Contents

1 DEPRECATION NOTE
2 Application Note Number
3 Revision History
4 Abstract
5 Summary
6 Step 1
7 Step 2
8 Appendix 1 Motherboard and Daughterboard ID?s
8.1 MOTHER BOARDS
8.2 DAUGHTER BOARDS
8.2.1 Basic and Low Frequency
8.2.2 WBX
8.2.3 SBX
8.2.4 CBX
8.2.6 TwinRX
8.2.6 TwinRX
8.2.7 Others (Legacy)
8.5 USB PID

This Application Note is deprecated. Use it at your own risk!

AN-503

Date	Author	Details
Date	Author	Details

2016-05-03 Tim Fountain Initial creation

This Application Note explains how to use an Ettus Research-branded USRP with LabVIEW, and in effect, convert it into an NI-USRP RIO.

NOTE: While this process is technically possible, NI/Ettus does *not* officially support it and does *not* guarantee providing any technical support for it. The user performs this process at their own risk. The process is documented here as a convenience to users.

This document outlines the steps necessary to modify an Ettus X310 + associated daughterboards (CBX, WBX, SBX or UBX) into the identical NI model (NI USRP-294x or NI USRP-295x). Note that you must have identical daughterboards in each X310 slot for LabVIEW to function. Identical daughterboards would be 2xCBX or 2xUBX for instance.

This document was created with version 15.0 of the NI USRP driver. In NI USRP 15.5 and later there will be a single LabVIEW vi that will conduct all 3 steps automatically.

There are 2 steps that need to be done to make an X310 into a USRP RIO.

The daughterboard IDs need to be burned into the EEPROM.

The default location for the utilities is

C:\Program Files (x86)\National Instruments\LabVIEW 2015\vi.lib\LabVIEW Targets\FPGA\USRP\niusrprio_tools.llb

Note - edit path accordingly if you have a different version of LabVIEW and/or you have installed the x64 version

Use the initialize Flash.vi to load the correct daughterboard ID?s and serial numbers. The vi is auto populated with the supported daughterboard ID?s, the complete list is included in appendix 1 for reference. The serial numbers are not critical bit can be matched to the physical daughterboard serial numbers which are found on a printed label on each daughterboard.

File Edit View Favorites Help File Edit View Favorites Help C:\Program Files (x86)\National Instruments\LabVIEW 2015\ Name Backup Persistent Device Data.vi bitReverse.vi Clear Flash.vi Examine Flash.vi getDeviceInfoFromScs.vi getRioInterfaceNumberFromFpgaRef.vi getRioInterfaceNumberFromRioString.vi Initialize Flash.vi Level Ranges.vi Restore Persistent Device Data.vi sampleProjectBitfilePath.vi Update Device Correction Data.vi updateChannelCorrectionDataFromFile.vi usrpDboardId.ctl Vrite Bitfile to Flash.vi Write EEPROM Header.vi	📴 LLB Manager	-		×	
C:\Program Files (x86)\National Instruments\LabVIEW 2015\ C:\Program Files (x86)\National Instruments\LabVIEW 2015\ Name N	File Edit View Favorites Help				
C:\Program Files (x86)\National Instruments\LabVIEW 2015\ Name Backup Persistent Device Data.vi bitReverse.vi Clear Flash.vi getDeviceInfoFromScs.vi getRioInterfaceNumberFromFpgaRef.vi getRioInterfaceNumberFromRioString.vi Initialize Flash.vi Level Ranges.vi Restore Persistent Device Data.vi sampleProjectBitfilePath.vi Update Device Correction Data.vi updateChannelCorrectionDataFromFile.vi usrpDboardId.ctl write Bitfile to Flash.vi Write EEPROM Header.vi	C 🔁 🏝 🕷 🖒 🗋 🗙 🖷	*			
Name Backup Persistent Device Data.vi bitReverse.vi Clear Flash.vi getDeviceInfoFromScs.vi getRioInterfaceNumberFromFpgaRef.vi getRioInterfaceNumberFromRioString.vi Initialize Flash.vi Level Ranges.vi Restore Persistent Device Data.vi sampleProjectBitfilePath.vi Update Device Correction Data.vi updateChannelCorrectionDataFromFile.vi usrpDboardld.ctl usrpDioardld.ctl Write Bitfile to Flash.vi Write EEPROM Header.vi	C:\Program Files (x86)\National Instruments\Lab	VIEW 20	015\ 🗸	Ē)
 Backup Persistent Device Data.vi bitReverse.vi Clear Flash.vi getDeviceInfoFromScs.vi getRioInterfaceNumberFromFpgaRef.vi getRioInterfaceNumberFromRioString.vi Initialize Flash.vi Level Ranges.vi Restore Persistent Device Data.vi sampleProjectBitfilePath.vi Update Device Correction Data.vi updateChannelCorrectionDataFromFile.vi usrpDboardId.ctl write Bitfile to Flash.vi Write Bitfile to Flash.vi Write EEPROM Header.vi 	Name				^
1	 Backup Persistent Device Data.vi bitReverse.vi Clear Flash.vi Examine Flash.vi getDeviceInfoFromScs.vi getRioInterfaceNumberFromFpgaRef.vi getRioInterfaceNumberFromRioString.vi Initialize Flash.vi Level Ranges.vi Restore Persistent Device Data.vi sampleProjectBitfilePath.vi Update Device Correction Data.vi usrpDboardId.ctl usrpRioPid.ctl Write Bitfile to Flash.vi Write EEPROM Header.vi 				*

