Zipper Deprecation Notice:

Beginning with OpenCPI Version 1.5, support for Lime Microsystems' Zipper card is now deprecated.

ALST4 Hardware Setup

Prerequisites

- A valid OpenCPI development system for Altera. These instructions were performed using Quartus 12.1.
- Windows system for downloading and extracting installation kit from Altera

Hardware Description

The ALST4 is the PCI-Express Altera development board for Stratix4. It can be purchased from Altera directly or one of their distributors. Altera's URL for the board containing user guides and reference manuals can be found here.

`https://www.altera.com/products/boards_and_kits/dev-kits/altera/kit-siv-gx.html`

There are two versions of the board, 230 and 530, which have different density FPGAs on them. Both are supported in OpenCPI under different platform names. The below table contains platform names and part numbers for the development boards.

OpenCPI Platform Name	Altera Kit Part Number	FPGA Part Number
alst4	DK-DEV-4SGX230N	EP4SGX230KF40C2
alst4x	DK-DEV-4SGX530N	EP4SGX530KH40C2

The setup for the two boards is the same. For simplicity, the remainder of this document will only refer to the ALST4 platform.

Setup Overview

To use this board within OpenCPI, the board must be configured to load the FPGA with an OpenCPI bitstream upon power up. There is flash memory on the board which can be loaded with a bitstream to load the FPGA on power up. The process for loading this flash is:

1. Download and extract Kit Installation archive from Altera (requires Windows)
2. Move extracted directory to Linux development PC
3. Set switches to factory defaults
4. Connect USB Blaster cable and power on board outside of chassis
   (a) See USB blaster cable installation for Linux
5. Run script to load flash
6. Power down board and set rotary switch to load OpenCPI bitstream from flash
7. Install board into PCIe chassis and verify OpenCPI bitstreams loads on power up

Details for implementing this procedure can be found in the sections below.

Download and extract Kit Installation archive from Altera (requires Windows)

The Kit Installation archive can be found on Altera's website. The Kit Installation is in the form of a Windows executable, so Windows is required to unpack it. The link for the kit is here :
`https://www.altera.com/products/boards_and_kits/dev-kits/altera/kit-siv-gx.html`

Download and run the latest version of the Kit Installation. As of May 2015, the latest version of the kit was 11.1 and the latest version of Quartus was 12.1. The result of running the executable is a directory called kits (Ex: C:/altera/11.1/kits/stratixIVGX_4sgx230_fpga) There are different kits for DK-DEV-4SGX230N and DK-DEV-4SGX530N, so make sure you download the correct one for your board.

Move extracted directory to Linux development PC

Copy the kits directory created in the previous step to your Linux development machine.Two environment variables are needed to run the loadFlash script:

1. OCPI_ALTERA_TOOLS_DIR
2. OCPI_ALTERA_KITS_DIR.

Set the OCPI_ALTERA_TOOLS_DIR variable to the directory of your installed Quartus tools, and set the OCPI_ALTERA_KITS_DIR variable to the location of where you copied the Kits directory.

An example of setting these variables is below:

```
$ export OCPI_ALTERA_TOOLS_DIR=/home/user/altera/12.1
$ export OCPI_ALTERA_KITS_DIR=/home/user/altera/11.1
```

Set switches to factory defaults

With the board powered down, set the switches as seen in the below diagrams. These are the factory default settings. Switch 5 on the board settings DIP is ON in order to provided the 100 MHz oscillator as the PCIE reference clock. The diagrams can be found in the User Guide, which can be downloaded from Altera at the link above.

Figure 4–1. Switch Locations and Default Settings on the Board Top

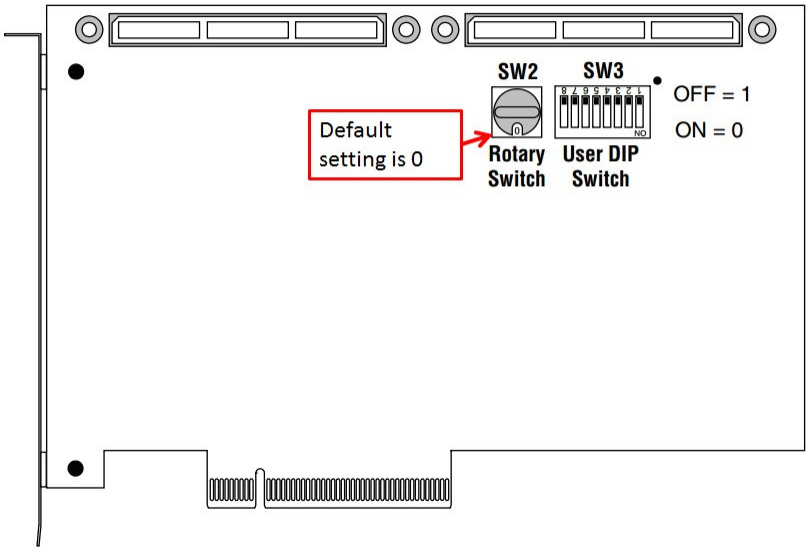
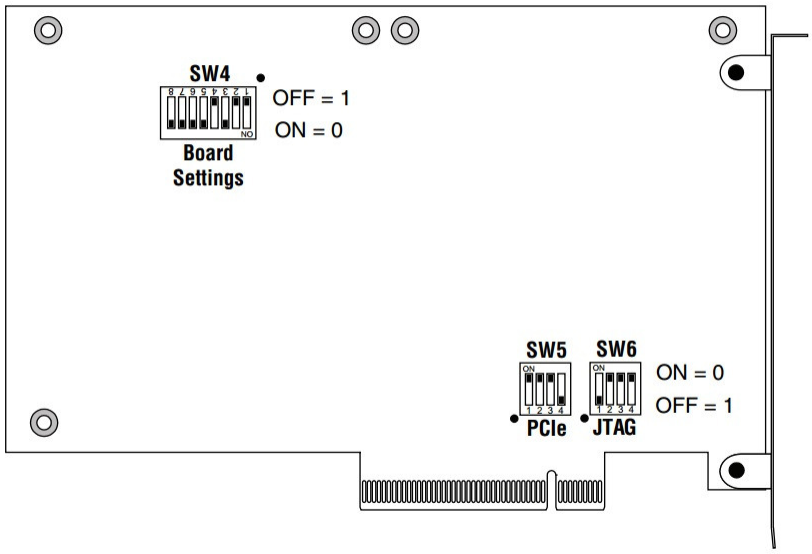


