myRIO-1950 JTAG Instructions

This document is intended to give instructions on how to use the myRIO-1950 JTAG port for lead users. This is not intended for the general public and should be distributed with caution. Using the JTAG port may cause permanent damage to the myRIO-1950 if improperly used.

1.0 General JTAG specifications

Logic Level: 1.8V[1]

Max JTAG Frequency: 10Mhz

Devices in JTAG Chain[2]: 1st) Zynq PS (Arm processor)
2nd) Zynq PL (FPGA Fabric)
3rd) LCMXO2-640 CPLD[2]

Number of JTAG Instruction Registers:
Zynq PS = 4
Zynq PL = 6
CPLD = 8

[1]NOTE that the JTAG pins are ESD sensitive. Additionally they may be damaged if overvoltage above 1.8V. Be sure to attach the VREF pin of the JTAG programmer properly. Do not force the JTAG programmer to use any voltage other than 1.8V on its IO.
[2]CAUTION: The CPLD is not intended for customer access over JTAG. If you modify the CPLD your myRIO-1950 will no longer function properly. Do not attempt to program the CPLD.

2.0 Physical Characteristics

When accessing JTAG on the myRIO-1950 be sure that header J5 is populated. If it is not then populate this header with a through hole 2x3 0.1" spaced standard header. Note that Pin 1 is labeled with a 1 and pin 6 is labeled with a 6.

Figure 1: JTAG Header Location

The following figure shows the JTAG header's pinout.

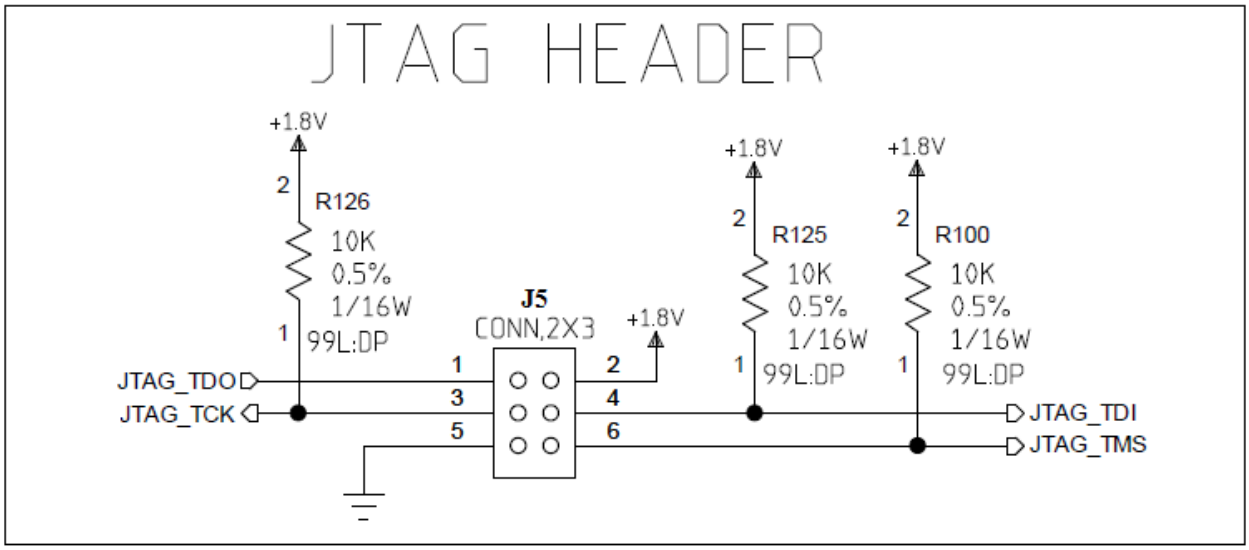

Figure 2: Pinnout of JTAG header

3.0 General Instructions:

1) Attach the JTAG programmer to the JTAG header, connecting TDO to TDO, TCK to TCK, TDI to TDI, TMS to TMS, GND to GND, and 1.8V to VREF.

2) Since this JTAG chain has multiple devices the JTAG software usually needs to know about all the devices in the chain to operate properly. Many times the software just needs to know the instruction register length of each of the devices in the chain. Here is an example of such configuration from the Xilinx XMD software (IR stands for instruction register):

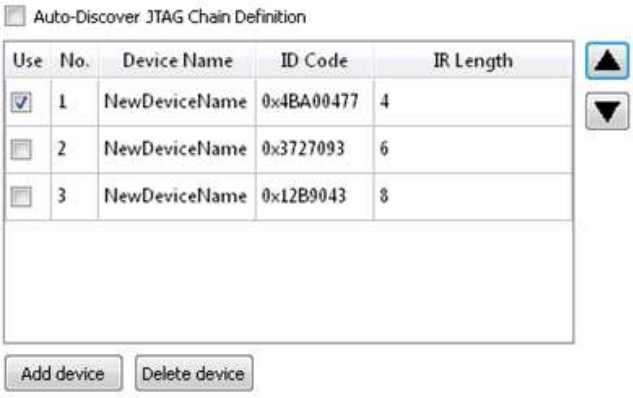

Figure 3: Example JTAG configuration

3) Configure the JTAG operating frequency to 10Mhz or slower.

4) After you configure these settings you should be able to communicate with any device in the JTAG chain. DO NOT USE SOFTWARE TO RECONFIGURE THE CPLD, you WILL CAUSE THE BOARD TO NO LONGER FUNCTION PROPERLY. This should not be a concern unless you are intentionally trying to program the CPLD.

