

BeScope Dual-Channel Oscilloscope User Guide and Hardware Reference Guide

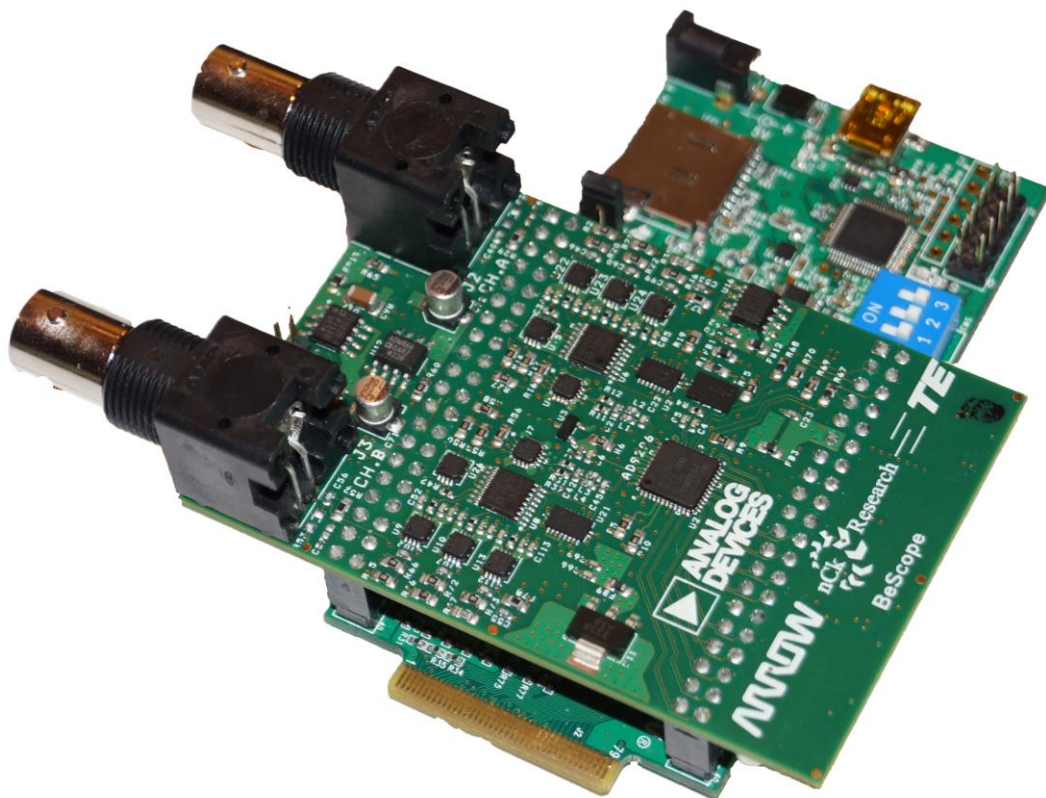


Table of Contents

Table of Contents	2
1 Introduction	4
2 Getting Started	4
2.1 Download and install the Quartus Programmer	4
2.2 Download the BeScope installation package	10
2.3 Assemble the BeScope	10
2.4 Program the BeMicro CV FPGA	11
2.5 Launch the System Console Socket Server	13
2.6 Launch the BeScope user interface	14
2.7 Verify the BeScope Operation	16
2.8 Explore the BeScope user interface	16
2.9 Batch File for easy launch	16
3 BeScope System technical overview	17
3.1 BeScope Analog Board	18
3.1.1 Component Placement Diagram	18
3.1.2 Board Components	18
3.1.3 Analog Signal Chain Description	20
3.1.3.1 Front-End Divider	20
3.1.3.2 Gain Stages	21
3.1.3.3 Low Capacitance Gain Mux	22
3.1.3.4 Single-ended to Differential Amp and Anti-aliasing Filter	23
3.1.3.5 Analog to Digital Converter (ADC)	24
3.1.4 Board Power	25
3.1.5 Clock & Timing	26
3.1.6 Miscellaneous	26
3.1.7 40 Pin Headers	26
3.1.8 Power Supply	29
3.1.9 On-board Clock Source	29
3.1.10 Board Components Reference	30
3.2 BeMicro CV Development Board	31
3.2.1 Qsys System	31
3.2.1.1 JTAG Master	33
3.2.1.2 Register Control	33

3.2.1.3	Trigger Control	34
3.2.1.4	ADC Interface	35
3.3	System Console Socket Server	35
3.4	BeScope user interface	35
4	Additional Information	36
4.1	Board Revision History	36

1 Introduction

This document describes the hardware, firmware, and software features of the BeScope Oscilloscope. The BeScope is a dual channel analog oscilloscope system, including passive scope probe, and graphical user interface. The BeScope supports complex waveform analysis in real time using high-speed FPGA logic. Waveforms are displayed on an easy-to-use Windows® Graphical User Interface (GUI).

2 Getting Started

In this chapter, we will download and install the software necessary to use the BeScope dual channel oscilloscope. Confirm that you have the items shown below.

List of required items

- Hardware
 - BeScope Dual Channel Oscilloscope Board
 - BeMicroCV Development Board
 - BNC Scope Probe
 - Mini USB Cable
- Computer
 - PC running Windows 7 32 or 64 bit Operating System
- Software
 - Quartus Programmer or Quartus 14.0 Software
 - BeScope Application Software
 - System Console Socket Server Script

2.1 Download and install the Quartus Programmer

In this step, we will download and install the Quartus Programmer to enable programming the Cyclone V FPGA on the BeMicro board with the BeScope firmware image.

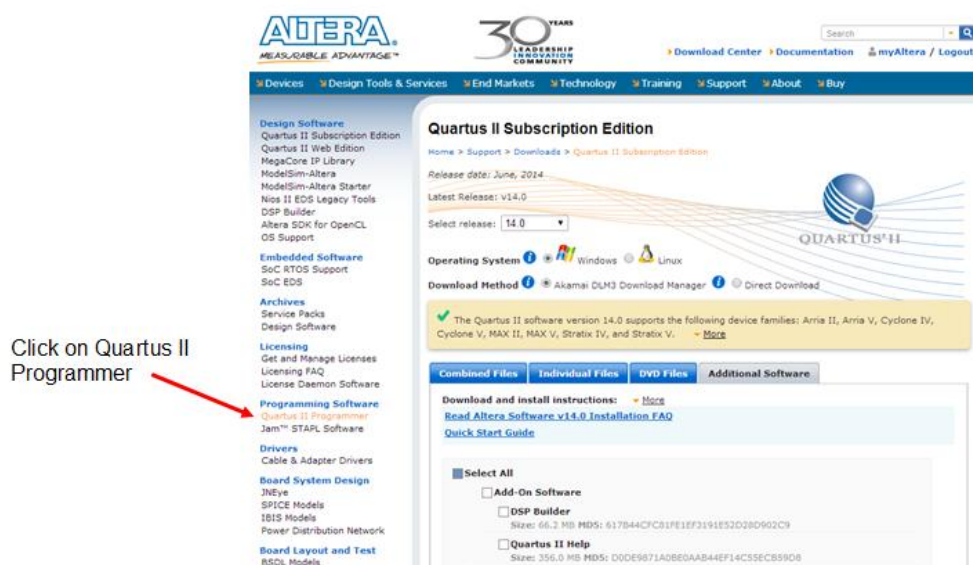


If you already have Quartus installed, you do not need to download or install the programmer. Proceed to step 2.3 *Assemble the BeScope*

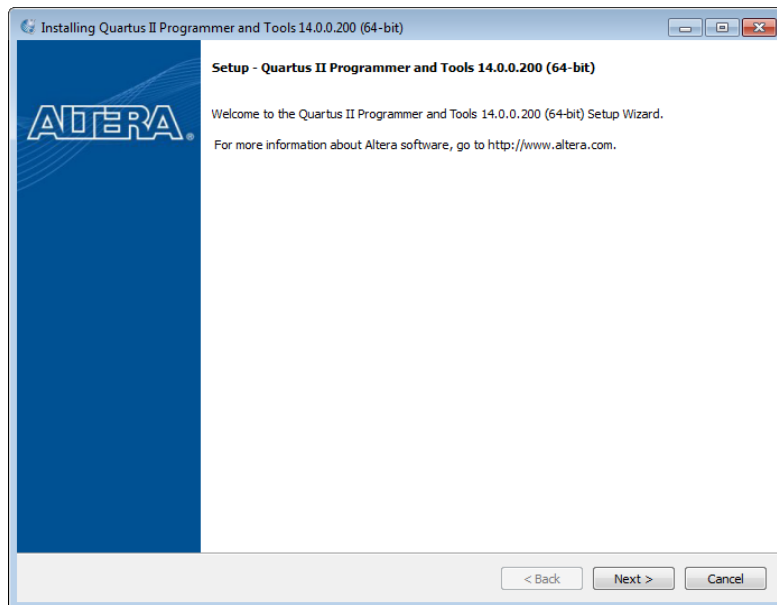
From the Altera website, main page, click on “Download Center”.



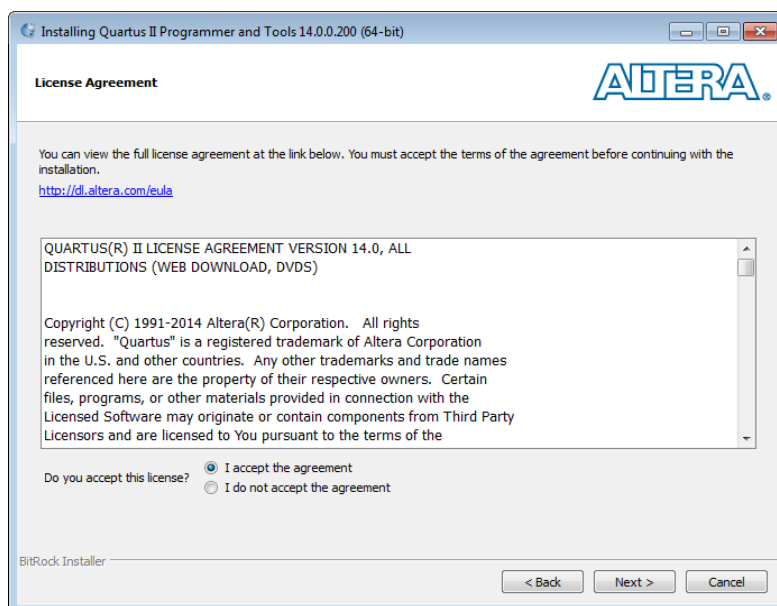
At the Download center, click on Quartus II Programmer and then click “Download Selected Files”.



