Versatile Express™ Configuration

Technical Reference Manual

ARM®

ARM DDI 0496B (ID062813)

Versatile Express Configuration
Technical Reference Manual

Release Information

The following changes have been made to this book.

Change history

Date	Issue	Confidentiality	Change
26 June 2012	A	Non-Confidential	First release
28 June 2013	B	Non-Confidential	Second release

Proprietary Notice

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.

Product Status

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

Web Address

`http://www.arm.com`

Conformance Notices

This section contains conformance notices.

Federal Communications Commission Notice

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

CE Declaration of Conformity

CE

The system should be powered down when not in use.

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

- Ensure attached cables do not lie across the card.
- Reorient the receiving antenna.
- Increase the distance between the equipment and the receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Note

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

Contents
Versatile Express Configuration Technical Reference Manual

Preface

This preface introduces the *Versatile Express™ Configuration Technical Reference Manual*. It contains the following sections:

About this book

This book is for the Versatile Express development system.

Intended audience

This document is written for experienced hardware and software developers to aid the development of ARM®-based products using the Versatile Express prototyping and development system.

Using this book

This book is organized into the following chapters:

Chapter 1 *Introduction*

Read this for an introduction to the Versatile Express development system.

Chapter 2 *Configuration Environment*

Read this for a description of the configuration environment of the Versatile Express system.

Chapter 3 *Power-on Sequence*

Read this for a description of the configuration environment of the Versatile Express development system.

Chapter 4 *Push-Button and Remote Resets*

Read this for a description of push-button and remote resets of the Versatile Express development system.

Chapter 5 *Configuration Files*

Read this for a description of the Versatile Express development system configuration files.

Chapter 6 *Updating Motherboard Firmware*

Read this for a description of how to update the firmware of the Versatile Express motherboard.

Chapter 7 *MCC Command-Line Interface*

Read this for a description of the MCC command-line interface of the Versatile Express configuration system.

Appendix A *Revisions*

Read this for a description of 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 *ARM Glossary*, `http://infocenter.arm.com/help/topic/com.arm.doc.aeg0014-/index.html`.

Typographical Conventions

Conventions that this book can use are described in:

- *Typographical*
- *Signals*.

Typographical

The typographical conventions are:

italic	Highlights important notes, introduces special terminology, denotes internal 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 >	Enclose replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0 <Rd>, <CRn>, <CRm>, <Opcode_2>`

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.
Lower-case n	At the start or end of a signal name denotes an active-LOW signal.

Additional reading

This section lists publications by ARM and by third parties.

See Infocenter, `http://infocenter.arm.com`, for access to ARM documentation.

ARM publications

This guide contains information that is relevant to the Versatile Express configuration process. See the following documents for other relevant information:

- *Custom Motherboard Design Guide for Versatile Express™* (ARM DS263-PRDC-013088)

  Note

  This document is available on the media supplied with the V2M-CP1 Programmer Module and on request from ARM support.

- *Motherboard Express™ µATX Technical Reference Manual* (ARM DUI 0447)
- *Design Assist Programmer Module (V2M-CP1) Technical Reference Manual* (ARM DUI 0495)
- *Versatile Express™ Boot Monitor Technical Reference Manual* (ARM DUI 0465)
- *LogicTile™ Express 3MG Technical Reference Manual* (ARM DUI 0449)
- *LogicTile™ Express 13MG Technical Reference Manual* (ARM DUI 0556)
- *CoreTile™ Express A5x2 (V2P-Ca5s) Technical Reference Manual* (ARM DUI 0541)
- *CoreTile™ Express A15x2 (V2P-CA15)Technical Reference Manual* (ARM DUI 0604)
- *CoreTile™ Express A9x4 (V2P-CA9)Technical Reference Manual* (ARM DUI 0448)

Feedback

ARM welcomes feedback on this product and its documentation.

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
- the number, ARM DDI 0496B
- the page numbers to which your comments apply
- a concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Chapter 1
Introduction

This chapter describes the scope of this guide, that describes the configuration process of the Versatile Express prototyping and development system.

It contains the following sections:

1.1 About the Versatile Express prototyping and development system

This guide describes the configuration process of the Versatile Express prototyping and development system.

Versatile Express is based on *Advanced Microcontroller Bus Architecture* (AMBA®) and uses the *Advanced eXtensible Interface* (AXI™) or custom logic for use with ARM processors.

The Versatile Express system has the following possible sets of component boards:

- A Motherboard Express, V2M-P1, with one or both of:
  - A CoreTile™ Express processor daughterboard.
  - A LogicTile™ Express FPGA daughterboard.
- A custom motherboard built under the ARM Design Assist Program with the V2M-CP1 Programmer Module. This configuration can use one daughterboard, either a CoreTile Express daughterboard, or a LogicTile Express FPGA daughterboard.

The Motherboard Express, V2M-P1, provides the ability to configure the Versatile Express system at power-up or reset. The custom motherboard built under the Design Assist Program does not provide this ability. If you use the custom motherboard, you must also use the V2M-CP1 Programmer Module to enable you to configure the system.

For more information on using the Versatile Express system with a custom motherboard, see the *Programmer Module (V2M-CP1) Technical Reference Manual* and *Custom Motherboard Design Guide for Versatile Express*™ and the Technical Reference Manual for the daughterboard you want to use.

For more information on using the Versatile Express system with the Motherboard Express, see the *Motherboard Express µATX (V2M-P1) Technical Reference Manual* and the Technical Reference Manuals for the daughterboards you want to use.

1.2 Target audience

The *Versatile Express™ Configuration Technical Reference Manual* is aimed at highly technical users of the Versatile Express family of ARM prototyping and development boards.

The document assumes that users are software and hardware developers, and that they have some experience of ARM products. The guide does not assume prior experience of Versatile Express.

The general user categories are:

- Software developers running applications in the Linux operating system.
- Software developers creating device drivers or applications that do not use an operating system. This category of user requires details of the *Serial Configuration Control* (SCC) registers present in the test chip on the CoreTile Express daughterboard, or in the FPGA on the LogicTile Express daughterboard. This information is in the relevant daughterboard TRM.
- Hardware and firmware designers developing custom peripherals using the Versatile Express family of ARM development and prototyping boards. The development system consists of one of the following combinations of Versatile Express boards:
  - Motherboard Express, V2M-P1, with one or both of:
    A CoreTile Express daughterboard.
    A LogicTile Express daughterboard.
  - A custom motherboard designed and manufactured under the ARM Design Assist Program, a V2M-CP1 Programmer Module, and either:
    A CoreTile Express processor daughterboard.
    A LogicTile Express FPGA daughterboard.

Other categories of user, such as operating system vendors and porters, debug tool vendors, or engineers responsible for benchmarking ARM-based hardware, are not specifically catered for in the documentation.

Chapter 2
Configuration Environment

This chapter describes the hardware configuration system of the Versatile Express system using either the Motherboard Express, V2M-P1, or a custom motherboard with the V2M-CP1 Programmer Module. It contains the following sections:

2.1 Configuration environment using the Motherboard Express, V2M-P1

This section describes the configuration environment and hardware of the Versatile Express system using the Motherboard Express, V2M-P1, CoreTile Express processor daughterboard, and LogicTile Express FPGA daughterboard.

2.1.1 Overview of the Motherboard Express, V2M-P1, configuration environment

The *Motherboard Configuration Controller* (MCC) is on the motherboard. Each daughterboard has a Daughterboard Configuration Controller. The MCC reads configuration files from a dedicated microSD card accessible as a USBMSD.

Figure 2-1 shows the configuration architecture.

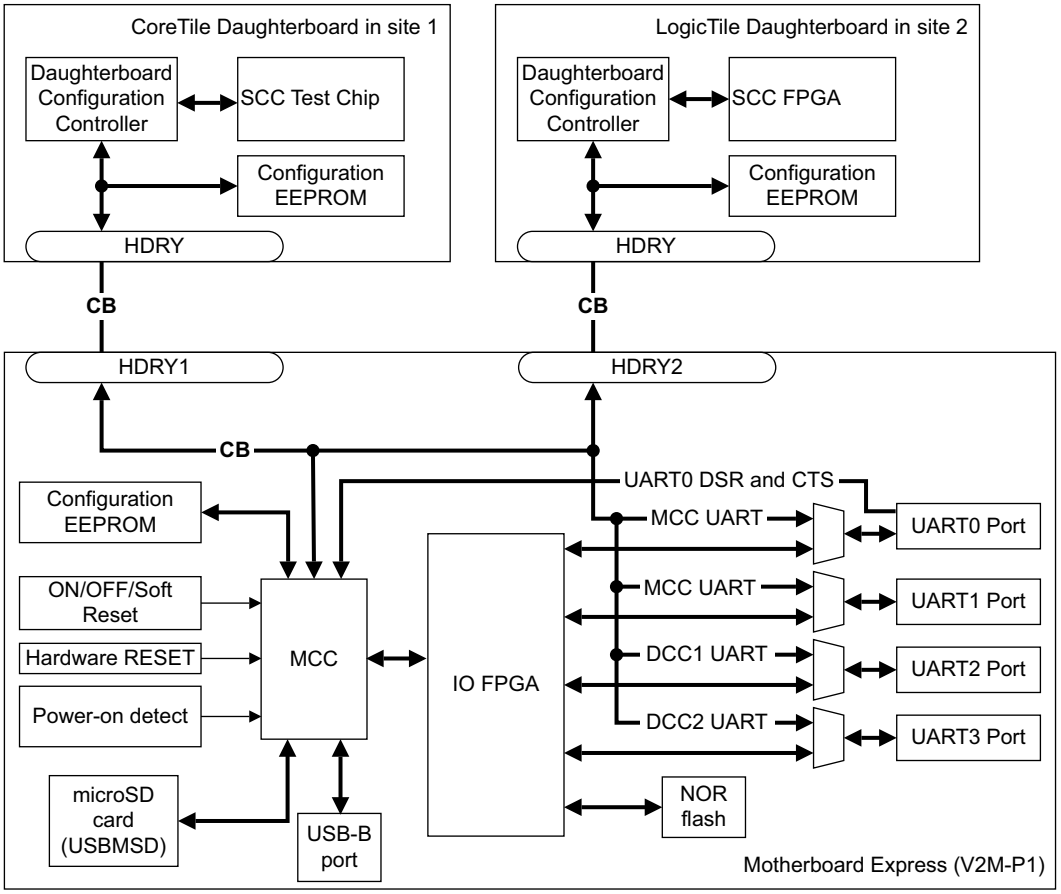

Figure 2-1 Motherboard Express, V2M-P1, configuration architecture

The configuration environment consists of the following hardware components:

- *Motherboard Configuration Controller* (MCC) on the Motherboard Express, V2M-P1.
- Daughterboard Configuration Controller on the CoreTile Express daughterboard and on the LogicTile Express daughterboard.
- Configuration microSD card or *Universal Serial Bus Mass Storage Device* (USBMSD) on the Motherboard Express, V2M-P1.
- Configuration EEPROM on the Motherboard Express, V2M-P1.