4.0 FPGA Pinnout and UCF File:

The following is a snapshot of a UCF file that defines the pinnout of the FPGA portion of the Zynq. Note that some of these pins are a part of SPI ports and an I2C port that goes to the onboard Dacs, ADCs, and accelerometer. These ports have timing requirements that are also included in this UCF snapshot.

```
##############################################################################
##############################################################################
#Top Level Clock
NET "Clk40"          LOC = K17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW; #40Mhz clock, Input

#User IO
NET "aUserSwitch_n"    LOC = F17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW; #BUTTON0, Input
NET "cFpgaLed[0]" LOC = P16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #LED0, output
NET "cFpgaLed[1]" LOC = P15  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #LED1, output
NET "cFpgaLed[2]" LOC = T19  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #LED2, output
NET "cFpgaLed[3]" LOC = R19  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #LED3, output

#Processor Reset
NET "system_reset_n"    LOC = J15  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;   #output

#Accelerometer (MMA8452)
NET "aAccelScl"       LOC = H17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output
NET "aAccelSda"       LOC = K18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aAccelInt_n"       LOC = L17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output

#ADC (ADS7952)
NET "aAiSpiCs_n"      LOC = F20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output
NET "aAiSpiClk"       LOC = F19  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;    #output
NET "aAiSpiMosi"      LOC = G18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output
NET "aAiSpiMiso"      LOC = G17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input

#DACs (DAC7562SDSC).
#There are 2 Dac devices selectable with the Cs[0] and Cs[1] pins.
#Note how this logic is inverted logic from a typical SPI port.
NET "aAoMxpSpiMosi_n"   LOC = G20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output
NET "aAoMxpSpiClk_n"    LOC = G19  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;    #output
NET "aAoMxpSpiCs[0]"    LOC = G15  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;     #output
NET "aAoMxpSpiCs[1]"    LOC = H15  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;    #output
NET "aAoMxpLdac"      LOC = N16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;    #output

#MXPA DIO
NET "aMxpAdio0"  LOC = V12  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio1"  LOC = T10  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio2"  LOC = U12  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio3"  LOC = R14  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio4"  LOC = T12  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio5"  LOC = U18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio6"  LOC = W13  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio7"  LOC = T15  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
```

```
NET "aMxpAdio8"  LOC = Y16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio9"  LOC = P14  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio10" LOC = P18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio11" LOC = T11  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio12" LOC = T14  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio13" LOC = N17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio14" LOC = V13  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpAdio15" LOC = W14  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aCom1Rx_n"   LOC = Y17 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input
NET "aCom1Tx_n"   LOC = Y14 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW; #output

#MXPB DIO
NET "aMxpBdio0"  LOC = P20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio1"  LOC = N20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio2"  LOC = U20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio3"  LOC = N18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio4"  LOC = W20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio5"  LOC = V20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio6"  LOC = Y19  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio7"  LOC = Y18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio8"  LOC = W16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio9"  LOC = V16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio10" LOC = R17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio11" LOC = R16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio12" LOC = R18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio13" LOC = T17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio14" LOC = V18  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aMxpBdio15" LOC = V17  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input/output
NET "aCom2Rx_n"   LOC = W19 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW; #input
NET "aCom2Tx_n"   LOC = W18 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW; #output

#Misc
NET "aDriveLow2"       LOC = F16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;   #output
NET "aDriveLow1"       LOC = L16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output
NET "aDriveLow0"      LOC = G14  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #output

NET "RESERVED0" LOC = H16  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;  #input
NET "RESERVED1" LOC = D20  | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW;

##############################################################################
#Onboard Timing Parameters
#Analog Input Spi Timing
NET "aAiSpiMiso" OFFSET = IN 6 ns VALID 9 ns BEFORE "Clk40";
NET "aAiSpiClk"  OFFSET = OUT 15 ns AFTER "Clk40";
NET "aAiSpiMosi" OFFSET = OUT 15 ns AFTER "Clk40";
NET "aAiSpiCs_n" OFFSET = OUT 15 ns AFTER "Clk40";
#Analog Output Spi Timing
NET "aAoMxpSpiCs[0]" OFFSET = OUT 20 ns AFTER "Clk40";
NET "aAoMxpSpiCs[1]" OFFSET = OUT 20 ns AFTER "Clk40";
NET "aAoMxpLdac" OFFSET = OUT 20 ns AFTER "Clk40";
NET "aAoMxpSpiClk_n" OFFSET = OUT 20 ns AFTER "Clk40";
```

```
NET "aAoMxpSpiMosi_n" OFFSET = OUT 20 ns AFTER "Clk40";
#Accelerometer I2c Timing
NET "aAccelSda" OFFSET = IN 5 ns VALID 10 ns BEFORE "Clk40";
NET "aAccelScl" OFFSET = OUT 20 ns AFTER "Clk40"; #Accelerometer I2c Parameter
NET "aAccelSda" OFFSET = OUT 20 ns AFTER "Clk40"; #Accelerometer I2c Parameter
##############################################################################
##############################################################################
```

Special Pinnout instructions:

1) In your FPGA drive aDriveLow0, aDriveLow1, and aDriveLow3 to a '0'. These are used to control some proprietary circuitry. If these are not driven low then the analog input's input impedance will be different than specified in the myRIO user manual.

2) Configure RESERVED pins as inputs. DO NOT DRIVE THESE PINS YOU MAY DAMAGE YOUR BOARD PERMENANTLY.