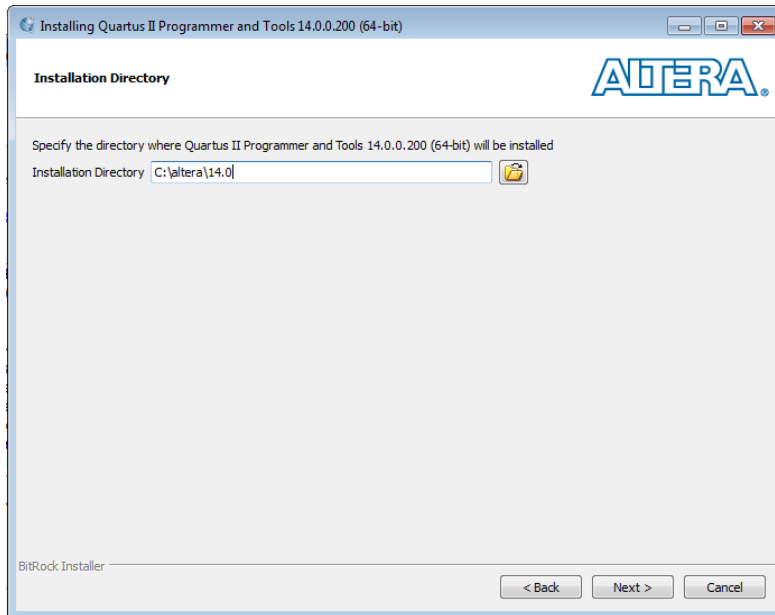
Note: you must have a myAltera account to download files. If you do not have an account, create one now. Download the installation executable to any convenient location on your hard drive, and run it. When the setup dialog appears, click “Next”.



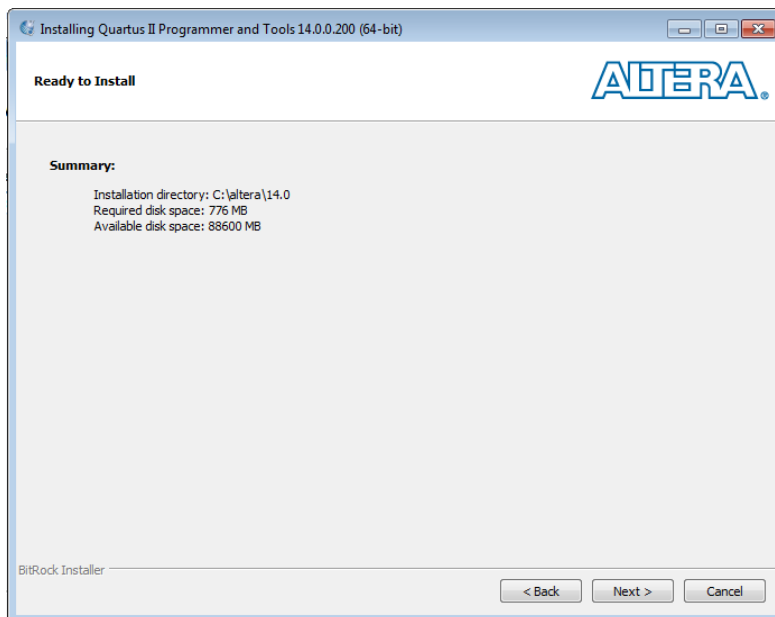
Select “I accept the agreement”, then click “Next”.



Specify a path to where you want the programmer installed, and click “Next”.



Click “Next” on the summary screen.



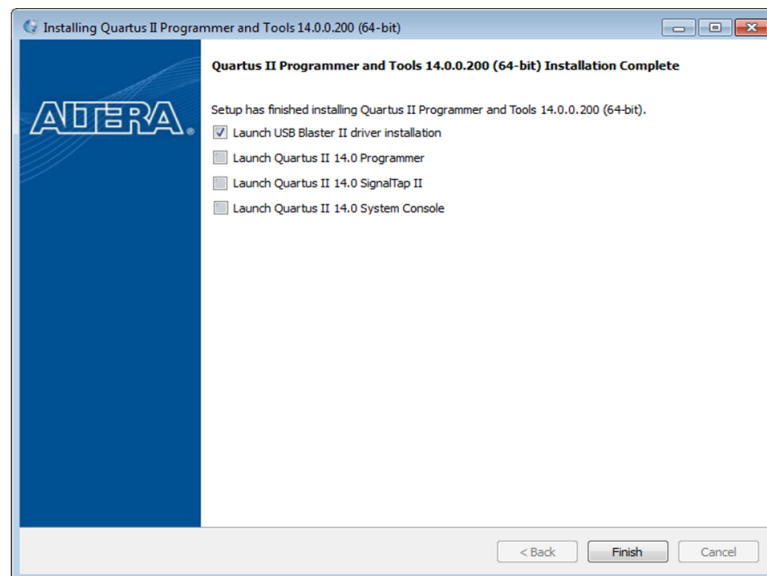
Since we don't need to launch any of the Quartus applications right now:

Uncheck Launch Quartus II 14.0 Programmer

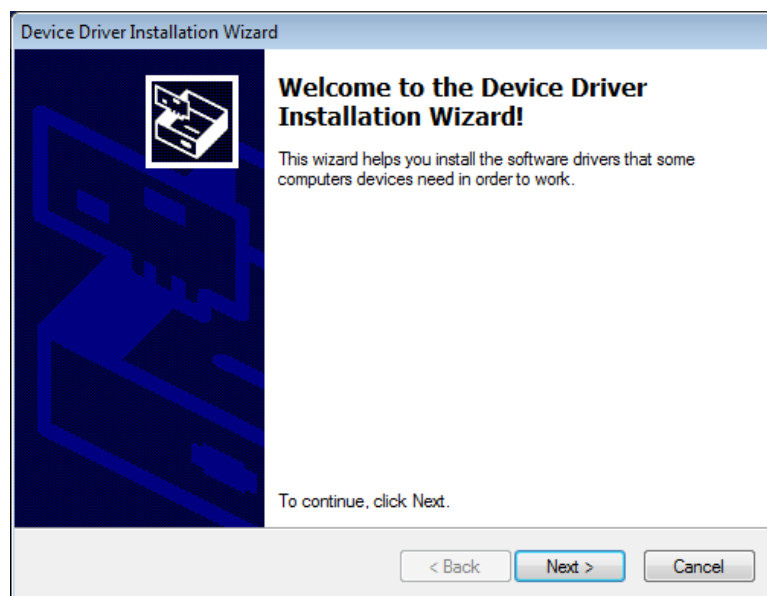
Uncheck Launch Quartus II 14.0 SignalTap II

Uncheck Launch Quartus II 14.0 System Console

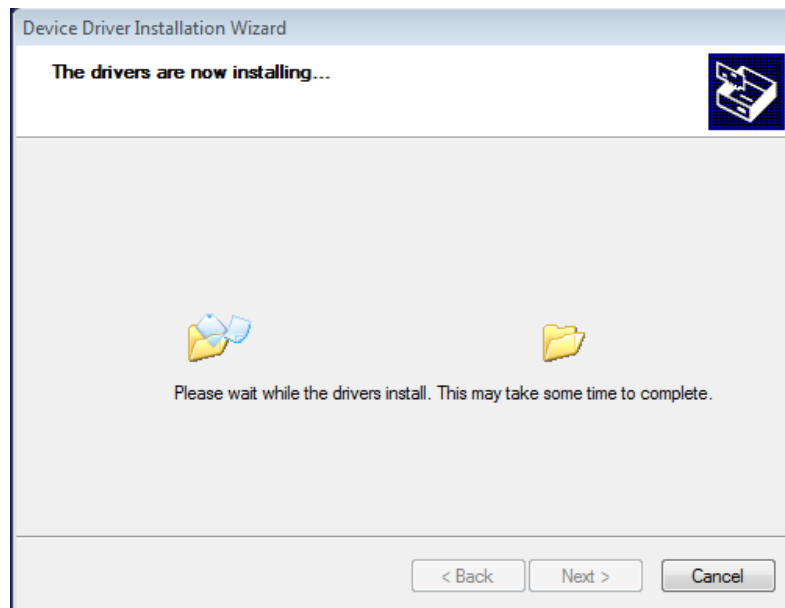
Then click "Next".



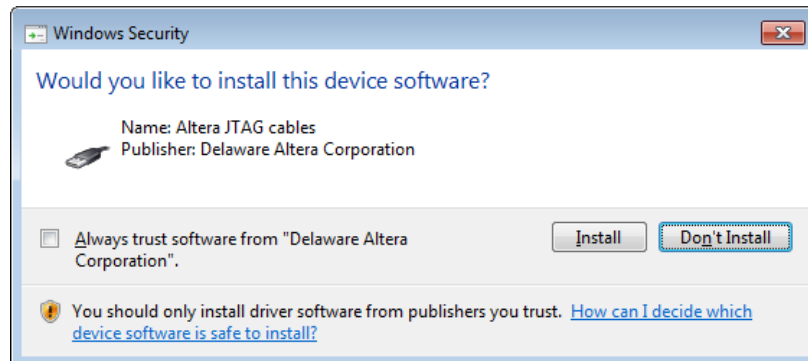
The USB Device driver installation wizard should open. Click "Next".



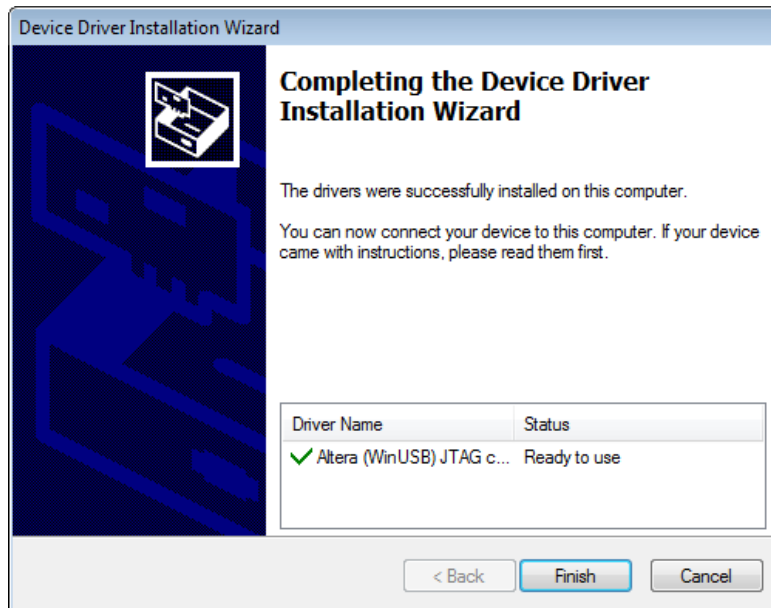
The installation status window should indicate the drivers are installing.



If you see a Windows Security dialog box, click "Install".



Lastly, you should see a successful installation notice. Click “Finish”



At this point, the installation of the Altera tools should be complete.

2.2 Download the BeScope installation package

Download the latest BeScope installation from the Arrow Electronics website:

http://www.arrow.com/solutions/bescope/downloads/BeScope_Installation_package_8-7-14.zip

After downloading the file, extract the zip file to **C:\Altera**. If you choose a different location, you will need to modify the System Console startup command.

