VadaTech AMC502

Software User Manual

January 2019

Version 5.0

Copyright

Notice

While reasonable efforts have been made to assure the accuracy of this document, VadaTech, Inc. assumes no liability resulting from any omissions in this document or from the use of the information obtained herein. VadaTech reserves the right to revise this document and to make changes periodically and the content hereof without obligation of VadaTech to notify any person of such revision or changes.

Electronic versions of this material may be read online, downloaded for personal use, or referenced in another document as a URL to the VadaTech Incorporated Web site. The text itself may not be published commercially in print or electronic form, edited, translated, or otherwise altered without the permission of VadaTech, Inc.

It is possible that this publication may contain reference to or information about VadaTech products (machines and programs), programming, or services that are not available in your country. Such references or information must not be construed to mean that VadaTech intends to announce such products, programming, or services in your country.

Trademarks

Revision History

Doc Revision	Description of Change	Revision Date
1.0	Document created	March 2015
2.0	Updated document to match latest partition mapping	May 2016
3.0	Added Slot Based IP section	September 2017
4.0	Added Application development and BSP sources sections	January 2019
5.0	Updated to the new template	January 2019

Table of Contents

Tables

1 Document Overview

1.1 Purpose

This document describes the AMC502 board software including the Linux BSP for the iMX6 Quad Core CPU. This document also provides details on how to program AMC502 FPGA image to QSPI flash.

1.2 Applicable Products

- AMC502

1.3 Document References

1.3.1 Specification

The AMC502 is compliant with the following specification:

- *PICMG® AMC.0 AdvancedMC Mezzanine Module (http://www.picmg.org)*
- *PICMG® AMC.1 AdvancedMC PCI Express and AS (http://www.picmg.org)*
- *PICMG® AMC.2 AdvancedMC Ethernet (http://www.picmg.org)*
- *PICMG® AMC.4 AdvancedMC Serial RapidIO (http://www.picmg.org)*
- *ANSI/VITA 57.1 FPGA Mezzanine Card (FMC) Standard (http://www.vita.com)*

1.3.2 Related Documents

The following documents provide information related to AMC502:

- *AMC502 Datasheet*
- *VadaTech AMC502 Hardware Reference Manual*
- *Xilinx Kintex-7 Datasheets and User's Guides*

1.4 Acronyms Used in this Document

Table 1: Acronyms

Acronym	Description
AMC	Advanced Mezzanine Card
BAR	Base Address Register
BIST	Built-In Self Test
BPI	Byte Peripheral Interface
BSP	Board Support Package
C2M	Carrier-to-Mezzanine (signal)
CGND	Chassis Ground
CLK	Clock
CPU	Central Processing Unit

Acronym	Description
DAC	Digital to Analog Converter
DDR3	Dual Data Rate 3 SDRAM
DIP	Dual In-line Package
DMA	Direct Memory Access
DMUX	De-multiplexer
DR	Data Ready
FMC	FPGA Mezzanine Card
FMC10G	Refers to VadaTech FMC106/107
FMC1G	Refers to VadaTech FMC102/103/104/105
FPGA	Field Programmable Gate Array
FRU	Field Replaceable Unit
GbE	Gigabit Ethernet
GND	Signal Ground
GTX	Virtex-6 Gigabit Transceiver
HPC	High Pin Count (FMC connector)
ioctl	Input/Output/Control
IP	Intellectual Property
IPMI	Intelligent Platform Management Interface
JSM	JTAG Switch Module
JTAG	Joint Test Action Group
LED	Light Emitting Diode
LPC	Low Pin Count (FMC connector)
LVCMOS	Low-Voltage Complementary Metal Oxide Semiconductor
LVDS	Low Voltage Differential Signaling
M2C	Mezzanine-to-Carrier (signal)
MAC	Media Access Controller
MB	Megabyte (2^20 bytes)
MIG	Memory Interface Generator
M-LVDS	Multi-point Low Voltage Differential Signaling
mmap	Memory Map
MMC	Module Management Controller (IPMI controller of AMC)
MMIO	Memory Mapped Input/Output
MUX	Multiplexer
n.c.	No connection
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect Express
PHY	Physical Layer Device
PICMG	PCI Industrial Computer Manufacturer's Group
PIO	Programmed Input/Output
PLL	Phase Locked Loop
SDRAM	Synchronous Dynamic Random Access Memory
SERDES	Serializer/Deserializer
SGMII	Serial Gigabit Media Independent Interface
SOL	Serial-Over-LAN
SRIO	Serial RapidIO
SSIF	SMBus System Interface
TCLK	Telephony Clock
USB	Universal Serial Bus

