OpenCPI
Matchstiq-Z1 Getting Started Guide

OpenCPI Release: v2.3.4

Revision History

Revision	Description of Change	Date
v1.1	Initial Release	3/2017
v1.2	Updated for OpenCPI Release 1.2	8/2017
v1.3	Updated for OpenCPI Release 1.3	2/2018
v1.4	Update descriptions and paths	9/2018
v1.5	Update for OpenCPI Release 1.5	4/2019
v1.6	Refer to installation doc when possible	1/2020

Table of Contents

1 References

The reference(s) in Table 1 can be used as an overview of OpenCPI and may prove useful. The installation guide is required since many of the steps mentioned here are defined there, especially in the section: Enabling OpenCPI Development for Embedded Systems. This document provides details for this system that can be applied to procedures defined there. It is best to use both documents at the same time. This document assumes a basic understanding of the Linux command line (or "shell") environment.

Title	Link
OpenCPI User Guide	`OpenCPI_User_Guide.pdf`
Acronyms and Definitions	`Acronyms_and_Definitions.pdf`
OpenCPI Installation Guide	`OpenCPI_Installation_Guide.pdf`

Table 1: References

2 Overview

This document provides steps for configuring a factory provided Epiq Solutions Matchstiq-Z1 SDR with the OpenCPI run-time environment for executing applications. **Note: Only the Z1 version of the Epiq Solutions Matchstiq product line is supported by OpenCPI.**

3 Prerequisites

It is assumed that the tasks defined in the "Enabling OpenCPI Development for Embedded Systems" section of the *OpenCPI Installation Guide* have been successfully performed. As mentioned there, support for the Matchstiq Z1 system is based on the *ocpi.assets* built-in project, using the *matchstiq_z1* OpenCPI HDL (FPGA) platform and the *xilinx13_3* OpenCPI RCC (software) platform. The software platforms are supported by the *ocpi.core* built-in project.

3.1 Vendor Software Setup

Also indicated in the installation guide, the tools used for software cross compilation are from the Xilinx Vivado 2013.4 SDK, and the tools used for the FPGA are any recent version of the Xilinx Vivado Webpack tools. The Linux Kernel is based on the Xilinx Linux kernel tagged *xilinx-v14.7*. The installation of these tools is described in the installation guide.

3.2 Building Required Projects

The standard built-in OpenCPI projects, as well as the OSP for this system, are built using the above instructions in the installation guide. This results in a single executable FPGA test application ready to run: *bias*, based on the *testbias* HDL assembly, both in the *assets* built-in project.

3.3 Hardware Setup

- **Epiq Solutions Matchstiq-Z1 SDR Kit**

  It is expected that this SDR kit includes a power supply, two SMA/SMB adapters, micro-USB to USB cable, micro-SD card installed internally (expected).

  A micro-USB connector on the back of the Matchstiq-Z1 provides access to the serial connection. To expose this micro-USB connector, the two screws in the back plate must be removed. Historically, this connector's attachment to the PCB has been extremely fragile, **so be careful when inserting/removing the mating cable**.

- **Micro-USB to Ethernet adapter**. To allow network access when plugged into the front panel micro-USB port. The OpenCPI software platform for the Matchstiq-Z1 is configured for DHCP. An Ethernet connection is required for developing OpenCPI in Network mode.

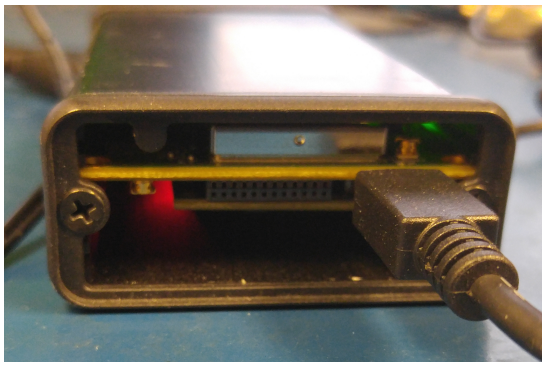

Figure 1: Connected Back Panel

On the front panel of the Matchstiq-Z1, there are three labeled SMB (50 Ohm) connectors: "RX" (receive), "TX" (transmit), and "GPS". From the factory, the Matchstiq-Z1 is provided with two SMB to SMA adapters. Due to the RF performance to the transceiver device, any RF COAX cables should be rated up to at least 3GHz.

