Lime Microsystems Limited

Surrey Technology Centre
Occam Road
The Surrey Research Park
Guildford, Surrey GU2 7YG
United Kingdom

Tel: +44 (0) 1483 685 063
e-mail: enquiries@limemicro.com

Universal Radio Development Platform User Guide

- User Guide for STREAM & UNITE7002 boards -

Version: 1.00 Last modified: 12/11/2015

REVISION HISTORY

The following table shows the revision history of this document:

Date	Version	Description of Revisions
12/11/2015	1.0	Initial version

Table of Contents

1 Introduction

The universal radio development platform, based on the Stream board and flexible, multi standard Lime transceiver boards, enables developers to implement their products for a wide variety of wireless communication applications efficiently and cost effectively. The main ideas are to:

- Accelerate the evaluation and development time.
- Experiment and evaluate new modulation schemes and wireless systems, operating over a wide frequency range.
- Easily modify and manufacture the platform for new designs using the Open Source database for the complete kit.

This document provides the following information:

- Design kit content description and first demo example
- Software installation, setup and programming of the Stream board.
- Example files for running the complete platform.

2 Complete Development Package

Complete design kit content for Stream and UNITE7002 board showed in *Figure 1*.

Figure 1 Stream & UNITE7002 complete package

Development kit content:

- Stream board
- UNITE7002 board
- 12 volt / 5 ampere power supply
- Micro-USB3 cable Type: A Male to Micro B Male
- Mini-USB2 cable Type: A Male to Mini B Male
- RF Cables SMA to U.FL
- Power cable with banana plugs for UNITE7002
- USB stick containing [link]:
    - Lime7Suite GUI and register setup for LMS7002M transceiver
    - FPGA bitstreams and project files
    - USB3 controller drivers
    - Waveforms
    - Windows drivers for UNITE7002

3 Stream Board Key Features

The STREAM development board provides a hardware platform for developing and prototyping high-performance and logic-intensive digital and RF designs using Altera's Cyclone IV FPGA and Lime Microsystems transceiver. The board provides a wide range of peripherals and memory interfaces to connect to Lime's current offering of transceiver evaluation boards and Open Source MyriadRF boards.

For more information on the following topics, refer to the respective documents:

- RFDIO connector RF evaluation boards, refer to MyriadRF project [link]
- Cyclone IV device family, refer to Cyclone IV Device Handbook [link]
- LMS7002M transceiver resources [link]
- LMS6002D transceiver resources [link]

Stream board features:

- **FPGA Features**
  - Cyclone IV EP4CE40F23C7N device in 484-pin FBGA
  - 39'600 LEs
  - 1134 Kbits embedded memory
  - 116 embedded 18x18 multipliers
  - 4 PLLs
- **FPGA Configuration**
  - JTAG mode configuration
  - Serial mode configuration via Cypress FX3
- **Memory Devices**
  - 2x64MB (16bit) SDRAM
- **Micro SD Card Socket**
  - SD card access via USB HOST controller
- **Ethernet**
  - USB HOST controller
  - 10/100/1000 Mb/s RJ45
  - 3 status LEDs
- **USB Interface**
  - Embedded Dual USB HOST 2nd generation
  - Cypress FX3 Supper Speed USB 3rd generation controller
- **Display**
  - DVI transmitter (TFP410), HDMI Jack
- **High-Speed mezzanine connectors**
  - FMC (FPGA Mezzanine Card) LPC connector for UNITE7002 board
  - RFDIO (FX10-80P) High speed connector for MyriadRF family boards
- **Connections**
  - Mictor E5346A Agilent Test Equipment connector
  - 12V DC power jack
  - GPIOs headers
  - 2x USB2A Jack

  - USB3B Micro Jack
- **Clock System**
  - 30.72MHz on board oscillator
  - Programmable clock generator for the FPGA reference clock input and RF mezzanine boards
  - Locking to external clock circuit
  - U.FL clock input
- **Board Size** 110mm x 124mm (4.3" x 4.9")

3.1 STREAM board overview

Stream board version 2, revision 1 picture with highlighted major connections showed in the *Figure* 2.

