

Arrow High Speed Data Acquisition Kit User Guide

Featuring the
Analog Devices DAQ2 Evaluation Board
and
Intel Arria 10 GX Development Kit

Table of Contents

1. INTRODUCTION.....	3
Theory of Operation	3
2. DATA FLOW/SIGNAL PATH	5
Transmit Path	5
Receive Path	5
3. HARDWARE	7
Kit Contents	7
2.1 Intel Arria 10 GX Development Kit.....	8
2.2 Analog Devices DAQ2 Evaluation Board	10
4. SOFTWARE	12
3.1 Downloading the github content	12
3.2 Applications	13
3.3 Files	13
4.1 HARDWARE SETUP	14
4.2 FIRMWARE SETUP	17
4.3 Navigate to the content folder.....	17
6.1 Start the DHCP Server	17
6.2 7. SOFTWARE SETUP.....	23
8. RUNNING THE DEMONSTRATION.....	24
8.1 Start the DHCP Server	24
8.2 Connect the IIO Oscilloscope	29
8.3 Create Plots	33
8.4 Shutdown.....	40
9. BUILDING THE HARDWARE AND SOFTWARE REFERENCE DESIGNS	42
9.3 Install the Intel PSG development tools.....	42
9.4 Build the hardware reference design.....	43
9.5 Create the hardware flash image	44
9.6 Software reference design resources.....	45
9.7 Build Linux for the Nios II processor.....	45
10.1 Create the software flash image.....	49
10.2 10. APPENDIX	50
More Information	50
Online resources	51
11. REVISION HISTORY	53

1. Introduction

The Arrow High-Speed Data Acquisition Kit is a complete development platform for creating solutions with the space-saving high-speed GSPS JESD204B serial interface.

Featuring Analog Devices' GSPS data converters and Intel's Arria 10 FPGA, the kit allows them to communicate across 4 serial lanes in each direction at speeds up to 40GSPS. Also provided are all the timing and power components required to both develop solutions and demonstrate the capabilities of JESD204B. With sampling speeds of up to 1GSPS, this is the ideal platform for applications requiring the capture and transportation of large amounts of data.

A one-year license for all IP and tools is included so that the user can begin development immediately.

The built-in demo mode allows for a rapid out of box (OOB) demonstration without the need for installing the Intel Quartus toolset.

Theory of Operation

1.1

The kit is configured to allow the user to send data from the Intel Arria 10 GX FPGA to the Analog Devices AD9144 DAC using the JESD204B serial interface and FMC (FPGA Mezzanine Card) connectors. It is setup to send data from the Analog Devices AD9680 ADC back to the Arria 10 by the same method. By attaching two RF cables, the analog outputs from the AD9144 can be looped back to the inputs of the AD9680 forming a complete closed system. Using Analog Devices' IIO Oscilloscope application on a host PC, the user can set the inputs generated by the Arria 10 and view the outputs on the return path.

The FPGA provides many capabilities in this kit. The primary means of configuring the DAC outputs and receiving the ADC inputs happens via dedicated IP blocks in the FPGA fabric. This also provides the gigabit transceivers necessary to communicate to the JESD204B capable DAC and ADC.

A wired Ethernet connection between the FPGA and the host PC is used to transfer the data to and from the host IIO Oscilloscope application and the Linux OS running on the NIOS processor in the FPGA.

The data sent out from the FPGA to the DAC is supplied from one of two sources: the FPGA can either create a digital sine wave internally via its Direct Digital Synthesizer block (DDS), or read in an externally supplied data file. From here the digital data is packaged and sent to the JESD204B IP block inside the FPGA, where it is configured per the JESD204B standard. The serialized data is then transmitted to the DAC over an FPGA Mezzanine Card (FMC) connection on four 10Gbps channels. Once the data has been received and decoded by the JESD204B DAC, the analog signals are sent to a pair of RF SMA connectors; this is defined as the transmit path.

For a closed loop system the signals from the DAC's SMA connectors are transmitted over RF cables to SMA connectors connected to the inputs of the ADC. The ADC takes the analog signals and converts to the serialized digital JESD204B format, then sends this to the FPGA over the FMC connection on four additional 10Gbps channels; the receive path.

User Guide

Once the data is captured in the FPGA it is sent over the gigabit Ethernet connection to the host PC running the IIO Oscilloscope application, where the results can be viewed in the time and frequency domains or as a constellation plot.

2. Data Flow/Signal Path

Here is a brief description of the signal as it flows from the FPGA across the JESD204B interface to the data converters and back. Refer to Figure 1.

Transmit Path

1. Using the IIO Oscilloscope application running on a host PC, along with a gigabit Ethernet cable connected from the PC to the Arria 10 FPGA, either a digital sine wave in the 0-500MHz range is generated from the FPGA's internal DDS frequency synthesizer blocks, or a data file with a digital signal representation is selected.
- 2.1 Inside the Arria 10 the digital signal is converted into the JESD204B format using Analog Devices' IP. It is then sent out across 4 lanes to an FMC connector on the Arria 10 development kit at a rate of 40Gbps (4 lanes @ 10Gbps each).
3. The data flows through the corresponding FMC connector on the AD-FMCDQA2-EBZ board and into the AD9144 multichannel high-speed DAC where it is converted from the JESD204B standard to the proper digital format prior to being converted to analog. The AD9144 DAC generates both I and Q signals used for quadrature amplitude modulation (QAM) of the RF signal, converts it to analog and sends each signal out via SMA RF connectors connected to RG316 RF coaxial cables.

2.2 Receive Path

4. The analog RF signals flow through the two coaxial cables and back into the AD-FMCDQA2-EBZ board through two additional SMA RF connectors routed to the AD9680 2-channel ADC. The signal is sampled and digitized and converted back into the JESD204B standard.
5. The data is sent back along 4 lanes also at a rate of 10Gbps per lane (40Gbps total) across the FMC connector to the Arria 10 GX kit where it is decoded by Arria 10 FPGA IP blocks.
6. The received data is sent to the host PC over the gigabit Ethernet connection. Using the ADI IIO Oscilloscope graphical user interface application on the host, the received data can be viewed in the time domain, frequency domain or as a polar plot.

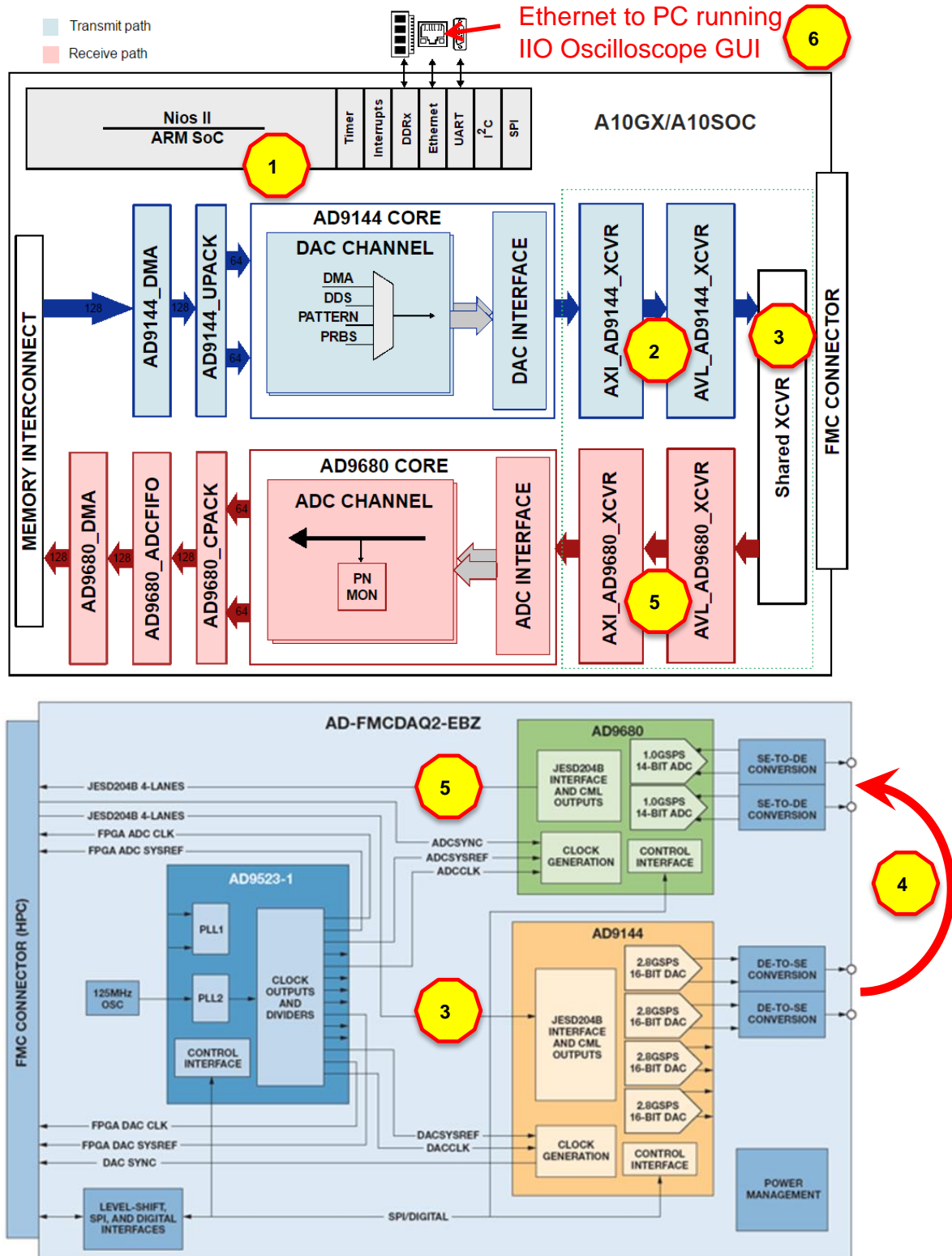


Figure 1: JESD204B Demonstration Platform Block Diagram

3. Hardware

Kit Contents

The Arrow High-Speed Data Acquisition Kit comes with the following components:

- 3.1
- Intel [Arria 10 GX Development Kit](#) p/n DK-DEV-10AX115S-A including power supply
 - Analog Devices [AD-FMCDQA2-EBZ](#) 'DAQ2' Evaluation Board
 - (2) [RG316 SMA RF cables](#)
 - (1) [5' CAT-6A 1000BASE-T GigE Ethernet cable](#)

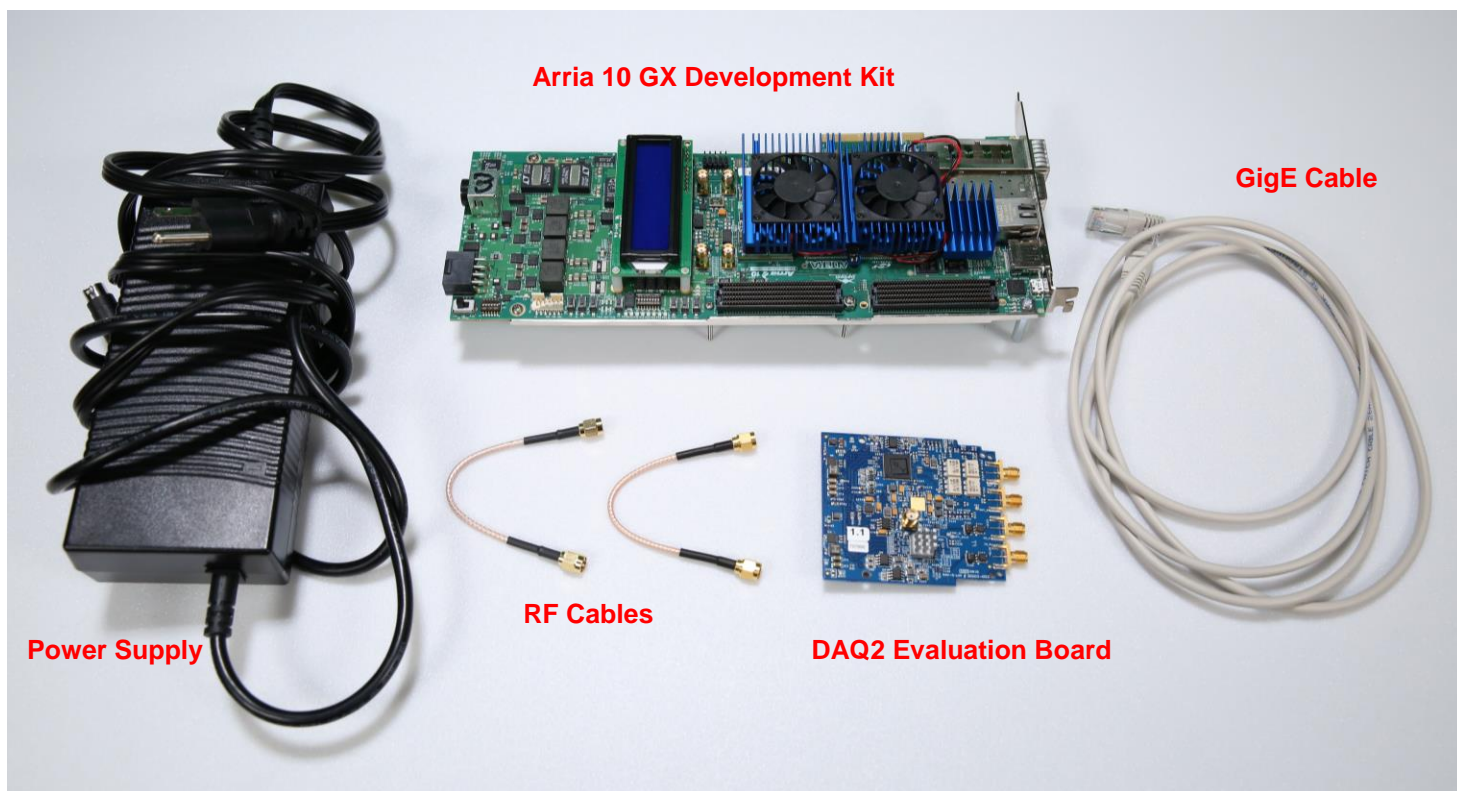


Figure 2: Arrow High-Speed Data Acquisition Kit Contents

Intel Arria 10 GX Development Kit

3.2



Figure 3: Arria 10 GX FPGA Development Kit (Power Supply not shown)

- https://www.altera.com/products/boards_and_kits/dev-kits/altera/kit-a10-gx-fpga.html

User guide: https://www.altera.com/content/dam/altera-www/global/en_US/support/boards-kits/arria10/FPGA/A10-FPGA-ES3-DK-UG_1.pdf

The Arria 10 GX FPGA Development Kit includes the following hardware.

