]**
21	**nTRST**	22	**TRACEDATA[4]**
23	**TRACEDAT[15]**	24	**TRACEDATA[3]**
25	**TRACEDAT[14]**	26	**TRACEDAT[2]**
27	**TRACEDATA[13]**	28	**TRACEDATA[1]**
29	**TRACEDATA[12]**	30	**GND**
31	**TRACEDATA[11]**	32	**GNDDETECT**
33	**TRACEDATA[10]**	34	**3V** reference
35	**TRACEDATA[9]**	36	**TRACECTL**
37	**TRACEDATA[8]**	38	**TRACEDATA[0]**

Note

- Pins 9, 11, 17, 19 and 21 have pullup resistors to **3V**.
- Pin 15 has a pulldown resistor to **GND**.

Related information

2.15.3 P-JTAG on page 2-45
2.15.5 16-bit Trace on page 2-45
2.15.6 Serial Wire Debug on page 2-45
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.2 Expansion connectors

The MPS2 and MPS2+ FPGA Prototyping Boards each provide two IDC header connectors that support user expansion.

The following figure shows the IDC expansion connector.

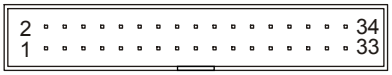

Figure 5-6 IDC expansion connector

The following two tables show the pin mappings for the IDC expansion connectors *EXP1*, component number J7, and *EXP2*, component number J8, which support general-purpose I/O expansion.

Table 5-6 Expansion connector EXP1, J7, signal list

Pin	Signal	Pin	Signal
1	**EXP0**	2	**EXP14**
3	**EXP1**	4	**EXP15**
5	**EXP2**	6	**EXP16**
7	**EXP3**	8	**EXP17**
9	**EXP4 SI Clock output**	10	**EXP18**
11	**GND**	12	**GND**
13	**EXP5 SI Clock output**	14	**EXP19 Clock output**
15	**EXP6**	16	**EXP20**
17	**EXP7**	18	**EXP21**
19	**EXP8**	20	**EXP22**
21	**EXP9**	22	**EXP23**
23	**EXP10**	24	**GND**
25	**EXP11 SI Clock output**	26	**3V**
27	**3V**	28	**GND**
29	**GND**	30	**3V**
31	**EXP12**	32	**EXP24**
33	**EXP13**	34	**EXP25**

Pins 9, 13, 14, and 25 of connector *EXP1* have source series terminating resistors. These resistors help to maintain the integrity of high slew-rate signals. Arm recommends these pins for use as clock outputs or sensitive data outputs in preference to other pins.

Table 5-7 Expansion connector EXP2, J8, signal list

Pin	Signal	Pin	Signal
1	**EXP26**	2	**EXP40**
3	**EXP27**	4	**EXP41**
5	**EXP28**	6	**EXP42**

Table 5-7 Expansion connector EXP2, J8, signal list (continued)

Pin	Signal	Pin	Signal
7	**EXP29**	8	**EXP43**
9	**EXP30 Clock output**	10	**EXP44 Clock output**
11	**GND**	12	**GND**
13	**EXP31 Clock output**	14	**EXP45**
15	**EXP32**	16	**EXP46**
17	**EXP33**	18	**EXP47**
19	**EXP34**	20	**EXP48**
21	**EXP35**	22	**EXP49 Clock input**
23	**EXP36**	24	**GND**
25	**EXP37**	26	**3V**
27	**3V**	28	**GND**
29	**GND**	30	**3V**
31	**EXP38**	32	**EXP50 Clock output**
33	**EXP39**	34	**EXP51**

Note

- Pins 9, 10, 13, 22 and 32 of connector *EXP2* have source series termination resistors. These resistors help to maintain the integrity of high-slew rate signals:
  - Arm recommends pins 9, 10, 13 and 32 for use as clock outputs or sensitive data outputs in preference to other pins.
  - Arm recommends pin 22 of connector *EXP2* for use as a clock input in preference to other pins.

Related information

2.4 User expansion port on page 2-28
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.3 CLCD connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide a female IDC connector that supports CLCD.

The following figure shows the CLCD connector.