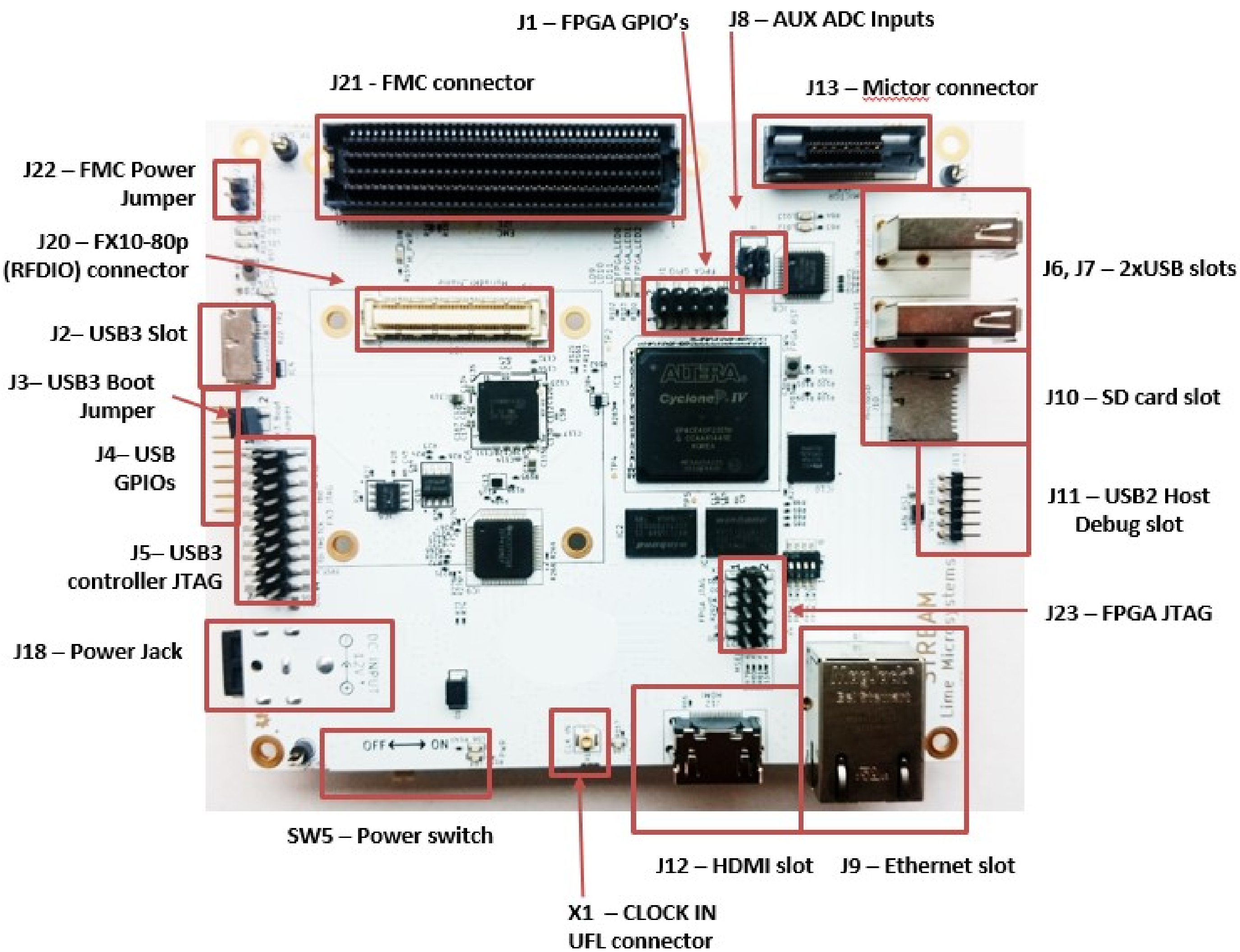

Figure 2 STREAM Development Board

Board components description showed in the *Table 1* and *Table 2*.

Table 1. Board components

Featured Devices		
Board Reference	***Type***	***Description***
IC1	FPGA	Cyclone IV EP4CE40F23C7N, 484-FBGA
J20	RFDIO connector	Provides digital data and control to MyriadRF boards
J21	FMC connector	Provides digital data and control to UNITE7002 board
IC6	USB3.0 microcontroller	Cypress FX3 Supper Speed USB 3rd generation controller
IC8	USB2.0 microcontroller	Embedded Dual USB HOST 2nd generation controller
IC9	IC	DVI transceiver
IC10	IC	Ethernet GbE Controller
Configuration, Status and Setup Elements		
Board Reference	***Type***	***Description***
J3	Jumper	USB3.0 boot from SPI flash IC5
J5	JTAG chain pinheader	USB3.0 microcontroller's debugging pinheader
J23	JTAG chain pinheader	FPGA programming pinheader for Altera USB-Blaster download cable
J11	Pinheader	USB2.0 microcontroller's debugging pinheader for VNC2 debug module
SW2	Button	USB3.0 microcontroller reset button
R1, R79, R2, R80, R3, R81, R4, R114	0 Ohm resistor	FPGA MSEL[3:0], by default Passive Serial Standard configuration scheme is selected
LD1, LD2, LD3	Status LED	User defined FX3 LEDs
LD4	Power LED	Illuminates when USB 5V power is present
LD9, LD10, LD11	Status LED	User defined FPGA LEDs
LD12, LD13	Status LED	User defined VNC2 LEDs
SW3	Button	Ethernet controller's reset button
R94, R95; R92, R93; R90, R91; R86, R87	0 Ohm resistor	Ethernet MODE[3:0], default GMII/MII mode
R96, R97; R100, R101; R108, R109	0 Ohm resistor	PHYAD[2:0], default physical address "00000"
SW4	Button	FPGA reset button
General User Input / Output		
Board Reference	***Type***	***Description***
J1	Pinheader	8 FPGA GPIOs
J4	Pinheader	USB3.0 controller's 6 GPIOs, USB2.0 controllers 3 GPIOs
J8	Pinheader	Two 12bit ADC inputs

Table 2 Board components

Memory Devices		
Board Reference	***Type***	***Description***
IC2, IC3	DDR2 memory	512Mbit DDR2 SDRAM with a 16-bit data bus
J10	microSD socket	microSD card
IC5	Flash memory	4Mbit flash, FX3 boot
IC15	Flash memory	16Mbit flash, FPGA configuration
Communication Ports		
Board Reference	***Type***	***Description***
J2	USB3.0 microB connector	
J6, J7	USB2.0 A socket	Two USB2.0 A type sockets
J12	HDMI connector	
J13	Connector	‘MICTOR’- Agilent test equipment interface
J9	Connector	RJ45 Ethernet connector
Clock Circuitry		
Board Reference	***Type***	***Description***
XO5	TCXO	E6245LF 30.72MHz oscillator
IC11	IC	Programmable clock generator for the FPGA reference clock input and RF boards
U2	IC	ADF4002 phase / frequency detector
X1	U.FL connector	RF connector for external clock
LD5	LED	Illuminates than onboard oscillator phase is locked to external clock
Power Supply		
Board Reference	***Type***	***Description***
J18	DC input jack	12V DC power supply
LD6, LD7, LD8	LED	Illuminates than board is powered on
J22	Jumper	Jumper to provide 12V DC to FMC connector

3.2 STREAM board architecture

The heart of the STREAM Development board is Altera Cyclone IV FPGA. It's main function is to transfer digital data between MyriadRF board or UNITE7002 to PC through USB or Ethernet ports. The block diagram for Stream board showed in the *Figure 3*.