- ON/OFF/Soft Reset and Hardware RESET buttons on the on the Motherboard Express, V2M-P1.
- USB-B port on the Motherboard Express, V2M-P1.
- Four UART ports on the Motherboard Express, V2M-P1.
- NOR flash on the Motherboard Express, V2M-P1.
- Power-on detect on the Motherboard Express, V2M-P1.
- Configuration EEPROM on the CoreTile Express daughterboard and on the LogicTile Express daughterboard.
- HDRY headers on the Motherboard Express, V2M-P1, CoreTile Express and LogicTile Express daughterboards.

You can use the Motherboard Express, V2M-P1, with the following configuration hardware options:

USB-B port connected to an external workstation

If an external workstation is connected to the USB-B port, the configuration memory is accessible as a *USB Mass Storage Device* (USBMSD). You can then use the workstation to copy configuration files and software images to the USBMSD configuration memory. See Chapter 5 *Configuration Files*.

MCC command-line operation with an external terminal

If an external workstation running a terminal emulator is connected to the UART0 port in Standby mode, you can use the MCC commands for low-level system debug. After startup and configuration, UART0 or UART1 can access the MCC. See Chapter 7 *MCC Command-Line Interface*.

The daughterboards can log startup information to UART2 and UART3. The UARTs that are enabled depend on the setting in the `config.txt` configuration file. See *config.txt generic motherboard configuration file* on page 5-5.

Boot Monitor application operation with an external terminal

If the USBMSD and NOR flash memory devices contain the correct configuration files, and the Boot Monitor application is selected as the boot image, the system starts up and communicates over UART0 in run mode to a workstation running a terminal emulator. See the *Versatile Express™ Boot Monitor Reference Manual*.

Note

- To eliminate the requirement to use multiple UART ports, the system uses UART0 as the MCC connection during standby, and as the Boot Monitor connection in run mode.
- When required, you can monitor the MCC in run mode through UART1 if MBLOG is enabled in the generic configuration file. Alternatively, remote UART0 control enables switching between System mode and MCC mode on UART0 in run mode. See *Remote reset of the Versatile Express development system* on page 4-7.

Standalone If the USBMSD and NOR flash memory devices contain the correct configuration files and boot image, pressing the **ON/OFF/Soft Reset** switch starts the system without the requirement for any external connections. See *Power-on sequence* on page 3-2.

Remote UART operation

You can use the UART flow-control signals to put the system into standby mode, or select if the UART0 is assigned to the MCC command line interface. See *Remote reset of the Versatile Express development system* on page 4-7 and Table 4-1 on page 4-3.

Remote USB operation

You can control the MCC through the motherboard USB port. 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.

See *config.txt generic motherboard configuration file* on page 5-5.

You initiate a command by putting a file into the USBMSD root directory through the USB port. You must use one of the following filenames:

`reboot.txt`	This reboots the system.
`reset.txt`	This resets the CoreTile.
`shutdown.txt`	This shuts down the system.

See Table 4-1 on page 4-3.

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

Note

The contents of the file have no effect. They can be empty files.

2.2 Configuration environment using the Design Assist custom motherboard

This section describes the configuration environment and hardware of the Versatile Express system using the custom motherboard built under the Design Assist Program, the V2M-CP1 Programmer Module, and either a CoreTile Express processor daughterboard, or a LogicTile Express FPGA daughterboard.

2.2.1 Overview of the V2M-CP1 and custom motherboard configuration environment

Figure 2-2 shows the configuration architecture of the Versatile Express system using the custom motherboard.

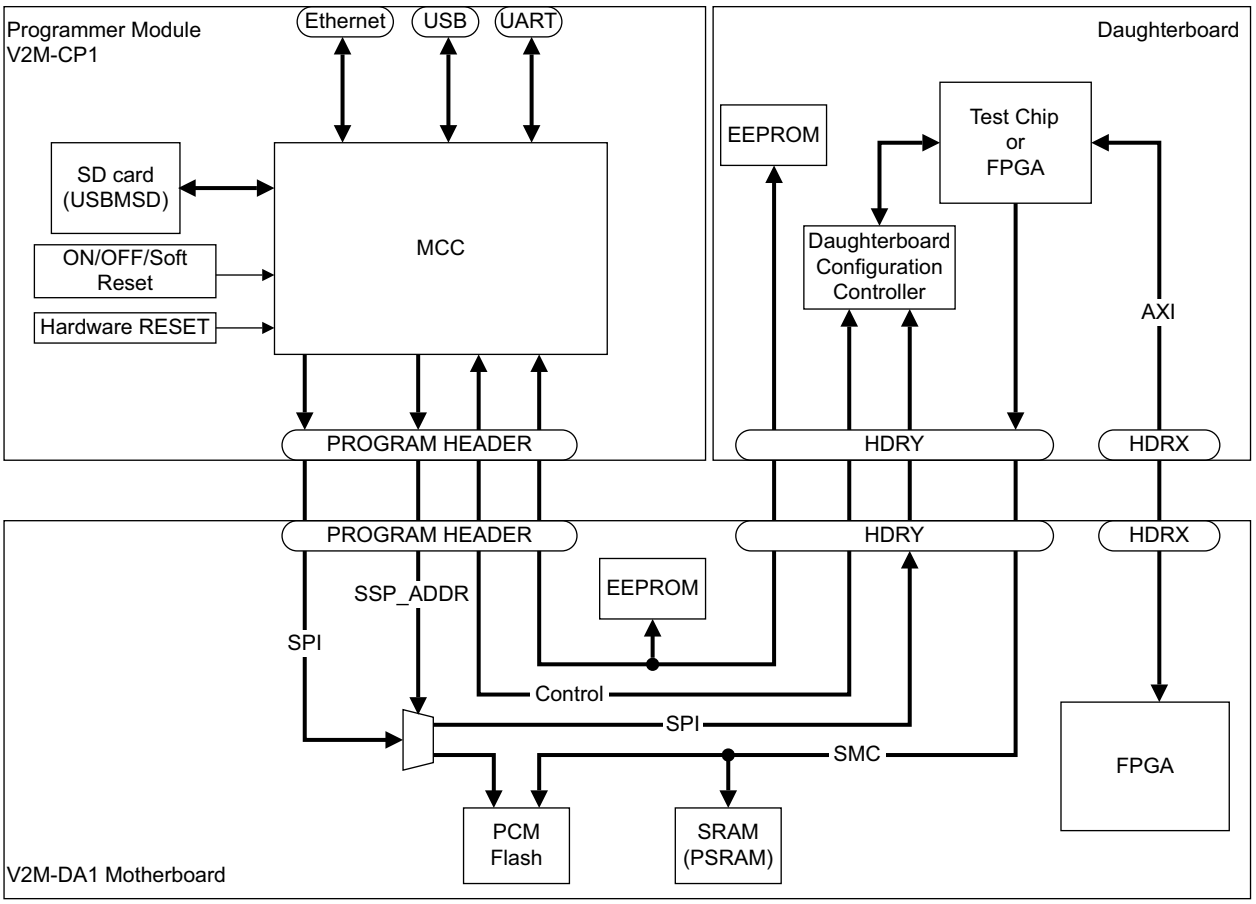

Figure 2-2 Custom motherboard configuration architecture

Note

The daughterboard can be either a CoreTile Express that contains a test chip, or a LogicTile Express that contains an FPGA.

The configuration environment consists of the following hardware components:

- *Motherboard Configuration Controller* (MCC) on the V2M-CP1.
- Daughterboard Configuration Controller on the CoreTile Express daughterboard, or on the LogicTile Express daughterboard, depending on what is fitted.
- Configuration microSD card or *Universal Serial Bus Mass Storage Device* (USBMSD) on the V2M-CP1 Programmer Module.

- ON/OFF/Soft Reset and Hardware RESET buttons on the on the V2M-CP1 Programmer Module.
- USB port on the V2M-CP1 Programmer Module.
- Ethernet port on the V2M-CP1 Programmer Module.

  Note

  The V2M-CP1 Programmer Module does not support the ethernet port.

- Configuration EEPROM on the custom motherboard.
- Configuration EEPROM on the CoreTile Express daughterboard, or on the LogicTile Express daughterboard, depending on what is fitted.
- NAND flash memory on the LogicTile Express daughterboard, if fitted.

Chapter 3
Power-on Sequence

This chapter describes the power-on sequence of the Versatile Express development system. It contains the following section:

- *Power-on sequence* on page 3-2.

3.1 Power-on sequence

Figure 3-1 shows an overview of the power-on sequence.