Figure 2: Connected Front Panel

- **Access to a network which supports DHCP. (Network Mode)**
- **Micro-SD card, 4GB+ (OPTIONAL, as it is possible to use internally installed card)**
- **Micro-SD card reader**

4 SD Card Setup

The Matchstiq-Z1 SDR is equipped with two SD card slots: one internal and one accessible via the front panel. It is expected that the SDRs are shipped from Epiq Solutions with an SD card installed in the internal slot that is loaded with their embedded environment. A feature of this SDR is that when an SD card is installed in the front panel SD slot, the SDR will automatically choose to operate from this SD card rather than the internal SD card. Therefore, a user can easily switch the SDR between operating in the Epiq Solutions or OpenCPI environment.

The Matchstiq-Z1's factory SD card has a non-default formatting and content, which *must* be maintained for proper operation. This guide assumes that the internal (factory) SD card is being use for OpenCPI and will be reinstalled in the front panel SD card slot. If the user desires the use of a new SD card, the user must ensure that it is initially imaged from the factory provided SD card, as there is a unique partition containing required content from the OEM.

The installation guide provides the procedure for creating a new SD card for OpenCPI and customizing some of files for your particular configuration. The usual way is to make a raw copy of the manufacturer supplied card to a new card, preserving formatting and content, and then removing most original files and copying files from OpenCPI.

To prepare for OpenCPI provided content to be placed onto the SD card, remove all factory files and directories from the ATLAS partition. The Matchstiq_z1 boots from the "ATLAS" partition of the SD Card so that is where the OpenCPI contents should be copied.

Any files/directories copied to the "ATLAS" partition will appear at /mnt/card on the Matchstiq-Z1.

All the files in this partition can be ignored. If space for files is required for your application, they can be deleted.

5 Hardware Setup

5.1 Establish a Serial Connection

The installation guide provides the procedure establishing a serial console. On these systems the console serial port operates at 115200 baud. The cable used is a micro-USB to USB-A cable to connect its console micro-USB port to the development host. Remember the micro-USB socket on the Matchstiq Z1 is **delicate!**.

5.2 Update U-boot Variables

1. Remove power from the Matchstiq-Z1 unit.
2. Insert the SD card into the front panel SD card slot.
3. Connect a terminal to the rear micro-USB connector of the Matchstiq-Z1 with a baud rate of 115200.
4. Apply power to the Matchstiq-Z1 with the terminal still connected and stop the boot process by hitting any key to enter the U-Boot terminal.
5. Run the following commands to setup the environment variables:
   - `setenv bootcmd 'ivmmc; run ocpiboot'`
   - `setenv ocpiboot 'setenv bootargs console=ttyPS0,115200n8 root=/dev/ram rw earlyprintk; setenv fdt_high ffffffff; setenv initrd_high 0x1000000; fatload mmc ${iv_mmc} ${dtbaddr} ${dtbfile}; fatload mmc ${iv_mmc} ${loadaddr} ${bootfile}; fatload mmc ${iv_mmc} 0x2000000 uramdisk.image.gz; bootm ${loadaddr} 0x2000000 ${dtbaddr}'`
     *Note: This should be a one-line command. Make sure there are no newlines.
   - `saveenv`
6. These U-Boot environment variables are now saved to the second partition of the SD card

Verify that the changes are correct by running the command "`env p`" and comparing to:

```
baudrate=115200
bootcmd=ivmmc;run ocpiboot
bootdelay=3
bootfile=uImage
defargs=setenv bootargs console=ttyPS0,115200n8 mem=240M iv_mb=${iv_mb} iv_io=${iv_io}
iv_bp=${iv_bp} iv_mmc=${iv_mmc} ${otherargs}
dtbaddr=0x02a00000
dtbfile=iveia-atlas-i-z7e.dtb
iv_io=205-00034-00-A0,,Atlas-II_GF_Carrier
iv_io_default=205-00034-00-A0,,Atlas-II_GF_Carrier
iv_io_ord=00034
iv_mb=205-00049-00-B1,A2WT9,Atlas-I-Z7e
iv_mb_ord=00049
iv_mmc=0
loadaddr=0x03000000
mmcdtload=fatload mmc ${iv_mmc} ${dtbaddr} ${dtbfile};fdt addr ${dtbaddr};fdt set
/chosen bootargs "${bootargs}";fdt ivclean ${iv_mb_ord}
mmcxload=axi_reset 1; fatload mmc ${iv_mmc} ${loadaddr} ${xloadfile};xload ${loadaddr}
${filesize}; axi_reset 0;
ocpiboot=setenv bootargs console=ttyPS0,115200n8 mem=240M root=/dev/ram rw earlyprintk;
setenv fdt_high ffffffff;  setenv initrd_high 0x1000000; fatload mmc ${iv_mmc} ${dtbaddr}
${dtbfile}; fatload mmc ${iv_mmc} ${loadaddr} ${bootfile};  fatload mmc ${iv_mmc} 0x2000000
uramdisk.image.gz; bootm ${loadaddr} 0x2000000 ${dtbaddr}
sdboot=run mmcxload;run defargs;fatload mmc ${iv_mmc} ${loadaddr} ${bootfile};run
mmcdtload;setenv fdt_high ffffffff;bootm ${loadaddr} - ${dtbaddr}
stderr=serial
```

```
stdin=serial
stdout=serial
xloadfile=xilinx.bit

Environment size: 1283/131068 bytes
```