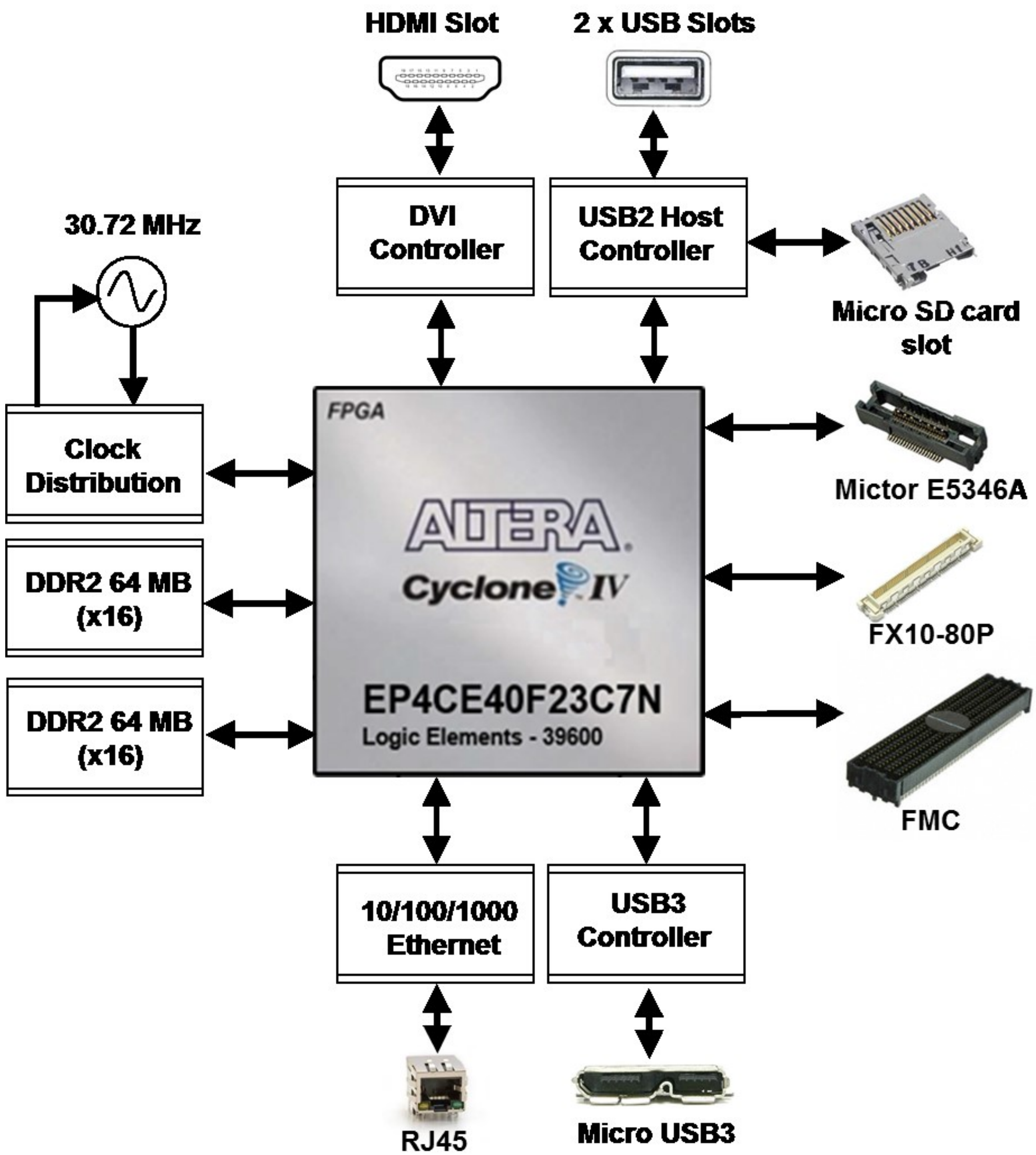

Figure 3 STREAM Development Board Block Diagram

3.2.1 LMS7002M based boards connectivity (FMC / RFDIO)

Stream board is designed to interface with UNITE7002 board via FMC connector and MyriadRF boards via to FX10-80P connector. The FX10-80P connector pinout has been standardized and known as RFDIO standard [link].

LMS7002M digital interface requires 12-bit data, *IQSEL_Enable*, *FCLK, MCLK, TXNRX* signals for each transmit and receive ports. The simplified interface block diagram showed in the *Figure 4*.