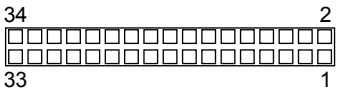

Figure 5-7 CLCD connector

Note

A mark on the board indicates pin 1.

The following table shows the pin mapping for the CLCD connector J9.

Table 5-8 CLCD connector, J9, signal list

Pin	Signal	Pin	Signal
1	**CLCD_T_SCL**	2	**CLCD_T_SDA**
3	**CLCD_CS**	4	**CLCD_RS**
5	**CLCD_WR_SCL**	6	**CLCD_RD**
7	**CLCD_RESET**	8	**CLCD_PD1**
9	**CLCD_PD2**	10	**CLCD_PD3**
11	**CLCD_PD4**	12	**CLCD_PD5**
13	NC	14	NC
15	NC	16	NC
17	NC	18	NC
19	NC	20	NC
21	NC	22	NC
23	NC	24	**GND**
25	**CLCD_BL_CTRL**	26	**3V**
27	**3V**	28	**GND**
29	**GND**	30	**3V**
31	**CLCD_SDO**	32	**CLCD_SDI**
33	**CLCD_T_CS**	34	**CLCD_T_SCK**

Note

Pins 1, 2, 5, 31, and 32 have 10k pullup resistors to 3V.

Related information

2.8 VGA and CLCD interfaces on page 2-32
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.4 USB 2.0 connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one USB 2.0 mini-B connector that provides a USB 2.0 Full Speed port. This port supports configuration file editing in the microSD, virtual UART access to the FPGA, and CMSIS-DAP FPGA debug.

The following figure shows the USB 2.0 mini-B connector.

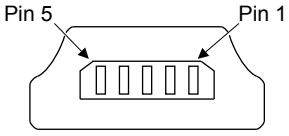

Figure 5-8 USB 2.0 mini-B connector

The following table shows the pin mapping for the USB 2.0 mini-B connector J14.

Table 5-9 USB 2.0 mini-B connector, J14, signal list

Pin	Signal	Pin	Signal
1	**MSD5V**	2	**MSD_DMRAW**
3	**MSD_DPRAW**	4	**MSD_ID**
5	**GND**	6	**GND_EARTH**

Note

- The **GND_EARTH** connection is the casing of the mini-B connector.

Related information

2.5 USB 2.0 Full Speed interface on page 2-29

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

3.2 Remote USB operation on page 3-51

5.5 UART connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one general-purpose female UART connector that supports access to the FPGA.

The general-purpose UART connector connects through a level-shifter to the FPGA. The meaning of the UART signals depends on the design which you implement in the FPGA.

The following figure shows the general-purpose UART connector.

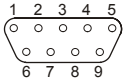

Figure 5-9 General-purpose female UART connector

The following table shows the pin mapping for the general-purpose UART connector J20.

Table 5-10 General-purpose UART connector, J20, signal list

Pin	Signal
1	NC
2	**SER_RX**
3	**SER_TX**
4	NC
5	**SER_GND**
6	NC
7	**SER_RTS**
8	**SER_CTS**
9	NC

Note

SER_RTS loops back to connect to **SER_CTS**.

Related information

2.7 UART interface on page 2-31
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.6 SPI connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one general-purpose SPI connector that supports access to the FPGA.

The general-purpose SPI connector connects to the FPGA. The meaning of the SPI signals depends on the design which you implement in the FPGA.

The following figure shows the general-purpose SPI connector.

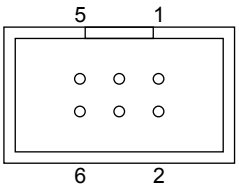

Figure 5-10 General-purpose SPI connector

The following table shows the pin mapping for the general-purpose SPI connector J21.

Table 5-11 General-purpose SPI connector, J21, signal list

Pin	Signal
1	**GND**
2	**SPI_SCK**
3	**SPI_MISO**
4	**SPI_nSS**
5	**3V_EXT**
6	**SPI_MOSI**

Related information

2.6 SPI interface on page 2-30

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.7 VGA connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one VGA connector that supports VGA and CLCD output.

The VGA connector connects to the VGA controller which connects to general-purpose pins on the FPGA. The availability of VGA or CLCD depends on the design which you implement in the FPGA.