- Arria 10 GX 10AX115S2F45I1SG2 FPGA
- 1 GB DDR4 SDRAM, 2GB DDR3 SDRAM, or RLDRAM3 (16 Meg x 36) (daughter cards)
- Two FMC loopback interfaces supporting gigabit transceivers, LVDS and single-ended I/Os
- One quad small-form-factor pluggable (QSFP) and one SFP+ port
- PCIe x8 edge connector
- Serial digital interface (SDI) channel
- Character LCD
- AC adapter power cables
- Ethernet and USB cables

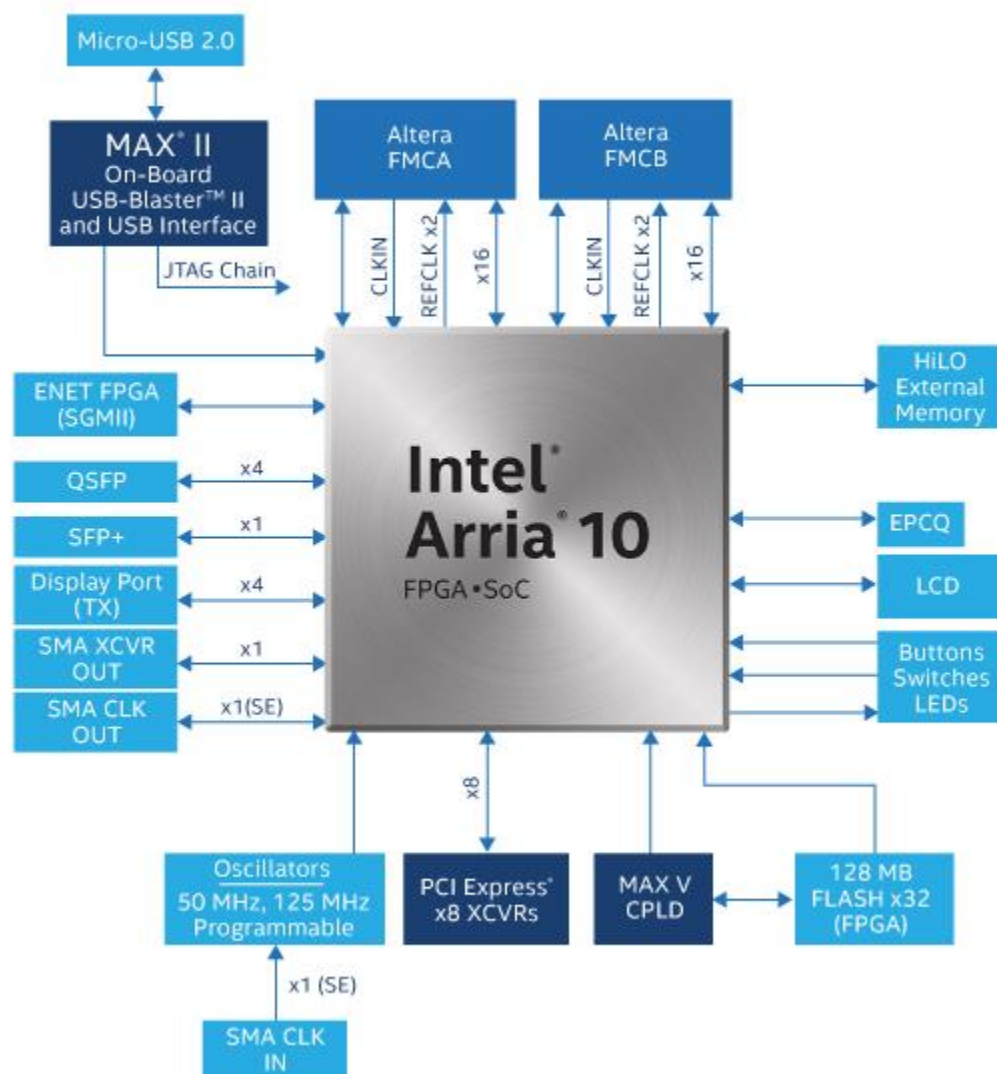


Figure 4: Arria 10 GX Development Board Block Diagram

Analog Devices DAQ2 Evaluation Board

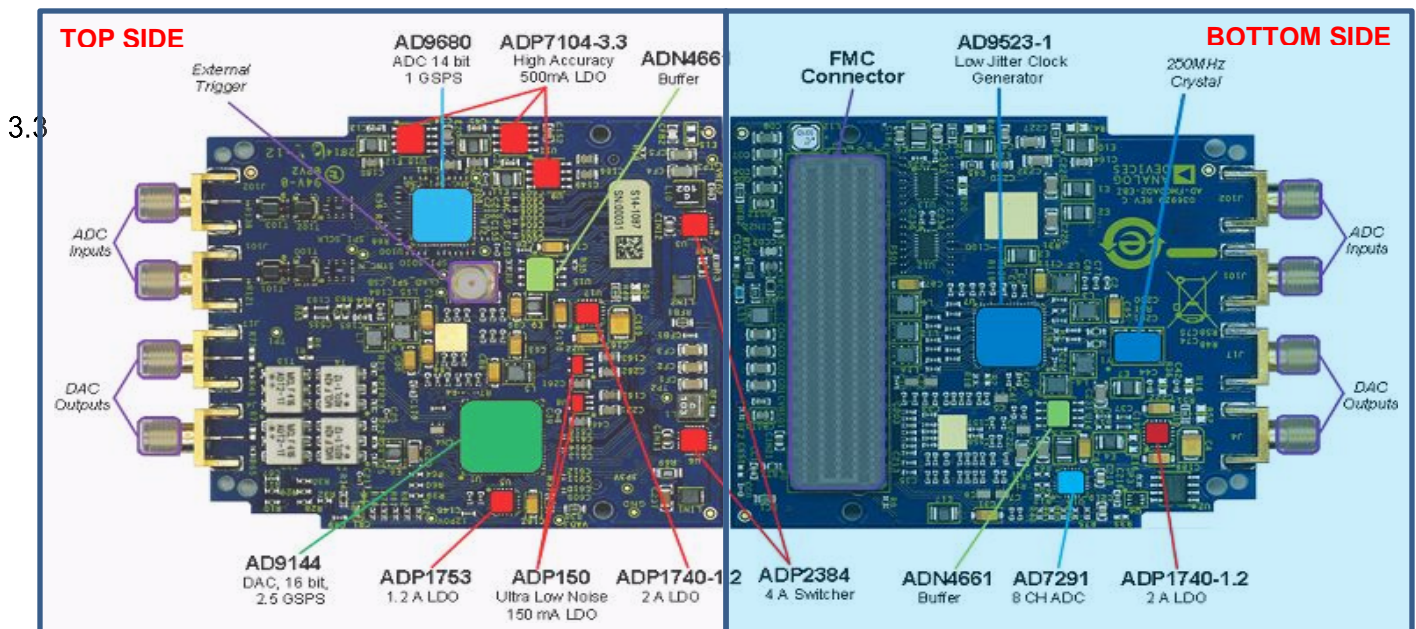


Figure 5: Analog Devices AD-FMCDQA2-EBZ Evaluation Board Top (Left) and Bottom (Right)

- <http://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/eval-ad-fmcdqa2-ebz.html>

The AD-FMCDQA2-EBZ Evaluation Board contains the following.

- AD9680 2-channel 14-Bit 1.0GSPS JESD204B ADC
- AD9144 4-Channel 16-Bit 2.8GSPS JESD204B DAC
- AD9523-1 14-Output 1GHz Clock Generator
- FPGA Mezzanine Card (FMC) High-Speed Connector

Also:

- AD7291 8-Channel 12-Bit ADC
- Miscellaneous Power and Timing components

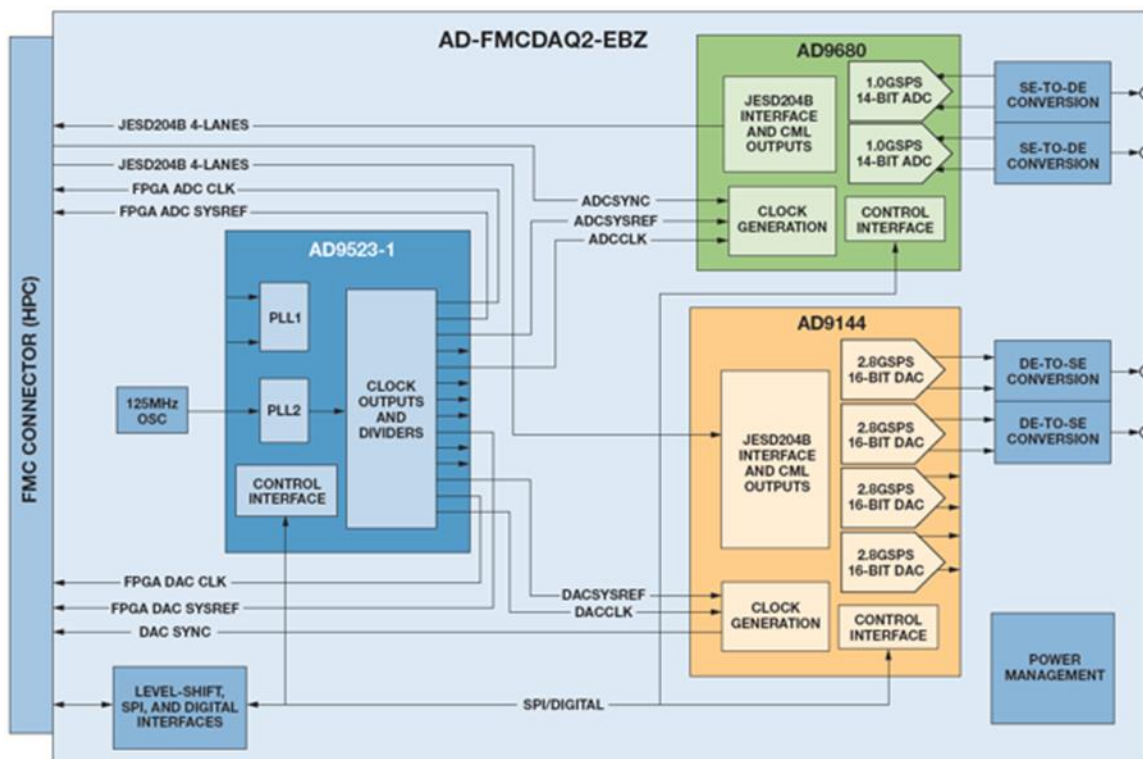


Figure 6: AD-FMCDAQ2-EBZ Evaluation Board Block Diagram

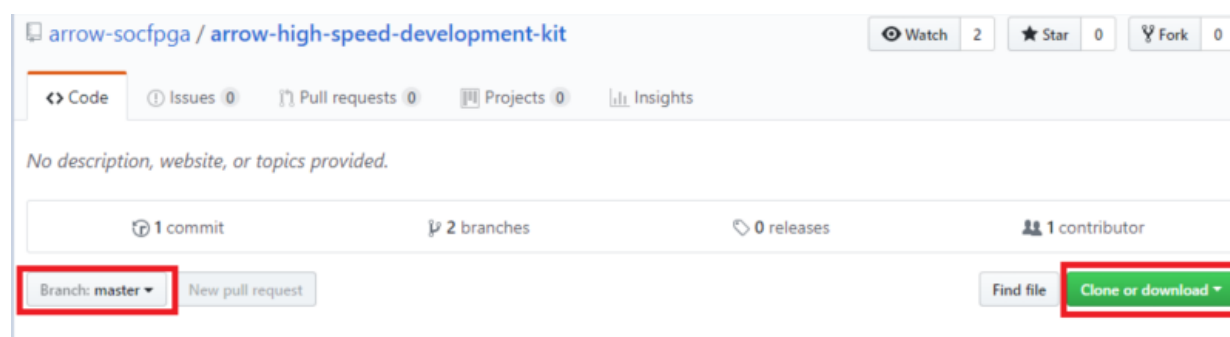
4. Software

All necessary software applications and files listed below are provided in the [arrow-socfpga github site](#). There is no need to install the Intel Quartus Prime Design Software to run this demonstration.

Downloading the github content

- Files necessary for the demonstration can be found on the **Arrow** github pages. Follow the instructions listed below to download this content

4.1



- Open a browser to <https://github.com/arrow-socfpga/arrow-high-speed-development-kit>
- Select the **2017_r1** branch.
- Download the repository as a zip file. Extract the zip file to a directory.
- cd to the **arrow-high-speed-development-kit/arria10gx_intel_PSG** sub-directory.

github_repo > arrow-high-speed-development-kit > arria10gx_intel_PSG			
Name	Date modified	Type	Size
Datasheets	10/17/2017 2:23 PM	File folder	
DHCP Server	10/17/2017 2:23 PM	File folder	
adi-osc-master-setup	10/17/2017 2:23 PM	Application	58,263 KB
Flash_devkit	10/17/2017 2:23 PM	Windows Batch File	6 KB
flash_files	10/17/2017 2:23 PM	Compressed (zipped) F...	18,619 KB
JESD204B_Demo	10/17/2017 2:23 PM	Windows Batch File	6 KB

Applications

- 4.2 • Analog Devices IIO Oscilloscope – **OSC.exe**
 - The ADI IIO Oscilloscope is an example application, which demonstrates how to interface different evaluation boards from within a Linux system. The application supports plotting of the captured data in four different modes (time domain, frequency domain, constellation and cross-correlation). The application also allows a user to view and modify several settings of the evaluation board's devices.
 - The IIO Oscilloscope application installer is provided in the downloaded content.
 - Optional web link: <https://github.com/analogdevicesinc/iio-oscilloscope/releases/download/v0.5-master/adi-osc-master-setup.exe>
- DHCP Server – **OpenDHCPServer.exe**
 - A simple DHCP server is required so that the network interface of the Arria 10 FPGA is assigned an IP address within the same subnet as the host PC.
 - The DHCP server application is provided in the downloaded content.
 - Optional web link: <https://sourceforge.net/projects/dhcpserver/>

Files

- 4.3 • Arria 10 Development kit firmware flash batch file – **Flash_devkit.bat**
 - The **Flash_devkit.bat** batch file temporarily assigns a static IP address to the host PC. It also dynamically assigns an IP address to the Arria 10 development kit. This will be used in Section 6 to flash the demo firmware into the kit.
- Firmware flash files – **flash_files.zip**
 - Contains the FPGA hardware image flash file
 - Contains the Nios II processor Linux flash file.
- Arria 10 JESD204B Demonstration batch file – **JESD204B_Demo.bat**
 - The JESD204B_Demo.bat batch file temporarily assigns a static IP address to the host PC. It also dynamically assigns an IP address to the Arria 10 development kit.
- DHCP Server configuration settings – **OpenDHCPServer.ini**
 - Required by the OpenDHCPServer application.
 - The configuration settings file is provided in the JESD204B_Demo.zip file.

5. Hardware Setup

The instructions in this section will guide you through the hardware setup of the JESD204B demonstration platform using the Arria 10 GX FPGA Development Kit and the Analog Devices DAQ2 JESD204B Evaluation Board.

1. Connect the AD-FMCDQAQ2-EVK board to the FMC A connector on the Arria 10 GX board.

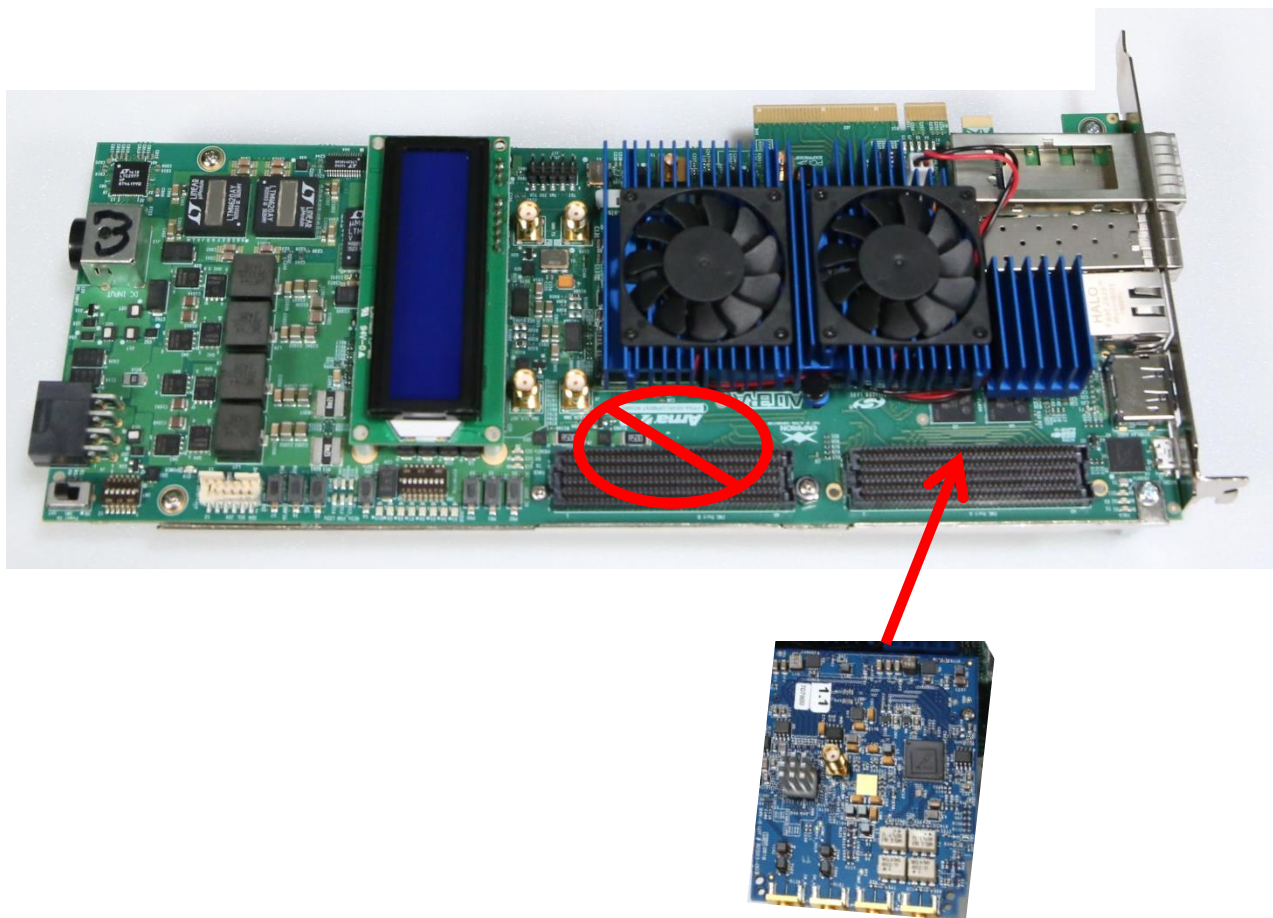


Figure 7: Connect AD-FMCDQAQ2-EVK board to FMC 'A' connector on Arria 10 GX board

2. On the AD-FMCDQAQ2-EVK hook up the two provided RG316 RF cables as shown

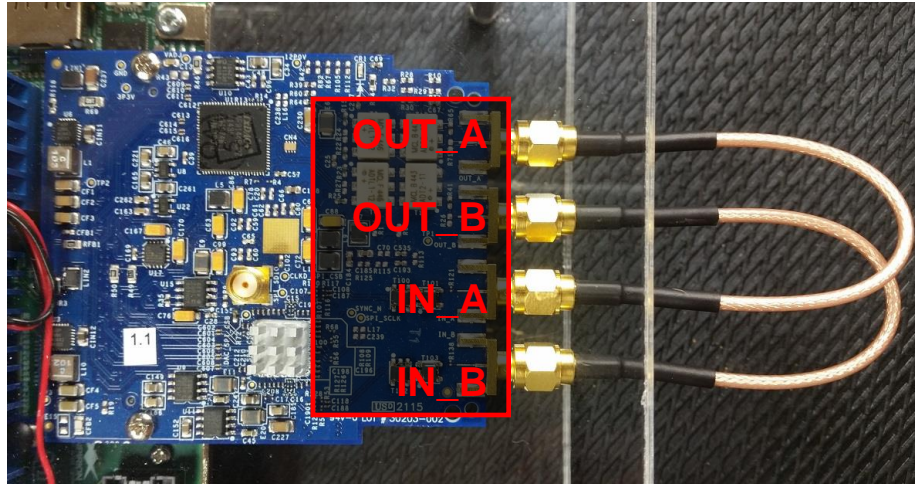


Figure 8: RF cable connections

3. Using the Cat-5e gigabit Ethernet cable connect the Arria 10 GX kit directly to the host PC via the RJ45 Ethernet port on each. Important: the use of a gigabit capable Ethernet patch cable is required due to the high data rates of the JESD204B interface.
4. Plug the included power supply into the Arria 10 GX board's 4-pin connector and into a 110V outlet.