Contents of the zip file:

1. System Console BeServer.tcl script
2. BeApp user interface application and source code
3. BeScope Hardware reference: Schematics & Bill of Materials
4. FPGA programming binary files and Quartus source file archive

2.3 Assemble the BeScope

Before the BeScope can be used to monitor analog waveforms, it is necessary to assemble the BeScope board onto the BeMicro CV Development Board.



It is important to check the orientation of the BeScope on the BeMicro CV board before applying power or plugging in the USB cable. The BNC connectors on the BeScope should point in the same direction as the SD card and 5V power connector on the BeMicro CV board. See Figure 2-1 for the proper orientation. Improper assembly will cause PERMANENT DAMAGE to the BeMicro CV board!

Once you have verified the assembly, use the provided USB cable to connect the BeScope to the host computer. Use the USB connector on the BeMicro CV board (**not** the connector on the BeScope.)

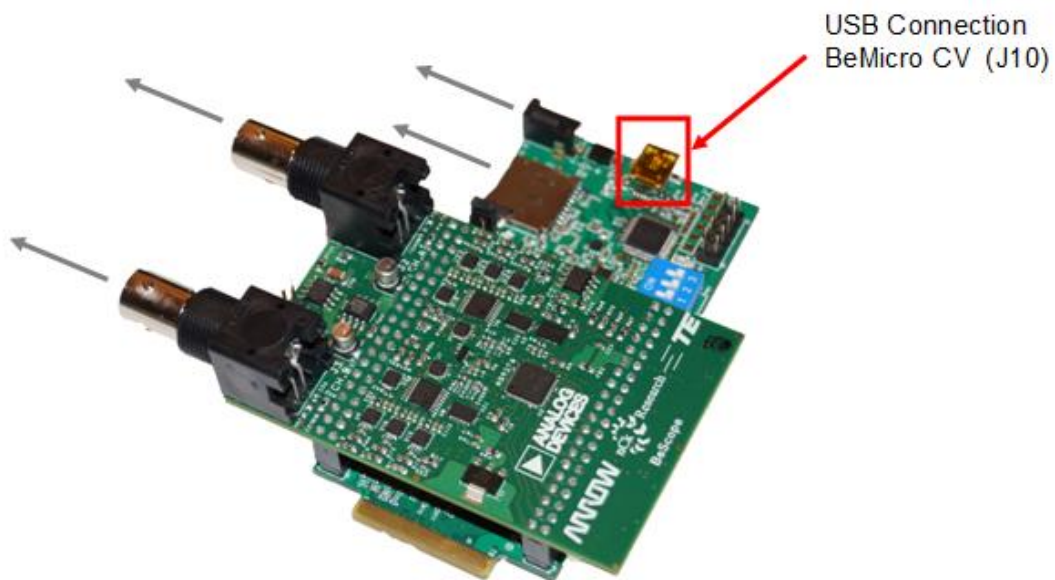


Figure 2-1: Assembly orientation on the BeMicro CV

2.4 Program the BeMicro CV FPGA

The configuration Flash memory on the BeMicro CV board must be programmed with the BeScope firmware.

1. Confirm the BeMicro CV is connected to the USB cable, and connected to a USB port on the Windows PC
2. Open the Altera Flash Programmer under Windows > All Programs > Quartus 14.0.0.200 > Quartus II Programmer and Tools 14.0.0.200 > Quartus II 14.0 Programmer

3. Verify that the hardware setup indicates USB-Blaster
4. Open the programmer configuration file using File > Open > C:\Altera\BeScope\Quartus\BeScope.cdf
5. Confirm that the programming chain looks like the image in Figure 2-2
6. Check the Program/Configure box as shown in Figure 2-2
7. Click Start
8. The progress bar will cycle several times as the programmer first configures the FPGA with a temporary image, then erases the flash, and finally programs the flash memory.
9. Confirm that the Progress is successful
10. Unplug the USB cable to force a reconfiguration of the FPGA.

The BeMicro CV is now programmed, and ready to communicate with the System Console server application.

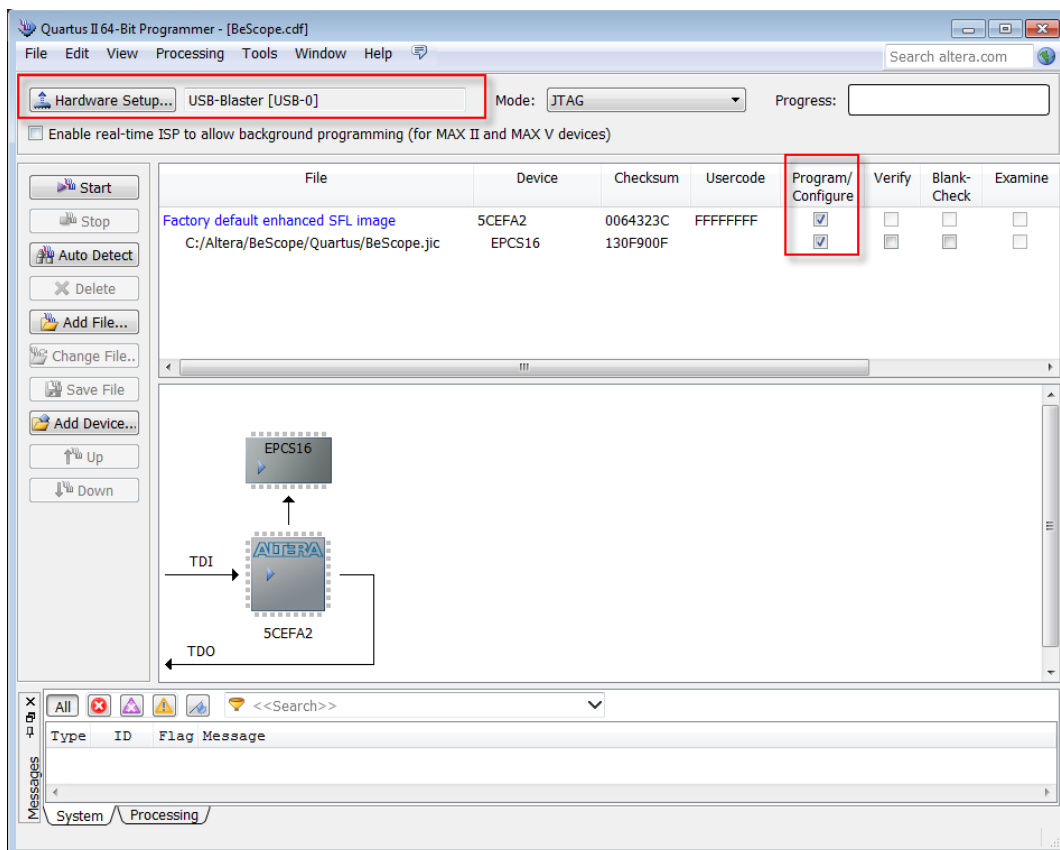


Figure 2-2: Quartus programmer setup

2.5 Launch the System Console Socket Server

The BeScope Graphical Interface uses TCP/IP sockets to communicate with the hardware. Before the BeScope GUI can communicate with the board, it is necessary to create a bridge between BeMicro CV Board JTAG master in the FPGA, and TCP/IP sockets.

Start the TCL script (pronounced tickle) by running the following command at a command prompt:

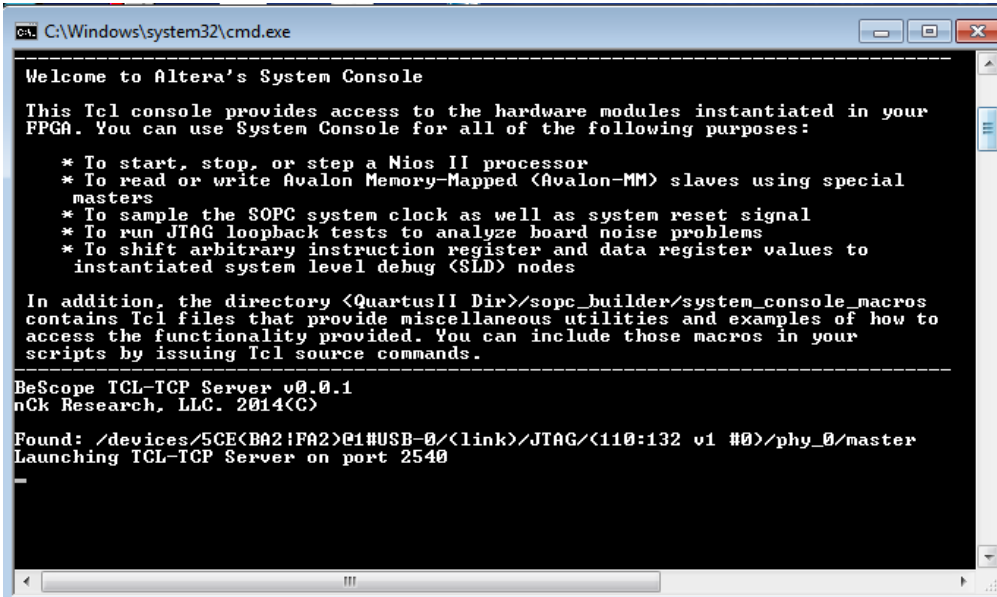
```
c:\altera\14.0\qprogrammer\sopc_builder\bin\system-console -cli -
-script c:\Altera\BeScope\BeServer.tcl
```



Note: If you have the full version of Quartus installed, you will need a slightly different command:

```
c:\altera\14.0\quartus\sopc_builder\bin\system-console -cil -
-script c:\Altera\BeScope\BeServer.tcl
```

It may take a minute at a blank screen, but then a message should appear. Confirm message in command window as shown in Figure 2-3.



```

C:\Windows\system32\cmd.exe

Welcome to Altera's System Console

This Tcl console provides access to the hardware modules instantiated in your
FPGA. You can use System Console for all of the following purposes:

* To start, stop, or step a Nios II processor
* To read or write Avalon Memory-Mapped (Avalon-MM) slaves using special
  masters
* To sample the SOPC system clock as well as system reset signal
* To run JTAG loopback tests to analyze board noise problems
* To shift arbitrary instruction register and data register values to
  instantiated system level debug (SLD) nodes

In addition, the directory <QuartusII Dir>\sopc_builder\system_console_macros
contains Tcl files that provide miscellaneous utilities and examples of how to
access the functionality provided. You can include those macros in your
scripts by issuing Tcl source commands.

-----
BeScope TCL-TCP Server v0.0.1
nCh Research, LLC. 2014(C)

Found: /devices/5CE(BA21FA2)01#USB-0/<link>/JTAG/<110:132 v1 #0>/phy_0/master
Launching TCL-TCP Server on port 2540
  