The following figure shows the VGA connector.

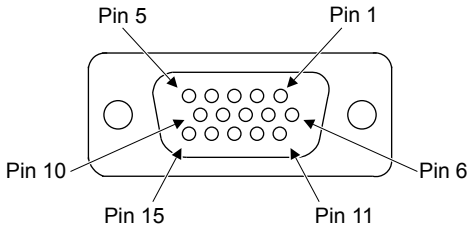

Figure 5-11 VGA connector

The following table shows the pin mapping for the VGA connector J10.

Table 5-12 VGA connector, J10, signal list

Pin	Signal	Pin	Signal
1	**RED**	2	**GREEN**
3	**BLUE**	4	NC
5	**GND**	6	**GND**
7	**GND**	8	**GND**
9	NC	10	**GND**
11	NC	12	NC
13	**HSYNC**	14	**VSYNC**
15	NC		

Related information

2.8 VGA and CLCD interfaces on page 2-32

1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17

1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.8 Audio connectors

The MPS2 and MPS2+ FPGA Prototyping Boards each provide two 3.5mm stereo jack connectors that connect to a low-power stereo audio codec. The two connectors provide an independent line-level stereo input and an independent line-level stereo output.

The audio codec connects to general-purpose pins on the FPGA. The availability of an audio controller depends on the design which you implement in the FPGA.

The following figure shows the audio connector. *Audio In* is component number J32 and *Audio Out* is component number J33 on the board.

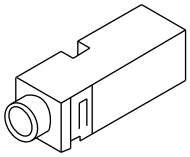

Figure 5-12 Audio connector

Related information

2.9 Audio interface on page 2-33
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.9 Ethernet connector

The MPS2 and MPS2+ FPGA Prototyping Boards provide one 100Mb/s Ethernet connector.

The Ethernet connector connects to the MAC/PHY on the MPS2 and MPS2+ FPGA Prototyping Boards. The MAC/PHY connects to general-purpose pins on the FPGA. The availability of the Ethernet function depends on the design which you implement in the FPGA.

The following figure shows the Ethernet connector J31.

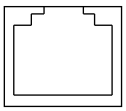

Figure 5-13 Ethernet connector

Related information

2.10 Ethernet interface on page 2-34
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

5.10 12V power connector

The MPS2 and MPS2+ FPGA Prototyping Boards each provide a Thru-hole DC power jack for connecting external power to the board.

Connect the external mains power supply unit, which Arm supplies, to the power jack.

Alternatively, you can connect an external 12V, +/- 10%, power supply to the power jack to supply power to the board.

Note

The center pin is the positive side of the power supply.

Related information

2.14 Power on page 2-42
1.3 Location of components on the MPS2 FPGA Prototyping Board on page 1-17
1.4 Location of components on the MPS2+ FPGA Prototyping Board on page 1-19

Appendix A
Specifications

This chapter contains the electrical specification of the MPS2 and MPS2+ FPGA Prototyping Boards and FPGAs.

It contains the following section:

- *A.1 Electrical specification* on page Appx-A-91.

A.1 Electrical specification

This electrical specification provides information on the current characteristics of the MPS2 and MPS2+ FPGA Prototyping Boards. It also enables you to implement your FPGA design without exceeding the power and current ratings of the board and the temperature rating of the FPGA.

This section contains the following subsection:

- *A.1.1 FPGA current requirements* on page Appx-A-91.

A.1.1 FPGA current requirements

You can calculate current requirements for your MPS2 or MPS2+ FPGA Prototyping Board FPGA image.

See *http://altera.com* for software to help you calculate the current requirements for your particular application. The following table shows the power available for each power domain, or group of power domains, of the FPGA.

Note

This table applies to both the Altera Cyclone 5CEA7 FPGA on the MPS2 FPGA Prototyping Board and to the Altera Cyclone 5CEA9 FPGA on the MPS2+ FPGA Prototyping Board.