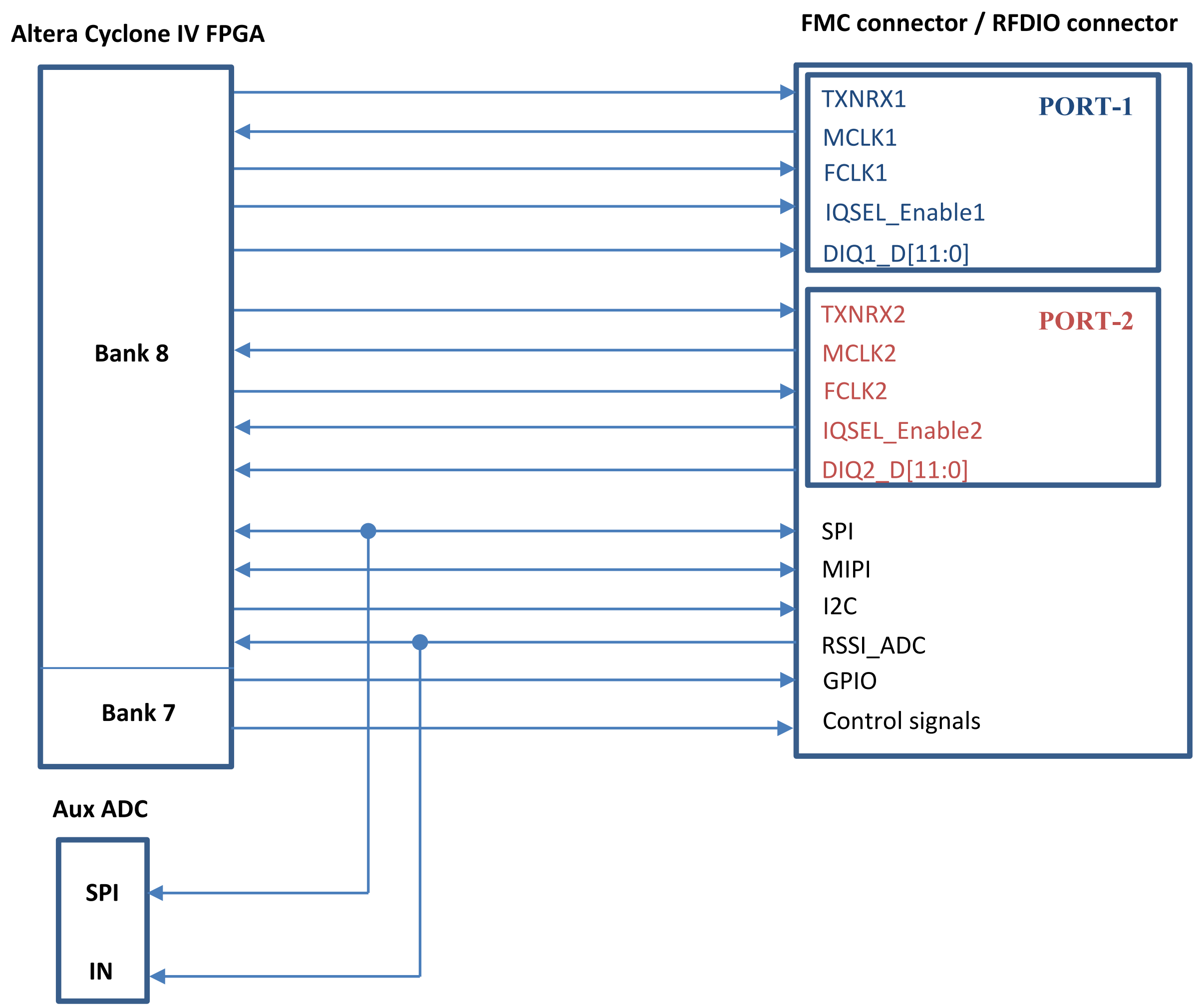

Figure 4 Simplified MyriadRF (RFDIO connector) and UNITE7002 (FMC connector) connection to FPGA block diagram

The interface and control signals are described below:

- **Digital Interface Signals:** MyriadRF and UNITE7002 boards are using same data bus *DIQ1_D[11:0]* and *DIQ2_D[11:0], IQSEL_Enable1* and *IQSEL_Enable2*, *FCLK1* and *FCLK2*, *MCLK1* and *MCLK2* signals to transfer data to / from FPGA. Indexes 1 and 2 indicate transceiver digital data *PORT-1* or *PORT-2*. Any of these ports can be used to transmit or receive data. By default *PORT-1* is selected as transmit port and *PORT-2* is selected as receiver port. The *FCLK#* is input clock and *MCLK#* is output clock for LMS7002M transceiver. *TXNRX* signals sets ports directions. For LMS7002M interface timing details refer to LMS7002M transceiver datasheet page 12-13. [link].

- **SPI Interface:** LMS7002M transceiver is configured via 4-wire SPI interface; *LMS_SPI_MOSI*, *LMS_SPI_MISO*, *LMS_SPI_SCK*, *LMS_SPI_CS*. The SPI interface controlled from FPGA Bank 8 (2.5V).

- **I2C Interface:** used to control external clock synthesizer on UNITE7002 board. The signals *LMS_I2C_CLK*, *LMS_I2C_DATA* connected to FPGA Bank 8 (2.5V).

- **MIPI Interface:** *MIPI_DATA*, *MIPI_SCK* – MyriadRF board micro controller programming interface connected to FPGA Bank 8 (2.5V).

- **RSSI_ADC Interface:** *RSSI_ADC_0*, *RSSI_ADC_1* – LMS7002M receiver power detector analog output. Index corresponds to MIMO channel. These signals are connected to 12-bit ADCs (U9, U10). ADCs are controlled via the SPI interface by FPGA signals: *LMS_SPI_SCK, LMS_SPI_MISO, ADC_SPI_CS0, ADC_SPI_CS1*. The signals connected to FPGA Bank 8 (2.5V).

- **GPIO signals:** *LMS_GPIO[6:0]* – are used only to control MyriadRF boards RF switches. The signals connected to FPGA Bank 7 (3.3V).

- **Control Signals:** these signals are used for optional functionality:

  - *LMS_RXEN, LMS_TXEN* – receiver and transmitter enable / disbale signals connected to FPGA Bank 8 (2.5V).
  - *LMS_RESET* – LMS7002M reset connected to FPGA Bank 7 (3.3V).
  - *MyriadPRSNT* – indication signal for MyriadRF board. If board is not connected to RFDIO port, signal is *"Logic High"*, if board is connected - signal is *"Logic Low"*.
  - *LMSS_iqsel1_dir, LMS_iqsel2_dir* – IQSEL_Enable direction control signal for UNITE7002 board digital buffers. By default *LMS_iqsel1_dir* is *"1"* and *LMS_iqsel2_dir* is *"0"*. This condition configures *IQSEL_Enable1* as transmitter (input signal) for LMS7002M; *IQSEL_Enable2* is configured as receiver (output signal) from LMS7002M.
  - *LMS_dio_dir_ctrl1, LMS_dio_dir_ctrl2* – digital data direction control for UNITE7002 board buffers, by default *DIQ1_D[11:0]* is transmitter data and *DIQ2_D[11:0]* is receiver data.
  - *LMS_dio_buff_oe* – UNITE7002 board digital buffers outputs enable signal, by default outputs are enabled. This signal is used to prevent short circuit between buffers outputs and LMS7002M outputs. Buffer outputs must be enabled after other signal directions are set.

- *LMS_SBEN* – is additional SPI enable pin used to control optional phase locked loop on UNITE7002 board. This controlled is used when frequency error of the crystal oscillator on UNITE7002 board has to be calibrated with external equipment. The *LMS_SBEN* is a *"chip select"* signal for SPI interface. The additional PLL is using same SPI lines as LMS7002M – *LMS_SPI_SCK, LMS_SPI_MISO*.

3.2.2 SDRAM

Stream board has two 64MB (16bit bus) SDRAM ICs (W9751G6KB [link]) connected to double data rate pins on Cyclone IV 1.8V Bank 3 and Bank 4. The memory can be used for data manipulation at high data rates between transceiver and FPGA. The memory is also used to load Linux operation system.

3.2.3 USB 3.0 controller

LMS 7 Suite software controls Stream board via the USB3 microcontroller (CYUSB3013 [link]). The data transfer to / from the board, SPI communication, FPGA configuration is done via the USB3 controller. The controller signals description showed below:

- FX3 digital data *FX3_DQ[15:0]* is connected to Cyclone IV 1.8V Bank 2.
- *FX3_CTL[12:0]* – FX3 control signals.
- *FX3 GPIO[5:0]* are available on J4 pinheader.
- *FX3_LED[2:0] (LD1, LD2, LD3)* – user defined debugging LEDs.
- *FX3_GPIO42, FX3_GPIO43, FX3_GPIO44* – connected to Cyclone IV 3.3V Bank 7 user defined GPIOs.
- *FX3_CONF_DONE, FX3_NSTATUS, FX3_DATA0, FX3_NCONFIG, FX3_DCLK* – are used to program FPGA via FX3 controller.
- *PMODE[2:0]* – boot options, by default boot from SPI and USB boot is enabled. If J3 jumper is present FX3 will boot from IC5 flash memory.
- *FX3 SPI* interface is used to program IC5 flash memory, boot from IC5 flash memory, read/write U1 flash memory. Also FX3 SPI is connected to FPGA 3.3V Bank 7. U1 flash memory is used to load FPGA configuration via FX3 controller.
- SW2 – resets FX3, all IOs are in tristate state during a hard reset.
- J5 – FX3 JTAG programming/debugging pin header.