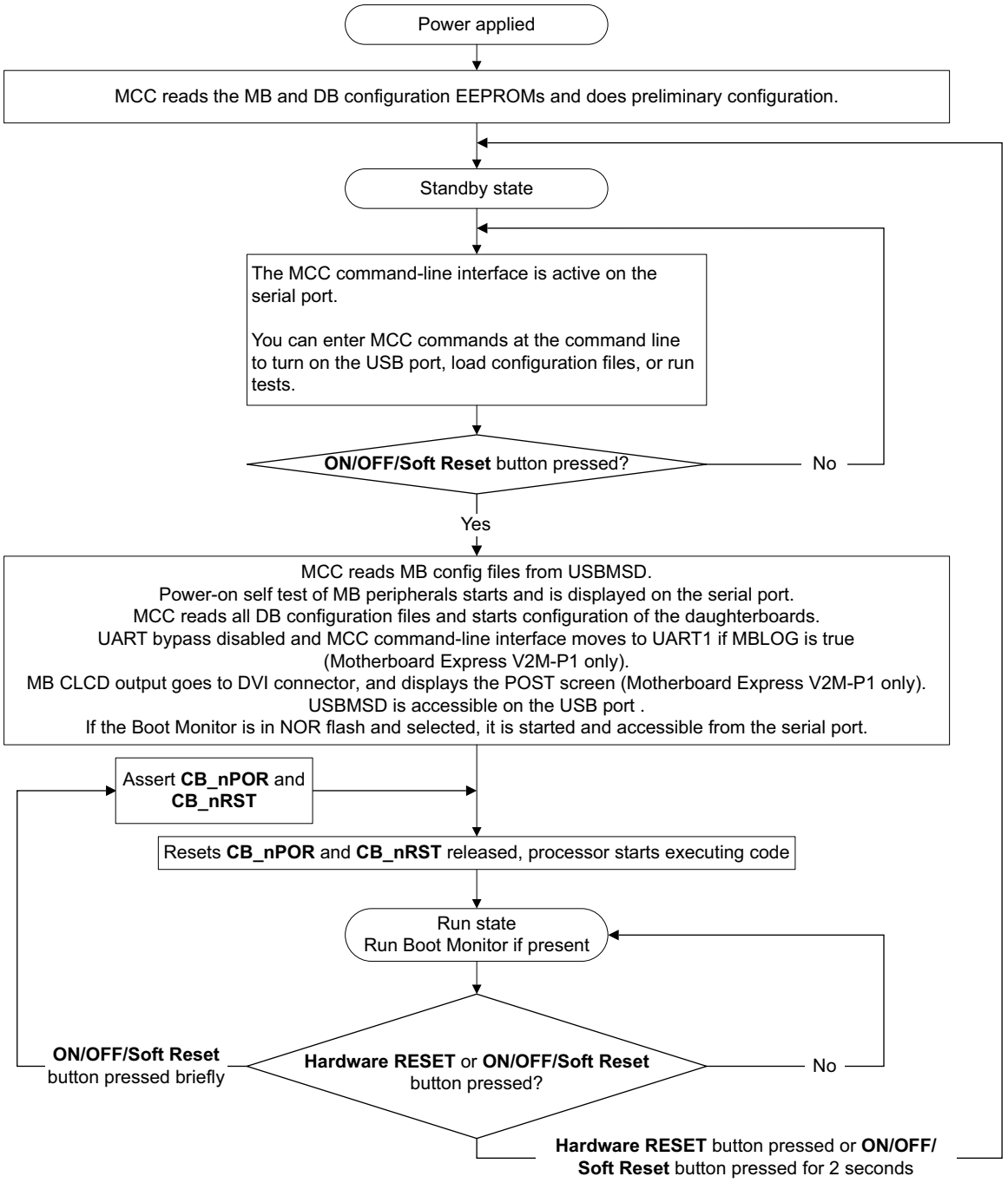

Figure 3-1 Overview of startup sequence

The *serial port* is *UART0* on the Motherboard Express, V2M-P1, and *UART* on the V2M-CP1 Programmer Module. See Figure 2-1 on page 2-2.

The *USB port* is *USB-B* on the Motherboard Express, V2M-P1, and *USB* on the V2M-CP1 Programmer Module. See Figure 2-2 on page 2-5.

3.1.1 Detailed configuration

If enabled in the `config.txt` file, a full system self-test and re-configuration is performed at power-on. See *config.txt generic motherboard configuration file* on page 5-5. The system power-on sequence is as follows:

1. Power is supplied to the system.
2. The MCC powers the EEPROMs on the motherboard and daughterboards, and reads them, to determine the HBI identification codes for the motherboard and any attached daughterboards. The HBI code is a unique number that identifies the board type and version.
3. The system enters standby mode.
4. The MCC command-line interface is enabled on the serial port. See Chapter 7 *MCC Command-Line Interface*.
5. If the `USB_ON` command is received on the serial port, the USBMSD memory card is enabled and you can connect a workstation to the USB-B port on the Motherboard Express, V2M-P1, or USB port on the V2M-CP1 Programmer Module, to drag-and-drop new configuration files.
6. The system stays in the standby state until the ON/OFF/Soft Reset button is pressed, or the serial port receives the `REBOOT` command.
7. The board configuration files are loaded:
   - The MCC reads the generic `config.txt` file.
   - The MCC searches the USBMSD `MB` directory for an `HBIxxxx` subdirectory that matches the motherboard HBI code from the motherboard EEPROM. See Chapter 5 *Configuration Files*.
   - The MCC searches the USBMSD for `SITE1\HBIxxx` and `SITE2\HBIxxx` subdirectories that match the HBI codes from the daughterboard EEPROMs.
8. The next steps depend on the configuration files:
   - If configuration subdirectories are found that match the HBI numbers for all boards present in the system, configuration continues, and the MCC reads the `board.txt` files.
   - If correct configuration files are not found, the MCC records the failure to a log file on the USBMSD and UART0. Configuration stops, and the system re-enters standby mode.
9. The board power supplies are measured.
10. The MCC configures the motherboard clocks, IOFPGA and MUXFPGA.

    Note

    This applies to the Motherboard Express, V2M-P1, only. The V2M-CP1 Programmer Module does not contain an IOFPGA or MUXFPGA.

11. If the MCC finds new software images, they are loaded into the NOR flash.

    Note

    This does not apply to the V2M-CP1 Programmer Module if NOR flash is not fitted.

12. The MCC performs a self-test of the motherboard peripherals and memory.

Note

This applies to the Motherboard Express, V2M-P1, only. The custom motherboard does not contain peripherals to test.

13. The MCC configures the daughterboard clocks and any new FPGA, PLD, or microcontroller images specified in the board configuration files:
    a. The MCC transfers clock settings to the Daughterboard Configuration Controller which programs the daughterboard clocks.
    b. The MCC updates Daughterboard Configuration Controller images.
    c. For daughterboard FPGAs, the image is transferred over the *Configuration Bus* (CB) to a Daughterboard Configuration Controller on the daughterboard that programs the image into local configuration memory.
    d. For JTAG devices and PLDs, the images are directly programmed into the target device by the MCC.
    e. Daughterboard FPGAs, if present, are configured by the local Daughterboard Configuration Controller that streams the data from the local configuration memory to the FPGA. On completion, each Daughterboard Configuration Controller indicates to the motherboard MCC that configuration is complete.
14. The MCC switches to the daughterboard SMB clock:
    a. The MCC loads the Test Chip or FPGA PLL configuration registers with the SCC values from the board configuration files and, for LogicTile Express daughterboards, the application note file.
    b. The MCC waits for the daughterboard PLLs to lock.
    c. The MCC switches the SMB clock to the daughterboard fitted to the custom motherboard, or in site 1 of the Motherboard Express, V2M-P1. This is the daughterboard that contains a processor.
    d. Each daughterboard returns **CB_READY** to signal the end of the configuration.
15. The power-on self-test is complete, and the POST screen in Figure 3-2 is displayed on the DVI-I interface. The USB MSD is enabled.

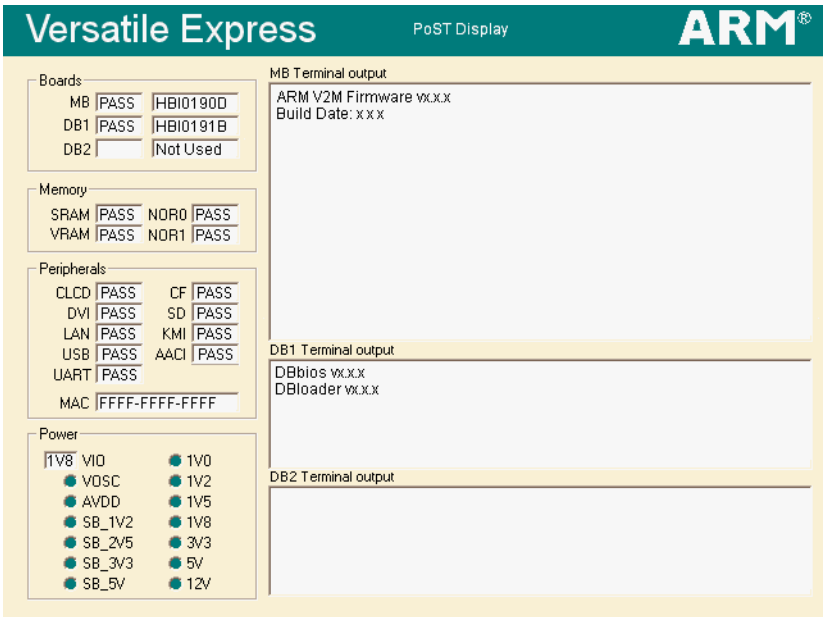

Figure 3-2 Power-on self test screen

Note

This applies to the Motherboard Express, V2M-P1 only.The V2M-CP1 does not contain a DVI-I interface.

16. The MCC releases the **CB_nPOR** and **CB_nRST** system resets, and enables the system to enter the run state, the UART bypasses are released.
17. The processors in the CoreTile Express daughterboard begin executing code.
18. Normal operation continues until a new event:
    - If the **ON/OFF/Soft Reset** is pressed briefly, the processors in the CoreTile Express daughterboard are reset.
    - If the **ON/OFF/Soft Reset** is pressed for two seconds the ATX power supply is powered down and the system enters the standby state. The MCC is still active and its command-line interface is enabled on UART0.
    - If the **Hardware Reset** is pressed, the ATX power supply is powered down and the system enters the standby state. The MCC is still active and its command-line interface is enabled on UART0.
    - An external reset request is received on the JTAG **nSRST** line from the debugger.

Example 3-1 shows an example boot log.