6 Configuring the runtime environment on the system

The installation guide provides the procedure for setting up and verifying the runtime environment. This system is initially set with "**root**" for user name and password. After a successful boot to PetaLinux, login to the system, using "**root**" for user name and password.

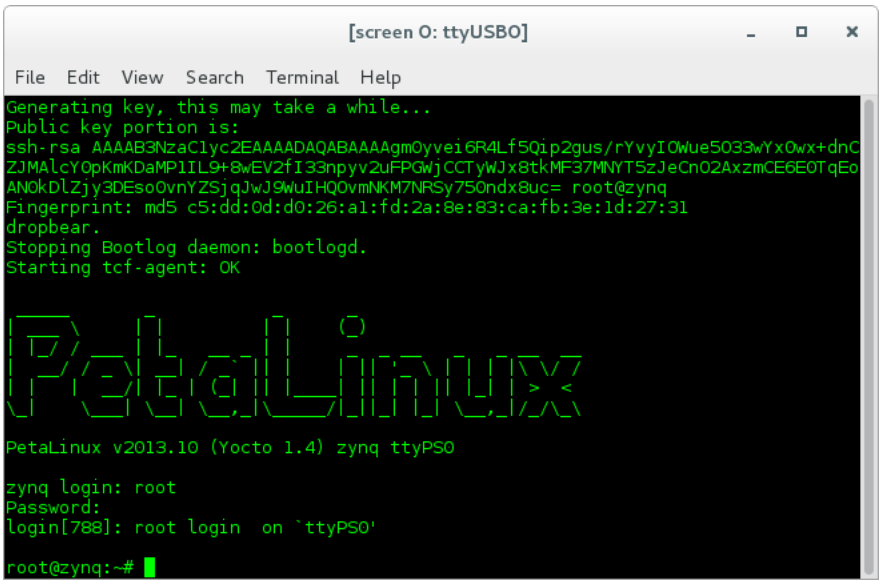

Figure 3: Successful Boot to PetaLinux

7 Run an Application

See the installation guide for running a small test application.

Appendices

A Intermittent Errors

Some tests have had "Segmentation Faults" or "Alignment Errors" in certain scenarios on the Z1. This seems to happen when both USB ports are used to simultaneously transmit a large amount of data, *e.g.* high log-level output to a USB serial console as well as NFS-mounted output files over a USB-to-Ethernet adapter. The default test setup avoids triggering this by limiting output that is fed to the user, but users should be aware of this issue if non-default test scenarios are attempted. If `ssh` is used to have all data routed through the USB-to-Ethernet adapter, this failure mode is avoided.

B Using ISE instead of Vivado with the Matchstiq-Z1

It is recommended that you use the default toolset (Xilinx Vivado) to build Matchstiq-Z1 bitstreams with OpenCPI. However, if you wish to use ISE instead, reference the README file in `assets/hdl/platforms/matchstiq_z1/ise_constraints/`, and perform the following steps:

1. Modify the target part in `assets/hdl/platforms/matchstiq_z1/matchstiq_z1.mk` to use the ISE alias:

   ```
   HdlPart_matchstiq_z1=xc7z020_ise_alias-1-clg484
   ```

2. Export the ISE constraints files found in `<assets/>hdl/platforms/matchstiq_z1/ise_constraints/` by modifying `ExportFiles` variable in `assets/hdl/platforms/matchstiq_z1/Makefile`:

   ```
   ExportFiles=ise_constraints/matchstiq_z1.ucf ise_constraints/matchstiq_z1.ut matchstiq_z1.mk
   ```

C Driver Notes

When available, the driver will attempt to make use of the CMA region for direct memory access. In use cases where many memory allocations are made, the user may receive the following kernel message:

```
alloc_contig_range test_pages_isolated([memory start], [memory end]) failed
```

This is a kernel warning, but does not indicate that a memory allocation failure occurred, only that the CMA engine could not allocate memory in the first pass. Its default behavior is to make a second pass, and if that succeeded, the end user should not see any more error messages. This message cannot be suppressed, but can be safely ignored. An actual allocation failure will generate unambiguous error messages.