3.2.4 Ethernet Controller

Stream board is equipped with Ethernet (Micrel KSZ9021GN [link]) port that can be used as alternative high speed data interface to PC. By default Ethernet port is configured to GMII/MII mod. 8-bits transmit data and 8-bits receive data are connected to FPGA Bank 5 (3.3V). The controller signals description showed below:

- *CLK125_EN* – enables/disables 125 MHz clock output from pin 55 (*CLK125_NDO*), by default it is *"0"* and clock output is disabled.
- *LED_MODE* – default *"0"*, tri-color dual LED mode
- Ethernet controller physical address is set to *"00000"*.
- SW3 – button, resets Ethernet controller

3.2.5 DVI controller

DVI controller (TI TFP410 [link]) 12-bit data bus *DVI_d[11:0]* and control signals are connected to FPGA Bank 1 (2.5V).

3.2.6 Mictor connector

Mictor 38-pin connector (J13) can be used as extension to Agilent Logic Analyzer equipment. 2x 16-bit data bus connected to FPGA Bank 6 (3.3V).

3.2.7 USB 2.0 Host

The USB 2.0 Host controller (VNC2-48L1B [link]) data bus *VIN_d[7:0]* connected to FPGA Bank 5 (3.3V).

USB2 controller can be programmed via FPGA using *VIN_debug*, *VIN_reset*, *VIN_prog* signals or via *"VNC2 Debug Module"* J11 connector

The microSD card is hosted by USB2 controller. The control commands are issued via the SPI interface.

3.2.8 Clock Distribution

Stream board has onboard 30.72 MHz TVCXO that is reference clock for FPGA (signal *CLK_FPGA2*) and for MyriadRF board (signal *CLK_IN*). See block diagram of the clock distribution system in *Figure 5*.

The optional clock generator (Si5351C [link]) can generate any reference clock frequency, starting from 8 kHz – 160 MHz, for MyriadRF and UNITE7002 boards digital interface clocks (*FCLK1, FCLK2*), FPGA reference clocks (*CLK_FPGA0, CLK_FPGA1*).

The onboard PLL (ADF4002 [link]) is used to synchronize onboard TVCXO with external equipment (via X1 U.FL connector) to calibrate frequency error. The ADF4002 is programmed by FX3 controller. The LD5 - illuminates when onboard oscillator frequency error is corrected.

X1 connector can also be used to supply external reference clock (fitting R151, removing R150).

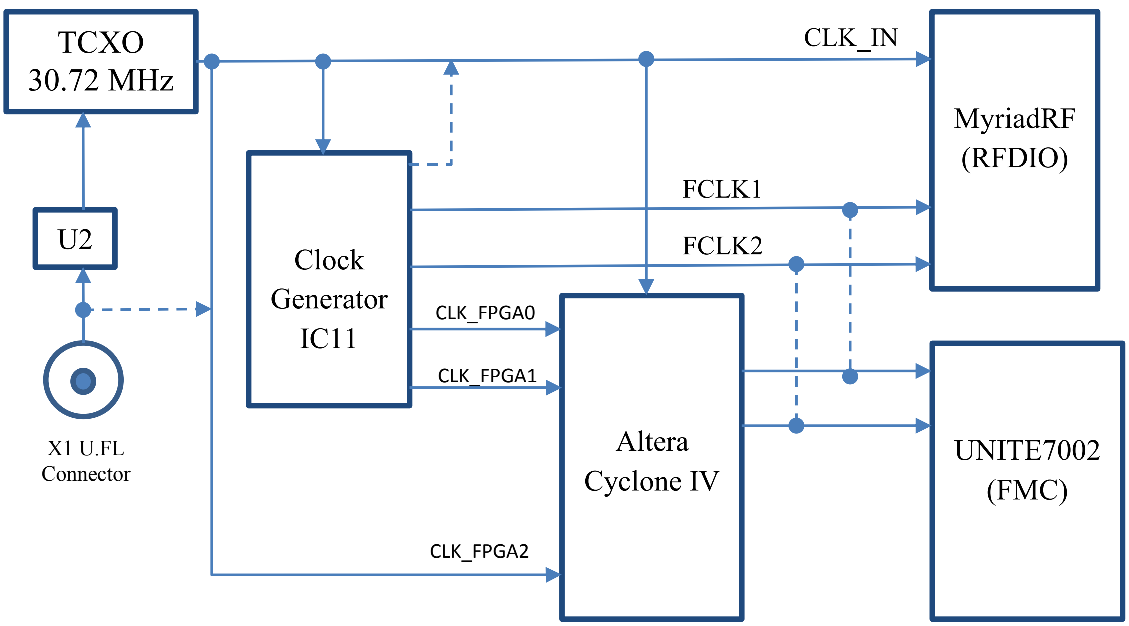

Figure 5 Clocks block diagram

4 Getting Started with Design Kit

The Stream and UNITE7002 design kit comes with LMS 7 Suite software, which enables the control of the LMS7002M transceiver, run the *"FFTviewer"* to analyse the ADC spectrum, load wanted waveforms to FPGA. Two example waveforms are available in the kit:

- Single tone generated in the digital domain by a programmable logic-based
- W-CDAM TM1 with 64ch waveform

The digital signals are driven from the Altera FPGA to the DAC within the LMS7002M to produce a complex analog I&Q output, then mixed with an adjustable frequency RF carrier through the quadrature modulator. The resulting RF signal is transmitted to the analyser through the TX-side. The incoming RF signal is converted to baseband through the quadrature de-modulator, digitized through the ADC and sent to the FPGA. The digitized signal can be analysed with Lime7Suite software.

4.1 DEMO Setup

The demo setup is showed in *Figure 6* . This demo uses single control software "*lms7suite.exe"* to control UNITE7002 board and Stream board.