Example 3-1 Boot log

```
1.          ARM V2M Boot loader vx.x.x
            HBI0xxxx build xxx
2.          ARM V2M Firmware vx.x.x
            Build Date: x x x

            Date: x x x x
            Time:     x x x
3.4.5.      Cmd>
6.          Powering up system...
            Daughterboard 1 detected.

7.8.        Switching on ATXPSU...
9.          ATX3V3: ON
            VIOset: 1.8V
            MBtemp: 28 degC

10.         Configuring motherboard (rev x, var x)...
            IOFPGA  config: PASSED
            MUXFPGA config: PASSED
            OSC CLK config: PASSED

11.         Testing SMC devices (FPGA build x)...
12.         SRAM 32MB test: PASSED
            VRAM  8MB test: PASSED
            LAN9118   test: PASSED
            USB & OTG test: PASSED
            KMI1/KMI2 test: PASSED
            MMC & SD  test: PASSED
            DVI image test: PASSED
            AACI AC97 test: PASSED
            CF card   test: PASSED
            UART port test: PASSED
            MAC addrs test: PASSED
```

```
            Reading Site 1 Board File \SITE1\HBI0191x\board.txt
13a.        Setting DB1 OSCCLKS...
13c.        DB1 SPI configuration complete.
13d.        DB1 JTAG configuration complete.

14a.        Writing SCC 0x40610000 with 0xXXXXXXXX
            Writing SCC 0x40610001 with 0xXXXXXXXX
            Writing SCC 0x40610002 with 0xXXXXXXXX
14b         DB1 SCC configuration complete.

14c.          DB1 SMC clock enabled.
14d.          Waiting for CB1_READY...
              Testing DB SMB clock...
15.           Configuring MUXFPGA for MB.
              Setting DVI mode for VGA.
              Enabling debug USB.
16.           Releasing Daughterboard resets.
              Switching off DCC log to UART2/3.
              Switching MCC log to UART1.

17.           ARM Versatile Express Boot Monitor
              Version:    Vx.x
              Build Date: x x x
              Tile Site 1: HBI-0191 Cortex A9
              Tile Site 2: Tile Not Fitted
              >
```

Chapter 4
Push-Button and Remote Resets

This chapter describes the push-button and remote resets. It contains the following sections:

4.1 Push-button resets of the Versatile Express development system

This section describes the push-button reset operation of the Versatile Express development system that uses either:

- The Motherboard Express, V2M-P1.
- The V2M-CP1 Programmer Module and custom motherboard.

4.1.1 Push-button reset operation

There are two push button switches on the back panel of the Motherboard Express, V2M-P1, and on the V2M-CP1 Programmer Module.This section describes the power-on, Hardware RESET, and Soft reset transitions of the Hardware RESET and ON/OFF/Soft reset buttons.

- Motherboard Express, V2M-P1:
  - The Hardware Reset button is the BLACK button.
  - The ON/OFF/Soft Reset button is the RED button.

  See the *Motherboard Express µATX Technical Reference Manual*.
- V2M-CP1 Programmer Module and custom motherboard:
  - The Hardware Reset button is button S2.
  - The ON/OFF/Soft Reset button is button S1.

  See the *Design Assist Programmer Module (V2M-CP1) Technical Reference Manual*.

ON/OFF/Soft Reset

The function of this button depends on how long it is pressed:

1. From Standby:

   Briefly press the **ON/OFF/Soft Reset** switch to power-on the system.
2. In Run mode:

   Briefly press the **ON/OFF/Soft Reset** switch to perform a software reset of the main ARM CPU in the Versatile Express system. This is typically:
   - A CoreTile Express daughterboard fitted in Site 1 of the Motherboard Express, V2M-P1.
   - A CoreTile Express daughterboard fitted in header HDRY of the custom motherboard.
3. Press and hold the **ON/OFF/Soft Reset** switch for more than two seconds to return the system to the standby state.

Hardware RESET

Press the **Hardware RESET** switch to return the system to the standby state. The MCC is still active and you can access it from:

- UART0 and USB-B on the Motherboard Express, V2M-P1.
- UART and USB on the V2M-CP1 Programmer Module.

Note

In standby mode, you can enable the USBMSD using the `USB_ON` command.

Figure 4-1 on page 4-3 and Table 4-1 on page 4-3 summarize the transitions between states using the push buttons.

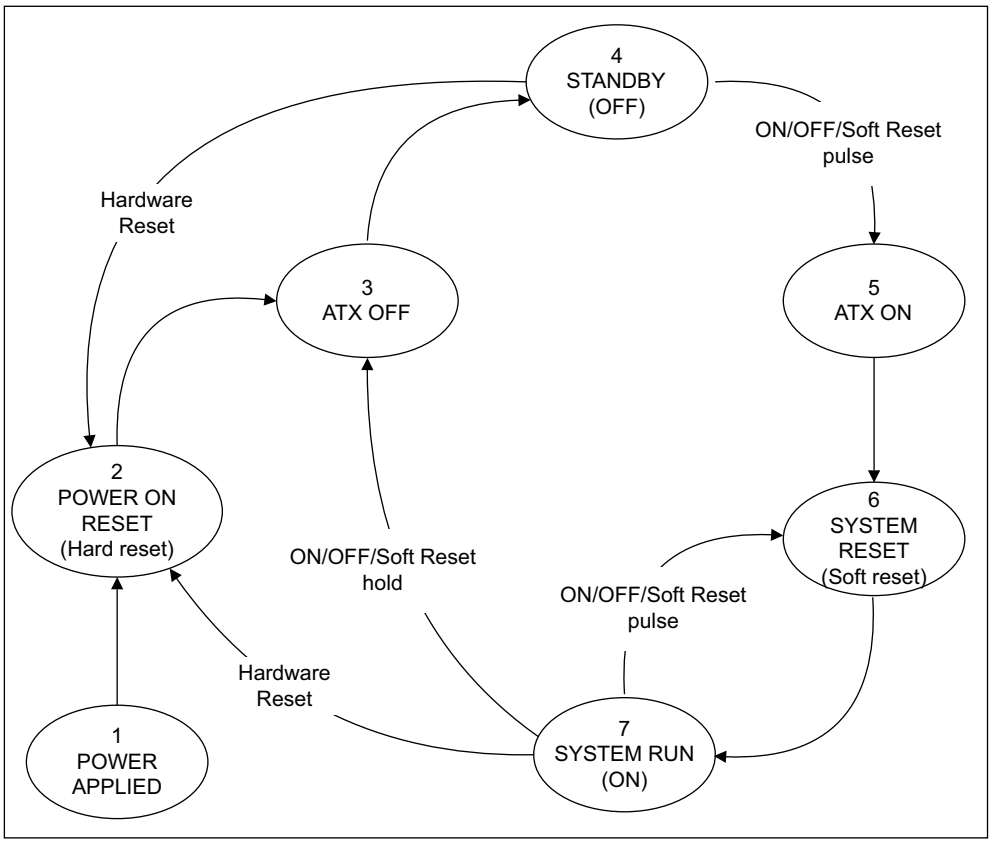

Figure 4-1 Push-button reset state diagram

Table 4-1 also shows how these same transitions can be achieved using remote USB or remote UART control. See Chapter 2 *Configuration Environment*.

Table 4-1 Soft and hard reset state transitions

Transition	Button press	Function	Remote operation, USB	Remote operation, UART/SERIAL
1 > 2 > 3 > 4	-	Power applied	-	-
4 > 5 > 6 > 7	ON/OFF/Soft Reset pulse	System running	`reboot.txt`	Reboot
7 > 6 > 7	ON/OFF/Soft Reset pulse	Soft reset	`reset.txt`	Reset
7 > 3 > 4	ON/OFF/Soft Reset hold	Standby (OFF)	`shutdown.txt`	Shutdown
7 > 3> 4 4 > 5 > 6 > 7	ON/OFF/Soft Reset hold ON/OFF/Soft Reset pulse	Reboot	`reboot.txt`	Reboot
4 > 2 > 3 > 4 or 7 > 2 > 3 > 4	Hardware Reset	Hard (Power ON) reset	-	-

Note

- *ON/OFF/SoftReset pulse* means push the ON/OFF/SoftReset button for less than two seconds.

- *ON/OFF/SoftReset hold* means hold the ON/OFF/SoftReset button for more than two seconds.

4.2 Configuration switches

This section describes the configuration switches present on the Motherboard Express, V2M-P1, and on the V2M-CP1 Programmer Module. These affect the Versatile Express initialization and configuration.

Note

Most of the configuration options are controlled by files on the USBMSD memory card. See Chapter 5 *Configuration Files*.

4.2.1 Configuration switches on the Motherboard Express, V2M-P1

The Motherboard Express, V2M-P1, has two configuration switches on the back panel.

The `config.txt` file contains `USERSWITCH` and `CFGSWITCH` entries for the virtual switch registers SYS_SW[7:0] and SYS_CFGSW[7:0] in the IO FPGA. These virtual switches are not used for system configuration, but they are available for the user application and Boot Monitor. See Chapter 4, *Programmers Model* of the *Motherboard Express µATX Technical Reference Manual*.

Boot script switch SW[0]

If SW[0] is ON, or the `config.txt` entry for `USERSWITCH[0]` is set to 1, SYS_SW[0] is set to 1.

If SYS_SW[0] is set to 1, the Boot Monitor runs its boot script at startup. See the *Versatile Express™ Boot Monitor Reference Manual*.

SYS_SW[0] can also be modified by a user application, but the change does not take effect until the next reset.

Remote UART0 control switch SW[1]

If SW[1] is ON, remote UART0 control is enabled and the flow-control signals on UART0 can control standby and the UART that is the MCC command line interface. This setting is typically used in test farms. See *Remote reset of the Versatile Express development system* on page 4-7.

Setting switch SW[1] to ON also sets SYS_SW[31] to 1. SYS_SW[31] is read-only by user applications.

Note

- The default setting for configuration switches SW[0] and SW[1] is OFF.
- If the switches are in the up position, they are OFF.
- See the *Versatile Express™ Boot Monitor Reference Manual* for other SYS_SW register bits that the Boot Monitor uses.

4.2.2 Configuration switch on the V2M-CP1 Programmer Module

The V2M-CP1 Programmer Module has the following configuration switch:

Remote UART control switch S3[2]

This switch provides the same functionality as switch SW[1] on the Motherboard Express, V2M-P1, except for the following differences:

- Only UART0 can be the MCC command line interface on the V2M-CP1 Programmer Module.

- On the Motherboard Express, V2M-P1, setting the remote UART0 to ON also sets SYS_SW[31] to 1 in the IOFPGA. This is not true on the V2M-CP1 Programmer Module because it does not contain an IOFPGA.

See *Configuration switches on the Motherboard Express, V2M-P1* on page 4-5.

Note

- The default setting for configuration switch S3[2] is OFF.
- If the switch is in the up position, it is OFF.
- See the *Versatile Express™ Boot Monitor Reference Manual* for other SYS_SW register bits that the Boot Monitor uses.

4.3 Remote reset of the Versatile Express development system

This section describes the remote reset of the Versatile Express system using either:

- The Motherboard Express, V2M-P1.
- The V2M-CP1 Programmer Module and custom motherboard.

Note

SW[1] on the Motherboard Express, V2M-P1, or S3[2] on the V2M-CP1 Programmer Module, must enable remote UART0 control. See *Configuration switches* on page 4-5.