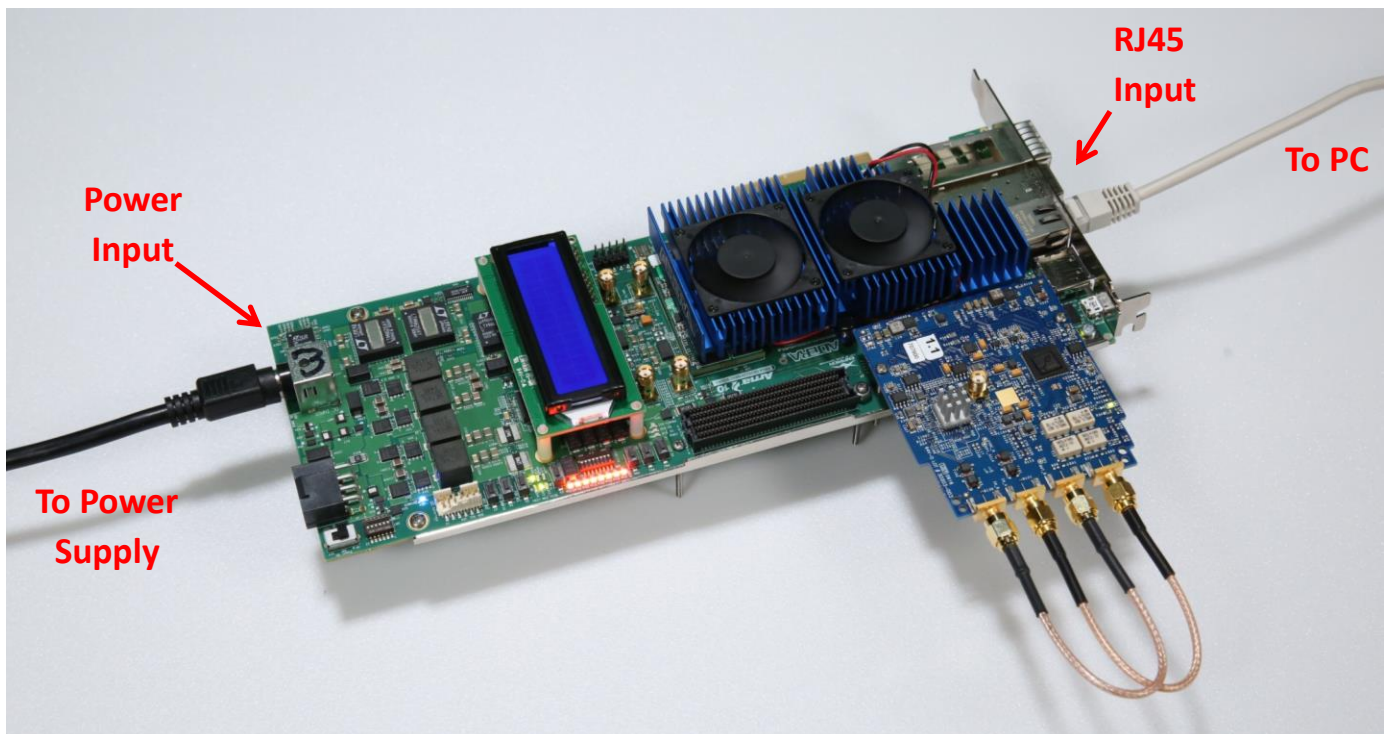


Figure 9: Power and Ethernet connections on Arria 10 GX Development Kit

5. Here is the complete setup

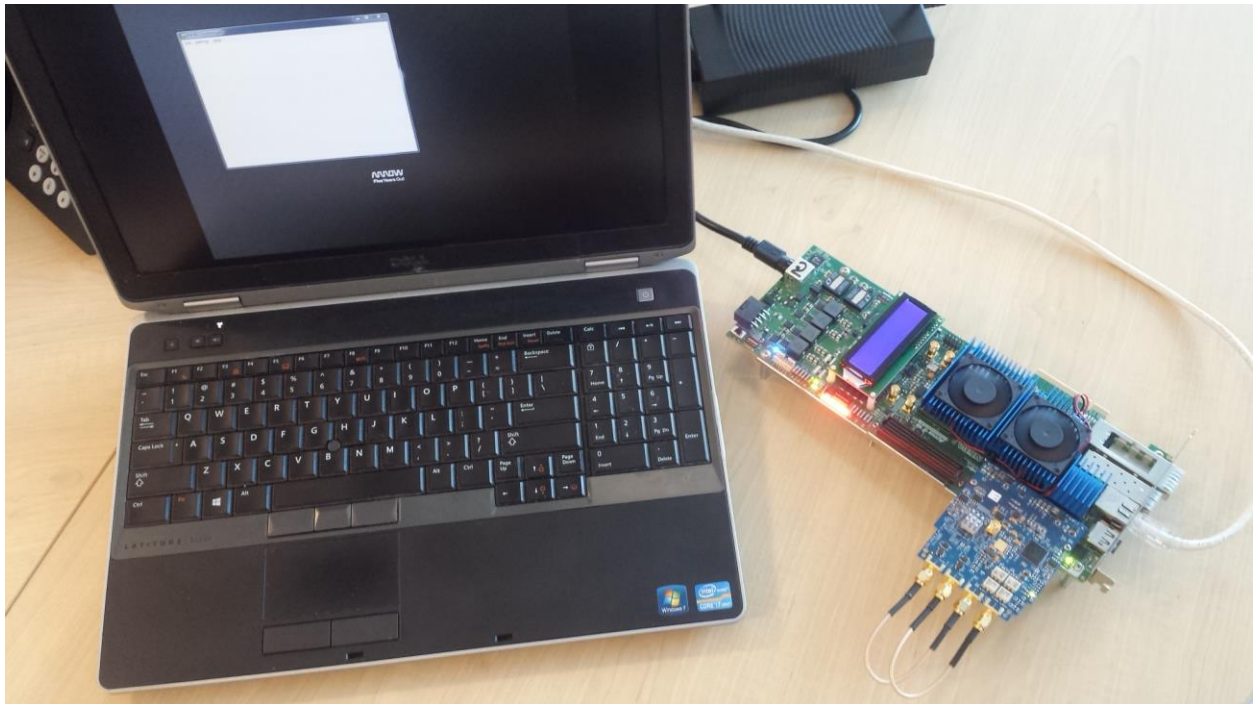


Figure 10: Complete Demo Setup

6. Firmware setup

The instructions in this section will guide you through flashing the demonstration firmware into the Arria 10 GX development kit.

Navigate to the content folder

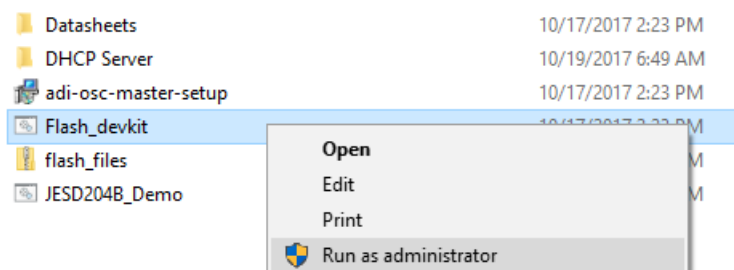
- Open the folder previously downloaded in section 4.1
- 6.1 • Unzip **flash_files.zip**
 - **daq2.flash** is the **FPGA** image flash file
 - **zImage.flash** is the Nios II processor **Linux** image flash file

Datasheets	10/17/2017 2:23 PM	File folder	
DHCP Server	10/19/2017 6:49 AM	File folder	
adi-osc-master-setup	10/17/2017 2:23 PM	Application	58,263 KB
daq2.flash	3/8/2017 12:24 PM	FLASH File	108,531 KB
Flash_devkit	10/17/2017 2:23 PM	Windows Batch File	6 KB
flash_files	10/17/2017 2:23 PM	Compressed (zipped) F...	18,619 KB
JESD204B_Demo	10/17/2017 2:23 PM	Windows Batch File	6 KB
zImage.flash	3/2/2017 6:45 AM	FLASH File	12,735 KB

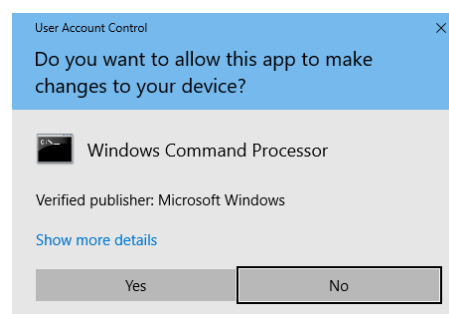
6.2

Start the DHCP Server

1. Flash_devkit.bat batch file as **administrator** by right-clicking the file and selecting **Run as administrator**.



2. Select **Yes** on the **User Account Control** dialog allowing the script to execute.



The “**Flash_devkit**” **Script**” window will open providing information about the script.

3. **Press any key** on the PC keyboard to continue.

```
Administrator: "Flash Arria 10 GX development kit"

--
--
-- Flash the JESD204B demo onto the Arria 10 GX Development kit
--
-- This script temporarily assigns the host PC a static IP address of
-- 192.168.0.1 and starts a DHCP server to provide the IP address
-- 192.168.0.2 to the Arria 10 GX board.
--
--
--
Press any key to continue . . .
```

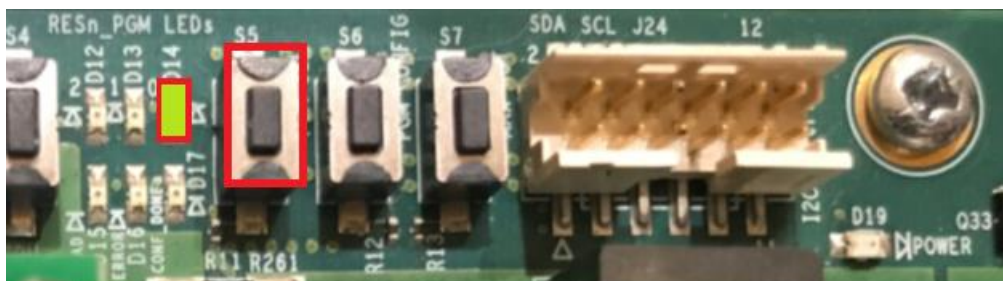
4. Move the power switch, **SW1**, on the Arria 10 GX kit to the **ON** position.

```
Administrator: "Flash Arria 10 GX development kit"

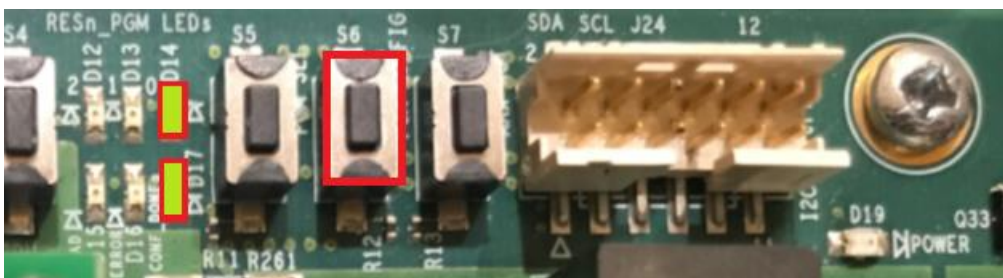
--
--
-- Power ON the Arria 10 GX board now if it is not already on.
--
-- Configure the dev kit to boot the Board Update Portal
--
-- 1. Press switch S5 until PGM_LED0 ( D14 - right most LED) is lit
-- 2. Press switch S6 to configure the board.
--
--
--
Press any key to continue . . .
```

5. Configure the Arria 10 GX development kit to boot the **Board Update Portal** (BUP)

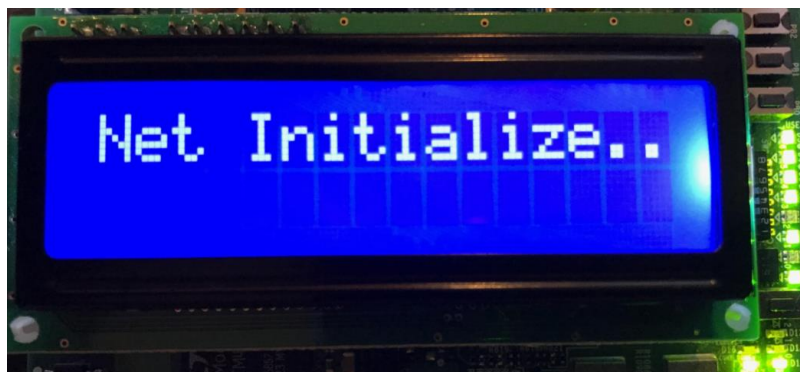
- Press switch S5 until PGM_LED0 (D14 - right most LED) is lit



- Press switch S6 to configure the board. The CONF_DONE LED (D17 – bottom right) will be lit to indicate the BUP image has been loaded into the FPGA



The text below will appear on the display to indicate that the BUP has been successfully loaded into the FPGA.



- Press any key on the keyboard to continue.

The script will query the PC Ethernet connections for 5 seconds and automatically report back the active interfaces.

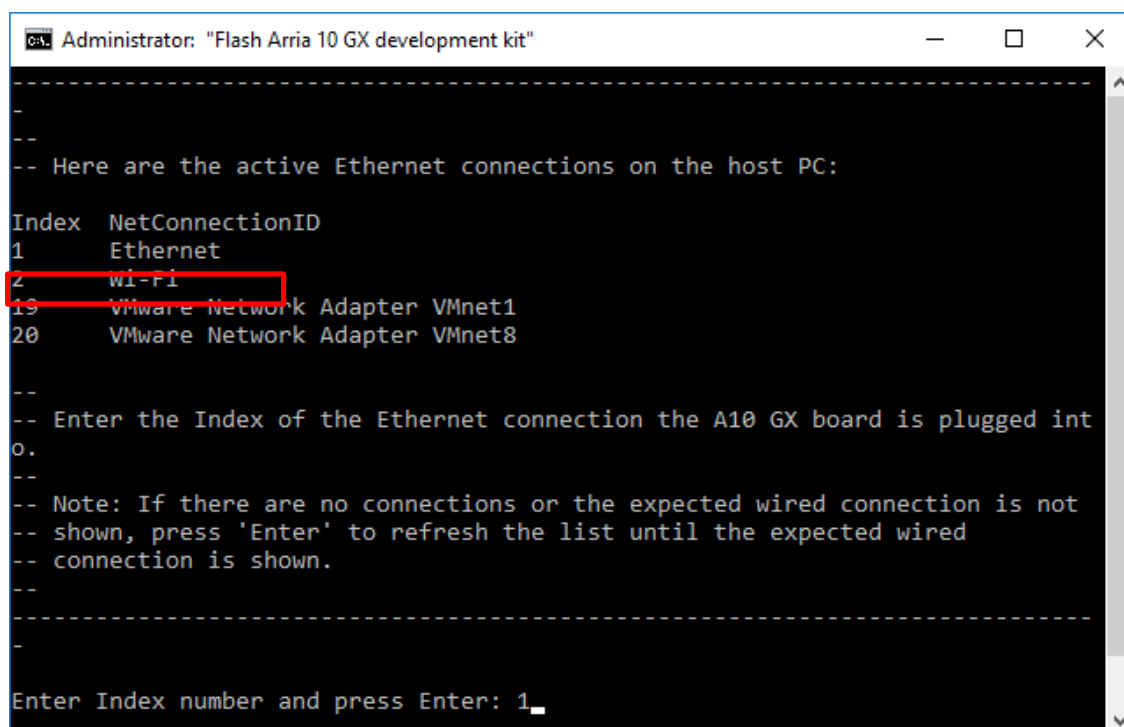


```

Administrator: "Flash Arria 10 GX development kit"
-----
--
-- Looking for active Ethernet connections...
--
-----
Waiting for 2 seconds, press a key to continue ...
  