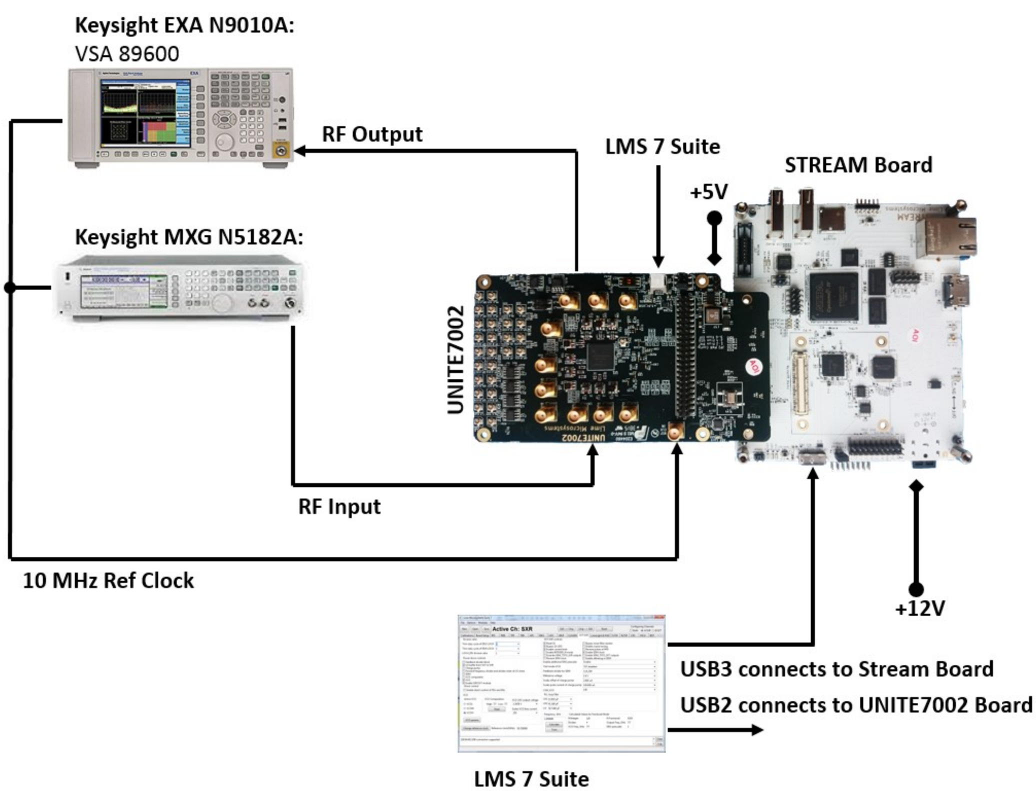

Figure 6 Demo Setup for LMS7002EVB and Stream board

4.2 DEMO Procedure

The DEMO procedure steps are showed below:

1. Connect the DEMO setup as shown in Figure 6. To measure Tx EVM the VSA89601B software is required.
2. Power up the kit.
3. Connect *lms7suite* software to the boards
4. Setup UNITE7002 board:
    a. Load the pre-set file
    b. Synchronize Analyser with UNITE7002 board
5. Calibrate TX path
6. Load the test waveform
7. Measure EVM for Tx Path
8. Run FFTviewer to analyse receiver
9. Calibrate Rx path

4.2.1 Power up the kit

The Stream board comes preprogrammed and ready to use. Once board is connected to power supply, toggle power switch (SW5) on the board. The LED1 starts flashing immediately indicating that USB3 microcontroller is ready for operation. The LED3 is constantly illuminating, indicating that FPGA is loading the bitstream. See
Figure 7.

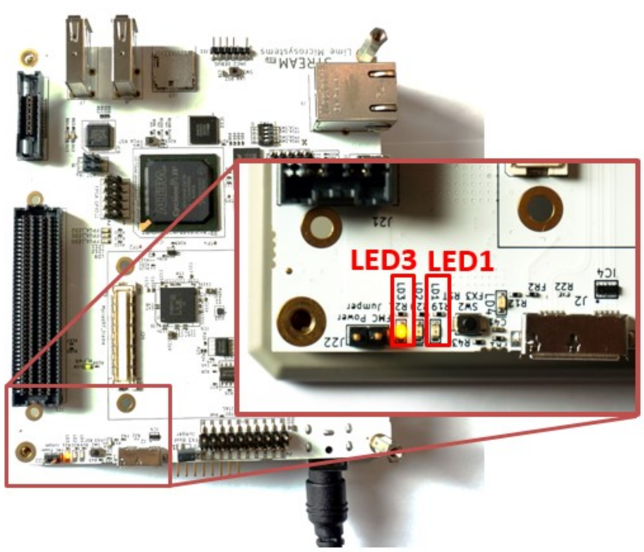

Figure 7 LED3 illuminates, indicating the FPGA programing procedure

When LED3 stops to illuminate, the board is ready to connect to LMS 7 Suite.

4.2.2 Connect *lms7suite* to the boards

When DEMO setup is ready, run the "*lms7suite.exe*" software, select **Options, Communication Settings** in top menu. New pop-up window should appear. Select COM port dedicated to the UNITE7002 board and **Cypress USB StreamExample (Stream)** name for Stream board. See figure *Figure 8.*

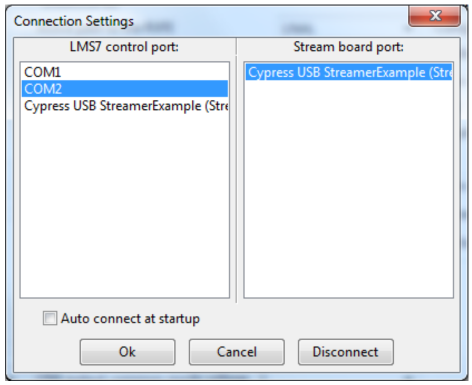

Figure 8 Comunicaton port selection for UNITE7002 and Stream boards

When boards are connected, you should see the indication in bottom of the main GUI window, see figure *Figure 9*.