The remote reset sequence is as follows:

- An external controller can toggle UART0 **DSR**, pin 6, HIGH for 100ms to put the motherboard into standby mode. This is equivalent to pushing the **Hardware RESET** button. Power cycling the board also places the system into standby mode.

  Note

  The duration of the DSR high pulse must be greater than or equal to 100ms.

- An external controller can remotely select whether the MCC or the system application uses UART0 in run mode. This overrides the `config.txt` entry for MBLOG and eliminates the requirement to use the second serial port on UART1.

  Set UART0 **CTS**, pin 8, LOW to select system mode, or set it HIGH to select MCC mode.

  Remote UART0 control requires a full null modem cable that is supplied with the motherboard. Figure 4-2 shows the wiring.

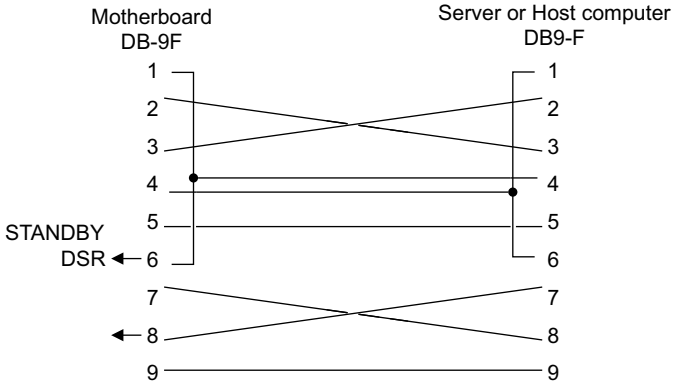

Figure 4-2 Modem cable wiring diagram

Note

The V2M-CP1 Programmer Module has reduced remote UART control. See *Configuration switch on the V2M-CP1 Programmer Module* on page 4-5.

You can achieve control of the DSR and CTS signals through control logic on the host computer.

Alternatively, you can use a custom terminal program, such as ARM `VETerminal.exe`, that is provided on the Versatile Express DVD. This program integrates the terminal output and control buttons into a single application.

Chapter 5
Configuration Files

This chapter describes the configuration files of the Versatile Express development system. It contains the following sections:

5.1 Overview of configuration files

The MCC reads configuration files from the dedicated microSD flash memory card on the Motherboard Express, V2M-P1, or custom motherboard, and uses the contents to configure the motherboard and daughterboards.

Because the motherboard flash memory is non-volatile memory, it is only necessary to load new configuration files if you change the system configuration. The USBMSD can store configuration files for multiple motherboard and daughterboard variants. The MCC uses the configuration files that match the boards in the system. The Motherboard Express, V2M-P1, is shipped with default configuration files.

If you connect a PC to the USB-B configuration port on the V2M-P1 Motherboard Express or the USB port on the V2M-CP1 Programmer Module, the configuration memory device appears as a *USB Mass Storage Device* (USBMSD), and you can add or delete files.

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

Figure 5-1 on page 5-3 is an example Motherboard USBMSD directory structure.

Caution

Files and directories are in an 8.3 file format, that is, the name of the file is a maximum of 8 characters long, and the extension is 3 characters. The following rules apply:

- File names must always be in lower case.
- Directory names must be in upper case.
- All configuration files must end in DOS line endings, that is, `0x0D`/`0x0A`.

Example 5-1 shows examples of valid and invalid 8.3 format file names.

Example 5-1 File names in 8.3 format

`board.txt`	This is a valid file name in 8.3 format.
`a224r1p0.txt`	This is a valid file name in 8.3 format.
`nametoolong.txt`	This is an invalid file name in 8.3 format. The name is longer than 8 characters.
`a224r1p0.text`	This is an invalid file name in 8.3 format. The extension is longer than 3 characters.

Figure 5-1 on page 5-3 shows a typical Motherboard Express, V2M-P1, USBMSD directory structure.

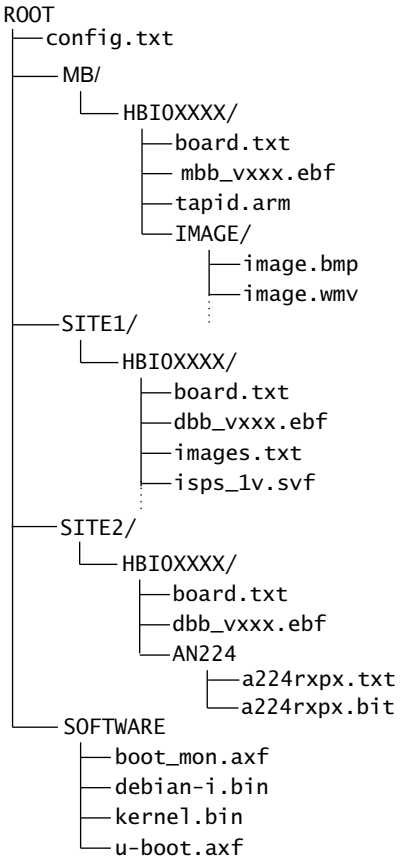

Figure 5-1 Typical Motherboard Express USBMSD directory structure

Figure 5-2 shows a typical custom motherboard directory structure.

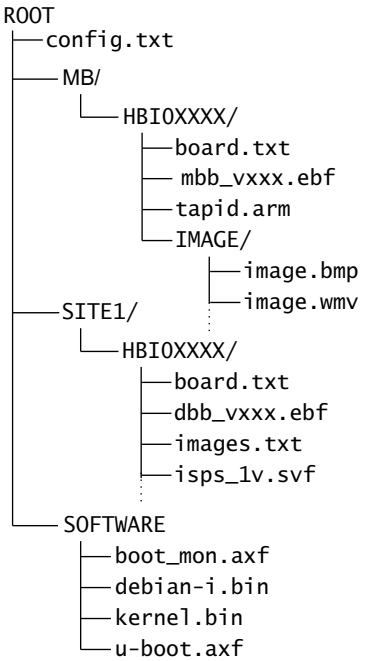

Figure 5-2 Typical custom motherboard USBMSD directory structure

The directory structure and file name format ensure that each image is matched to the correct target device that the board configuration EEPROMs define:

- For CoreTile Express boards, there is a single `board.txt` and a single `image.txt` file. For LogicTile Express boards, there is a `board.txt` file plus an Application note configuration file. These files contain the image files and clock settings for the daughterboards.
- `config.txt` is the generic configuration file for all motherboards. It also contains configuration information for all daughterboards.
- The `MB` directory contains subdirectories for any motherboard variants that might be present in the system. The subdirectory name matches the HBI codes for the specific motherboard variants. This file contains image files and clock settings for the motherboard.
- The `SITE1` directory contains subdirectories for processor boards that you can fit to the custom motherboard or to Site 1 of the Motherboard Express, V2M-P1. These are typically CoreTile Express daughterboards, but you can also load a LogicTile Express daughterboard with an FPGA image that enables it to function as a processor board.
- The `SITE2` directory contains subdirectories for boards that you can place in daughterboard Site 2 of the Motherboard Express, V2M-P1. These are typically LogicTile Express daughterboards.

  Note

  This applies only to the Motherboard Express, V2M-P1.

- The `SOFTWARE` directory contains application files that you can load to the NOR flash on the motherboard. The files that are actually loaded depend on the `NORxFILE` settings in the `SITEx/HIBxxx/image.txt` file.

See the documentation for your CoreTile Express and LogicTile Express daughterboards for any information that is not specified in this document, for the specific board configuration files.

5.2 config.txt generic motherboard configuration file

You can use the USB-B configuration port on the Motherboard Express, V2M-P1, or the USB port on the custom motherboard, to update the config.txt generic board configuration file from your workstation to the root directory of the USBMSD flash.

Example 5-2 shows a configuration file that you can load to the microSD card on the Motherboard Express, V2M-P1.

Note

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

Example 5-2 Motherboard ExpressV2M-P1 config.txt file

```
TITLE: Versatile Express Configuration File

[CONFIGURATION]
AUTORUN: FALSE                  ;Auto Run from power on
TESTMENU: FALSE                 ;MB Peripheral Test Menu

UPDATE: FALSE                   ;Force JTAG and FPGA update to DBs
VERIFY: FALSE                   ;Force FPGA verify to DBS

SLIDESHOW: FALSE                ;CLCD Slideshow
MUXFPGA: MB                     ;MB or DB1 or DB2
DVIMODE: VGA                    ;VGA or SVGA or XGA or SXGA or UXGA

MBLOG: TRUE                     ;LOG MB MICRO TO UART1 in run mode
DBLOG: TRUE                     ;LOG DB MICRO TO UART2/3 in run mode

USERSWITCH: 00000000            ;UserSwitch[7:0] in binary
CONFSWITCH: 00000000            ;Configuration Switch[7:0] in binary
ASSERTNPOR: FALSE               ;External resets assert nPOR
WDTRESET: RESETMB               ;Watchdog reset options NONE/RESETMB/RESETDB/
PCIMASTER: DB1                  ;Port Failover DB1/DB2/NONE
MASTERSITE: DB1                 ;Boot Master DB1/DB2
MEMORYMAP: LEGACY               ;Select Memory map LEGACY/AS1
USB_REMOTE: FALSE               ;Selects remote command via USB

MACADDRESS: OxFFFFFFFFFFFF      ;MAC Address
```

Example 5-3 shows a configuration file that you can load to the microSD card on the custom motherboard.

Example 5-3 Example custom motherboard config.txt file

```
TITLE: Versatile Express Configuration File

[CONFIGURATION]
AUTORUN: FALSE                  ;Auto Run from power on
TESTMENU: FALSE                 ;MB Peripheral Test Menu

UPDATE: FALSE                   ;Force JTAG and FPGA update to DBs
VERIFY: FALSE                   ;Force FPGA verify to DBS
```

```
ASSERTNPOR: FALSE              ;External resets assert nPOR
MEMORYMAP: LEGACY              ;Select Memory map LEGACY/AS1
USB_REMOTE: FALSE              ;Selects remote command via USB

MACADDRESS: OxFFFFFFFFFFFF     ;MAC Address
```

CONFIGURATION section

The [CONFIGURATION] section of the config.txt file defines generic settings that you can use for any motherboard variant.

AUTORUN, TESTMENU, UPDATE, VERIFY, SLIDESHOW, MBLOG, DBLOG, and ASSERTNPOR are control values that you can set to TRUE or FALSE. Set to FALSE for default operation.