The letter revision on the X310 motherboard should be to set the HW current version (rev A = 1, B =2, etc). Set the oldest compatible version to the current HW version too. For revisions 6 and below (A-F), the 3.3v device model must be set. For revisions 7 and above (G and H), use the 1.6v device model must be set.

Initialize Flash.vi Front Panel Labview 20	
File Edit View Project Operate Tools Wind Image: State of the st	F SH
Inititialize a USRP RIO's Flash Select the USRP model (which daughterboards, bandwidth, and hardware revision), and specify daughterboard serial numbers (not required) and hardware revision information. A completed run implies successful Flash initialization (an error dialog will display if there are errors).	^
Select USRP RIO Device	
USRP-2940R (WBX, 40 MHz BW, 3.3v) Daughterboard Serial Numbers	ш
fake0 fake1	
Hardware Revision Information Hardware Current Version 7 Hardware Oldest Compatible Version	•

IQ imbalance corrections need to be loaded into the EEPROM.

The easiest way to load the IQ imbalance corrections into the daughterboard EEPROM is to create a sample USRP-RIO project. From the default LabVIEW windows, click create new project and select NI-USRP Simple Streaming:

📴 Create Project		_		×
Choose a starting point for the project	ct:			
All Templates Desktop	•))))•	NI-USRP Simple Streaming Sample Projects Implements basic streaming functionality to stream data to or from a single USRP. Mo Information	re	^
Sample Projects CompactRIO Desktop NI-579X	+))))+	NI-USRP Simple Streaming with Time Sample Projects Implements time-based streaming functionality to stream data to or from one or more More Information NI-USRP Multi-device Synchronized Simple Streaming Sample Projects	e USRPs.	
Real-Time SoftMotion		Implements streaming functionality to synchronously stream data to or from multiple More Information	USRPs.	
Additional Search				
Keyword				~
		Next Cancel	Help	

Once the project has been created, navigate to the utilities in the project window and open the Self Correct RX IQ Offset.vi:



This .vi will automatically calculate the optimal IQ imbalance correction factors and load them into the factory default location on the daughterboard EEPROM. Note there is no method at this time to load DC offset corrections without sending the unit back to the factory.