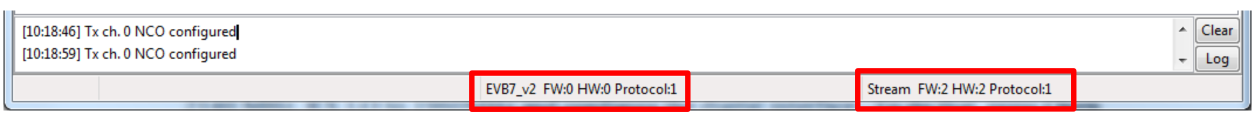

Figure 9 lms7suite board connections indication

__NOTE__: If **Communication Setting** window shows up as empty, install windows drivers for the board. Please follow the procedures described in the chapter *"5.2 USB3 Windows driver installation procedure"* and *"6.1 USB2 Windows driver installation procedure"*

4.2.3 Setup UNITE7002 Board

When boards are connected to the lms7suite software, the registers for the LMS7002M can be configured:

a. Load the register pre-set file for the LMS7002M transceiver. This will set Tx PLL LO to 2140 MHz, RX LO to 1960MHz and configure the digital interface. To do that, press **Open** button in the GUI front panel. See *Figure 10*.

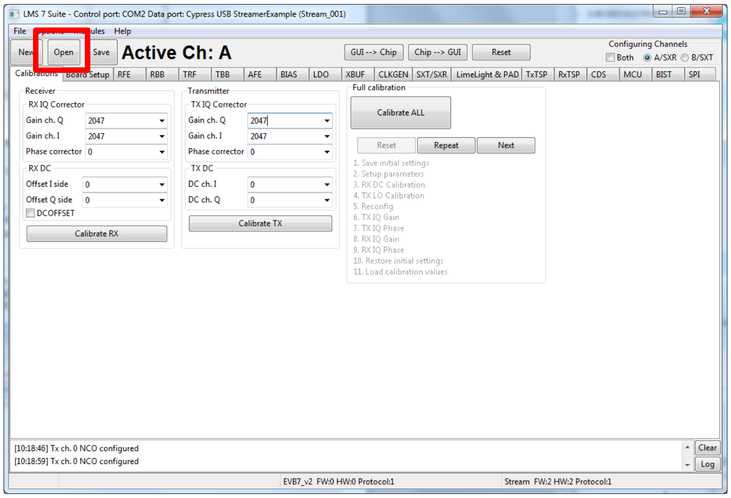

Figure 10 Select the pre-set file for transceiver

Select the ***wfm_tx_rx_61MHz.ini*** file in the **..\Design Kit Software Package\GUI** location, and press **OPEN**.

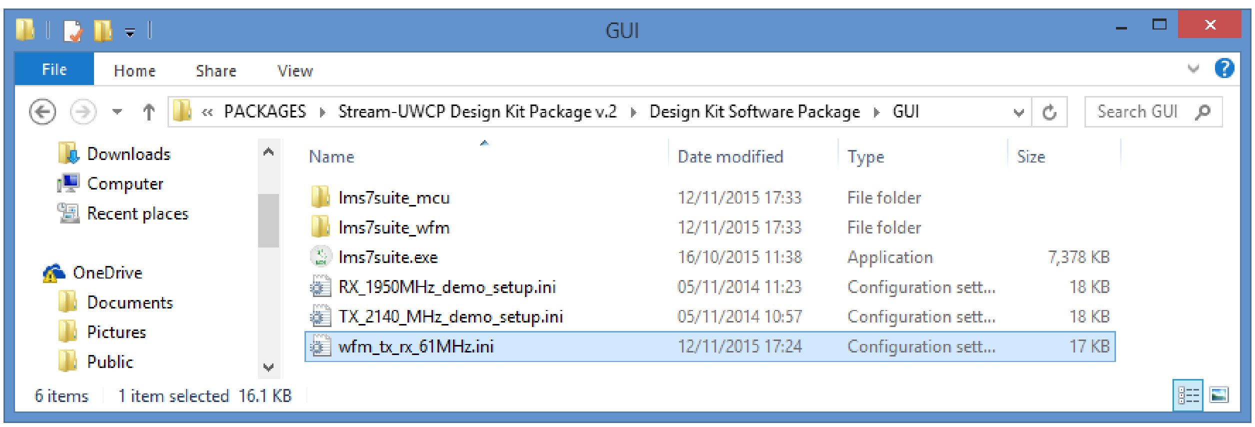

Figure 11 Select the *wfm_tx_rx61MHz.ini* file

In order for the changes to take effect press **GUI→ Chip**, as shown below in *Figure 12*.

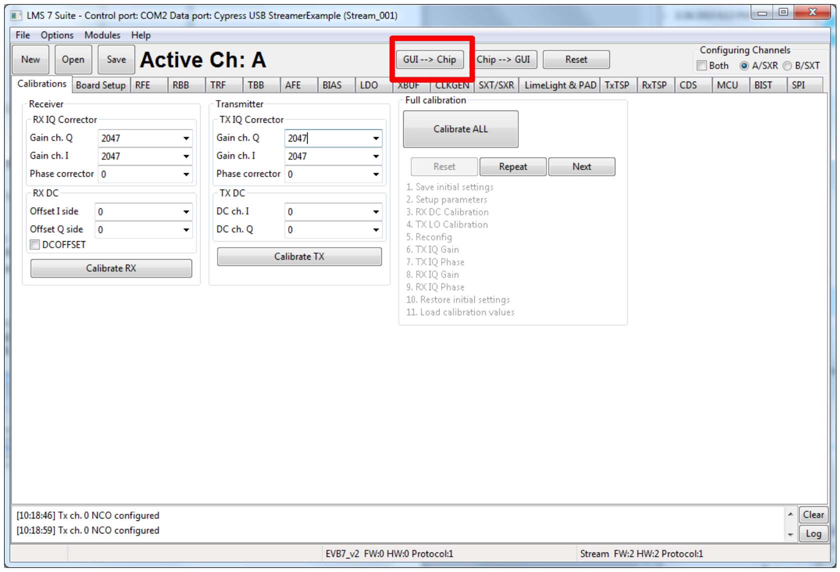

Figure 12 Load register setup to LMS7002M

At this point you should see the TX LO at 2140 MHz on analyser screen.

NOTE: If TX LO appears to be not locked, select the **B/SXT** channel in top right of the GUI, go to **SXR/SXT** tab and press **Calculate** and **Tune**.

NOTE: The index in the *.ini file name indicates the interface speed between LMS7002M and FPGA. To be able to run supplied waveforms files with GUI please select file *wfm_tx_rx_61MHz.ini.*

b. The UNITE70002 has to be synchronized with analyser in order to correct frequency error. To do that, connect 10 MHz reference signal coming from analyser to X18 connector on the LMS7002 board. Select the **Bord Setup** tab in GUI and press **Synchronize**. See *Figure 10* The LD2 should light up, which indicates that board is synchronized.