```

Figure 2-3: System Console command window

If you see the message that a master was found, and that the TCL-TCP server is Launching, then you are ready to run the BeScope user interface. If you do not see this message, confirm that you programmed the BeMicro board successfully in step 2.4 *Program the BeMicro CV FPGA*. If you get a java exception that the Path

cannot be found during the `open_service` command, this is a result of the PC not being able to communicate with the JTAG master inside the FPGA. Possible causes and suggested solutions are listed below:

- **USB JTAG server in use.** Close any other Altera applications that use USB
- **Incorrect image programmed.** Confirmed that you have the correct image loaded in the FPGA.

```

C:\Windows\system32\cmd.exe

Welcome to Altera's System Console
This Tcl console provides access to the hardware modules instantiated in your
FPGA. You can use System Console for all of the following purposes:

* To start, stop, or step a Nios II processor
* To read or write Avalon Memory-Mapped (Avalon-MM) slaves using special
  masters
* To sample the SOPC system clock as well as system reset signal
* To run JTAG loopback tests to analyze board noise problems
* To shift arbitrary instruction register and data register values to
  instantiated system level debug (SLD) nodes

In addition, the directory <QuartusII Dir>/sopc_builder/system_console_macros
contains Tcl files that provide miscellaneous utilities and examples of how to
access the functionality provided. You can include those macros in your
scripts by issuing Tcl source commands.

BeScope TCL-TCP Server v0.0.1
nCh Research, LLC. 2014(C)

Found:
java.lang.Exception: open_service: Path cannot be found
while executing
"open_service master $nm
  
```

Figure 2-4: BeApp Connection Dialog – With Error

2.6 Launch the BeScope user interface

Locate the BeApp in the extracted folder (C:\Altera\BeScope) and double click the BeApp.exe application. The BeScope user interface should open. When the window appears, click File > Connect To Device as shown in Figure 2-5.

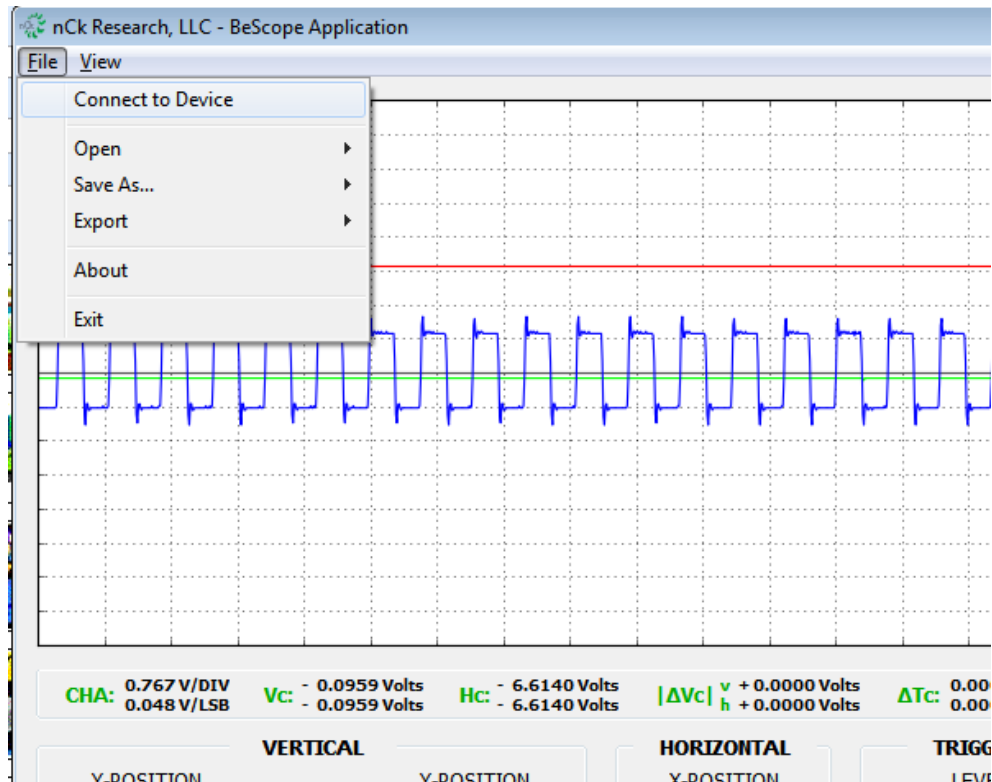


Figure 2-5: BeApp Connection Dialog

If the connection is successful, you will get a status message in the lower left corner indicating *BeMicro/BeScope - Connected*.

If you get a communication error, as shown in Figure 2-6, you will need to diagnose the connection. Confirm that the socket server launched. In some cases, leaving the programming window open can tie up the USB port. Try closing other Altera applications, if any are open.

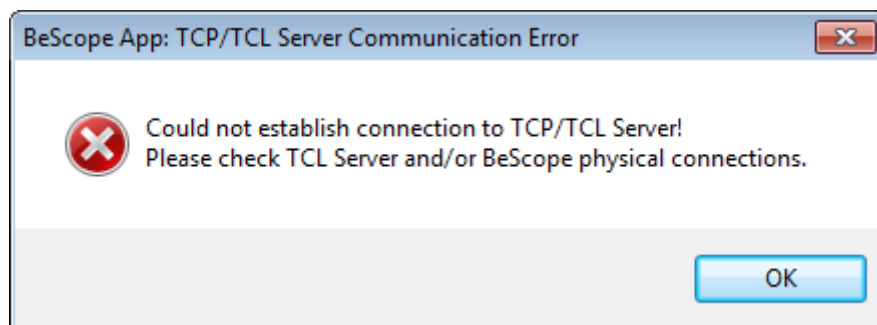


Figure 2-6: BeApp Communication Error Dialog Box

2.7 Verify the BeScope Operation

Connect the scope probe to Channel A. Using the spring-tip on the scope probe, probe pin 1 of J6 (labeled TEST FREQ). Turn on the Probe Source in the user interface as shown in Figure 2-7. You should now see a square wave appear in the waveform window.

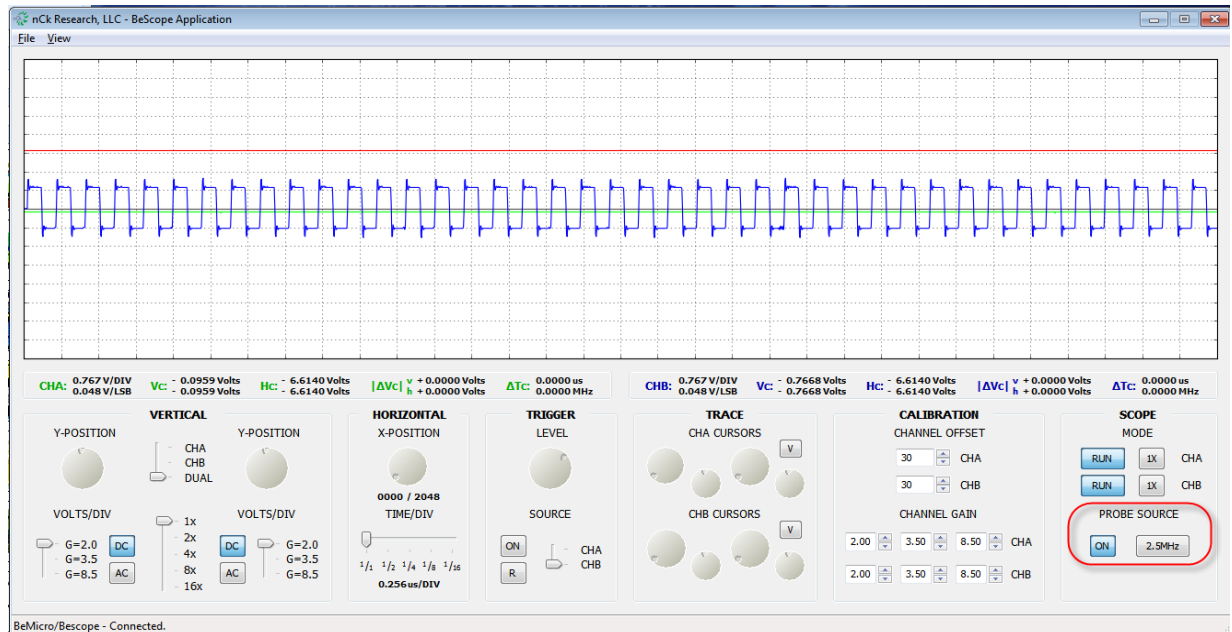


Figure 2-7: BeApp User Interface

If you see a waveform, you are now ready to use your oscilloscope.

2.8 Explore the BeScope user interface

Now that you have installed the necessary components for the BeScope, take a few moments to explore the user interface. Adjust the gain, use cursors to make a measurement, and turn on triggering and adjust the trigger level. More details on the user interface can be found in section 3.4 *BeScope user interface*

2.9 Batch File for easy launch

A batch file is provided to launch the BeApp and the Be Server as described above in sections 2.5 and 2.6. Because various installations may have different paths, this batch file may need customizing for your system. The batch file is located in the installation folder `c:\Altera\BeScope\BeScope.bat`

3 BeScope System technical overview

This section describes the hardware and software components of the BeScope system. The BeScope system consists of the following major component blocks:

- BeScope Analog board
 - 50 MHz analog bandwidth
 - 250 MSPS per channel using AD9286 with simultaneous sampling
 - Three independent gain stages for optimal performance
 - Input range adjustable up to $\pm 60V$ (10x probe setting, lowest gain)
 - Resolution as low as 12 mV/LSB (1x probe setting, highest gain)
 - 2.5 MHz and 5 MHz square wave generator output
 - Power via USB or via user-provided 5V supply
- BeMicro CV Development board
 - Provides the FPGA platform for communication and data acquisition
 - One Cyclone V E FPGA (5CEFA2F23C8N)
 - 25,000 LEs
 - 1,760 Kbit (Kb) M10K and 196 Kb MLAB memory
 - 4 fractional phase locked loops (PLLs)
 - 50 18x18-bit multipliers
 - 1 Hard Memory Controller
 - 1Gbit DDR3 SDRAM (x16)
 - Integrated USB Blaster for JTAG communication
- Altera Socket server
 - Altera provided application that runs on the host PC
 - Handles the messaging between the BeScope hardware and the user interface
 - Accepts TCP-IP socket
- User interface
 - Graphically displays waveforms on the screen
 - Allows setting gain on the BeScope hardware
 - Controls the on-board waveform generator
 - Controls trigger settings
 - Provides cursor and basic waveform measurement tools
 - Uses the Python platform-independent development environment
 - Source code included

3.1 BeScope Analog Board

3.1.1 Component Placement Diagram

Figure 3-1 shows a block diagram of the BeScope oscilloscope.