Acronym	Description
VADJ	Adjustable Voltage (power rail)
VIO	I/O Voltage (power rail)
VREF	Reference Voltage (power rail)
XAUI	Ten Gigabit Attachment Unit Interface

1.5 Conventions Used

The following conventions are used in this document:

⚡ ***WARNING - Important information, when ignored can cause harm. serious injury or death to the User is described next to this symbol***

⚠ ***CAUTION - Important information, when ignored can cause serious damage to the device is described using this symbol***

ℹ ***NOTE - Important information useful to the reader is described next to this symbol***

`Command` ***Any CLI commands are described with this font style***

2 Linux BSP Software Overview

The AMC502 comes pre-configured with a boot loader (U-Boot) and Linux kernel/file system in flash memory.

The CPU RS-232 port can be used to gain access to the serial console. The serial port setup is 115200-8-N-1-NOFLOW.

2.1 Linux Boot Sequence

When the board boots up the sequence is as follows:

1. U-Boot loader starts up and initializes SDRAM, Flash, GbE ports
2. U-Boot script automatically loads/starts the 'active' Linux kernel from flash
3. Linux initializes devices and mounts 'active' root file system from flash
4. File system init scripts run
5. The Linux login prompt is presented.

2.2 Memory Map and Linux Device Nodes

The 8GB NAND Flash (u65) memory map and corresponding Linux device files are as follows:

Table 2: NAND Flash Memory Map

Address	Size	Part	Description	Linux Device
0x000000000000	128 MB	Nand Flash (U65)	Reserved	/dev/mtd0
0x000008000000	8 MB	Nand Flash (U65)	Reserved	/dev/mtd1
0x000008800000	64 MB	Nand Flash (U65)	Linux Kernel Image A	/dev/mtd2
0x00000c800000	2500 MB	Nand Flash (U65)	Root File System A	/dev/mtd3
0x0000a8c00000	8 MB	Nand Flash (U65)	Reserved	/dev/mtd4
0x0000a9400000	64 MB	Nand Flash (U65)	Linux Kernel Image B	/dev/mtd5
0x0000ad400000	2.500 MB	Nand Flash (U65)	Root File System A	/dev/mtd6
0x000149800000	2920 MB	Nand Flash (U65)	Reserved	/dev/mtd7

The Boot Flash (U63) memory map and corresponding Linux device files are as follows:

Table 3: Boot Flash Memory Map

Address	Size	Part	Description	Linux Device
0x000000000000	1920 KB	Boot Flash (U63)	U-boot	/dev/mtd8
0x0000001e0000	64 KB	Boot Flash (U63)	U-boot Environment A	/dev/mtd9
0x0000001f0000	64 KB	Boot Flash (U63)	U-boot Environment B	/dev/mtd10
0x000000200000	2048 KB	Boot Flash (U63)	Etc file system	/dev/mtd11

The FPGA QSPI Flash (U24) memory map and corresponding Linux device files are as follows:

Table 4: FPGA QSPI Memory Map

Address	Size	Part	Description	Linux Device
0x000000000000	128 MB	Nand Flash (U65)	Reserved	/dev/mtd0

2.3 Default IP Address

eth0 (AMC Port 0): `192.168.1.252`

These IP addresses can be used to `ssh` into the board to get a command prompt or to use `scp` for transferring files. To change them edit the `/etc/rc.d/rc.conf` file.

2.4 Slot based IP Address

Slot based IP automatically generates the unique IP for an Ethernet interface based on the

Slot number of the AMC in the chassis. This feature calculates the fourth octet of the IP by adding the slot number to the base value of the fourth octet and the rest of the first three octets are applied as is.

Slot based IP is a result of cooperation of different components of AMC design which are described below:

1. Module Management Controller, which accepts OEM IPMI commands and provides the slot info for the AMC.
2. Hardware support, which allows the IPMI interface between MMC and Payload CPU.
3. The slot based IP script at the OS level which calculates the fourth octet and the resultant IP address, which is applied to the corresponding Ethernet interface.

The slot based IP would be applicable only if the fourth octet is '0' (zero). The octet can be made zero by modifying the `IPADDR0` variable in the `/etc/rc.d/rc.conf` file.

For example if the IPADDR0 in the `/etc/rc.d/rc.conf` is `192.168.1.252`, it's in slot 2 and the base value is set to 200, then changing the IP in the interface file to `192.168.1.0`, would result in the `eth0` having an address of `192.168.1.202`.

User can set the base value for the fourth octet by assigning the value to the variable `IP_OCT4_BASE` in `/etc/rc.d/rc.conf` file.