NAME	ID
B200	0x7737
B210	0x7738
B200mini	0x7739
B205	0x773a

Flex 2400 Tx MIMO A

0x001b

Name (as reported by Unknown Basic LF	UHD)	RX ID 0xfff1 0x0001 0x000f	TX ID 0xfff0 0x0000 0x000e	Notes
Name (as reported by WBX LO WBX WBX + Simple GDB WBX v3 WBX v3 + Simple GDB WBX v4 WBX v4 + Simple GDB WBX-120 WBX-120 + Simple GDB	UHD)	RX ID 0x0051 0x0053 0x0057 0x0057 0x0063 0x0063 0x0063 0x0081	TX ID 0x0050 0x0052 0x004f 0x0056 0x004f 0x0062 0x004f 0x0080 0x004f	Notes (Not registered in UHD)
Name (as reported by SBX SBX v4	UHD)	RX ID 0x0054 0x0065	TX ID 0x0055 0x0064	Notes v3
SBX v5 SBX-120		0x0069 0x0083	0x0068 0x0082	
Name (as reported by	UHD)	RX ID	TX ID	Notes
CBX-120		0x0067	0x0066	V3
OBX 120		0,0000	0,0004	
Name (as reported by	UHD)	RX ID	TX ID	Notes
UBX v0.3		0x0074	0x0073	Prototype
UBX v0.4		0x0076	0x0075	Prototype
UBX-40 v1		0x0078	0x0077	
UBX-160 V1		0X007A	0x0079	
Name (as reported by TwinRX v1.0	UHD)	RX ID 0x0091	TX ID 0xffff (Notes (0x90 Reserved)
Name	ID	Note	s	
DBS Rx	0x000)2		
TV Rx	0x000	03		
Flex 400 Rx	0x000)4		
Flex 900 Rx	0x000)5		
Flex 1200 Rx	0x000	06		
Flex 2400 Rx	0x000)7		
Flex 400 Tx	0x000)8		
Flex 900 Tx	0x000)9		
Flex 1200 Tx	0x000	Ja Na		
)d		
DBS TX CIRINOU DBSBX2	0x000			
Flex 400 Rx MIMO A	0x001	-		
Flex 900 Rx MIMO A	0x001	15		
Flex 1200 Rx MIMO A	0x001	16		
Flex 2400 Rx MIMO A	0x001	17		
Flex 400 Tx MIMO A	0x001	8		
Flex 900 Tx MIMO A	0x001	19		
Flex 1200 Tx MIMO A	0x001	la		

Flex 400 Rx MIMO B	0x0024
Flex 900 Rx MIMO B	0x0025
Flex 1200 Rx MIMO E	3 0x0026
Flex 2400 Rx MIMO E	3 0x0027
Flex 400 Tx MIMO B	0x0028
Flex 900 Tx MIMO B	0x0029
Flex 1200 Tx MIMO B	0x002a
Flex 2400 Tx MIMO B	0x002b
Flex 2200 Rx MIMO E	3 0x002c
Flex 2200 Tx MIMO B	0x002d
Flex 1800 Rx	0x0030
Flex 1800 Tx	0x0031
Flex 1800 Rx MIMO A	0x0032
Flex 1800 Tx MIMO A	0x0033
Flex 1800 Rx MIMO E	3 0x0034
Flex 1800 Tx MIMO B	0x0035
TV Rx Rev 3	0x0040
DTT754	0x0041
DTT768	0x0042
TV Rx MIMO	0x0043
TV Rx Rev 2 MIMO	0x0044
TV Rx Rev 3 MIMO	0x0045
TVRX2	0x0046
WCDMA US	0x004d
WCDMA EU	0x004e
XCVR2450 Tx - No D	iv 0x0059
XCVR2450 Tx	0x0060
XCVR2450 Rx	0x0061
Bitshark Rx	0x0070
B150 v1 TX	0x0071
B150 v1 RX	0x0072
Name	ID Notes
E300 AD9364 RevB	0x0100
E310 AD9361 RevB	0x0110
E300 AD9364 RevC	0x0101
E310 AD9361 RevC	0x0111
E330	0x0120

Name	ID	Notes
Ettus Research	0x2500	
National Instruments	0x3923	
Cypress Semiconductor	0x04b4	

Name	ID	Notes
FX2	0x8613	
FX3	0x00f3	
FX3 (Re-enumerated)	0x00f0	
B100	0x0002	
B200	0x0020	
B210	0x0020	
B200 (NI)	0x7813	
B210 (NI)	0x7814	
B200mini	0x0021	
B205	0x0022	