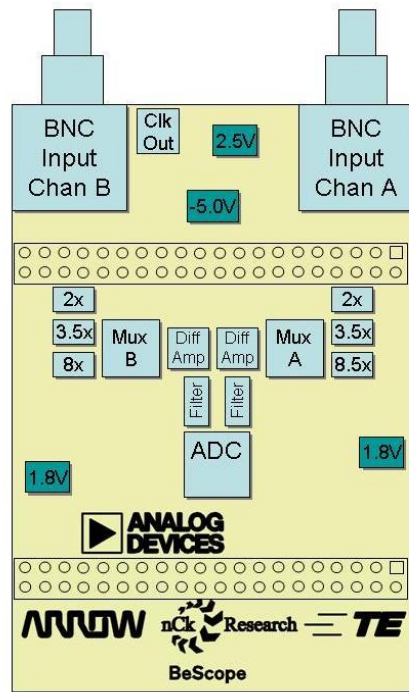


Figure 3-1: BeScope Block Diagram

3.1.2 Board Components

This section introduces the major components on the BeScope Dual Channel Oscilloscope. Figure 3-2 and

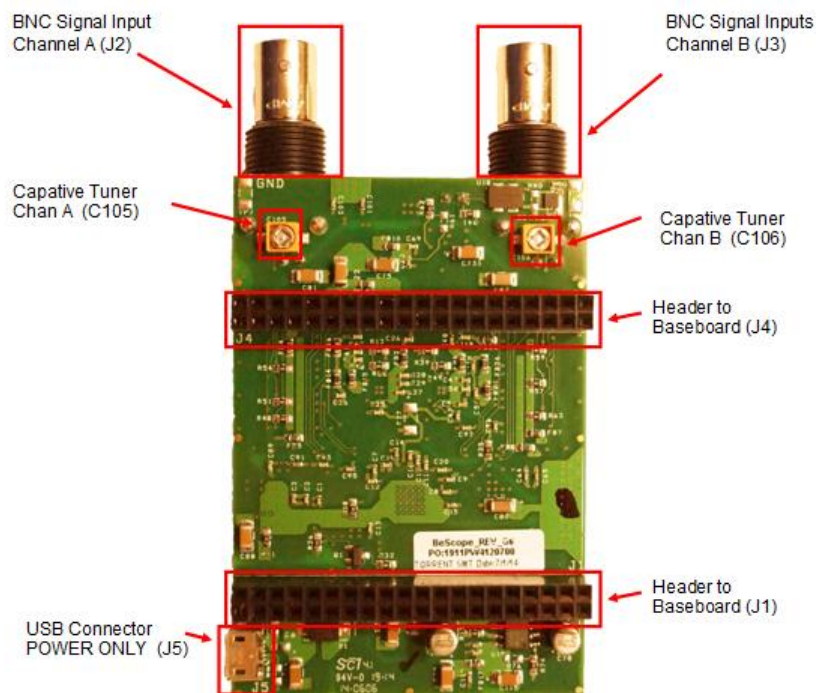


Figure 3-3 illustrate the component locations.

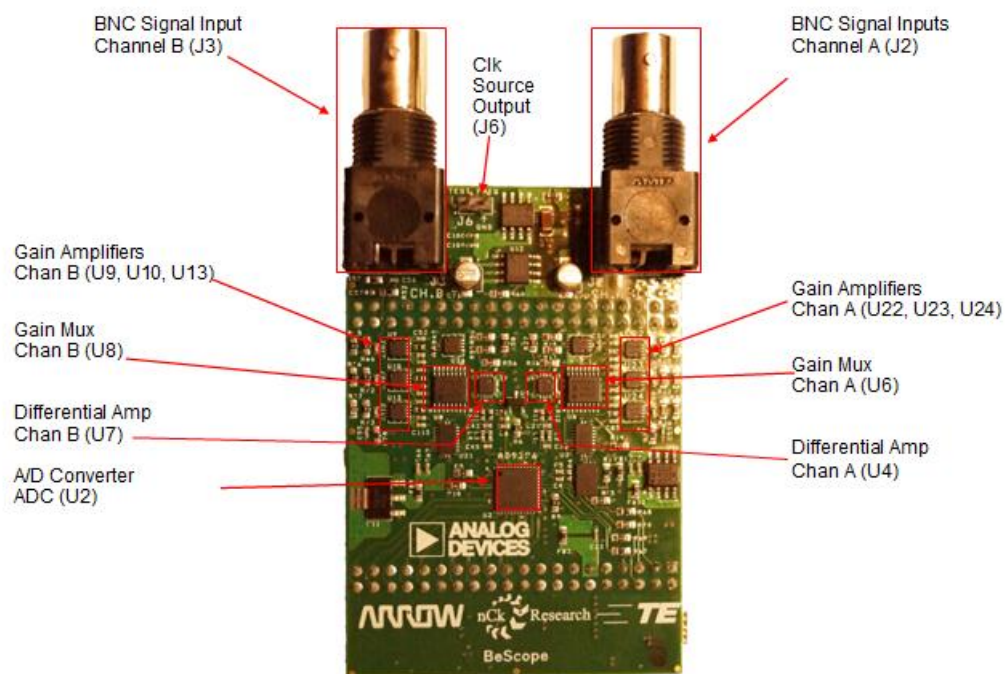


Figure 3-2: Major component locations, Top View

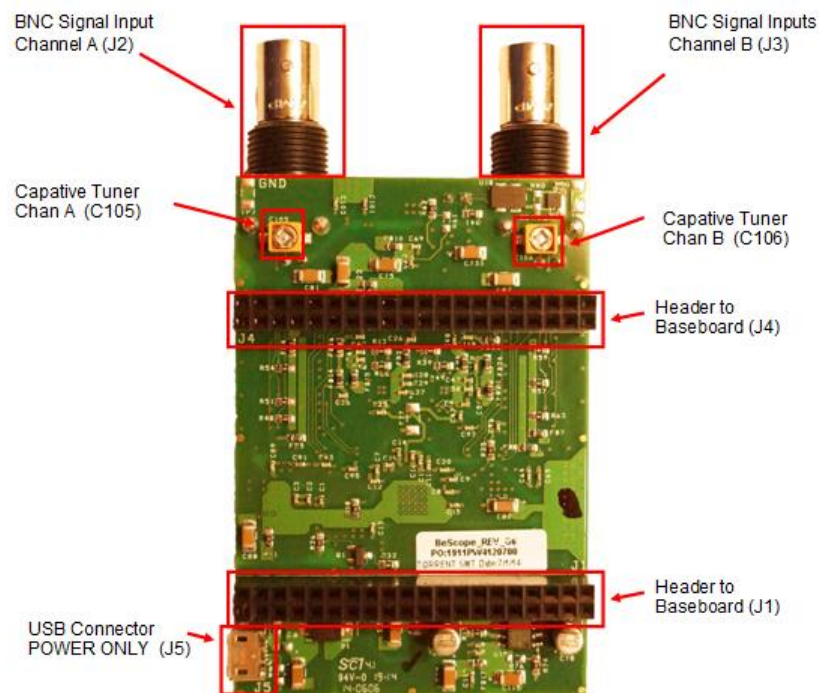


Figure 3-3: Major component locations, Bottom View

3.1.3 Analog Signal Chain Description

The primary function of the BeScope board is to convert an analog input voltage level to a digital signal. Figure 3-4 shows the signal path for one of the channels.

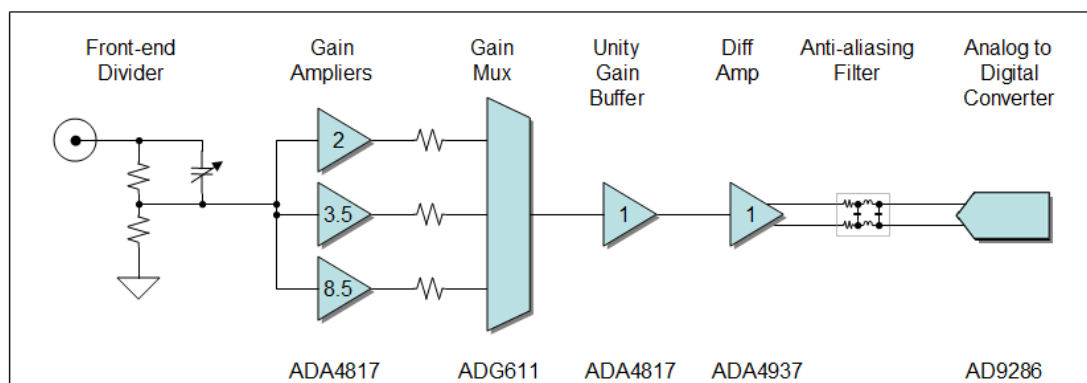


Figure 3-4: Basic signal path diagram

The Basic components in the analog signal chain are:

- Front-End Divider
- Gain Stages
- Gain Mux (Low Capacitance Analog Switch)

- Single-ended to Differential Amp
- Anti-aliasing Filter
- Analog to digital converter

These components will be described in more detail in the following sections.

3.1.3.1 Front-End Divider

The input of the BeScope is DC coupled. The input impedance is 1Meg Ohm, and the input signal is scaled by a resistor divider network 49.9k/953K for a 20:1 divider ratio.

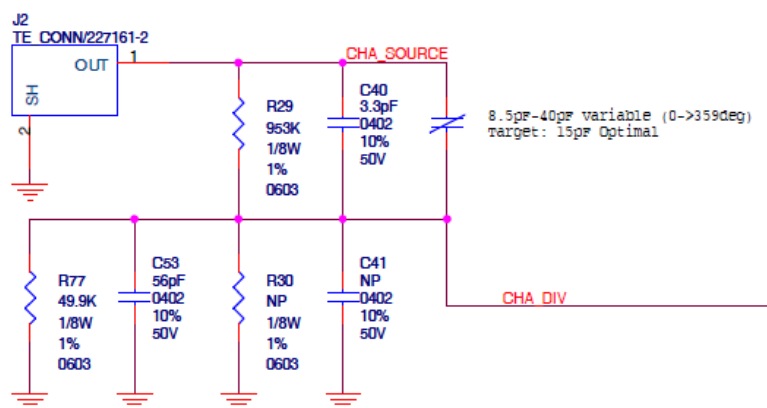


Figure 3-5: Schematic view: Front-end Divider

3.1.3.2 Gain Stages

The BeScope uses a 3 amplifier gain stage, whereby each gain setting has it's own amplifier for improved Gain and Bandwidth performance. Each amplifier is either turned on or off by their respective power down (/PD) pins, thereby alleviating any interference with each other.

The output of each gain amplifier, is connected to a low capacitance analog switch, configured as a mux.