Figure 4–2. Switch Locations and Default Settings on the Board Bottom



Connect USB Blaster cable and power on board

Connect USB type B connector to USB port on board and type A connector to OpenCPI development host. Power on board with the external power supply provided with the kit.

Enter a terminal and source the OpenCPI environment for x86. Run the following script to test the JTAG connection: opencpi/hdl/scripts/probeJtag

The output should look like this:

```
==== Probing for Altera JTAG ports:
Found cable "USB-Blaster [3-2]" with serial number "91d28408"
==== Probing for Xilinx JTAG ports:
Discovered ports are:
Look at /tmp/ocpibitstream5211.1.* for details.
```

For the purposes of this guide, ignore any messages related to Xilinx.

If the output doesn't look like this, refer to the USB blaster cable installation for Linux

Run loadFlash script

The loadFlash script is located at opencpi/tools/cdk/scripts. You must pass it a pre-built ALST4 .bitz file generated by OpenCPI and the serial number (in hex) of the JTAG cable from probeJtag script above. This step with take approximately 10 minutes to complete.

alst4 syntax

```
$ loadFlash alst4 <OpenCPI Bitstream File> <JTAG_CABLE_SERIAL_NUMBER>
```

alst4 output

```
$ [user@opencpi-test1 scripts]$ ./loadFlash alst4 /home/user/ocpi/opencpi/hdl/platforms/alst4/
testbias_alst4_base.bitz 91d28408
Loading the flash memory on the alst4 platform attached to the JTAG pod with ESN 91d28408
Loading from file:  /home/user/ocpi/opencpi/hdl/platforms/alst4/testbias_alst4_base.bitz
Found cable "USB-Blaster [2-1.3]" with serial number 91d28408
Using cable "USB-Blaster [2-1.3]" with serial number 91d28408.
The bitstream file "/home/user/ocpi/opencpi/hdl/platforms/alst4/testbias_alst4_base.bitz" is compressed.
Expanding it.
Bitstream file decompressed into "/tmp/ocpibitstream5351.sof"
```

```
Converting bitstream file "/tmp/ocpibitstream5351.sof" to flash format in "/tmp/ocpibitstream5351.flash"
using sof2flash
Loading factory bitstream "/home/user/altera/11.1/kits/stratixIVGX_4sgx230_fpga/factory_recovery/
s4gx230_fpga_bup.sof" to use for indirectly writing flash memory.
JTAG Loading of the factory bitstream succeeded.  Now using it to write the flash.
real 1m25.590s
user 0m0.801s
sys 0m0.591s
Flash programming is complete.  You must power-cycle the system to use it.
```

Use the "ocpihdl search" command after power cycling to confirm success.

Modify Linux kernel boot arguments to reserve PCIE memory

If you want the Linux OpenCPI driver to use more than 128 KB of RAM, then you will need to reserve a block of memory during the Linux kernel boot, using the memmap paramater. Reference the README in the driver directory of the OpenCPI install location for its usage. Regardless of desired RAM size, all CentOS 7 users are recommended to make the grub2 modifications (for 128 MB) specified therein. Make any grub/grub2 modifications and reboot the development system before continuing.

Power down board and set rotary switch to load OpenCPI bitstream from flash

With the board powered down, set SW2 to 1. See figure above for location of SW2 on board. This setting will cause the board to load the bitstream from the flash location we just programmed.

Install board into PCIe chassis and verify OpenCPI bitstream loads on power up

With the development system powered down, install the PCIe card. Reconnect the USB cable and then power on the system. To check the board was installed correctly, enter a terminal and source the OpenCPI environment for x86. Then load the PCIe driver using (requires sudo privileges).

```
$ ocpidriver load
```

Perform a container check using ocpirun -C. The result should look like this:

```
[user@opencpi-test1 opencpi]$ ocpirun -C
Available containers:
# Model      Platform    OS     OS Version  Name
0 rcc         x86_64     linux   c6         rcc0
1 hdl         alst4                         PCI:0000:02:00.0
```

Known Issue

See alst4_getting_started_guide.pdf "Known Issues" section for known alst4 issues.

ALST4 Card Support

The OpenCPI alst4 platform includes support for use of the Zipper/Myriad RF card in one or both HSMC slots on the Stratix IV board. Note that for proper function, the hardware modifications specified in [1] are necessary. If using a Zipper/Myriad RF with the alst4, ensure that the Myriad RF card is connected to the Zipper via HSMC.

References

[1] Required Modifications for Myriad-RF 1 and Zipper Daughtercards
Required_Modifications_for_Myriad-RF_1_Zipper_Daughtercards.pdf (included in cards doc directory)