2.5 Linux BSP Field Upgrade

The BSP binary release includes a field upgrader package which can be used to upgrade the board from the Linux command prompt. This upgrader uses a ping-pong partition style upgrade using the 'A' and 'B' partitions seen in the memory map above. This ensures a consistent upgrade with minimal risk should a failure occur. But it does mean that any customizations that you've applied to the file system will need to be re-applied after the upgrade except /etc file system which is standalone partition and all files changed in this location will be preserved during upgrade.

To use the upgrade package, transfer the upgrade tar file to the **/upgrade** directory using **scp**. Then untar the package using the following command:

```
tar xvzf amc502-bsp-upgrade-releaseX.tgz
```

Then to invoke the upgrade type:

```
./upgrade
```

To verify the BSP version after rebooting the board, type:

```
bsp-version
```

3 Programming BSP via U-Boot

Programming the board via U-Boot is recommended only when some other factor is preventing upgrade via Linux.

In order to program via U-Boot, power up the board and interrupt the auto-boot sequence by pressing ENTER. This will drop you to a U-Boot prompt. Then you will need to setup the U-Boot networking environment.

To set up the U-Boot networking, use the following commands from the U-Boot prompt:

```
=> setenv ipaddr <board ip address>

=> setenv netmask <board netmask>

=> setenv gatewayip <optional gateway router>

=> setenv serverip <tftp server ip>

=> setenv ethact XXX      (XXX is Gem.e000b000, Gem.e000c000)
```

Next, ping through the selected port first to the gateway router (if you have one) and next to the TFTP server to ensure that you have network connectivity.

```
=> ping <gateway router>

=> ping <tftp server>
```

Then proceed on to placing the upgrade images onto your TFTP server and using U-Boot to download them to the board and program the partitions. Use extreme care when entering commands and do not progress from one step to the next if any failure occurs.

WARNING: If the U-Boot binary partition of the flash becomes erased or corrupted and the board is reset/power-cycled before reprogramming a valid image.

3.1 Programming the U-Boot Image

- `run up_uboot`

This will download `amc502/u-boot.imx` image from TFTP server and reprogram it.

NOTE: If the board does not already have a valid U-Boot image programmed on it then the board will not boot.

3.2 Programming the Linux Kernel Image

- `run up_kernela (for upgrading Kernel-A)`
- `run up_kernelb (for upgrading Kernel-B)`

This will download `amc502/uImage` image from TFTP server and reprogram it.

3.3 Programming the Root File System

- `run up_roota (for upgrading Rootfs-A)`
- `run up_rootb (for upgrading Rootfs-B)`

This will download `amc502/rootfs.ubifs.img` image from TFTP server and reprogram it.

3.4 Programming the Etc File System

- `run up_etc`

This will download `mc502/ etcfs.jffs2` image from TFTP server and reprogram it.

3.5 Completing an Upgrade via U-Boot

After the necessary partitions are upgraded please issue the `reset` command from U-Boot to ensure that the upgrades take effect. The `bsp-version` command can be used within Linux to check the BSP version and confirm the success of the upgrade.

4 Programming FPGA Image from iMX6 CPU

In order to program FPGA image from iMX6 CPU download FPGA image to /upgrade directory and do the following:

- `[root@amc502 upgrade]# fpga_upgrade amc502_fpga.bin`

```
NOTE: This upgrade script DOES upgrade/downgrade
the FPGA Configuration in QSPI flash......

Please ensure that you have specified the proper image.
Do you wish to continue with the upgrade (y/n)? y

s25fl512s spi3.0: s25fl512s (65536 Kbytes)
Creating 1 MTD partitions on "s25fl512s":
0x000000000000-0x000004000000 : "s25fl512s"
Writing FPGA image to /dev/mtd12...
Erasing blocks: 12/12 (100%)
Writing data: 2987k/0k (100%))
Verifying data: 2987k/0k (100%))
FPGA Configuration ... SUCCESS
```

The fpga_upgrade script will attach QSPI flash to iMX6 CPU, program provided image into it and trigger FPGA to configure itself from the QSPI Flash using the new image. Refer to AMC502 Hardware reference manual for more information about FPGA QSPI Flash Upgrade Sequence.

5 BSP Sources

The sources package is named as amc502_bsp_source_releaseX.tgz where ‘X’ is the release number.