Shown in Figure 3-6 is one of the gain amplifiers

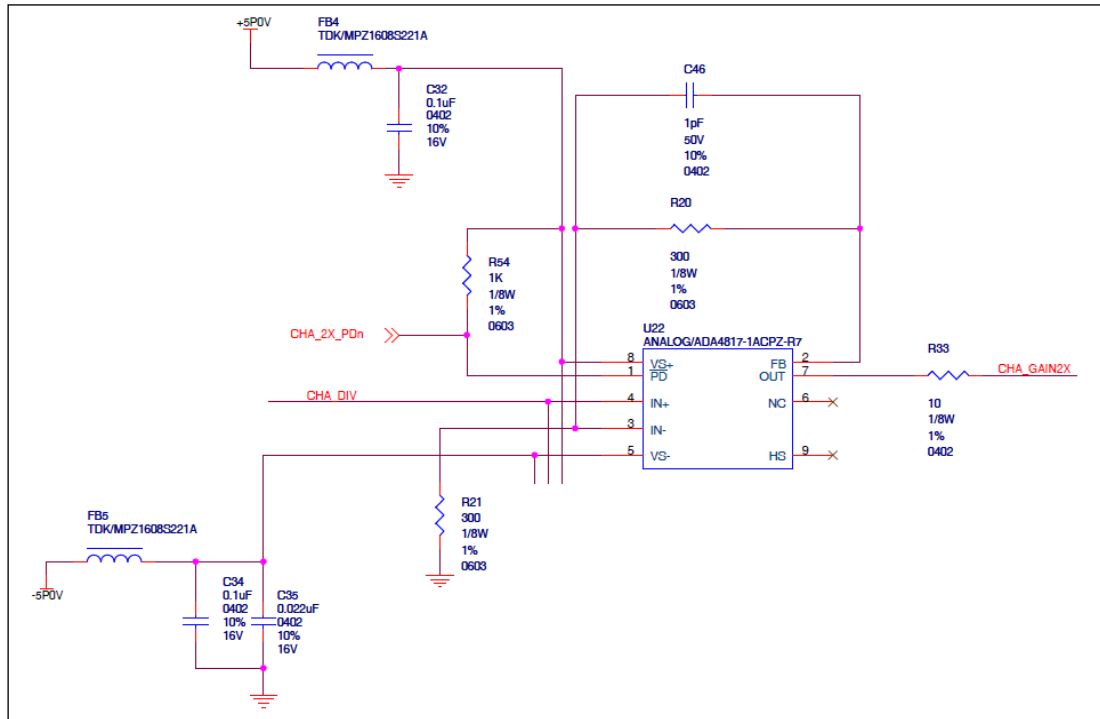


Figure 3-6: Schematic view: Gain stages

3.1.3.3 Low Capacitance Gain Mux

The outputs of the three amplifiers are sent to a low capacitance analog switch, configured as a mux, which selects one of the signal based on signals from the FPGA logic.

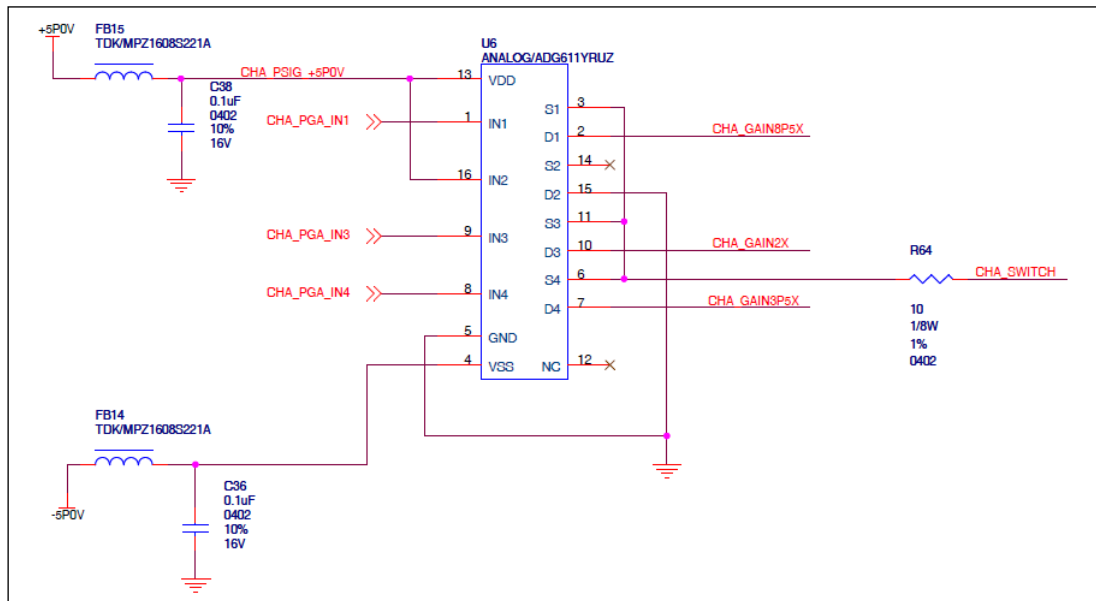


Figure 3-7: Schematic view: Gain Mux

3.1.3.4 Single-ended to Differential Amp and Anti-aliasing Filter

The output from the Gain Mux is routed through a unity gain buffer stage, before final routing to a single-ended to differential gain stage, that will drive the high speed Analog to Digital Converter.

This is done to eliminate any errors caused by the interaction between the Gain Mux, and the input of the single-ended to differential amplifier.

A low noise single-ended to differential amplifier, with a gain of 1.8, and output common mode voltage of 1.4Vdc is used, to maximize the dynamic range on the ADC. The input structure is configured to alleviate any mismatch between both sides of the differential output. The output common mode voltage (Vocm) is derived from the common mode voltage (Vcm) output from the ADC.

On the output of the single-ended to differential amplifier, is a third order 125Mhz, low pass filter, which reduces the noise bandwidth of the amplifier, and isolates the amplifier outputs from the ADC inputs.

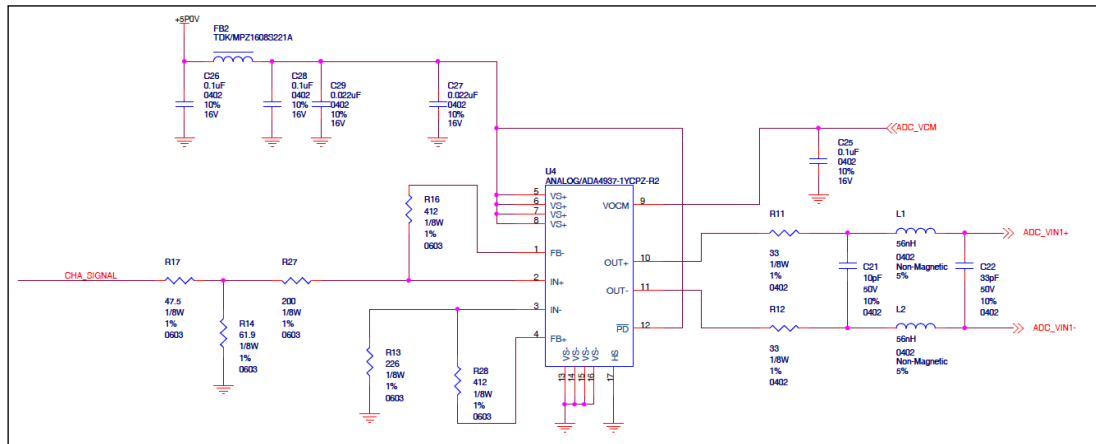


Figure 3-8: Schematic view: Differential Amp and Anti-aliasing Filter

3.1.3.5 Analog to Digital Converter (ADC)

The Analog-to-Digital converter (ADC) is an 8 bit, dual channel 250MSPS converter, with an input analog bandwidth of 700Mhz.

This ADC can also be set up as a single channel, 500MSPS converter, when configured in interleave mode, although this feature is not enabled on this platform.

The ADC is capable of measuring a 1.2Vp-p input signal, with a common mode input of 1.4Vdc, all fully under software control by the BeMicro CV board.

An external 1.0Vdc, reference voltage is provided by a shunt type Vref.

The main clock is provided a differential 250Mhz oscillator.

The BeMicro CV board, provides all the configuration control and output data processing to/from the ADC.

See Figure 3-9.

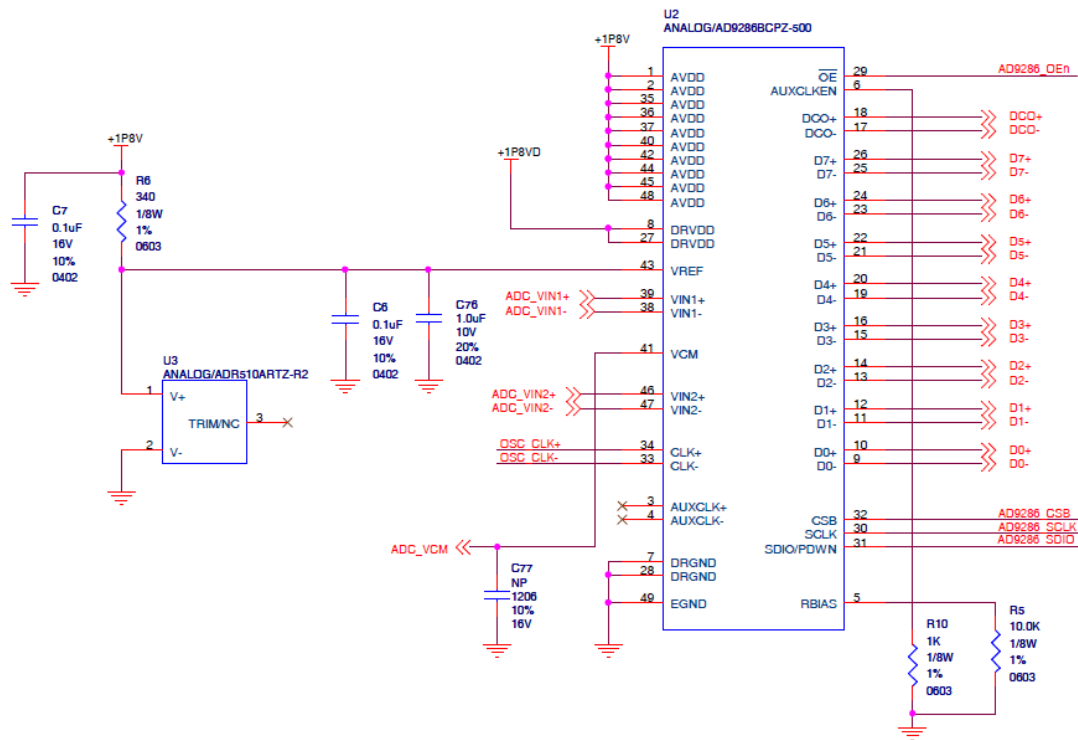


Figure 3-9: Schematic view: Analog to Digital Converter

3.1.4 Board Power

The BeScope can be powered by the BeMicro CV board directly, or when needed by an external 5Vdc supply, connected to the BeScope board via J5.