```

Note: The name of the connections may vary. ('Local Area Connection' or 'Ethernet' are common.)

6. Enter the **Index** number of the Ethernet connection which the A10 GX board is plugged into and press **Enter**. (Index = 1 for "Ethernet" in the below example.)



```

Administrator: "Flash Arria 10 GX development kit"
-----
--
-- Here are the active Ethernet connections on the host PC:
Index  NetConnectionID
1      Ethernet
2      Wi-Fi
19     VMware Network Adapter VMnet1
20     VMware Network Adapter VMnet8
--
-- Enter the Index of the Ethernet connection the A10 GX board is plugged into.
--
-- Note: If there are no connections or the expected wired connection is not
-- shown, press 'Enter' to refresh the list until the expected wired
-- connection is shown.
--
-----
Enter Index number and press Enter: 1_
  
```

Status messages will appear with information concerning assigning the static IP address to the host and starting the DHCP server. These messages will only appear for a few seconds and then disappear. The entire process can take up to a minute to assign an IP address.

```

Administrator: "Flash Arria 10 GX development kit"

-----
--
--
-- Assigning static IP address: 192.168.0.1 to local host...
--
-- Starting DHCP server...
-- (You will see a few status messages here for just a momment.)
--
-----
--
--
Open DHCP Server Version 1.64 Windows Build 1041 Starting...
Logging: Normal
Warning: No IP Address for DHCP Static Host 00:ff:a4:0e:ef:99 specified
DHCP Range: 192.168.0.2-192.168.0.254/255.255.255.0
Server Name: LM1310
Detecting Static Interfaces..
Warning: 192.168.0.1 UDP Port 67 already in use
No Static Interface ready, Waiting...
  
```

Additional DHCP server status messages will appear with information concerning assigning the static IP address to the Arria 10 GX. Upon successful DHCP allocation, verify the static IP address 192.168.0.2 was assigned to the Arria 10 GX.

7. **Press any key** once the status message **"Host xx:xx:xx:xx:xx:xx (Hostxxxxxxxxxxxx) allotted 192.168.0.2 for 3600 seconds"** appears. (Note: The "xx" values will vary depending on the host PC MAC address.)

```

Administrator: "Flash Arria 10 GX development kit"

-----
--
--
-- There will be a short delay (~40 seconds) while the DHCP address is issued.
--
-- Upon succesful DHCP allocation, verify DHCP server output on last line:
-- "... (Hostxxxxxxxxxxxx) allotted 192.168.0.2 for 36000 seconds"
--
-- Press any key to continue after IP address is allotted . . .
--
-----
Retrying failed Listening Interfaces..
Lease Status URL: http://127.0.0.1:6789
Listening On: 192.168.0.1
DHCPDISCOVER for 00:07:ed:2a:03:ec () from interface 192.168.0.1 received
Host 00:07:ed:2a:03:ec (Host0007ed2a03ec) offered 192.168.0.2
DHCPPREQUEST for 00:07:ed:2a:03:ec () from interface 192.168.0.1 received
Host 00:07:ed:2a:03:ec (Host0007ed2a03ec) allotted 192.168.0.2 for 36000 seconds
  
```


8. Open a web browser to connect to the Board Update Portal

- Open a web browser with the URL set to 192.168.0.2
- Select daq2.flash for the Hardware File Name
- Select zImage.flash for the Software File Name
- Press the Upload button to begin the Upload

The files are large and the upload process can take up to 30 minutes to complete.



Board Update Portal Arrial 10 GX FPGA Development Kit

This Board Update Portal web page is being served by a design running in the FPGA on your development board. This page, in coordination with the FPGA design serving it, allows you to write new FPGA images to the flash on your board and provides links to useful information on the Altera® website. The FPGA design contains a Nios® II processor and the Triple Speed Ethernet media access control (MAC) MegaCore® function. When you install the development kit design files on your system, the design files for the Board Update Portal FPGA design are installed in the `...kits/arrial10GX_10AX115SF45_fpga/examples/board_update_portal` directory. This design is one example of how to remotely update an FPGA system over Ethernet. Remote update can be accomplished without a webserver, and it can also be used to update just the firmware of an embedded FPGA system. Please see application note [AN429: Remote Configuration Over Ethernet with the Nios II Processor \(PDF\)](#) to learn more about remote update.

Instructions on preparing your own .sof/.elf files for uploading to flash via the Board Update Portal are available [here](#).

Upload New Designs to User Portion of Flash Memory

Hardware File Name:

daq2.flash

Software File Name:

zImage.flash

Kit Specific Resources

- [Arrial® 10 GX FPGA Development Kit](#)
- [High-speed serial solutions](#)
- [Arrial 10 FPGAs](#)

General Design Resources

- [Board Design Resource Center](#)
- [Licensing](#)
- [Software Download Center](#)
- [Technical Support Center](#)
- [Development kits](#)
- [Embedded processing](#)
- [Altera Forum](#)
- [Altera Wiki](#)

Other Utilities

- [Factory Restore](#)

The following message will appear when uploading is complete



Board Update Portal FPGA Development Kit, Arria® 10 GX Edition

Upload Complete!

Press button PGM_SEL (S5) until PGM_LED 1 is lit, then press button PGM_CONFIG (S6) to configure the FPGA with the new image.

Copyright © 1995-2015 Altera Corporation. All Rights Reserved.

ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.

7. Software Setup

The instructions in this section will guide you through the software setup of the JESD204B demonstration.

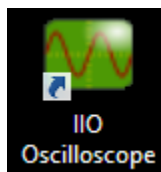
Open the folder previously downloaded in section 4.1

Datasheets	10/17/2017 2:23 PM	File folder	
DHCP Server	10/19/2017 9:58 PM	File folder	
adi-osc-master-setup	10/17/2017 2:23 PM	Application	58,263 KB
daq2.flash	3/8/2017 12:24 PM	FLASH File	108,531 KB
Flash_devkit	10/17/2017 2:23 PM	Windows Batch File	6 KB
flash_files	10/17/2017 2:23 PM	Compressed (zipped) F...	18,619 KB
JESD204B_Demo	10/17/2017 2:23 PM	Windows Batch File	6 KB
zImage.flash	3/2/2017 6:45 AM	FLASH File	12,735 KB

Figure 11: Extracted JESD204B Archive

1. Install the Analog Devices IIO Oscilloscope application by double-clicking the **adi-osc-master-setup.exe** installer. Accept **ALL** the defaults and accept the License Agreement. Also check the box labeled 'Create a desktop icon'.

After installing the IIO Oscilloscope there will be a new **IIO Oscilloscope** Desktop icon.



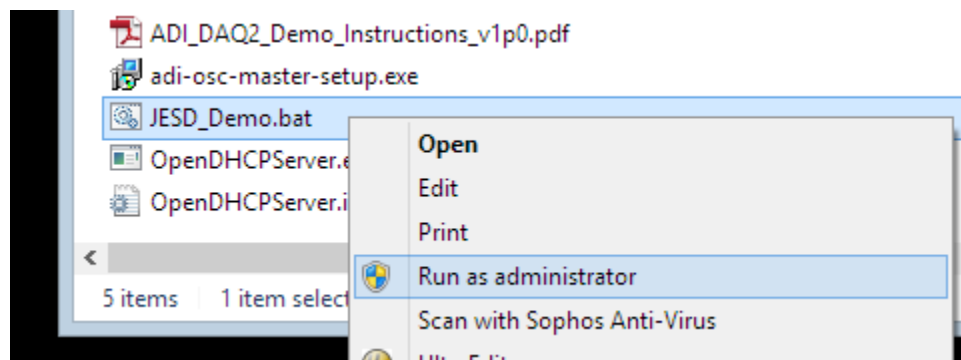
8. Running the Demonstration

This section describes how to run the JESD204B demonstration including powering up the hardware and making a connection to the system with the IIO Oscilloscope application. An initial configuration of the transmitters is described and details on creating plots of the received data are covered.

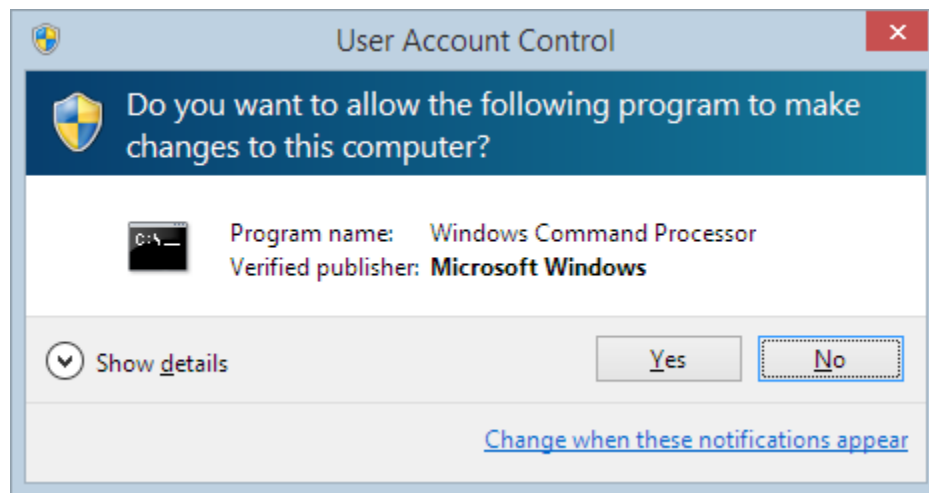
Start the DHCP Server

9. Launch the **JESD204B_Demo.bat** batch file as **administrator** by right-clicking the file and selecting **Run as administrator**.

8.1

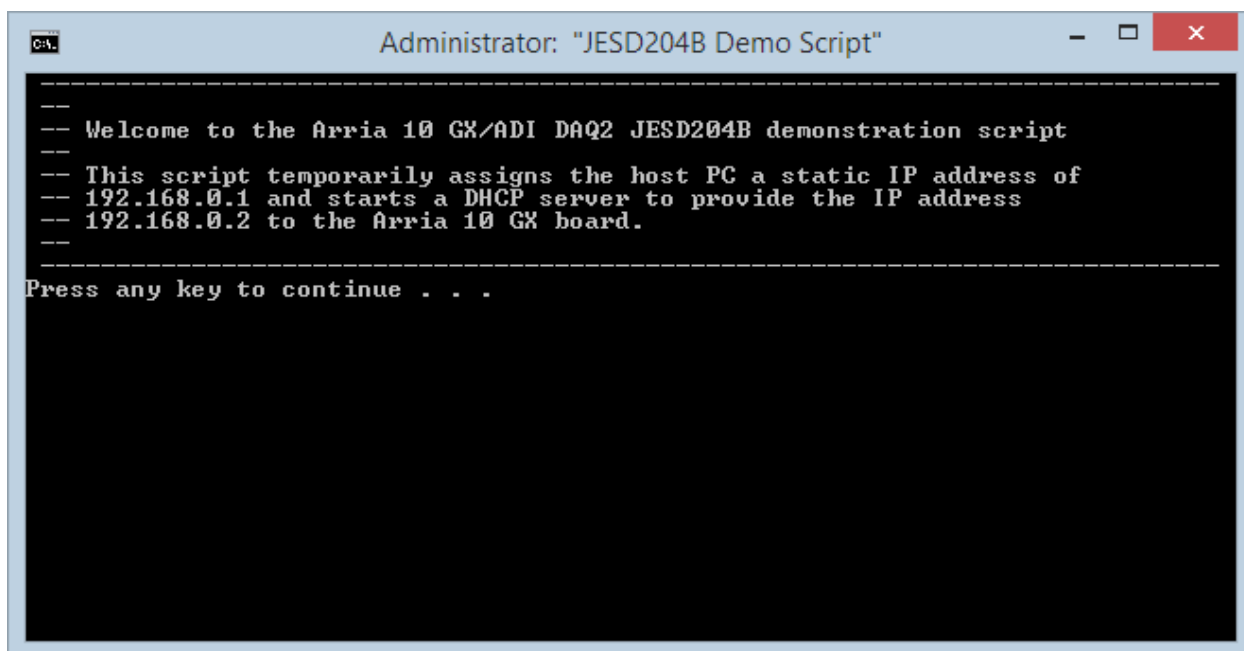


10. Select **Yes** on the **User Account Control** dialog allowing the script to execute.

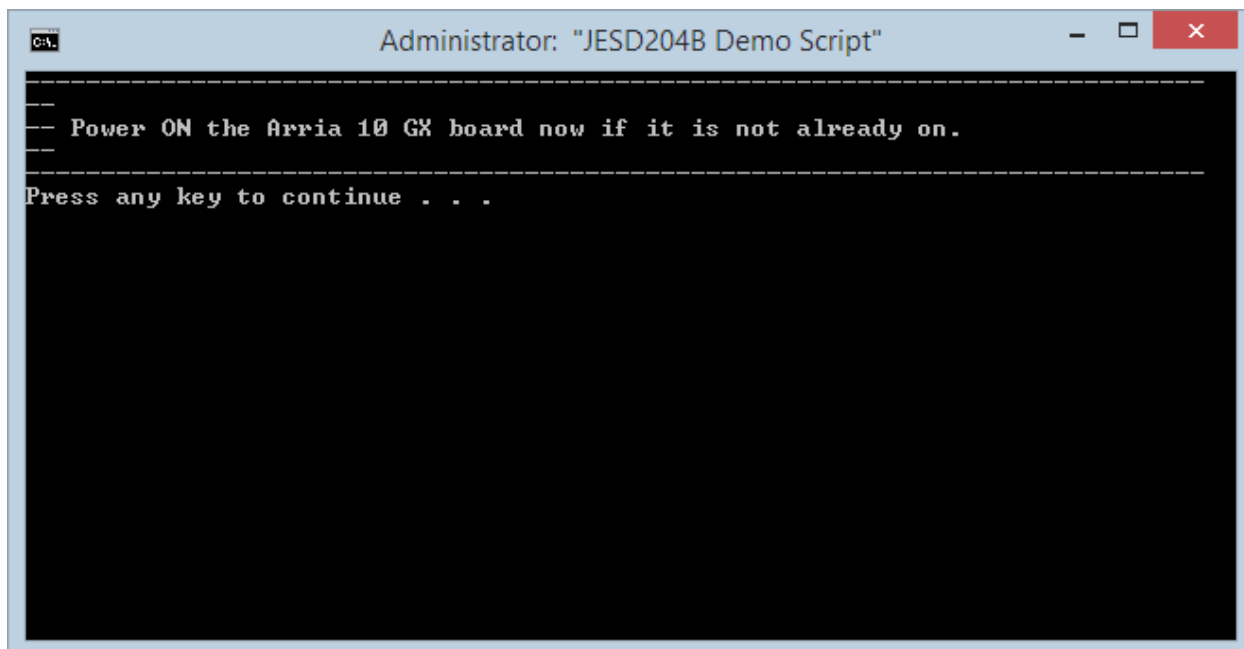


The “JESD204B Demo Script” window will open providing information about the script.

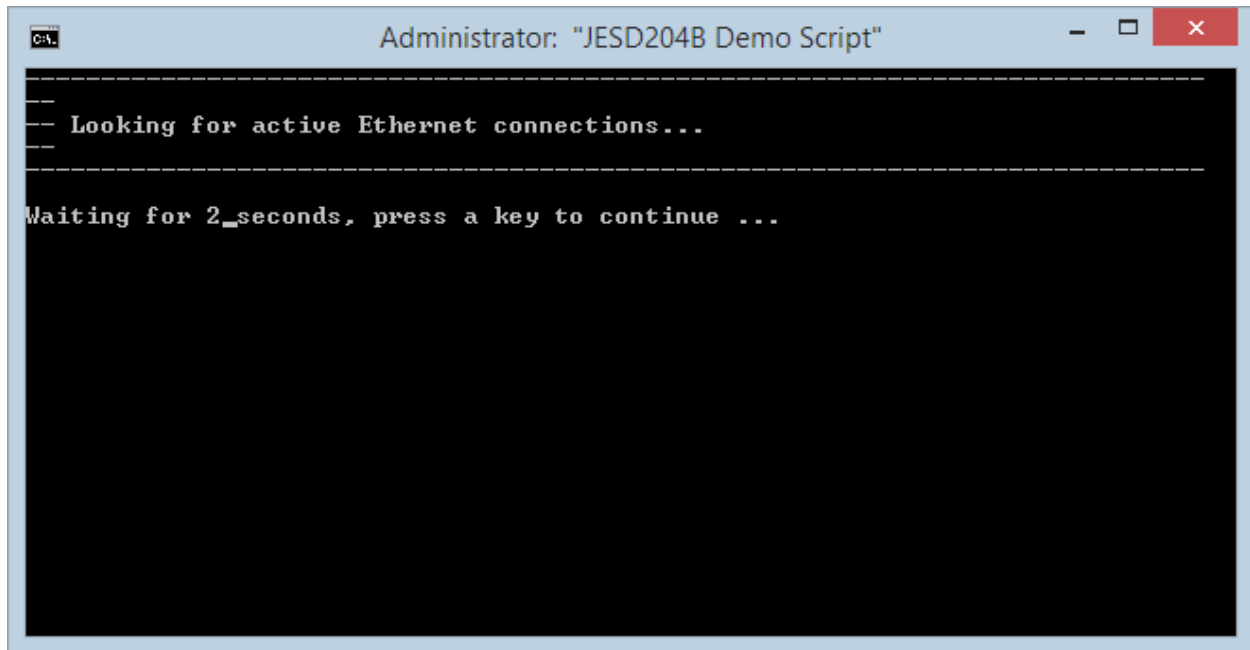
11. **Press any key** on the PC keyboard to continue.