Table A-1 Available FPGA current

FPGA power domain	Board power supply	Voltage	Maximum available current from board power supply	Comment
VCCO_3V0 VCCPD_3V0 VCCPGM VREF_3V	3V	3V	50mA	-
VBATT	VBAT	3V	-	-
VCC	1V1	1V1	2.5A	-
VCCAUX VCCO_2V5 VCCPD_2V5 VREF_2V5 VCCA_FPLL	2V5	2V5	1.5A	An analog filter on the MPS2 and MPS2+ FPGA Prototyping Boards smooths the supply to the VCCA_FPLL domain.

Caution

- Some FPGA designs might exceed the current and temperature rating of the board. Therefore you must estimate the power requirements of such designs, using tools such as *Quartus II* before implementation.
- The FPGA package has a thermal resistance of 13.7° per Watt. The recommended maximum FPGA junction temperature is 85°C.

Related information

2.14 Power on page 2-42

Appendix B
Revisions

This chapter describes the technical changes between released issues of this book.

It contains the following section:

- *B.1 Revisions* on page Appx-B-93.

B.1 Revisions

This appendix describes the technical changes between released issues of this book.

Table B-1 Issue A

Change	Location	Affects
No changes, first release	-	-

Table B-2 Differences between issue A and issue B

Change	Location	Affects
Added application notes AN399 and AN400.	*Additional reading* on page 9	All board revisions.

Table B-3 Differences between issue B and issue C

Change	Location	Affects
Added CMSIS-DAP FPGA debug.	Throughout document	All board revisions.
Corrected SCC register descriptions.	*4.3 Register summary* on page 4-62 *4.4 SCC register descriptions* on page 4-63	All board revisions.
Updated example configuration files.	*3.6.3 Contents of the MB directory* on page 3-57	All board revisions.

Table B-4 Difference between issue DDI0525C and issue 100112-0100-03

Change	Location	Affects
Document number changed from Arm DDI0525 to Arm 100112. No technical changes to document.	Throughout document	All board revisions.

Table B-5 Differences between issue 100112-0100-03 and issue 100112-0200-04

Change	Location	Affects
Added the MPS2+ FPGA Prototyping Board so the document now describes both the MPS2 and MPS2+ FPGA Prototyping Boards. The additional references now include the FPGA on the MPS2+ FPGA Prototyping Board, the Altera Cyclone 5CEA9 FPGA.	*1.3 Location of components on the MPS2 FPGA Prototyping Board* on page 1-17 *1.4 Location of components on the MPS2+ FPGA Prototyping Board* on page 1-19 *2.1 Overview of the MPS2 and MPS2+ hardware* on page 2-22 *A.1.1 FPGA current requirements* on page Appx-A-91 Throughout document.	All board revisions.

Table B-6 Differences between issue 100112-0200-04 and issue 100112-0200-05

Change	Location	Affects
Added information about use of the Arduino Adapter Board in the *User expansion port* section. Added the application note for the Arduino Adapter Board to the *Additional Reading* section.	*2.4 User expansion port* on page 2-28 *Additional reading* on page 9	All board revisions.

Table B-7 Differences between issue 100112-0200-05 and issue 100112-0200-06

Change	Location	Affects
Added operating temperature statement.	*1.1.2 Operating temperature* on page 1-13	All board revisions.
Added safety warnings.	*1.1.1 Ensuring safety* on page 1-13 *2.14 Power* on page 2-42	All board revisions.
Added another example images configuration file.	*3.6.3 Contents of the MB directory* on page 3-57	RevB and RevC boards.

Table B-8 Differences between issue 100112-0200-06 and issue 100112-0200-07

Change	Location	Affects
No technical changes to document.	-	All board revisions.
Update document title to match product name change. Update text and diagrams to match change of document title and product name change.	Throughout document.	All board revisions.
Added information about remote USB operation.	*3.2 Remote USB operation* on page 3-51. *2.5 USB 2.0 Full Speed interface* on page 2-29.	All board revisions.

Table B-9 Differences between issue 100112-0200-07 and issue 100112-0200-08

Change	Location	Affects
Added information about remote USB operation with version 3 of the firmware.	*3.2 Remote USB operation* on page 3-51.	All board revisions.

Table B-10 Differences between issue 100112-0200-08 and issue 100112-0200-09

Change	Location	Affects
Updated CE Conformance Notice.	*Conformance Notices* on page 3.	All board revisions.