Powered by Cyclone V BeMicro

When the BeScope is powered by the BeMicro CV board, the 5Vdc (VCC5P0) is provided via J1-11, and the 3.3Vdc (VCC3P3) is provided via J4-1 and J4-2).

Powered by External Power Supply

When the BeScope is powered by an external power supply, using a Micro-USB connector J5, the external 5Vdc is diode-or'd with the 5Vdc, from the BeMicro CV board, to provide additional current on the 5Vdc.

Whether powered only by the BeMicro CV board, or external power supply, the VCC5P0, is filtered and becomes the main +5P0V for the BeScope.

The +5P0V, is subsequently used to generate several voltages on the BeScope as outlined below.

1P8VD	1.8Vdc Vdd for the ADC
1P8V	1.8Vdc general board power
2P5V	2.5Vdc general board power
-5P0V	-5Vdc for Channel A gain amplifiers
-5P0V_B	-5Vdc for Channel B gain amplifiers

3.1.5 Clock & Timing

ADC Clock

The ADC clock is provided by a 250Mhz, LVDS clock oscillator.

Test Frequency Oscillator

A test frequency oscillator is provided, which allows you to switch between two test frequencies of 2.5Mhz and 5Mhz. The test frequency oscillator is controller on/off by a high side power switch, all of which are controlled by the BeMicro CV board.

3.1.6 Miscellaneous

There are two 2.5Vdc-to-5.0Vdc logic logic level translation devices between the BeMicro CV board and the Gain Mux.

3.1.7 40 Pin Headers

The BeScope connects to the BeMicro baseboard through two 2x20 headers.

Table 3-1 and Table 3-2 summarize the BeScope signal names, header pin assignments, BeMicro CV signal names, and FPGA pin number for the 40 Pin Headers J1 and J4.

J1 Pin Number	BeScope Signal Name	BeMicro CV Signal Name	Cyclone V FPGA Pin Number
1	N/C	GPIO_01	T22
2	N/C	GPIO_02	T15
3	ADC_CSBn	GPIO_03	R22
4	N/C	GPIO_04	R15
5	ADC_SDIO	GPIO_05	R21
6	ADC_SCLK	GPIO_06	R16
7	ADC_OEn	GPIO_07	P22

8	ADC_SDOn	GPIO_08	R17
9	N/C	DIFF_TX_5+	N20
10	N/C	DIFF_TX_5-	N21
11	VCC5P0	VCC5P0	na
12	GND	GND	na
13	N/C	DIFF_TX_6+	M22
14	N/C	DIFF_TX_6-	L22
15	N/C	DIFF_TX_7+	M20
16	N/C	DIFF_TX_7-	M21
17	N/C	DIFF_TX_8+	K21
18	N/C	DIFF_TX_8-	K22
19	N/C	DIFF_TX_9+	T19
20	N/C	DIFF_TX_9-	T20
21	D7+	DIFF_RX_9+	T18
22	D7-	DIFF_RX_9-	T17
23	D6+	DIFF_RX_8+	L19
24	D6-	DIFF_RX_8-	L18
25	D5+	DIFF_RX_7+	K17
26	D5-	DIFF_RX_7-	L17
27	D4+	DIFF_RX_6+	N19
28	D4-	DIFF_RX_6-	M18
29	N/C	VCC3P3	na
30	GND	GND	
31	DCO+	DIFF_RX_5+	N16
32	DCO-	DIFF_RX_5-	M16
33	D3+	DIFF_RX_4+	U10
34	D3-	DIFF_RX_4-	T9
35	D2+	DIFF_RX_3+	R9
36	D2-	DIFF_RX_3-	T10
37	D1+	DIFF_RX_2+	U12
38	D1-	DIFF_RX_2-	U11
39	D0+	DIFF_RX_1+	R11
40	D0-	DIFF_RX_1-	R10

Table 3-1: Board Reference Information for 40 Pin Header J1

J4 Pin Number	BeScope Signal Name	BeMicro CV Signal Name	Cyclone V FPGA Pin Number
------------------	------------------------	---------------------------	---------------------------------

1	VCC3P3	VCC3P3	
2	VCC3P3	VCC3P3	
3	N/C	I2C_SDA	G1
4	N/C	I2C_SCL	G2
5	CHA_3P5X_PDn	GPIO_A	V10
6	CHA_2X_PDn	GPIO_B	P8
7	GND	GND	
8	GND	GND	
9	GND	GND	
10	GND	GND	
11	CHA_8P5X_PDn	LVDS_TX_E4-	R7
12	CHA_IN1	LVDS_TX_E4+	P7
13	CHA_IN3	LVDS_TX_E3-	W8
14	CHA_EN	LVDS_TX_E3+	W9
15	CHA_IN4	LVDS_TX_ECLK-	U6
16	N/C	LVDS_TX_ECLK+	V6
17	GND	GND	
18	GND	GND	
19	N/C	LVDS_TX_E2-	U7
20	N/C	LVDS_TX_E2+	U8
21	N/C	LVDS_TX_E1-	AA7
22	N/C	LVDS_TX_E1+	AB7
23	N/C	LVDS_TX_E0-	AB6
24	N/C	LVDS_TX_E0+	AB5
25	GND	GND	
26	GND	GND	
27	MON_FS	LVDS_TX_O4-	AA8
28	N/C	LVDS_TX_O4+	AB8
29	MON_EN	LVDS_TX_O3-	AA10
30	CHB_EN	LVDS_TX_O3+	AA9
31	CHB_IN2	LVDS_TX_OCLK-	Y10
32	CHB_IN1	LVDS_TX_OCLK+	Y9
33	GND	GND	
34	GND	GND	
35	CHB_IN4	LVDS_TX_O2-	R12
36	CHB_3P5X_PDn	LVDS_TX_O2+	P12
37	CHB_2X_PDn	LVDS_TX_O1-	AB10
38	CHB_8P5X_PDn	LVDS_TX_O1+	AB11
39	N/C	LVDS_TX_O0-	Y11
40	N/C	LVDS_TX_O0+	AA12

Table 3-2: Board Reference Information for 40 Pin Header J4

3.1.8 Power Supply

The BeScope board can get its power from two locations; either the 40 pin headers J1 and J4, or the micro USB connector, J5. Note the USB connector on this board is only used for power; no data transfer is possible using this connection. Table 3-3 summarizes the various power rails on the BeMicro CV Board.

Schematic Signal Name	Voltage (V)	Description
VCC5P0	5.0	Power from the headers or the micro USB connector
VCC3P3V	3.3	3.3V from headers
+5P0V	5.0	Filtered 5V power
-5P0V	-5.0	-5V for OpAmps, Channel A
-5P0V_B	-5.0	-5V for OpAmps, Channel B
+1P8VD	1.8	Digital power for ADC
+1P8VD	1.8	Analog power for ADC
+2P5V	2.5	2.5V I/O buffers

Table 3-3: BeScope Power Rails

3.1.9 On-board Clock Source

The BeScope board has an on-board clock source for self diagnostics and testing. This clock source is a dual frequency SIT8033AC. The frequency is selectable between 2.5MHz and 5.0 MHz using the `mon_fs` bit of the Control register. In addition, this clock source can be completely powered down to eliminate possible noise coupling. The `mon_en` pin of the control register controls the ADP195 load switch to turn off the 2.5V rail to the clock source.

The output from the clock source is a square wave, and can be found on Pin 1 of header J6. J6 Pin 2 is a conveniently located ground pin.

3.1.10 Board Components Reference

This section describes the BeScope board components. Table 3-4 lists the devices on the BeScope Board along with Manufacturer Part Numbers and website information.

Board Reference	Component	Manufacturer	Manufacturer Part Number	Manufacturer Website
U2	Dual ADC Pipelined 500Msps	Analog Devices	AD9286BCPZ-500	www.analog.com
U9,U10,U13, U22,U23,U24,	OP Amp Single-ended	Analog Devices	ADA4817-1ACPZ-R7	www.analog.com
U4,U7	OP Amp Differential	Analog Devices	ADA4937-1YCPZ-R2	www.analog.com
U6,U8	Analog Switch Quad SPST	Analog Devices	ADG611YRUZ	www.analog.com
U12,U14	Charge Pump Inverting Regulator	Analog Devices	ADM8660ARZ	www.analog.com
U27	Power Switch	Analog Devices	ADP195ACPZ-R7	www.analog.com
U11	LDO Regulator 1.8V 1.6A	Analog Devices	ADP3338AKCZ-1.8-R7	www.analog.com
U16	LDO Regulator 1.8V 0.3A	Analog Devices	ADP7102ARDZ-1.8-R7	www.analog.com
U17	LDO Regulator 2.5V 0.3A	Analog Devices	ADP7102ARDZ-2.5-R7	www.analog.com
U3	Precision Voltage Ref 1V	Analog Devices	ADR510ARTZ-R2	www.analog.com
D1	Diode Schottky 60V 3A	Diodes_Inc	B360A-13-F	www.diodes.com
Q1	MOSFET N-CH 20V 2A	Fairchild	FDN327N	www.fairchildsemi.com
U20,U21	Voltage Level Translator	Fairchild	FXMA108BQX	www.fairchildsemi.com
C105,C106	Cap Trimmer Ceramic	Murata	TZB4P400BA10R00	www.murataamericas.com
U18	Two Frequency Oscillator	SiTime	SIT8033AC-31-25A-5.00000	www.sitime.com

Table 3-4: Component Information for Devices on BeScope Oscilloscope

3.2 BeMicro CV Development Board

The BeMicro CV Board provides the base platform and digital capabilities of the BeScope system. Note: there may be other boards released in the future that are compatible with the BeScope analog board. This document only describes operation with the BeMicro CV Development Board.

The Analog to Digital converter on the BeScope translates the conditioned analog signals to digital signals, and transmits them over nine LVDS pairs (eight data pairs, plus a clock pair) to the Cyclone V FPGA on the BeMicro CV board.

The Logic in the FPGA captures the LVDS signals, and stores the data sample in on-chip RAM. When a Buffer has been collected, it can be transmitted to the user interface via the JATG master and System Console.

3.2.1 Qsys System

The Main logic design is implemented in the Altera Qsys system integration tool. The Qsys system integration tool saves significant time and effort in the FPGA design process by automatically generating interconnect logic to connect intellectual property (IP) functions and subsystems. Figure 3-10 shows the Qsys system for the BeScope Oscilloscope.