12. Move the power switch, **SW1**, on the Arria 10 GX kit to the **ON** position and **press any key** on the keyboard. Upon power-up, the FPGA will be programmed from the preconfigured CFI flash on the Arria 10 GX kit. The 8 red LEDs will be illuminated upon successful FPGA configuration.

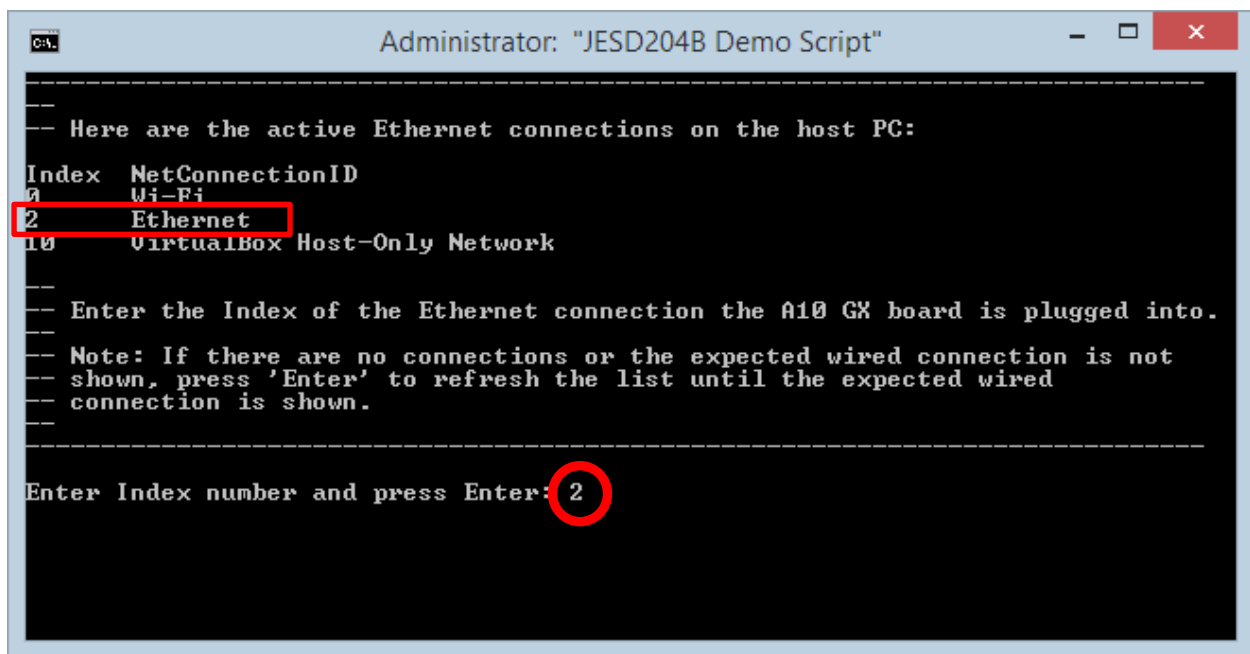


The script will query the PC Ethernet connections for 5 seconds and automatically report back the active interfaces.



Note: The name of the connections may vary. ('Local Area Connection' or 'Ethernet' are common.)

13. Enter the **Index** number of the Ethernet connection which the A10 GX board is plugged into and press **Enter**. (Index = 2 for "Ethernet" in the below example.)



Status messages will appear with information concerning assigning the static IP address to the host and starting the DHCP server. These messages will only appear for a few seconds and then disappear.

```

Administrator: "JESD204B Demo Script"

--
-- Assigning static IP address 192.168.0.1 to local host...
--
-- Starting DHCP server...
--   (You will see a few status messages here for just a moment.)
--
-----
Open DHCP Server Version 1.64 Windows Build 1041 Starting...
Logging: Normal
Warning: No IP Address for DHCP Static Host 00:ff:a4:0e:ef:99 specified
DHCP Range: 192.168.0.2-192.168.0.254/255.255.255.0
Server Name: 114070
Detecting Static Interfaces..
Lease Status URL: http://127.0.0.1:6789
Listening On: 192.168.0.1

```

14. Press the **"CPU RSTn"** button on the Arria 10 GX board forcing the NIOS processor into reset and reloading of the Linux kernel. The button is located between the 2 green and 8 red illuminated LEDs.

```

Administrator: "JESD204B Demo Script"

--
-- Press the "CPU RSTn" button on the Arria 10 GX board.
--   (It is the button between the 2 green and 8 red illuminated LEDs.)
--
--   There will be a short delay (~40 seconds) while the Linux kernel
--   boots and the DHCP address is issued.
--
--   Upon succesful DHCP allocation, verify DHCP server output on last line:
--   "...<Hostxxxxxxxxxxxx> allotted 192.168.0.2 for 36000 seconds"
--
-- Press any key to continue after IP address is allotted . . .
--
-----

```

Additional DHCP server status messages will appear with information concerning assigning the static IP address to the Arria 10 GX. Upon successful DHCP allocation, verify the static IP address 192.168.0.2 was assigned to the Arria 10 GX.

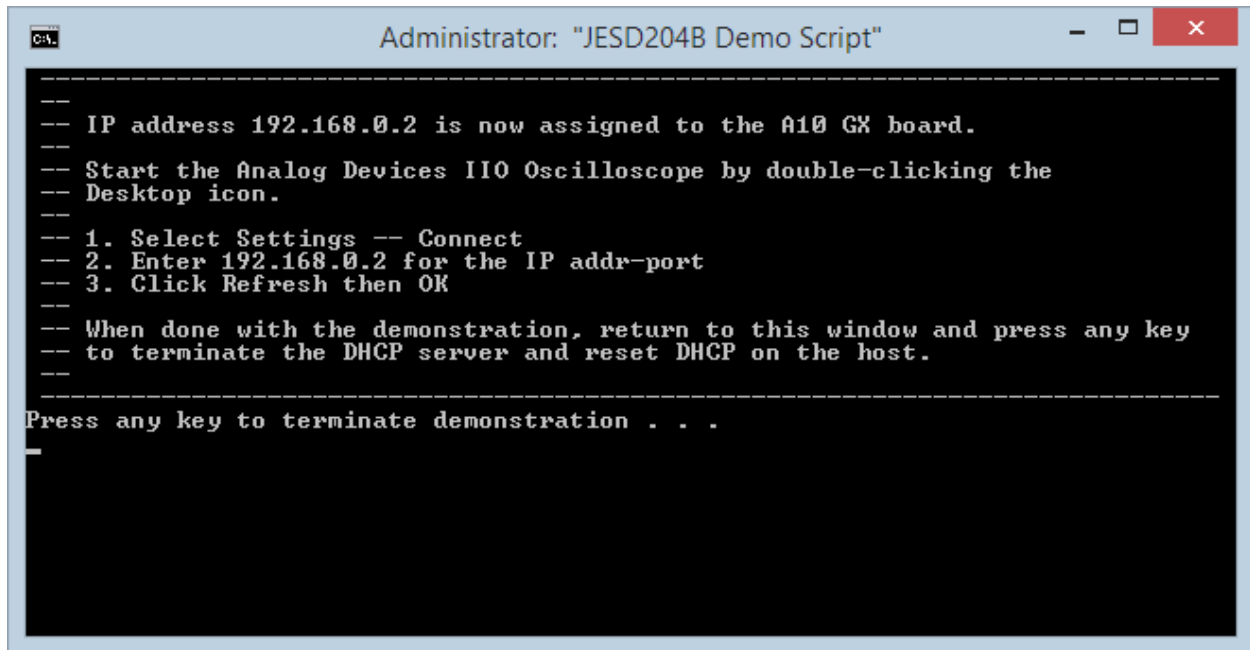
15. **Press any key** once the status message **"Host xx:xx:xx:xx:xx:xx (Hostxxxxxxxxxxxx) allotted 192.168.0.2 for 3600 seconds"** appears. (Note: The "xx" values will vary depending on the host PC MAC address.)

```
Administrator: "JESD204B Demo Script"

-- Press the "CPU RSTn" button on the Arria 10 GX board.
--   <It is the button between the 2 green and 8 red illuminated LEDs.>
--
--   There will be a short delay <~40 seconds> while the Linux kernel
--   boots and the DHCP address is issued.
--
--   Upon succesful DHCP allocation, verify DHCP server output on last line:
--   "...<Hostxxxxxxxxxxxx> allotted 192.168.0.2 for 36000 seconds"
--
-- Press any key to continue after IP address is allotted . . .
-----
Network changed, re-detecting Static Interfaces..
Lease Status URL: http://127.0.0.1:6789
Listening On: 192.168.0.1
Network changed, re-detecting Static Interfaces..
Lease Status URL: http://127.0.0.1:6789
Listening On: 192.168.0.1
DHCPDISCOVER for b2:94:3d:6e:11:8f (<) from interface 192.168.0.1 received
Host b2:94:3d:6e:11:8f <Hostb2943d6e118f> offered 192.168.0.2
DHCPREQUEST for b2:94:3d:6e:11:8f (<) from interface 192.168.0.1 received
Host b2:94:3d:6e:11:8f <Hostb2943d6e118f> allotted 192.168.0.2 for 36000 seconds
```

The script output now provides additional information about running the demonstration.

Note: **DO NOT** press any additional keys while in this window or close this window until the demonstration is completed as it will terminate the DHCP server and sever the connection between the PC and the Arria 10 GX FPGA. See section 8.4.



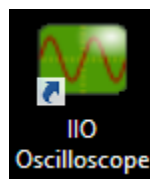
8.2 Connect the IIO Oscilloscope

There are two components to IIO Oscilloscope application that is run on the host PC: configuration and plotting.

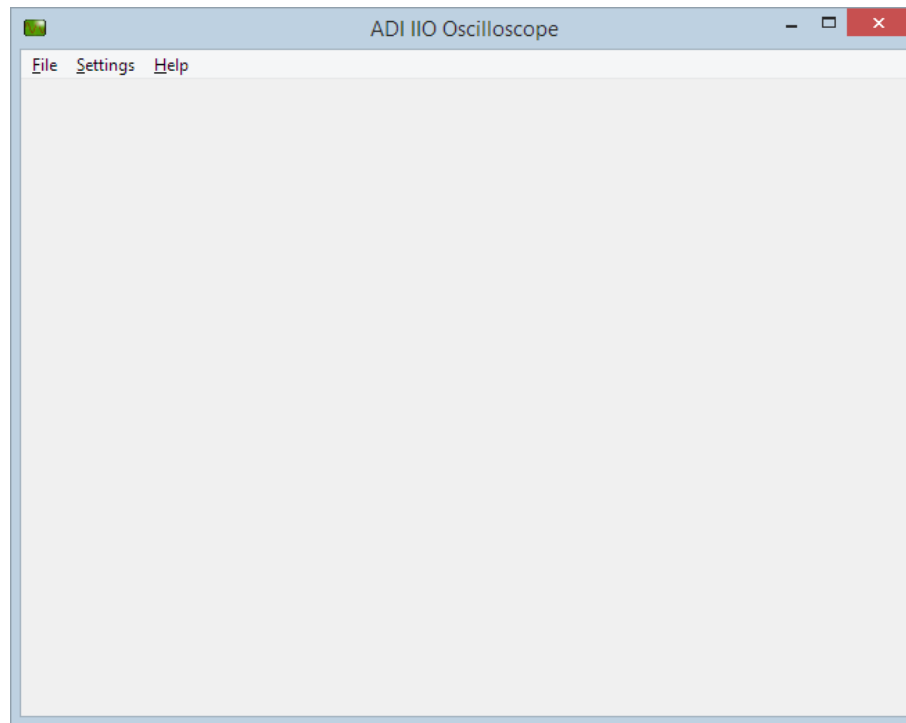
The configuration dialogs provide for choosing the digital signal data that is to be sent to the AD9144 DAC as well as putting the AD9380 ADC into test modes. The AD9144 data is either “tones” that are the output generated by the Arria 10 DDS frequency synthesizer blocks or data files containing digital representations of a signal.

The plotting component of the IIO Oscilloscope application allows the user to view the data received by the Arria 10 after the signal has made a complete loop from the FPGA, through the JESD204B interface to the DAC, through the RF cable back into the ADC, again through a JESD204B interface and back into the FPGA. The plots can display data in the time or frequency domain as well as constellation and cross correlation plots.

1. Launch the **IIO Oscilloscope** application by double-clicking the icon on the Desktop.



2. The IIO Oscilloscope application will start with a blank window.



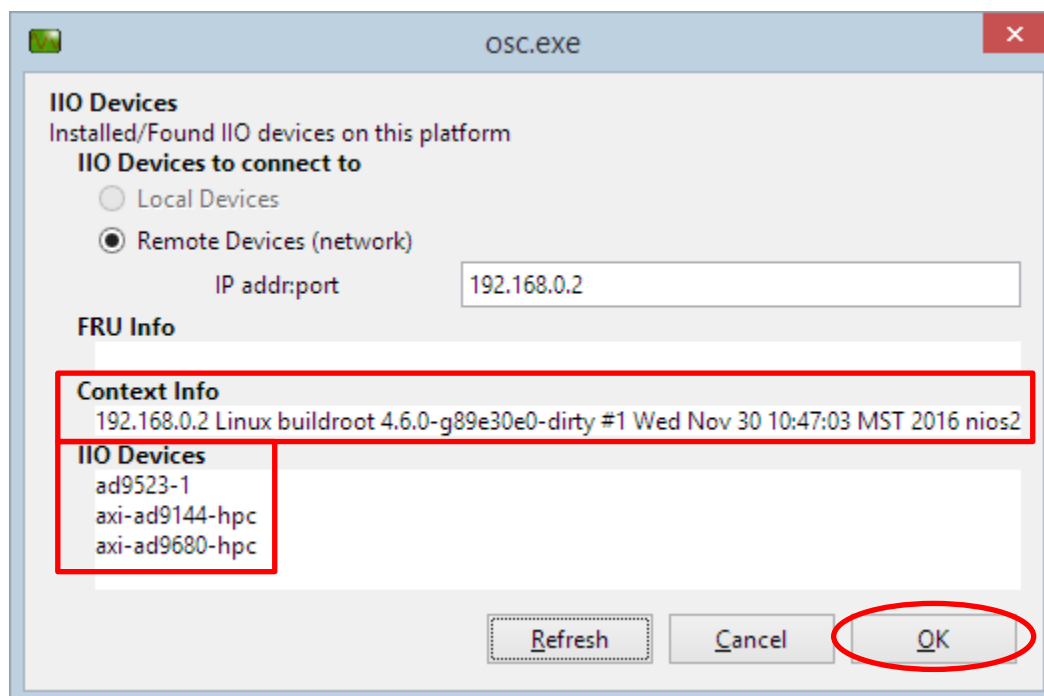
3. Establish a network connection from the IIO Oscilloscope application by selecting **Settings -> Connect**



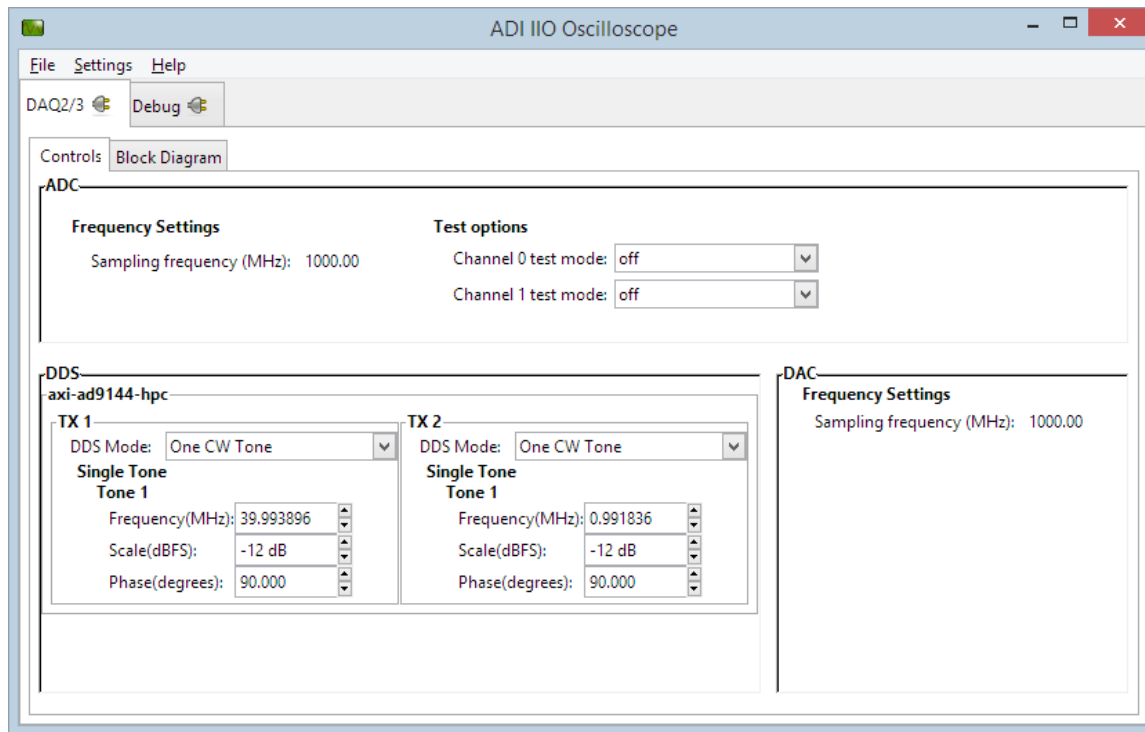
4. Enter the IP address **192.168.0.2** into the **IP addr:port** field and click **Refresh**.