AUTORUN
: Forces the motherboard to enter the power-up sequence when power is applied, or the **Hardware RESET** button is pressed.

TESTMENU
: Display the test menu on the terminal display connected to UART0. This is typically only used for test purposes. It runs the full motherboard peripheral self-test on startup.

UPDATE
: Force updates of the daughterboard JTAG devices and the FPGAs. This is typically only used for test purposes. It forces all of the images in the system to be updated from the micro SD card.

VERIFY
: Force verification of the daughterboard FPGAs. This is typically only used for test purposes. This performs a verify on the daughterboard Nand Flash.

SLIDESHOW
: Display a series of images to the CLCD display. This is typically only used for test purposes.

  Note

  This setting is valid only for the Motherboard Express, V2M-P1, config.txt file.

MUXFPGA
: Selects the device to which to supply video and audio to the DVI-I connector: DB1, default, DB2, or MB.

  DB1 is the CoreTile Express daughterboard fitted to Site 1 of the Motherboard Express, V2M-P1, and typically has a CLCD controller. DB2 is the LogicTile Express in Site 2, and you can use it if a custom CLCD controller is implemented in the FPGA. The MB CLCD controller is used if neither of the daughterboards has its own controller.

  Note

  This setting is valid only for the Motherboard Express, V2M-P1, config.txt file.

DVIMODE
: Selects the default resolution for the DVI-I output as VGA, SVGA, XGA, SXGA or UXGA.

  Note

  This setting is valid only for the Motherboard Express, V2M-P1, config.txt file.

MBLOG
Specify as TRUE to have the output from the MCC sent to the UART1 in run mode.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

DBLOG
Specify as TRUE to have the output from `DB1` Daughterboard Configuration Controller sent to UART2 and the output from `DB2` Daughterboard Configuration Controller sent to UART3 in run mode.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

USERSWITCH
Load the soft user switch. Your applications can read this. See the *Motherboard Express µATX Technical Reference Manual*.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

This switch is not used for system configuration, but you can use it for application code, for example Boot Monitor.

CONFSWITCH
Load the soft configuration switch. Your applications can read this. See the *Motherboard Express µATX Technical Reference Manual*.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

This switch is not used for system configuration, but you can use it for application code.

ASSERTNPOR
Determines whether the **nPOR** reset signal is also asserted when an external source asserts **nRST**. You can generate this through a JTAG **nSRST** signal, or by pressing the **ON/OFF/Soft Reset** briefly in run mode.

WDTRESET
Selects the action to be performed when the Watchdog peripheral on the MotherBoard generates a reset. **RESETMB** reboots the motherboard, and **RESETDB** resets the daughterboard fitted to Site 1 of the Motherboard Express, V2M-P1. NONE ignores the reset.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

PCIMASTER
Selects the PCI failover mode. DB1 selects Site 1 of the Motherboard Express, V2M-P1, as the upstream port. DB2 selects Site 2 as the upstream port. NONE selects failover not enabled.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

`MACADDRESS` This entry can specify a new value for the LAN MAC address. The new value overwrites the default LAN MAC address that the LAN controller uses.

The default LAN MAC address is permanently stored in an EEPROM associated with the LAN controller. To restore the default LAN MAC address, set `MACADDRESS` to `0xFFFFFFFFFFFF`.

`MASTERSITE` Selects the site of the Motherboard Express, V2M-P1, where the processor core is to boot from.

Note

This setting is valid only for the Motherboard Express, V2M-P1, `config.txt` file.

`MEMORY_MAP` Selects the ARM Legacy map or the ARM *Cortex-A Series* memory map.

`USB_REMOTE` Enables remote commands through USB.

5.3 Contents of the motherboard directory

The motherboard directory, that is, the directory named `MB`, contains configuration directories for the supported motherboard variants. Each subdirectory name matches the HBI number of the motherboard.

The Motherboard Express, V2M-P1, HBI directory contains:

- A single `board.txt` file that contains information specific to the motherboard variant.
- An `IMAGE` subdirectory containing bitmap (`.bmp`) and audio files (`.wav`) that you can use through the slide show application and POST Screen.
- Image files for the FPGAs, PLDs, and microcontrollers present on the Motherboard Express, V2M-P1. These have `.bit`, `.ebf`, or `.svf` extensions.
- A `tapid.arm` file that list the TAP controller register lengths. This is used for configuring JTAG devices.

The custom motherboard HBI directory contains:

- A single `board.txt` file that contains information specific to the motherboard variant.
- Image files for the microcontroller present on the motherboard. These have `.ebf` extensions.
- A `tapid.arm` file that list the TAP controller register lengths. This is used for configuring JTAG devices.

Example 5-4 shows a typical Motherboard Express `board.txt` file.

Example 5-4 Typical Motherboard Express, V2M-P1, `board.txt` file

```
BOARD: HBI0190
TITLE: MotherBoard Configuration File

[MCCS]
MBBIOS: mbb_v217.ebf            ;MB BIOS IMAGE

[FPGAS]
MBIOFPGA: io_b105.bit           ;REQUIRED TO ALLOW UPDATE FROM VECD v1.x
MBIOFPGA1: io_b105.bit          ;MB LEGACY IOFPGA IMAGE FOR SITE 1 BOOT MASTER
MBIOFPGA2: io_b205.bit          ;MB LEGACY IOFPGA IMAGE FOR SITE 2 BOOT MASTER
MBIOFPGA3: io_b111.bit          ;MB A SERIES IOFPGA IMAGE FOR SITE 1 BOOT MASTER
MBIOFPGA4: io_b211.bit          ;MB A SERIES IOFPGA IMAGE FOR SITE 2 BOOT MASTER
MBMUXFPGA: mux_b1c.bit          ;MB MUXFPGA IMAGE

[OSCCLKS]
TOTALOSCCLKS: 6
OSC0: 50.0                      ;OSC0 MB SMB clock in MHz
OSC1: 23.75                     ;OSC1 MB CLCD clock in MHz
OSC2: 24.0                      ;OSC2 IOFPGA REFCLK in MHz
OSC3: 24.0                      ;OSC3 Reserved
OSC4: 24.0                      ;OSC4 SB_GCLK in MHz
OSC5: 24.0                      ;OSC5 Reserved
```

Example 5-5 on page 5-10 shows a typical custom motherboard `board.txt` file.

Example 5-5 Typical custom motherboard `board.txt` **file**

```
BOARD: HBI0226
TITLE: V2M-DA1 MotherBoard Configuration File

[MCCS]
MBBIOS: mbb_v217.ebf              ;MB BIOS IMAGE

[FPGAS]

[OSCCLKS]
TOTALOSCCLKS: 0
```

MCCS section

The `MCCS` section lists each MCC on the motherboard and the image to load. These are the motherboard configuration controllers.

FPGAS section

The `FPGAS` section lists the FPGAs on the motherboard and the images to load. These are the IO FPGA and DVI multiplexer MUX FPGA.

Note

The custom motherboard does not use this section.

OSCCLKS section

The `OSCCLKS` section lists the number of programmable clocks and the frequency for each. See the *Motherboard Express µATX Technical Reference Manual*.

Note

The custom motherboard does not use this section.

5.4 Contents of the directory for CoreTile Express boards

The CoreTile Express boards are fitted to the custom motherboard or to SITE1 of the Motherboard Express, V2M-P1. The SITE1 directory contains configuration directories for the supported daughterboard variants. Each subdirectory in SITE1 has a name in the form `HBIxxxxx`, and this matches the HBI code of the daughterboard.

5.4.1 Overview of the directory for CoreTile Express boards

The HBI subdirectories are typically for CoreTile Express daughterboards, but you can load a LogicTile Express daughterboard with a custom FPGA image that enables it to function as a processor board. If a LogicTile Express daughterboard is fitted to the custom motherboard, or to Site 1 of the Motherboard Express, V2M-P1, its HBI directory uses the format that *Contents of the directory for LogicTile Express boards* on page 5-16 describes.

For CoreTile Express daughterboards, each HBI subdirectory contains:

- A single `board.txt` file that contains information specific to the daughterboard variant.
- An `image.txt` file containing flash images.
- Image files for the FPGAs, PLDs, and Daughterboard Configuration Controllers present on the daughterboard. These have `.bit`, `.svf`, or `.ebf` extensions.

5.4.2 board.txt file

Example 5-6 shows the contents of a typical `board.txt` file for a daughterboard fitted to the custom motherboard or Site 1 of the Motherboard Express, V2M-P1.

Example 5-6 Typical `board.txt` file for Site 1

```
BOARD: HBIXXXX
TITLE: V2P-CA9 test build 1

; Do not place comments within the [SECTION] blocks.

[DCCS]
TOTALDCCS: 1                    ;Total Number of DCCS (Max:8)
M0FILE: dbb_vXXX.ebf            ;DCC0 Filename
M0MODE: MICRO                   ;DCC0 Programming Mode

[FPGAS]
TOTALFPGAS: 0                   ;Total Number of FPGAS (Max:8)
F0FILE: NONE                    ;FPGA0 Filename
F0MODE: NONE                    ;FPGA0 Programming Mode

[TAPS]
TOTALTAPS: 6                    ;Total Number of TAPS (Max:32)
T0NAME: STM32TMC                ;TAP0 Device Name
T0FILE: NONE                    ;TAP0 Filename
T0MODE: NONE                    ;TAP0 Programming Mode
T1NAME: STM32CM3                ;TAP1 Device Name
T1FILE: NONE                    ;TAP1 Filename
T1MODE: NONE                    ;TAP1 Programming Mode
T2NAME: ispCLOCK5610V           ;TAP2 Device Name
T2FILE: XXX.svf                 ;TAP2 Filename
T2MODE: PLD                     ;TAP2 Programming Mode
T3NAME: ispCLOCK5610V           ;TAP3 Device Name
T3FILE: XXX.svf                 ;TAP3 Filename
```

```
T3MODE: PLD                     ;TAP3 Programming Mode
T4NAME: XC2C64A                 ;TAP4 Device Name
T4FILE: XXX.svf                 ;TAP4 Filename
T4MODE: PLD                     ;TAP4 Programming Mode
T5NAME: XC2C64A                 ;TAP5 Device Name
T5FILE: XXX.svf                 ;TAP5 Filename
T5MODE: PLD                     ;TAP5 Programming Mode

[OSCCLKS]
TOTALOSCCLKS: 3                 ;Total Number of OSCCLKS (Max:8)
OSC0: XX.X                      ;OSC0 Frequency in MHz (EXTSAXICLK)
OSC1: XX.X                      ;OSC1 Frequency in MHz (CLCDCLK)
OSC2: XX.X                      ;OSC2 Frequency in MHz (TCREFCLK)

[SCC REGISTERS]
TOTALSCCS: 3       ;Total Number of SCC registers defined
SCC: 0x000 0xXXXXXXXX           ;SCC general read/write register address/value
SCC: 0x004 0xXXXXXXXX           ;SCC general read/write register address/value
SCC: 0x008 0xXXXXXXXX           ;SCC general read/write register address/value
```

DCCS section The DCCS section lists each Daughterboard Configuration Controller on the daughterboard and the images to load.