The source tree includes the following top level directories:

Table 5: Top Level Directories

Directory	Contents
auto_build.sh	Script that performs a complete build of the BSP
kernel	This directory contains the customized Linux kernel for the AMC502 board. The amc502_make script found here can be used for building it. The make_release packager picks up the kernel image (Image) from the kernel/arch/arm/boot/uImage location within this tree, expecting that it has already been built (binary not included in source package).
make_release	This script is used to package up all of the binaries into the production and field upgrade packages. This script can be used by the customer to package their own customized version of the BSP for their own internal use if desired. In this way a simple field upgrade to convert the stock VadaTech reference design to the customer’s application-specific design is possible. This script assumes that binaries are available for everything needed and that any changes to the root file system binaries have already been merged in. It does not do any building of source code.
u-boot	This directory contains the U-Boot bootloader sources customized for the AMC502 board. The amc502_make script can be used to build it. The make_release packager picks up the U-Boot image from this location.
ltib	Root file system builder
tools	This directory contains various tools used to create the production, upgrade packages and tools that were not created during rootfs build.
fpga	FPAG image and tool to get u-boot readable image
iomux	NXP iMX6 IOMUX source files

6 Application Development

In order to build applications for AMC502 you need to prepare host system where you can build the applications from sources. You can setup your building environment in the hosts mentioned below

- Redhat: 7.3, 8.0, 9.0, Enterprise release 4
- Fedora Core: 1, 2, 3, 4, 5, 8
- Debian: 3.1r0 (stable), unstable, 4.0
- Suse: 8.2, 9.1, 9.2, 10.0, 10.2, 10.3
- Ubuntu 6.10, 7.04, 7.10, 8.04

To develop the applications follow these steps:

1. Download the `amc502_bsp_source_releaseX.tgz` package from VadaTech into your build host.

2. Extract it and change the directory to amc502:

```
tar xvf amc502_bsp_source_releaseX.tgz
cd amc502
```

3. You have to install some libraries needed to build the project. If you already have them installed then skip this step

```
sudo yum install lzo
sudo yum install lzo-devel
sudo yum install tcl
```

NOTE: Use apt-get to install the libraries if you are using Debian based host.

4. At first you need to build the LTIB which will download the cross compiler to build the rootfs and later it will be used to compile kernel, u-boot and the other applications for AMC502. Run the ./ltib and follow steps during it's execution:

```
cd ./ltib
./ltib
```

NOTE: LTIB may stop the execution and provide you to add some privilege credentials into sudoers file like this

```
"… To configure this, as root using the command "/usr/sbin/visudo",and add the
following line in the User privilege section:
username ALL = NOPASSWD: /bin/rpm, /opt/freescale/ltib/usr/bin/rpm …"
```

Just add this line into the `sudoers` file and rerun `./ltib` command

5. Move back to the project's root folder and run

```
./autobuild.sh -v
```

You will be notified about build completion at the end.

6.1 Building U-BOOT

To build U-Boot:

- Go to the VadaTech AMC502 BSP's **`u-boot`** directory
- Run: **`./amc502_make`**

The resulting U-Boot binary will be called u-boot.bin in the u-boot directory. Refer to upgrade section for instructions on how to flash this image to the board.

6.2 Building kernel

To build the Linux Kernel:

- Go to the VadaTech AMC502 BSP's **`kernel`** directory
- Run: **`./amc502_make`**

6.3 Packaging a release

The previous sections described how to build individual components from the BSP. Doing individual component upgrades is fine for most development activity, but you may desire to also package the components into a complete release in the same manner as the field upgrade packages are provided by VadaTech. The release package is easier to upgrade the board with since you can use the Linux command line upgrade script. This method also ensures that all the binaries on the board match to a given release configuration and avoids potential human errors.

To package a release:

1. Edit tools**`/bsp-version/bsp-version.dat`** file and add a suffix of your choosing to help identify the filesystem version as being different from VadaTech's release.
2. Make sure that U-Boot, Linux kernel are also built with any customizations of your requirement.
3. From the BSP root directory run **`./make_release release`***`XXSS`* where *`XX`* is the VadaTech BSP number you are using as a baseline and *`SS`* is your own suffix. This will make the full release packages.

Look in the `/tmp/amc502_bsp_release`***`XXSS`*** directory for the upgrade package which you can transfer to the board and run the upgrade as described earlier in the manual.

Contact VadaTech

Technical Support

If you have purchased the VadaTech product through our distributor network, contact your distributor for any technical assistance. If you require further technical support, you can contact VadaTech technical support team from VadaTech Customer Support site.

Locations

VadaTech Corporate Office	European Sales Office	Asia Pacific Sales Office
198 N, Gibson Road Henderson, NV 89014 USA Email: info@vadatech.com Telephone: +1 702 896 3337 Fax: +1 702 896 0332	VadaTech House, Bulls Copse Road, Southampton, SO40 9LR UK Email: info@vadatech.com Telephone: +44 2380 016403	7th Floor, No 2, Wenhu Street, Neihu District, Taipei 114, Taiwan Email: info@vadatech.com Telephone: +886 2 2627 7655 Fax: +886 2 2627 7792