5. A successful connection is verified by observing the **Context Info** and **IIO Devices** fields as shown here. Click **OK**.



The ADI IIO Oscilloscope window will now be populated as shown (some values may differ):



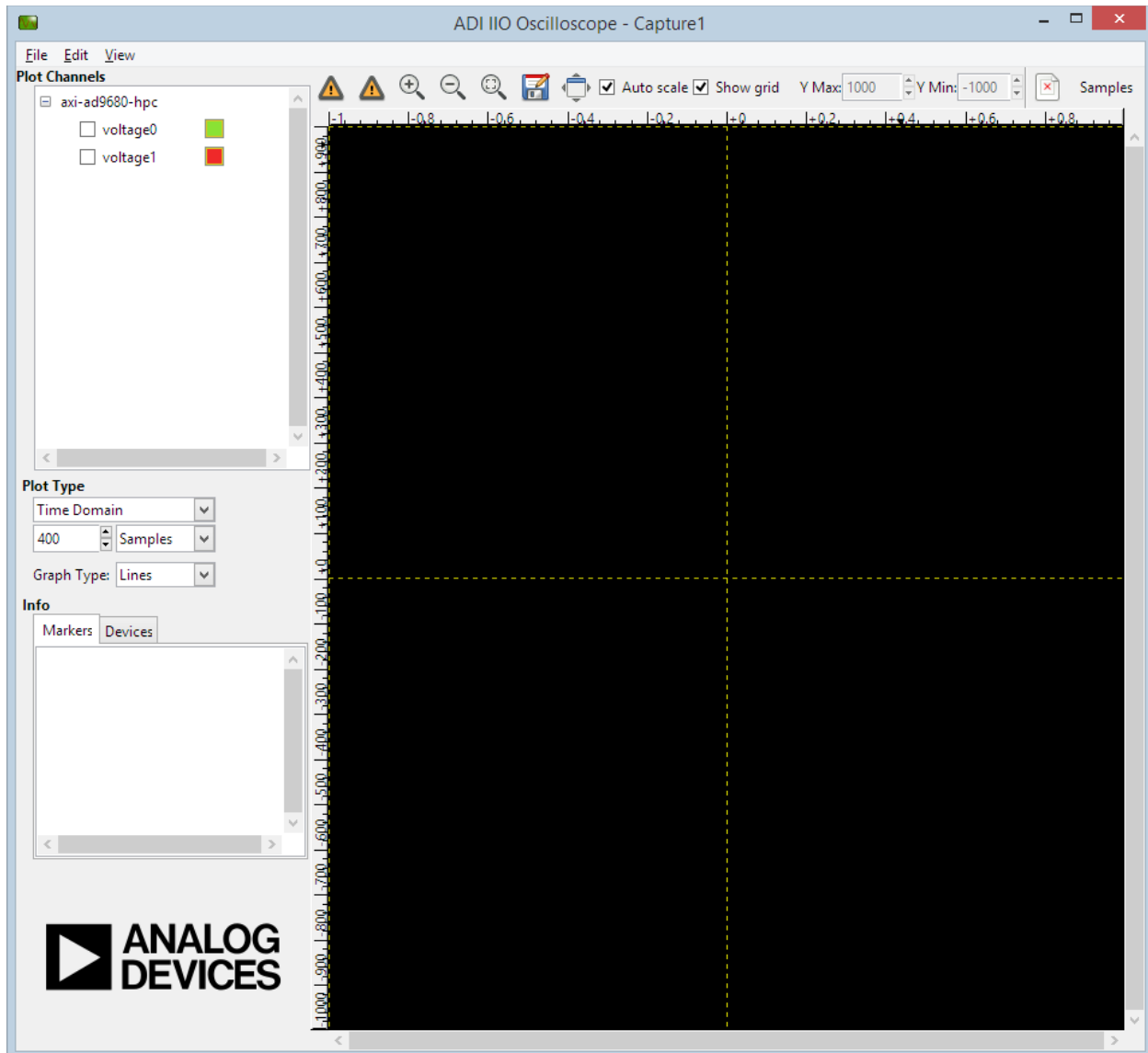
6. Enable the FPGA IP to generate a 40 MHz sine wave for the AD9144 **TX 1** channel by highlighting the **Frequency(Mhz)** field and typing in '**40**'. Set amplitude by typing '**-12**' in **Scale(dBFS)** window then press **Enter**. Note: The frequency requested will be adjusted to closest match possible that the FPGA DDS IP blocks can synthesize.

Note: TX 2 is not enabled for this demo setup. Set 'DDS Mode' to 'Disable' just as a reminder.

Create Plots

1. In the **ADI IIO Oscilloscope** window, using the pull-down menu select **File -> New Plot**.
A new **ADI IIO Oscilloscope – Capture 1** window will open:

8.3

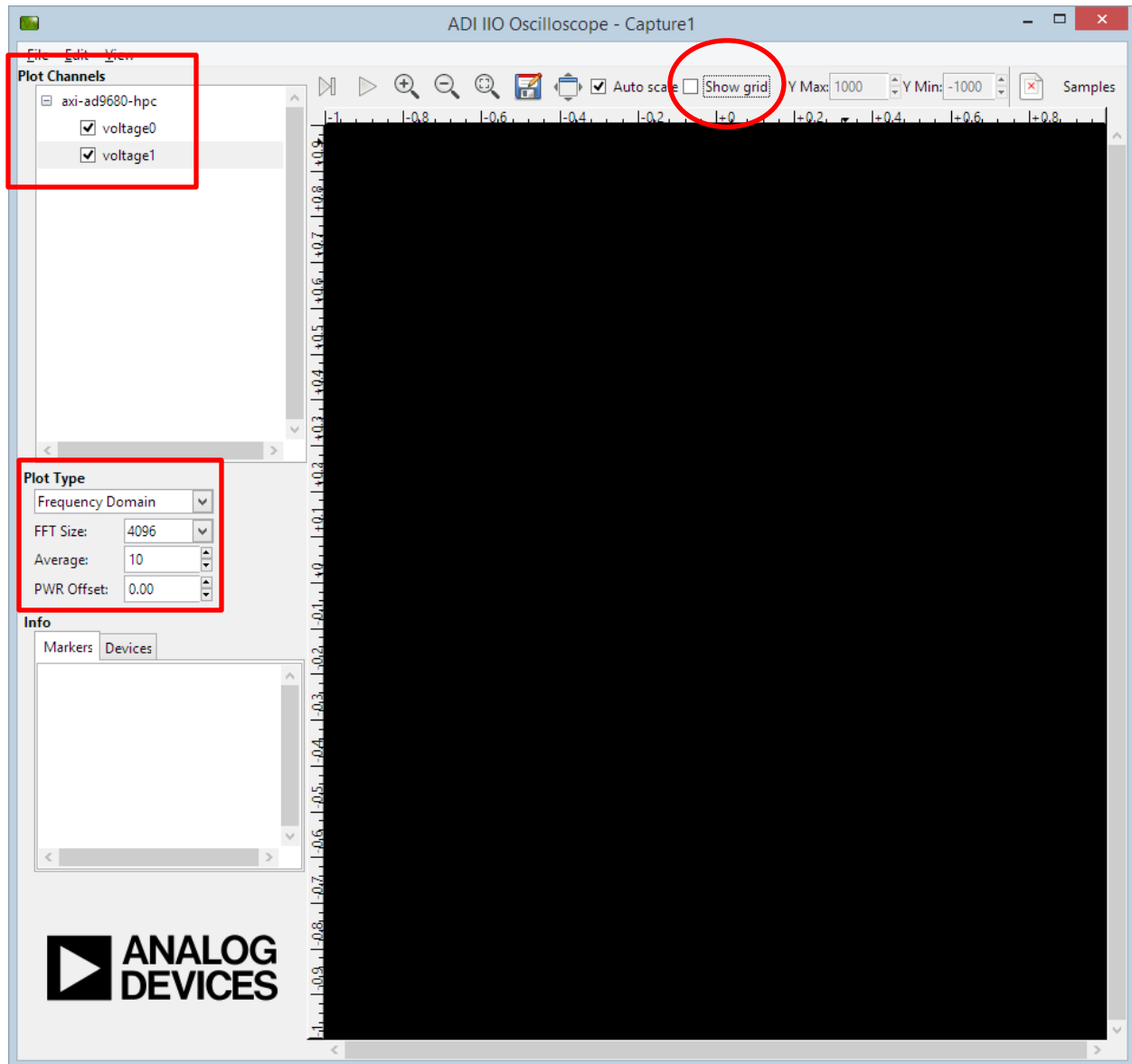



2. Check the selection boxes next to **voltage0** and **voltage1** in the **Plot Channels** panel. These equate to the I & Q channels respectively for the Analog Devices AD9144 DAC.
3. Change the **Plot Type** to **Frequency Domain** via the drop-down list.
4. Change the **FFT Size** to **4096**
5. Set the **Average** to **10**

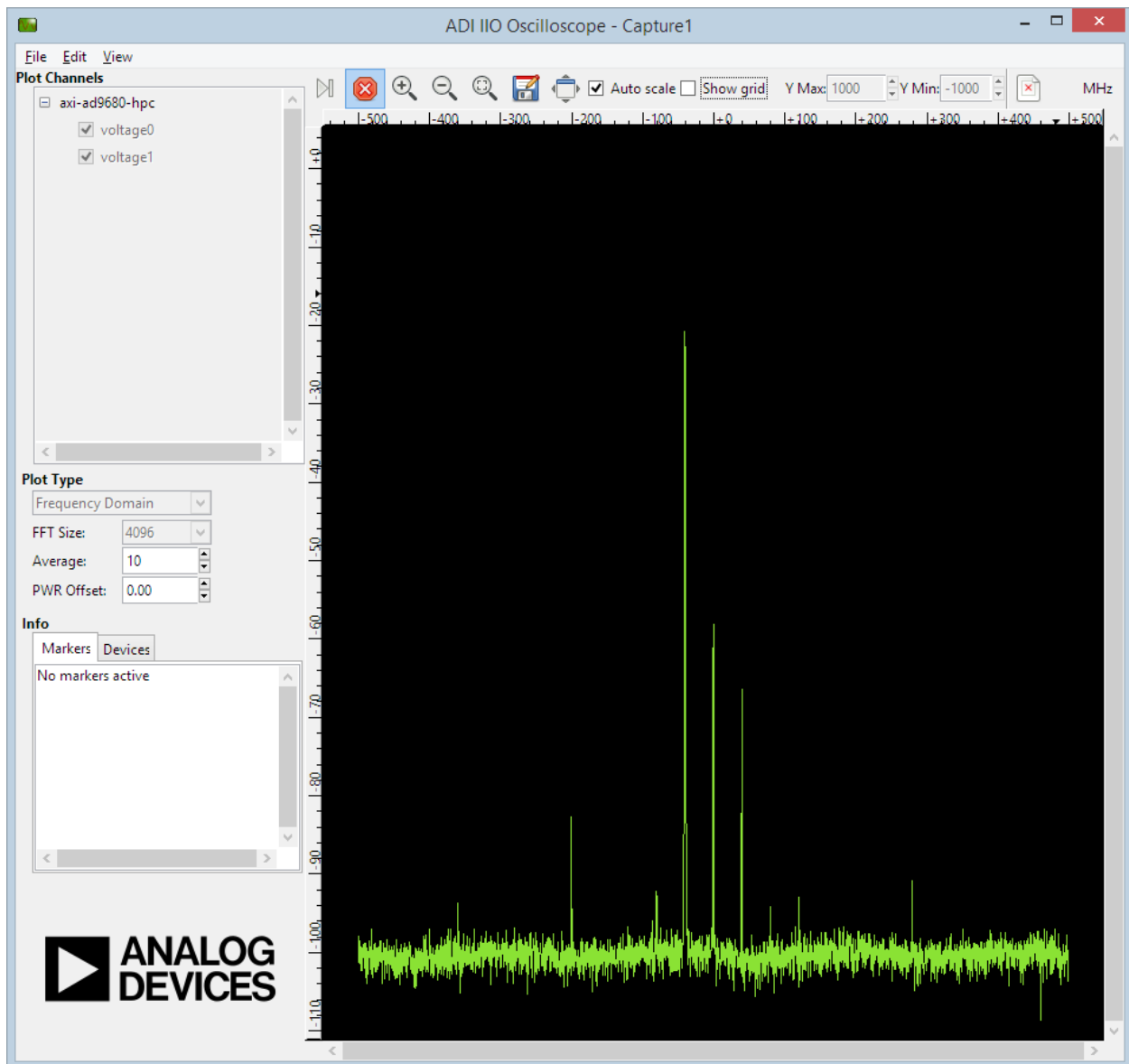
User Guide

- Uncheck the **Show Grid** selection box near the top of the window.

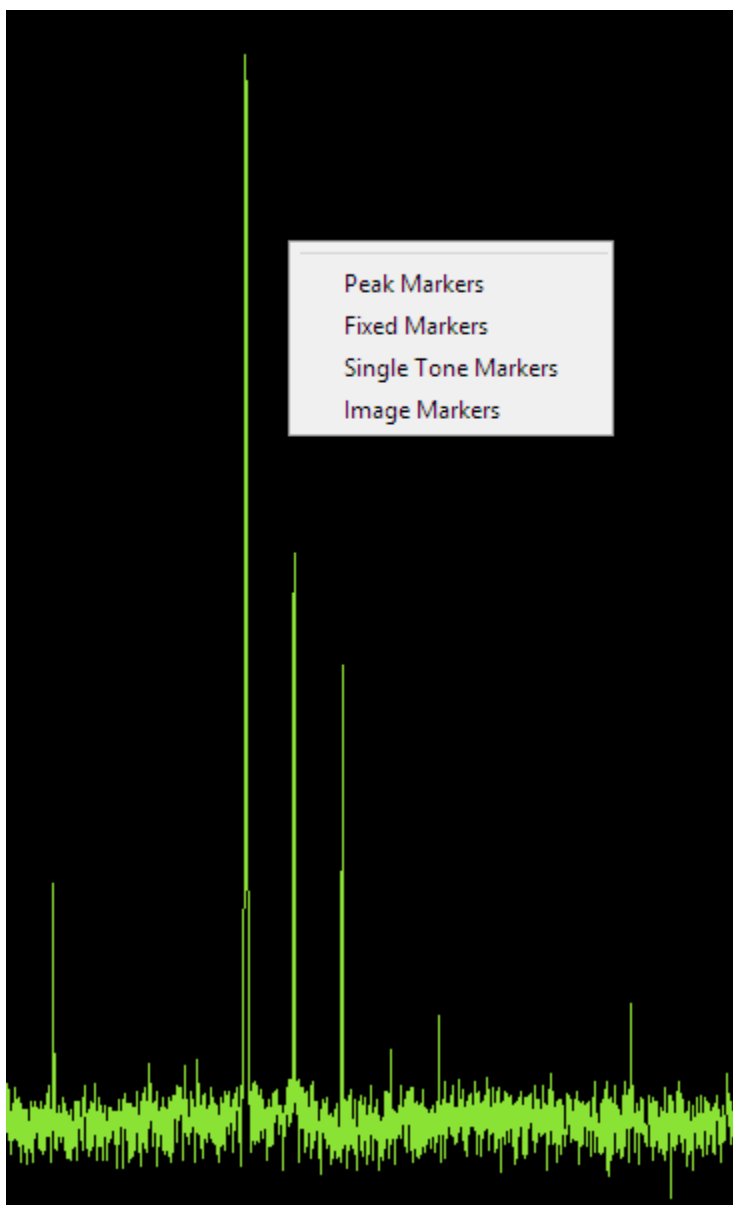
The ADI IIO Oscilloscope – Capture 1 window should now look like this:



- Click on the “play” button  near the top of the window to start the oscilloscope capture.