FPGAS section The FPGAS section lists the FPGAs on the daughterboard, the images to load and the FPGA programming mode. See *Application note .txt file* on page 5-17.

TAPS section The TAPS section lists information about the JTAG TAP controllers. This is defined in manufacturing, but you can update image files from the Versatile Express DVD.

OSCCLKS section The OSCCLKS section lists the number of programmable clocks and their frequencies. See the application note documentation and the Technical Reference Manual for the daughterboard that is fitted.

Note

OSCCLKS have an operating range of 2MHz-230MHz with a resolution of better than 1%. This means that the actual oscillator frequency is within 1% of the specified frequency. The oscillator stability is 50 ppm. You must not operate clocks outside of the recommended range.

You can change the clock values at run time using the MCC command line interface, see CFG in Table 7-3 on page 7-4, or through writing application code to the SYS-CFG registers, see the *Motherboard Express µATX Technical Reference Manual*.

SCC REGISTERS section

The SCC REGISTERS section lists serial configuration controller registers present on the daughterboard, and the values to load into the registers.

The format is as follows:

```
SCC: 0xAAA (12 bit address) 0xDDDDDDDD (32 bit data)
```

5.4.3 images.txt file

Example 5-7 shows the contents of a typical images.txt file for a daughterboard fitted to the custom motherboard or Site 1 of the Motherboard Express, V2M-P1. This example applies to the ARM Legacy memory map. See the Memory maps section in the Technical Reference Manual for the daughterboard that is fitted.

Example 5-7 Example images.txt file for Site 1

```
TITLE: Versatile Express Images Configuration File

[IMAGES]
TOTALIMAGES: 7                      ;Number of Images (Max : 32)
NOR0UPDATE: AUTO                    ;Image Update:NONE/AUTO/FORCE
NOR0ADDRESS: BOOT                   ;Image Flash Address
NOR0FILE: \SOFTWARE\BM_V307.axf     ;Image File Name

NOR1UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR1ADDRESS: 40000000               ;Image Flash Address
NOR1FILE: \SOFTWARE\u-boot.axf      ;Image File Name

NOR2UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR2ADDRESS: 41000000               ;Image Flash Address
NOR2FILE: \SOFTWARE\kernel.bin      ;Image File Name
NOR2LOAD: 60008000                  ;Image Load Address
NOR2ENTRY: 60008040                 ;Image Entry Point

NOR3UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR3ADDRESS: 42000000               ;Image Flash Address
NOR3FILE: \SOFTWARE\debian-i.bin    ;Image File Name
NOR3LOAD: 420000000                 ;Image Load Address
NOR3ENTRY: 42000000                 ;Image Entry Point

NOR4UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR4ADDRESS: 43F80000               ;Image Flash Address
NOR4NAME: U-Boot-Environment        ;Image Name
NOR4FILE: \SOFTWARE\u-boot-e.bin    ;Image File Name
NOR4LOAD: 0                         ;Image Load Address
NOR4ENTRY: 0                        ;Image Entry Point

NOR5UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR5ADDRESS: 40000000               ;Image Flash Address
NOR5NAME: BOOTSCRIPT                ;Image Name
NOR5FILE: \SOFTWARE\bootscr.txt     ;Image File Name

NOR6UPDATE: AUTO                    ;IMAGE UPDATE:NONE/AUTO/FORCE
NOR6ADDRESS: 44000000               ;Image Flash Address
NOR6NAME: \SOFTWARE\selftest.axf    ;Image Name
```

Note

A Versatile Express system using a custom motherboard might not have any user FLASH memory because the system boots from AXI. In this case, you must set the parameter TOTALIMAGES to 0.

The images.txt file lists user applications in the SOFTWARE directory to load to the NOR flash memory. You can use the following settings:

NORxUPDATE The update mode for this image:

- NONE turns auto update off.
- AUTO enables auto update on power-up, if the file date stamp or filename has changed.
- FORCE forces the image to be updated on every power-up.

NORxADDRESS The address in NOR flash that the image is programmed into.Specify the base address for the flash device 0x40000000, for NOR 1, or 0x44000000, for NOR 2, to enable the image to be programmed at the first free block in the device specified.

If you must program the image at a fixed address, then you must specify it here. If the location is not free, or insufficient space is available, then an error is generated.

The BOOT flag forces the image to be programmed at the location 0x40000000, overwriting any image that is already programmed at that address.

Note

These addresses apply to the ARM Legacy memory map. See the Technical Reference Manual for the daughterboard that is fitted.

NORxNAME The destination filename in NOR flash. This is an optional parameter. If it is not specified, then the name becomes the file name without the .xxx suffix.

NORxFILE The source filename of the NOR flash image in the microSD card, usually the SOFTWARE directory.

NORxLOAD The load address of the image in system memory. This is normally only required for a binary image.

NORxENTRY The entry point for execution of the image in system memory. This is normally only required for a binary image.

5.4.4 elf image

Example 5-8 shows that to program a new elf image into NOR2, you must specify the following:

Example 5-8 Programming a new elf image

```
NOR6UPDATE: AUTONOR6ADDRESS: 0x44000000NOR6FILE: \SOFTWARE\example.axf
```

The load address and entry point are obtained from the elf image. An elf image is defined by the.axf suffix.

5.4.5 binary image

Example 5-9 on page 5-15 shows that to program a new binary image into NOR2, you must specify the following:

Example 5-9 Programming a new binary image

```
NOR6UPDATE: AUTONOR6ADDRESS: 0x44000000NOR6FILE: \SOFTWARE\example.binNOR6LOAD:
0x10000000NOR6ENTRY: 0x10008000
```

Note

Example 5-8 on page 5-14 and Example 5-9 apply to the ARM Legacy memory map. See the Technical Reference Manual for the daughterboard that is fitted.

5.5 Contents of the directory for LogicTile Express boards

The LogicTile Express boards are typically located in Site 2 of the Motherboard Express, V2M-P1. The SITE2 directory contains configuration directories for the supported daughterboard variants. Each subdirectory name matches the HBI code of the daughterboard.

5.5.1 Overview of the contents of the directory for LogicTile Express boards

Each HBI subdirectory contains:

- A single `board.txt` file that contains information that is specific to that daughterboard variant.

  Because the full daughterboard configuration is dependent on the FPGA image, for the specific application note, the `board.txt` file specifies an application note directory that contains additional configuration information.

- Image files for the Daughterboard Configuration Controllers present on the daughterboard. These have `.ebf` extensions.

- One or more application-note subdirectories that contain images to load for a specific function.

  The subdirectory name is in the format `AN`*xxx* where *xxx* identifies a specific application note. The subdirectory contains the following files:

  `a`*xxx*`r`*npm*`.txt` Contains additional configuration information that is specific to this application note.

  `a`*xxx*`r`*npm*`.bit` Is the image to load into the FPGA, and `r`*npm* is a version code.

  `a`*xxx*`r`*npm*`.svs` For PLD images.

5.5.2 board.txt file

Example 5-10 shows the contents of a typical `board.txt` file for a daughterboard fitted to Site 2 of the Motherboard Express, V2M-P1.

Example 5-10 Example `board.txt` file for Site 2

```
BOARD: HBIXXXX
TITLE: FPGA V2F-XXXX
[APPLICATION NOTE]
APPNOTE: ANXXX\aXXXr0p0.txt     ;points to app notes in this HBI directory
                                ;can be edited to point to diff app note

[DCCS]
TOTALDCCS: 1                    ;Total Number of DCCS (Max:8)
M0FILE: dbb_vXXX.ebf            ;DCC0 Filename
M0MODE: MICRO                   ;DCC0 Programming Mode
```

APPLICATION NOTE section

The `APPLICATION NOTE` section references an application note subdirectory. For LogicTile Express boards, there can be multiple application notes that define different images to load to the FPGA. See Example 5-12 on page 5-17.

DCCS section

The DCCS section lists each Daughterboard Configuration Controller on the daughterboard and the images to load.

Example 5-11 shows the daughterboard board.txt file for Site 2 with more than one Daughterboard Configuration Controller.

Example 5-11 Example board.txt file for Site 2 with more than one Daughterboard Configuration Controller

```
BOARD: HBIXXXX
TITLE: FPGA V2F-XXXX

[APPLICATION NOTE]
TOTALALAPP: 2                   ; Total Number of Application notes (max:8)
A0FILE:  ANXXX\XXXr0p0.txt      ; Application note board file for first DCC
A1FILE:  ANXXX\XXXr0p0.txt      ; Application note board file for second DCC

[DCCS]
TOTALDCCS: 2                    ;Total Number of DCCS (Max:8)
M0FILE: dbb_vXXX.ebf            ;Filename for first DCC
M0FILE: MICRO                   ;Programming Mode
M1FILE: dbb_vXXX                ;Filename for second DCC
M1MODE: MICRO                   ;Programming Mode
```

5.5.3 Application note .txt file

Example 5-12 shows the contents of a typical application note a*xxxrnpm*.txt file.

Example 5-12 Typical application note .txt file for Site 2

```
BOARD: HBI0192
TITLE: AN224

[FPGAS]
TOTALFPGAS: 1                   ;Total Number of FPGAS (Max:8)
F0FILE: a224r0p0.bit            ;FPGA0 Filename
F0MODE: FPGA                    ;FPGA0 Programming Mode

[OSCCLKS]
TOTALOSCCLKS: 6                 ;Total Number of OSCCLKS (Max:8)
OSC0: 90.0                      ;OSC0 Frequency in MHz (ACLK)
OSC1: 23.75                     ;OSC1 Frequency in MHz (CLCD)
OSC2: 133.0                     ;OSC2 Frequency in MHz (ZBTRAM)
OSC3: 33.0                      ;OSC3 Frequency in MHz (ExtS ACLK)
OSC4: 50.0                      ;OSC4 Frequency in MHz (SMB)
OSC5: 50.0                      ;OSC5 Frequency in MHz (Not used)

[SCC REGISTERS]
TOTALSCCS: 2                    ;Total Number of SCC registers defined
SCC: 0x000 0x01234567           ;SCC general read/write register address/value
SCC: 0x004 0x89ABCDEF           ;SCC general read/write register address/value
```

FPGAS section

The FPGAS section lists the FPGAs on the daughterboard, the images to load and the FPGA programming mode.