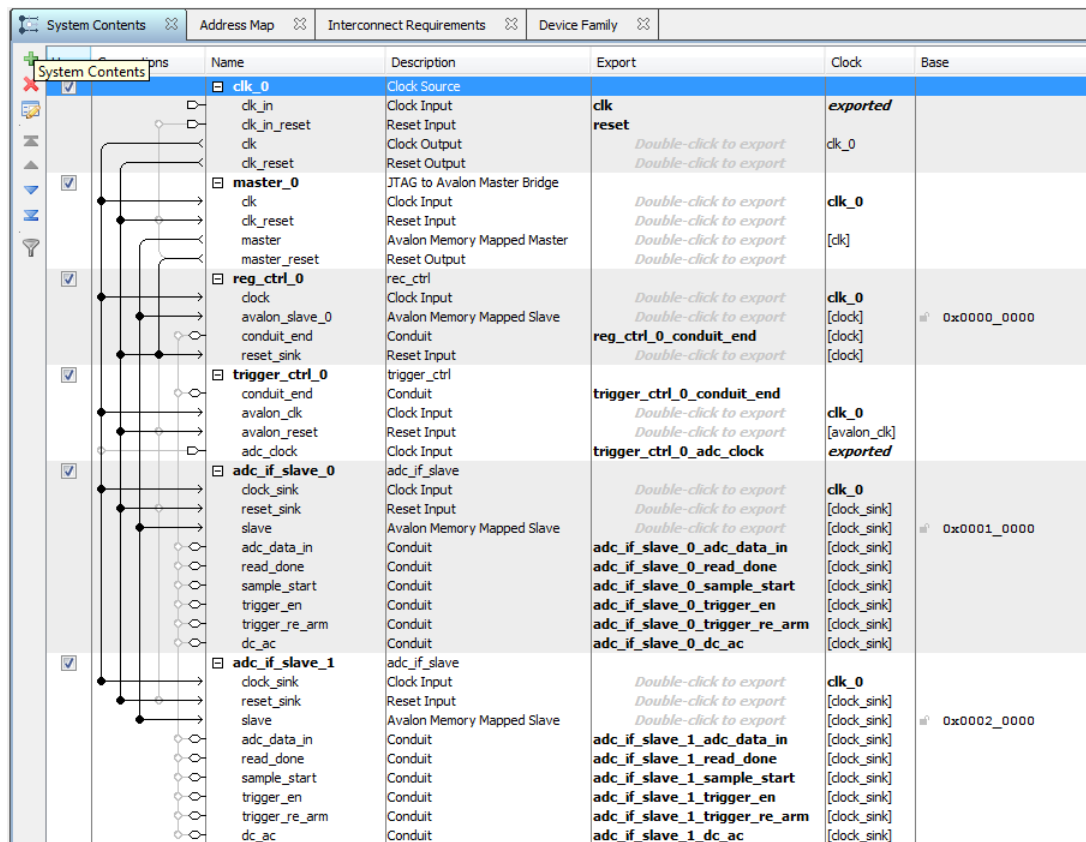


Figure 3-10: Qsys Interconnect Diagram

There are 5 Main components in the Qsys system. The main blocks are:

- JTAG Master
- Register Control
- Trigger Control
- ADC Interface A
- ADC Interface B

Figure 3-11 shows a simplified view of the blocks and their interconnections. The operation of the blocks will be described in more detail in the following sections.

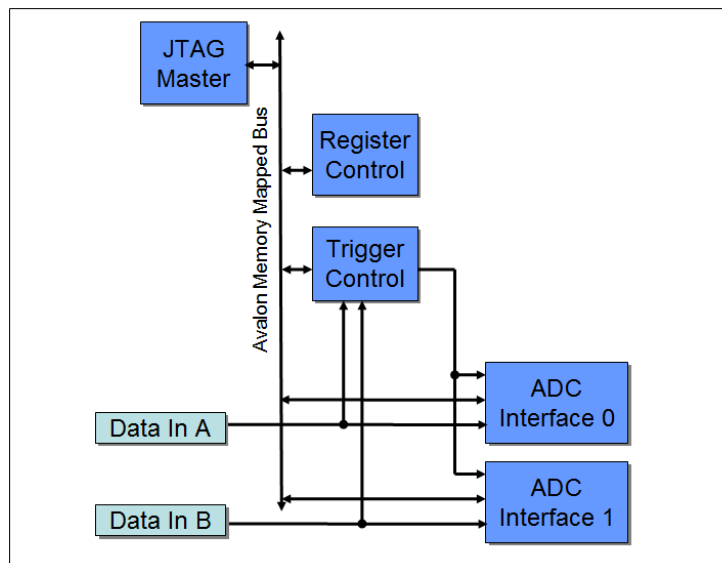


Figure 3-11: Qsys Functional Block Diagram

3.2.1.1 JTAG Master

The Altera JTAG to Avalon Master Bridge core provides a connection between host systems and Qsys systems via the JTAG interface. Host systems can initiate Avalon Memory-Mapped (Avalon-MM) transactions by sending encoded streams of bytes via the JTAG interface. The core supports read and write transactions, but not burst transactions. The complete documentation for the JTAG Master Bridge can be found in the Altera Embedded Peripheral IP User Guide

(http://www.altera.com/literature/ug/ug_embedded_ip.pdf)

3.2.1.2 Register Control

This block contains general purpose control and status registers for the BeScope platform. The Register Control block controls the Gain settings, on board clock source, triggering, and AC/DC operation. The graphical Interface sets these values automatically. The bit definitions of the control register are defined in Table 3-5

Field Name	Description	Bit Position
cha_gain	Channel A Gain Setting. Selection for Low, Medium, and High gain. As initially released, these settings are: Gain 2.0 = b1101 Gain 3.5 = b0111 Gain 8.5 = b1110	[3:0]
chb_gain	Channel B Gain Setting. Selection for Low, Medium, and	[7:4]

	High gain. As initially released, these settings are: Gain 2.0 = b1101 Gain 3.5 = b1110 Gain 8.5 = b0111	
dc_ac_a	AC/DC Operation Channel A. Logic 0 sets DC operation, Logic 1 sets AC operation	[8]
dc_ac_b	AC/DC Operation Channel B. Logic 0 sets DC operation, Logic 1 sets AC operation	[9]
mon_en	Turns on the power to the on-board clock source. Clock is on when this is set to logic 1.	[10]
mon_fs	Selects the output frequency of the on-board clock source. Logic 0 will result in a 2.5MHz output, Logic 1 results on 5MHz output. Mon_en must be on for the output to function.	[11]
trig_lvl	Sets the Trigger Level. This is an 8-bit value, stored in absolute ADC samples.	[23:16]
trig_src	Sets the Trigger source. Logic 0 sets Channel A as the trigger source, Logic 1 sets Channel B as the trigger source	[29]
trig_dir	Sets the Trigger Direction Logic 0 sets falling edge, Logic 1 sets Rising Edge	[30]
trig_en	Enables the trigger.	[31]

Table 3-5: Control Register offset 0x0 Bit Definitions

3.2.1.3 Trigger Control

The Trigger Control logic is responsible for detecting the trigger condition, and generating trigger strobes to the ADC interface blocks. The trigger enable, trigger level, and trigger direction are set in the user interface and stored in the Register Control block.

The data from the ADC is sent to both the ADC interface block and the Trigger Control block. The Trigger Control then compares the value of the Trigger Level setting with the incoming data stream to generate a trigger strobe. The trigger logic applies a mild amount of smoothing on the incoming data values to ensure accurate triggering

3.2.1.4 ADC Interface

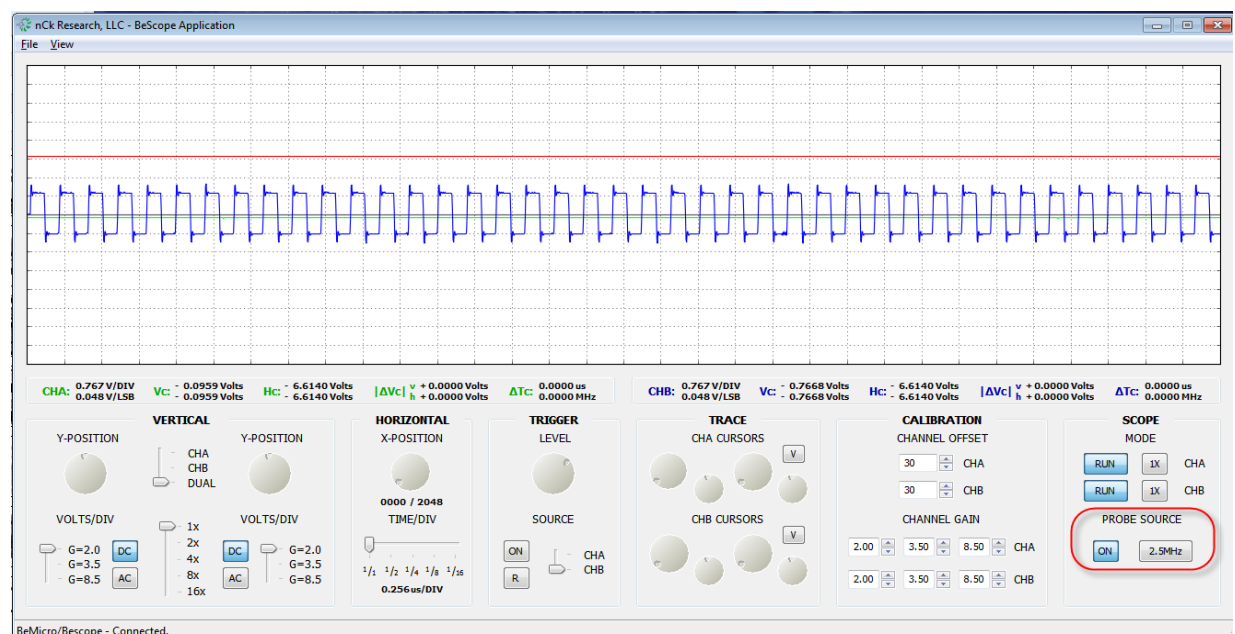
The ADC interface is responsible for storing the samples acquired by the data converter. The samples are stored in an on-chip dual-port RAM, with write access from the ADC converter side, and the read access is from the user interface. In normal (non-triggered) mode, the stored samples are read by the user interface as fast as possible. In triggered mode, the ADC Interface indicates the buffer is valid by storing the pattern 0xAAAAAAAA to the first location in memory. When the user interface reads this buffer in trigger mode, if the first value is not 0xAAAAAAAA, the buffer is discarded.

3.3 System Console Socket Server

The Altera System Console communicates with Altera hardware over the JTAG interface. System Console is a tool from Altera that provides visibility into your system, which allows faster debugging and time to market for your FPGA.

3.4 BeScope user interface

The BeScope user interface was written in the Python programming language. Python is a widely used general-purpose, high-level programming language. This section describes the operation of the user interface.



4 Additional Information

This chapter provides additional information about the document.

4.1 Board Revision History

The Table 4-1 lists the versions of all releases of the Arrow Electronics BeScope dual channel oscilloscope.

Release Date	Version	Description
July 2014	Initial release	Limited production run

Table 4-1:Board Revision History

4.2 Document Revision History

Date	Version	Description
July 2014	1.0	Initial Release

Table 4-2:Document Revision History