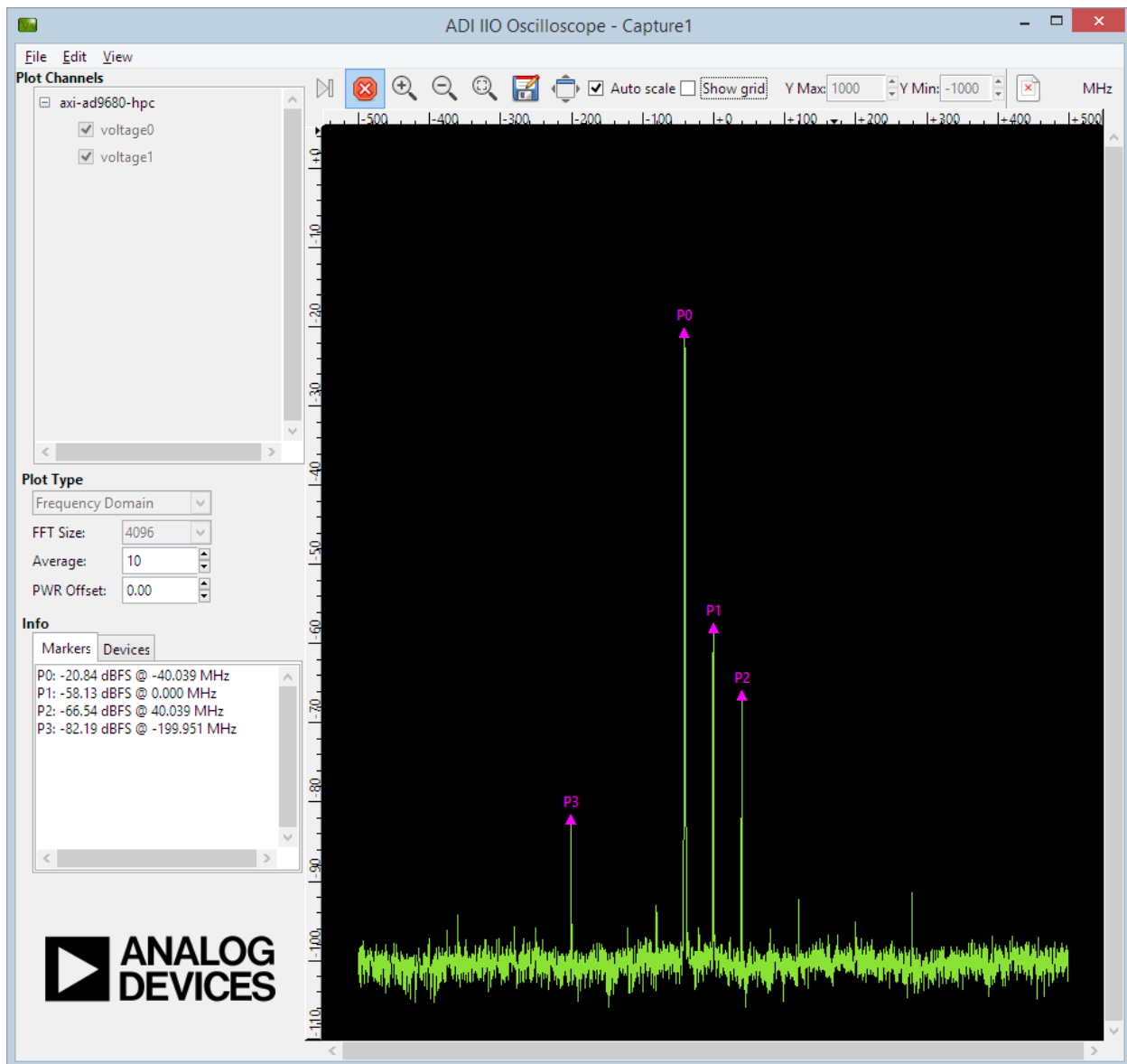



8. Set markers on the plot by right-clicking in the middle of the plot and selecting the marker of choice. (**Peak Markers** is a good choice...)



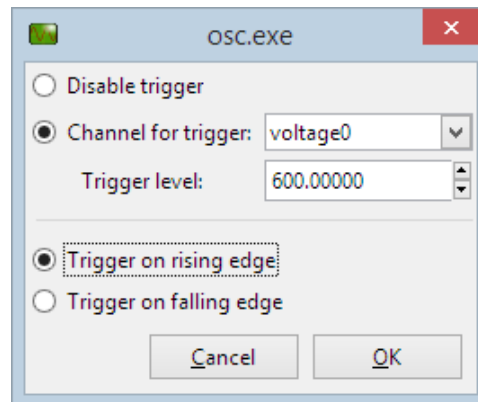
User Guide

Once markers are set, they will persist on the plot and the details of each marker are displayed in the **Info** panel.

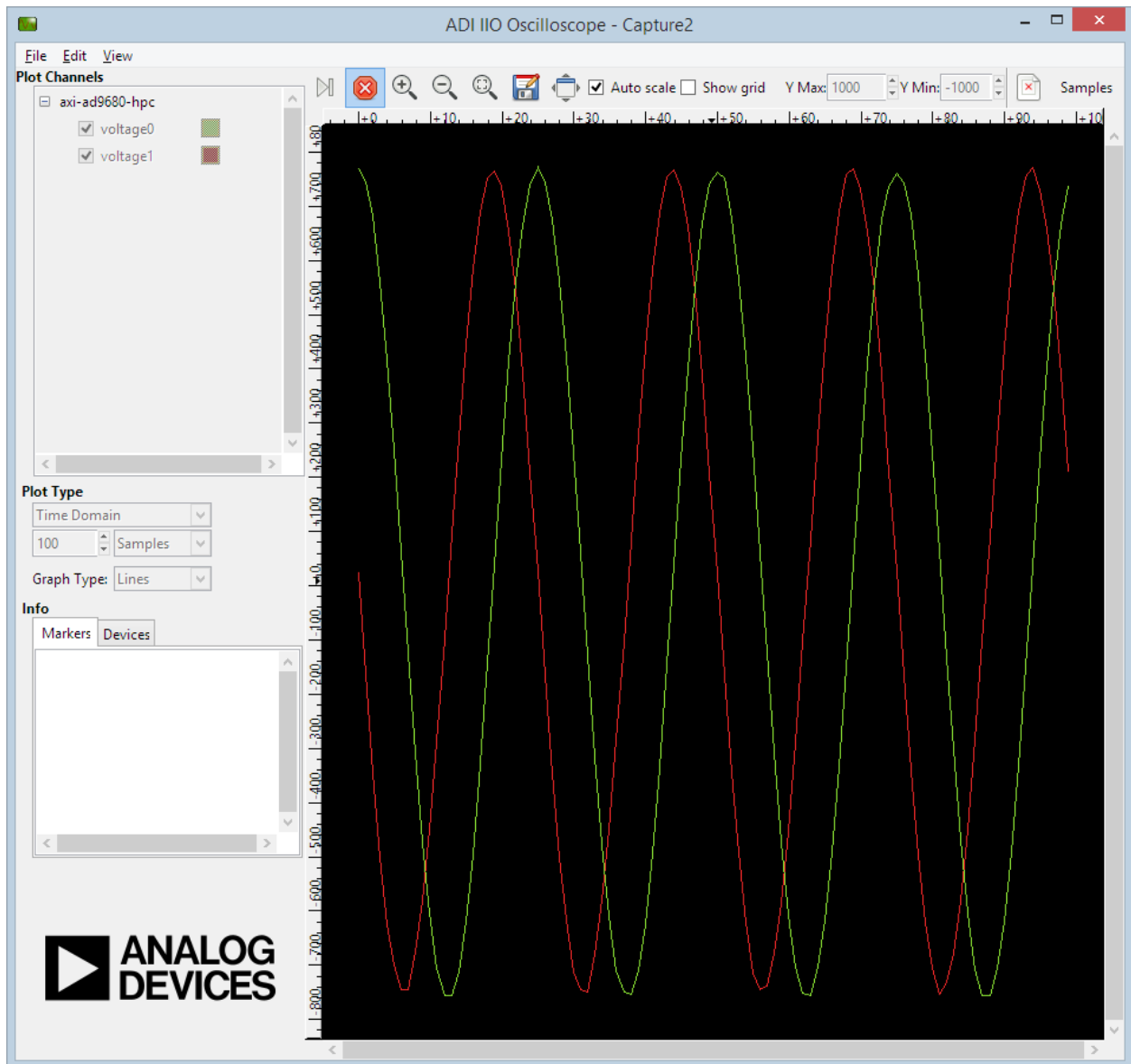



9. Add a second plot by clicking the  icon in the upper-right corner of the Capture 1 window. Set up the new **ADI IIO Oscilloscope – Capture 2** window to plot in the **Time Domain** with **100** Samples on both voltage channels.
10. Right-click on the **axi-ad9680-hpc** in the **Plot Channels** panel and select **Trigger settings**.

11. Select the radio button for **Channel for trigger: voltage0** and set the **Trigger level** to **600**. Click **OK**.



12. Click “play” to observe the new time domain plot.



13. Stop the capture by clicking the “stop” icon  in each of the capture windows as desired. (A plot must be stopped in order to make changes to its capture settings.)
14. From here, play around with different settings in the ADI IIO Oscilloscope ADC/DAC control window and change the **DDS Single Tone: Tone 1 Frequency (MHz)** and observe the changes in the plots.
15. Change the **DDS Mode** to **TWO CW Tones** and observe 2 frequencies in the plots.

Feel free to poke around and play with any of the settings and plotting options. Note that due to the large amount of high speed data being received by the FPGA and plotted, sometimes the plots momentarily hang. They do return to functioning once you stop and restart the plot(s).

Shutdown

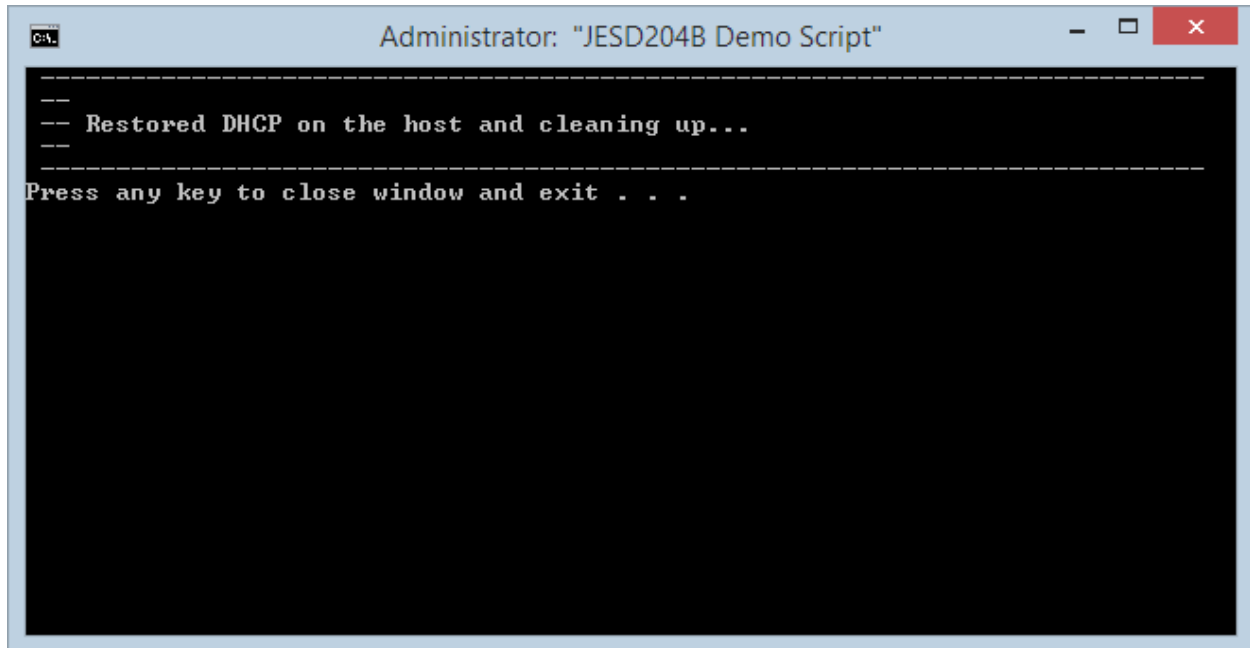
1. When done with the demo, **close** the **Capture** windows and **close** the **ADI IIO Oscilloscope** window.
2. **IMPORTANT** – To properly reset the host PC Ethernet network interface to DHCP, in the **JESD204B Demo Script** window running the DHCP server, **press any key** to properly terminate the script. This will force the DHCP server to stop and the batch file will restore the DHCP settings on the host.

8.4

```
Administrator: "JESD204B Demo Script"

--
-- IP address 192.168.0.2 is now assigned to the A10 GX board.
--
-- Start the Analog Devices IIO Oscilloscope by double-clicking the
-- Desktop icon.
--
-- 1. Select Settings -- Connect
-- 2. Enter 192.168.0.2 for the IP addr-port
-- 3. Click Refresh then OK
--
-- When done with the demonstration, return to this window and press any key
-- to terminate the DHCP server and reset DHCP on the host.
--
-----
Press any key to terminate demonstration . . .
_
```

3. **Press any key** to close the final window completing the demonstration script.



9. Building the Hardware and Software Reference Designs

This section describes how to obtain the source code and build the hardware and software portions of the reference design. These reference designs provide a starting point for the user to create their own custom designs.

The designs were created and are maintained by Analog Devices Inc.

Hardware reference design resources

The following Analog Devices Inc. resources are available

9.1

- HDL User Guide - <https://wiki.analog.com/resources/fpga/docs/hdl>
- Latest Release Notes - <https://github.com/analogdevicesinc/hdl/releases>
- HDL Help & Support - <https://ez.analog.com/community/fpga>

9.2 Install the Intel PSG development tools

The Quartus II Prime 16.1 Standard or Pro tools are required to build the hardware reference design

- Download the tools from <https://www.altera.com/downloads/download-center.html> and install
- Obtain licenses for Quartus and the Intellectual Property cores included in the reference design.