FxMODE selects the programming mode.

- FPGA_STREAM:
  - This is a debug mode that programs the FPGA directly from the microSD card without using NAND memory.
- FPGA:
  - This mode uses the daughterboard NAND memory to configure the FPGA. If the LogicTile contains more than one FPGA, this mode configureS the FPGAs sequentially.
- FPGA_PCM:
  - This is Parallel Configuration Mode. It enables parallel configuration of FPGAs from daughterboard NAND memory.

Note

FPGA is the default mode on Versatile Express LogicTiles.

FPGA_PCM is the fastest configuration mode. ARM recommends that you use this mode to configure the FPGAs on the daughterboard.

OSCCLKS section

The OSCCLKS section lists the number of programmable clocks and their frequencies. See the application note documentation and the Technical Reference Manual for the daughterboard that is fitted.

Note

OSCCLKS have an operating range of 2MHz-230MHz with a resolution of better than 1%. This means that the actual oscillator frequency is within 1% of the specified frequency. The oscillator stability is 50 ppm.You must not operate clocks outside of the recommended range.

You can change the clock values at run time using the MCC command line interface, see CFG in Table 7-3 on page 7-4, or by writing application code to the SYS-CFG registers, see the *Motherboard Express µATX Technical Reference Manual*.

Note

This does not apply to the custom motherboard.

SCC REGISTERS section

The SCC REGISTERS section lists serial configuration controller registers present on the daughterboard.

The format is SCC: 0xAAA, 12-bit address, 0xDDDDDDDD, 32-bit data.

5.6 Contents of the SOFTWARE directory

The SOFTWARE directory contains applications that you can load into the NOR flash memory.

You can create new applications and load them into the NOR flash on the motherboard. Application images are typically boot images or demo programs and have an .axf extension. Typical applications in this directory are:

bm_vxxx.axf	The boot monitor application.
debian-i.bin	A Linux binary image.
kernel.bin	A Linux kernel binary image.
u-boot.axf	A Linux boot loader.

Chapter 6
Updating Motherboard Firmware

This chapter describes how to update the motherboard firmware. It contains the following section:

6.1 Updating motherboard firmware

This following procedure explains how to update the motherboard firmware:

1. Apply power to the motherboard.
2. Power-up the motherboard using the ON/OFF/Soft Reset push button, or type `USB_ON` in the serial terminal.
3. Copy the updated firmware, `mbb_vxxx.ebf`, to the `MB/HBI0xxxx/` USBMSD directory. See Figure 5-1 on page 5-3.
4. Edit the `MBBIOS: mbb_v2xx.ebf` line in the motherboard `board.txt` file to match the new firmware version. See Example 5-4 on page 5-9.
5. Press the Hardware Reset button, to reset the system.
6. Press the ON/OFF/Soft Reset button, to power-up and load the new firmware.

See *Push-button resets of the Versatile Express development system* on page 4-2.

Note

- Motherboard Express, V2M-P1:
  - The Hardware Reset button is the BLACK button.
  - The ON/OFF/Soft Reset button is the RED button.
- V2M-CP1 Programmer Module:
  - The Hardware Reset button is button S2.
  - The ON/OFF/Soft Reset button is button S1.

See the *Motherboard Express µATX Technical Reference Manual* or the *Design Assist Programmer Module (V2M-CP1) Technical Reference Manual* for the location of these switches.

Chapter 7
MCC Command-Line Interface

This chapter describes the *Motherboard Configuration Controller* (MCC) command-line interface of the Versatile Express development system. It contains the following sections:

7.1 MCC main menu

This section describes the command-line interface to the MCC.

To enter MCC commands, use a terminal emulator connected to either:

- UART0 if the system is in standby mode.
- UART1 if the system in run mode.

Note

The V2M-CP1 Programmer Module does not support the MCC main menu using UART1.

The terminal emulator settings must be:

- 38.4kBaud.
- 8N1, that is, 8 data bits, no parity, 1 stop bit.
- No hardware or software flow control.

Table 7-1 shows the MMC main menu commands.

Table 7-1 MCC main command menu

Command	Description
`CAP file_name [/A]`	Capture serial data to the file *file_name*. Use the `/A` option to append data to an existing file.
`FILL file_name [nnnn]`	Create a file *file_name* filled with text. `nnnn` specifies the number of lines to create. The default value is 1000.
`TYPE file_name`	Display the content of text file *file_name*.
`DEL file_name`	Delete file `file_name`.
`DIR [mask]`	Displays a list of files in the directory.
`FORMAT [label]`	Format Flash memory card and optionally give it a label.
`USB_ON`	Enable configuration USB-B port.
`USB_OFF`	Disable configuration USB-B port.
`SHUTDOWN`	Shutdown ATX PSU, but leave MCC running. It returns to Standby mode.
`REBOOT`	Power cycle the system and reboot.
`RESET`	Reset daughterboards using the **CB_nRST** reset signal.
`FLASH [0/1]`	Change to the flash menu. Use the 0 or 1 option to select the flash device to use.
`DEBUG`	Change to the Debug submenu. **Note** The Debug submenu is only available in run mode.
`EEPROM`	Change to the eeprom menu.
`HELP` or ?	Display this help text.

7.2 Flash menu

Enter FLASH at the main menu to switch to the flash submenu and directly manage the flash images stored in NOR flash. The flash device used depends on the parameter.

Note

The Flash submenu is only available in standby mode.

A Versatile Express system using a custom motherboard might not have any user flash memory because the system boots from AXI. In this case, the system does not support the flash commands.

Table 7-2 shows the flash commands.

Table 7-2 Flash commands

Command	Description
AREAS	List flash areas
IMAGES	List flash images
INFO *image_name*	List information for *image_name*
READ *image_name* [*file_name*]	Read *image_name* to *file_name*
EXIT or QUIT	Return to main menu
HELP or ?	Display help information

7.3 Debug menu

Enter DEBUG at the main menu to switch to the Debug submenu.

Note

The Debug submenu is only available in run mode.

Table 7-3 shows the debug commands.

Table 7-3 Debug commands

Command	Description
DATE	Displays current date.
TIME	Displays current time.
DELAY	Tests delay timers.
BIOSPRG *filename*	Program MBbios backup flash from file *filename*.
CFG R *para site position DCC device*	Read SPI configuration command. *para* is the parameter to read and is one of: • OSC • V • I • TEMP • SCC • PWR • MUX • DVI *site* selects the motherboard or one of the daughterboards: • MB • DB1 • DB2 *position* is the position of the board in the stack. *DCC* is the Daughterboard Configuration Controller performing the task. *device* is the identifier for the device on the board. For example, the command: CFG R OSC MB 0 0 1 displays: clock read = 23.750 MHz
CFG W *para site position DCC device data*	Write SPI configuration command. *data* is the data value to write to the specified parameter. For example, the command: CFG W OSC MB 0 0 1 24.0 sets MB OSC 1 to 24.0 MHz. Other parameters are the same as for CFG R.

Table 7-3 Debug commands (continued)

Command	Description
`CFG W DVI display`	Write DVI configuration command. *display* selects the graphic display standard and is one of: • VGA • SVGA • XGA • SXGA • UXGA **Note** The V2M-CP1 Programmer Module does not support this command.
`CFG W MUX site`	Write MUX configuration command.
`PCICFG R address B/W`	Read from a PCI-Express register. *address* is the register address. B or W selects Byte or Word format-default double word. **Note** The V2M-CP1 Programmer Module does not support this command.
`PCICFG W address data B/W`	Write to a PCI-Express register. **Note** The V2M-CP1 Programmer Module does not support this command.
`PCIEE R address`	Read from PCI-Express EEPROM. **Note** The V2M-CP1 Programmer Module does not support this command.
`PCIEE W address data`	Write to PCI-Express EEPROM. **Note** The V2M-CP1 Programmer Module does not support this command.
`PCIFILE R filename`	Read PCI-Express EEPROM and write to file. **Note** The V2M-CP1 Programmer Module does not support this command.
`PCIFILE W filename`	Write to PCI-Express EEPROM from file. **Note** The V2M-CP1 Programmer Module does not support this command.
`EXIT` or `QUIT`	Returns to main menu.
`HELP` or `?`	Displays help information.

7.4 EEPROM menu

Enter `EEPROM` at the main menu to switch to the EEPROM submenu. The contents of the motherboard and daughterboard EEPROMs identify the specific board variant and might contain data to load to the other devices on the board.

Caution

You must not modify the EEPROM values that are programmed in production.

Table 7-4 shows the EEPROM commands.

Table 7-4 EEPROM commands

Command	Description
`CONFIG [0/1/2][position] filename`	Writes configuration file to EEPROM
`READCF [0/1/2][position]`	Read configuration EEPROM
`ERASECON`[0/1/2][position]	Erase configuration section of EEPROM
`ERASEDEV[0/1/2]`[position]	Erase device section of EEPROM
`WRITEBOARD[1/2][position]`	Write board configuration file to EEPROM
`READBOARD [1/2]`[position]	Read board configuration file from EEPROM
`READRANGE [0/1/2][position][start][end]`	Read EEPROM between start and end
`ERASERANGE[0/1/2][position][start][end]`	Erase EEPROM between start and end
`READIMAGES`	Read images stored in Motherboard EEPROM
`ERASEIMAGES`	Erase images stored in Motherboard EEPROM
`ERASEIMAGE[image_id]`	Erase image stored in Motherboard EEPROM
`EXIT` or `QUIT`	Return to main menu
`HELP` or ?	Display this help

Appendix A
Revisions

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

Table A-1 Issue A

Change	Location	Affects
No changes, first release	-	-

Table A-2 Differences between issue A and issue B

Change	Location	Affects
Clarified information on FPGA programming modes	*Application note .txt file* on page 5-17	All versions