Build the hardware reference design

- open a **nios2 16.1 command shell**
- 9.3 ○ navigate to a selected directory
- git clone <https://github.com/analogdevicesinc/hdl.git>
- cd hdl
- git checkout **hdl_2017_r1**

```

/cygdrive/c/arrow_hsdk/hdl
-----
Altera Nios2 Command Shell [GCC 4]
Version 16.1, Build 203
-----

a08473@LM1310 /cygdrive/c/intelFPGA/16.1
$ cd "C:\arrow_hsdk"

a08473@LM1310 /cygdrive/c/arrow_hsdk
$ git clone https://github.com/analogdevicesinc/hdl.git
Cloning into 'hdl'...
remote: Counting objects: 40926, done.
remote: Compressing objects: 100% (165/165), done.
remote: Total 40926 (delta 102), reused 123 (delta 54), pack-reused 40704
Receiving objects: 100% (40926/40926), 11.05 MiB | 3.50 MiB/s, done.
Resolving deltas: 100% (28930/28930), done.

a08473@LM1310 /cygdrive/c/arrow_hsdk
$ cd hdl

a08473@LM1310 /cygdrive/c/arrow_hsdk/hdl
$ git checkout hdl_2017_r1
Switched to a new branch 'hdl_2017_r1'
Branch hdl_2017_r1 set up to track remote branch hdl_2017_r1 from origin.

a08473@LM1310 /cygdrive/c/arrow_hsdk/hdl
$

```

- **make -C projects/daq2/a10gx**
- The project can take up to an hour to build. All build information is saved to the **daq2_a10gx_quartus.log** test file

```

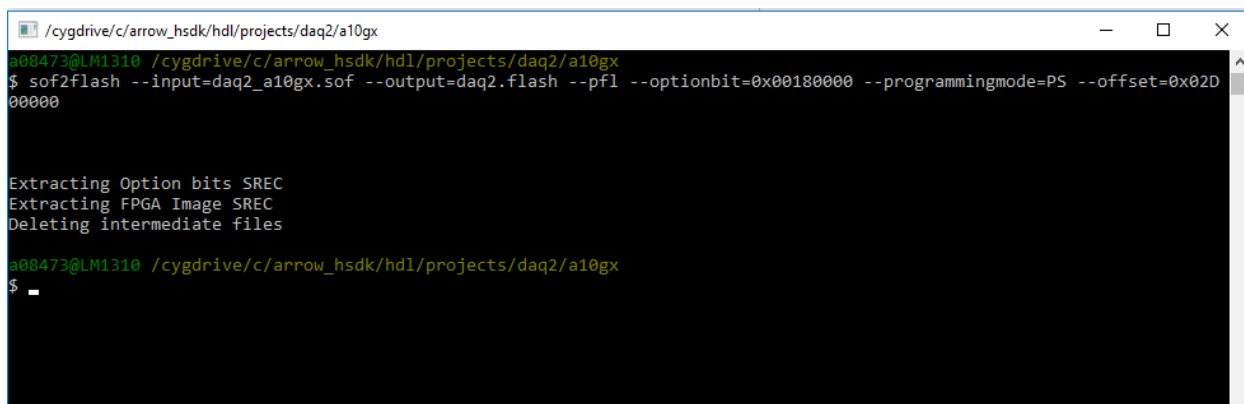
/cygdrive/c/arrow_hsdk/hdl

a08473@LM1310 /cygdrive/c/arrow_hsdk/hdl
$ make -C projects/daq2/a10gx
make: Entering directory `C:/arrow_hsdk/hdl/projects/daq2/a10gx'
rm -rf *.log *_INFO.txt *_dump.txt db *.asm.rpt *.done *.eda.rpt *.fit.* *.map.* *.sta.* *.qsf *.qpf *.qws *.sof *.cdf *
.sld *.qdf hc_output system_bd hps_isw_handoff hps_sdram*.csv *ddr3*.csv incremental_db reconfig_mif *.sopcinfo *.jdi
*.pin *_summary.csv *.dpf
quartus_sh --64bit -t system_project.tcl >> daq2_a10gx_quartus.log 2>&1

```

Create the hardware flash image

- From the current hdl directory cd to projects/daq2/a10gx
- 9.4
- `sof2flash --input=daq2_a10gx.sof --output=daq2.flash --pfl --optionbit=0x00180000 --programmingmode=PS --offset=0x02D00000`



```
/cygdrive/c/arrow_hsdk/hdl/projects/daq2/a10gx
a08473@LM1310 /cygdrive/c/arrow_hsdk/hdl/projects/daq2/a10gx
$ sof2flash --input=daq2_a10gx.sof --output=daq2.flash --pfl --optionbit=0x00180000 --programmingmode=PS --offset=0x02D00000

Extracting Option bits SREC
Extracting FPGA Image SREC
Deleting intermediate files

a08473@LM1310 /cygdrive/c/arrow_hsdk/hdl/projects/daq2/a10gx
$
```

Use the Board Update Portal to upload the image to flash as described in chapter 5

Software reference design resources

The following Analog Devices Inc. resources are available

- Nios II Linux <https://wiki.analog.com/resources/tools-software/linux-drivers/platforms/nios2>

9.5

Build Linux for the Nios II processor

The following instructions assume that the user has a Linux host available. The build for the Arrow High Speed Development kit was tested using a host with Ubuntu 14.04 LTS

9.6

- open shell on the Linux host. cd to a <project_directory>
- git clone <https://github.com/analogdevicesinc/linux.git>

```
soceds@ubuntu: ~/daq2_project
soceds@ubuntu:~/daq2_project$ git clone https://github.com/analogdevicesinc/linux.git
Cloning into 'linux'...
remote: Counting objects: 5768417, done.
remote: Compressing objects: 100% (12/12), done.
Receiving objects: 0% (33823/5768417), 15.01 MiB | 155.00 KiB/s
```

- cd linux/
- git checkout altera_4.9

```
soceds@ubuntu: ~/daq2_project_4.9/linux
soceds@ubuntu:~/daq2_project_4.9$ cd linux
soceds@ubuntu:~/daq2_project_4.9/linux$ git checkout altera_4.9
M      arch/nios2/configs/adi_nios2_defconfig
soceds@ubuntu:~/daq2_project_4.9/linux$
```

get the nios2 root file system

- wget http://wiki.analog.com/_media/resources/tools-software/linux-drivers/platforms/nios2/rootfs_nios2.cpio.gz -P arch/nios2/boot/rootfs_cpio.gz

move the rootfs to the appropriate folder

- mv arch/nios2/boot/rootfs_cpio.gz/rootfs_nios2.cpio.gz
arch/nios2/boot/rootfs.cpio.gz


```
soceds@ubuntu: ~/daq2_project_4.9/linux
soceds@ubuntu:~/daq2_project_4.9/linux$ wget http://wiki.analog.com/_media/resources/tools-software/linux-drivers/platforms/nios2/rootfs_nios2.cpio.gz -P arch/nios2/boot/rootfs_cpio.gz
Connecting to wiki.analog.com (wiki.analog.com)|37.61.203.15|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3098543 (3.0M) [application/octet-stream]
Saving to: 'arch/nios2/boot/rootfs_cpio.gz/rootfs_nios2.cpio.gz'

100%[=====] 3,098,543 1.42MB/s in 2.1s

2017-11-08 09:45:44 (1.42 MB/s) - 'arch/nios2/boot/rootfs_cpio.gz/rootfs_nios2.cpio.gz' saved [3098543/3098543]

soceds@ubuntu:~/daq2_project_4.9/linux$ mv arch/nios2/boot/rootfs_cpio.gz/rootfs_nios2.cpio.gz arch/nios2/boot/rootfs.cpio.gz
```

- **get nios toolchain from codesourcery.** download advanced package IA32 GNU/Linux
- <https://sourcery.mentor.com/GNUToolchain/release3275>



Mentor Embedded Portal

ALREADY REGISTERED?

Email

Password

[Forget your password?](#)

NOT REGISTERED?

Sourcery CodeBench Lite 2016.11-32 for Nios II GNU/Linux

Status: Release

This is a fully-validated release.

This release was made on 20 December 2016.

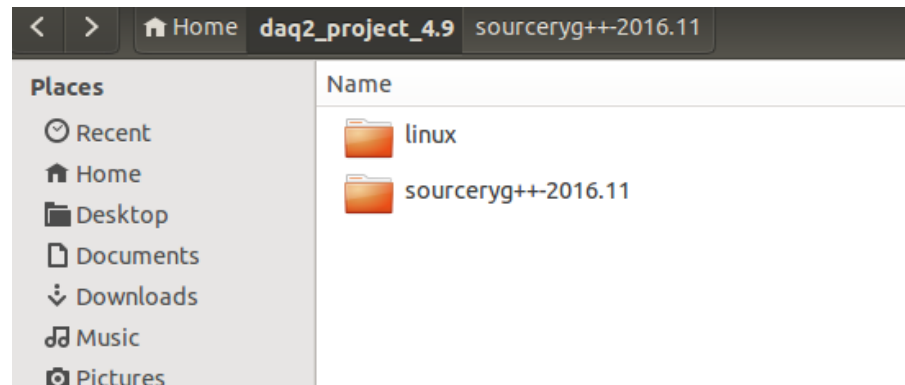
Software

Download	MD5 Checksum
Recommended Packages	
IA32 GNU/Linux Installer	03e8a702dac8edfd1641c46bc8e95564
IA32 Windows Installer	04dad3ec1f73da0a67e6a9d5d4e14b54
Advanced Packages	
IA32 GNU/Linux TAR	0197e48f1ccc8f23fa3388d7daa17907
IA32 Windows TAR	3e49c64093e5fa8948622f8422e140b9
Source TAR	700cb35bfb9488de5281268a1c161039

What's in this release?

The datasheet provides information about key components of Sourcery CodeBench Lite 2016.11-32.

- untar into the <project directory>



- export ARCH=nios2
- export CROSS_COMPILE=<project directory>/sourceryg++2016.11/bin/nios2-linux-gnu-

```
soceds@ubuntu: ~/daq2_project_4.9/linux
soceds@ubuntu:~/daq2_project_4.9/linux$ export ARCH=nios2
soceds@ubuntu:~/daq2_project_4.9/linux$
soceds@ubuntu:~/daq2_project_4.9/linux$
soceds@ubuntu:~/daq2_project_4.9/linux$ export CROSS_COMPILE=../sourceryg++-2016
.11/bin/nios2-linux-gnu-
soceds@ubuntu:~/daq2_project_4.9/linux$
```

- make adi_nios2_defconfig
- cp arch/nios2/boot/dts/a10gx_daq2.dts arch/nios2/boot/devicetree.dts

```
soceds@ubuntu: ~/daq2_project_4.9/linux
soceds@ubuntu:~/daq2_project_4.9/linux$ make adi_nios2_defconfig
#
# configuration written to .config
#
soceds@ubuntu:~/daq2_project_4.9/linux$ cp arch/nios2/boot/dts/a10gx_daq2.dts ar
ch/nios2/boot/devicetree.dts
soceds@ubuntu:~/daq2_project_4.9/linux$
```

- make zImage

```
soceds@ubuntu: ~/daq2_project_4.9/linux
soceds@ubuntu:~/daq2_project_4.9/linux$ make zImage
scripts/kconfig/conf  --silentoldconfig Kconfig
CHK      include/config/kernel.release
CHK      include/generated/uapi/linux/version.h
CHK      include/generated/utsrelease.h
CC      kernel/bounds.s
CHK      include/generated/bounds.h
CHK      include/generated/timeconst.h
CC      arch/nios2/kernel/asm-offsets.s
```

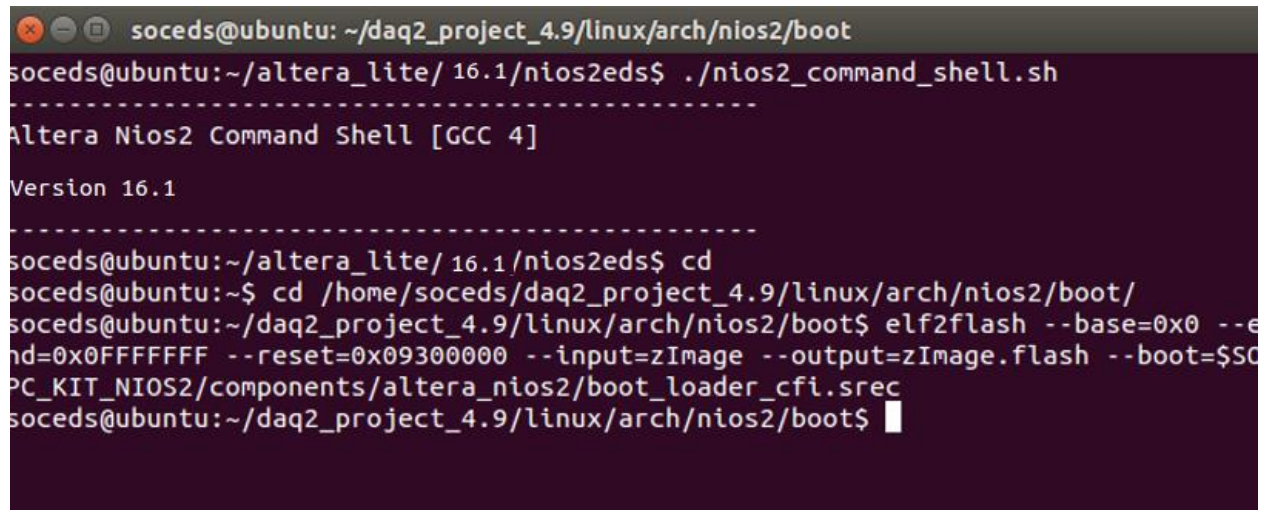
- the zImage file is located in arch/nios2/boot

```
soceds@ubuntu: ~/daq2_project_4.9/linux
LD      arch/nios2/lib/built-in.o
LD      virt/lib/built-in.o
LD      virt/built-in.o
LD      vmlinux.o
MODPOST vmlinux.o
GEN      .version
CHK      include/generated/compile.h
UPD      include/generated/compile.h
CC      init/version.o
LD      init/built-in.o
KSYM     .tmp_kallsyms1.o
KSYM     .tmp_kallsyms2.o
LD      vmlinux
SYSMAP   System.map
OBJCOPY  arch/nios2/boot/vmlinux.bin
GZIP     arch/nios2/boot/vmlinux.gz
LDS      arch/nios2/boot/compressed/vmlinux.lds
AS       arch/nios2/boot/compressed/head.o
CC       arch/nios2/boot/compressed/misc.o
LD       arch/nios2/boot/compressed/piggy.o
LD       arch/nios2/boot/compressed/vmlinux
OBJCOPY  arch/nios2/boot/zImage
Kernel: arch/nios2/boot/zImage is ready
soceds@ubuntu:~/daq2 project 4.9/linux$
```

Create the software flash image

9.7

- open a nios2 16.1 command shell
- cd <project_directory>/linux/arch/nios2/boot
- elf2flash --base=0x0 --end=0xFFFFFFFF --reset=0x09300000 --input=zImage --output=zImage.flash --boot=\$SOPC_KIT_NIOS2/components/altera_nios2/boot_loader_cfi.srec



```
soceds@ubuntu: ~/daq2_project_4.9/linux/arch/nios2/boot
soceds@ubuntu:~/altera_lite/16.1/nios2eds$ ./nios2_command_shell.sh
-----
Altera Nios2 Command Shell [GCC 4]
Version 16.1
-----
soceds@ubuntu:~/altera_lite/16.1/nios2eds$ cd
soceds@ubuntu:~$ cd /home/soceds/daq2_project_4.9/linux/arch/nios2/boot/
soceds@ubuntu:~/daq2_project_4.9/linux/arch/nios2/boot$ elf2flash --base=0x0 --end=0xFFFFFFFF --reset=0x09300000 --input=zImage --output=zImage.flash --boot=$SOPC_KIT_NIOS2/components/altera_nios2/boot_loader_cfi.srec
soceds@ubuntu:~/daq2_project_4.9/linux/arch/nios2/boot$
```

Use the Board Update Portal to upload the image to flash as described in chapter 5.

10. APPENDIX

More Information

Intel Arria 10 GX FPGA

<https://www.altera.com/products/fpga/arria-series/arria-10/overview.html>

10.1 Intel Arria 10 GX FPGA Development Kit

https://www.altera.com/products/boards_and_kits/dev-kits/altera/kit-a10-gx-fpga.html

Analog Devices AD9144 Quad 16-bit 2.8GSPS DAC

<http://www.analog.com/en/products/digital-to-analog-converters/da-converters/AD9144.html>

Analog Devices 9680 Dual 14-bit 1.25GSPS ADC

<http://www.analog.com/en/products/analog-to-digital-converters/high-speed-ad-10msps/ad9680.html>

Analog Devices AD-FMCDQA2-EBZ Evaluation Board

<http://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/eval-ad-fmcdqa2-ebz.html>

Online resources

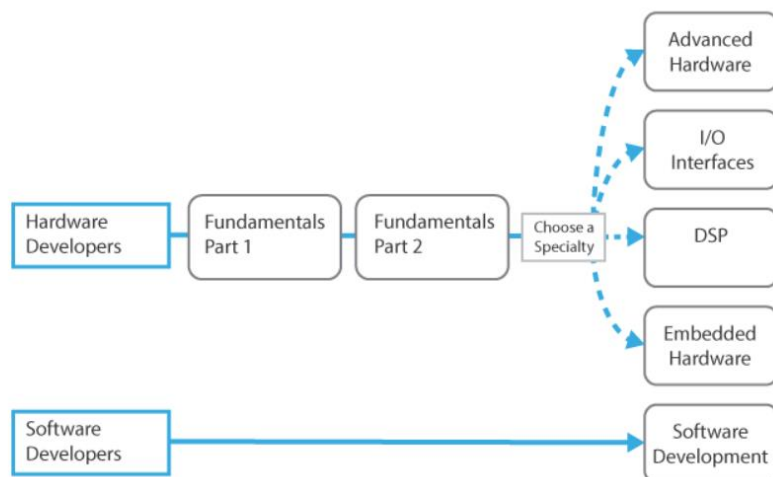
Intel PSG Training Curricula

10.2

<https://www.altera.com/support/training/curricula.html>

Intel FPGA Curricula Introduction

The table below shows the curriculum flow for each of the specialties listed on the menu to the left. Software engineers may proceed directly to the courses listed in the Software Development curriculum. Hardware engineers should first review the courses listed under the Fundamentals Part 1 and Part 2 category and take any that are unfamiliar before moving on to the courses listed under their specialty.



Intel PSG DSP Overview

<https://www.altera.com/solutions/technology/dsp/overview.html>

Intel HLS Compiler Overview

<https://www.altera.com/products/design-software/high-level-design/intel-hls-compiler/overview.html>

Intel FPGA SDK for OPENCL

<https://www.altera.com/products/design-software/embedded-software-developers/opencl/overview.html>

11. REVISION HISTORY

Revision	Changes	Date
1.0	Pre-release	2/8/2017
2.0	Initial release	6/7/2017
3.0	Added instructions on downloading and flashing image from Github	10/17/2017