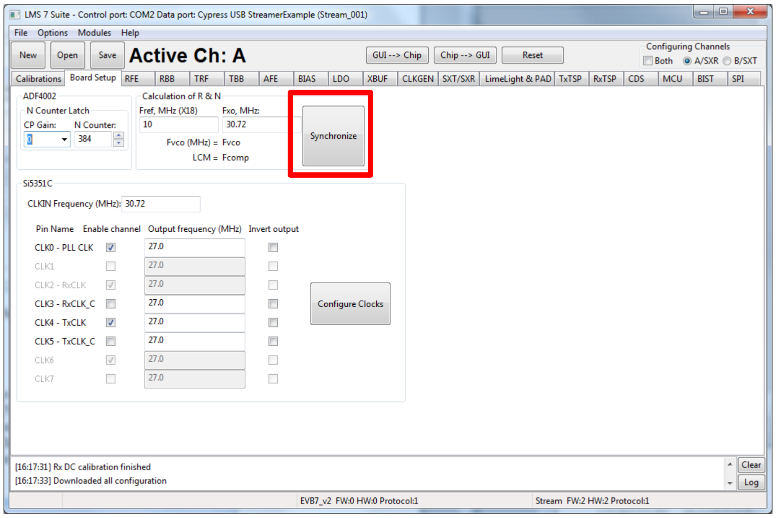

Figure 13 Board level setup

4.2.4 Calibrate TX path

The LO leakage and IQ imbalance have to be calibrated for the LMS7002M transceiver in order to get optimum performance for Tx EVM measurement. The IQ imbalance calibration is done by generating CW and adjusting IQ phase/gain error for IQ mismatch. Th LO leakage calibration is doem by adjust DC offset registers. The internal test NCO can be enabled for this purpose. To do this, select **TxTSP** tab in *lms7suite* and select the **Test Signal** as input for Tx path, as showed in figure below.

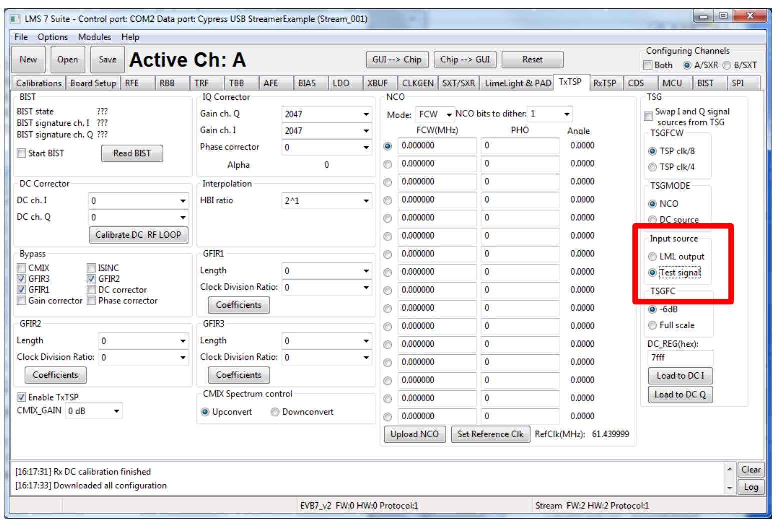

Figure 14 Enable the test NCO

__NOTE__**:** Before configuring **TxTSP** tab, select the **A/RXT** channel in top right of the GUI, On the transmitter output you should see the wanted CW with 3.8MHz offset from LO, unwanted SSB on the other side of spectrum and LO leakage. See *Figure 15*.

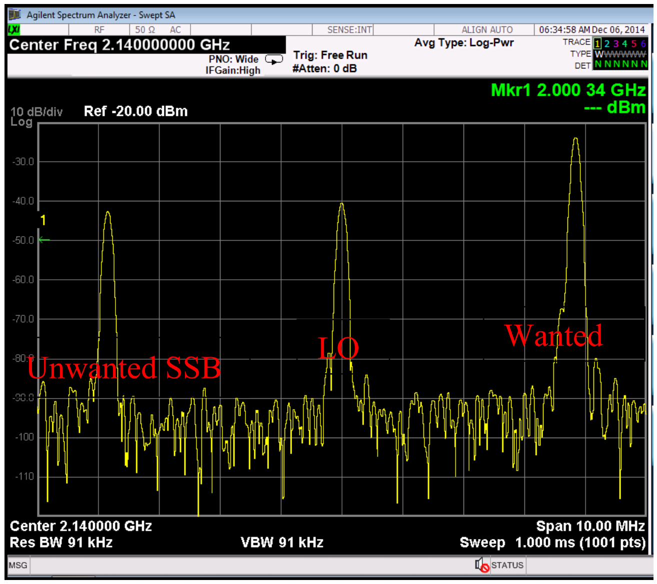

Figure 15 Not calibrated Tx Output

To do the LO leakage calibration, select **TxTSP** tab in the lms7suite GUI and adjust the **DC Corrector** settings for channel I and Q separately to get minimum LO leakage.

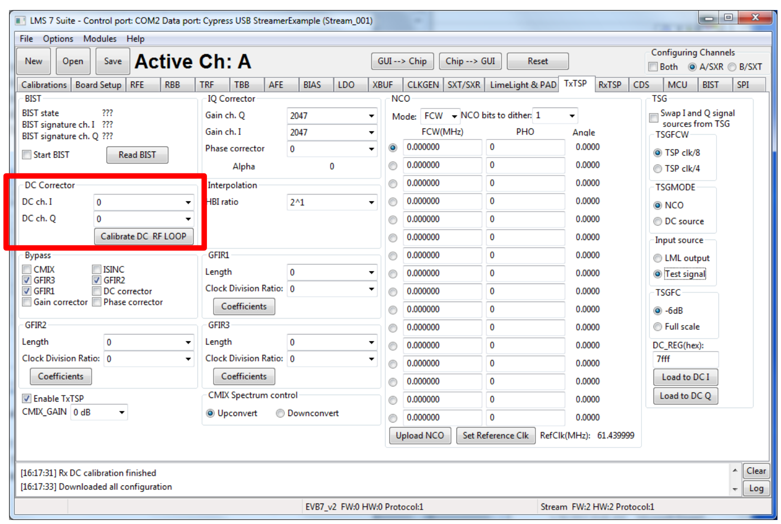

Figure 16 DC offset block control

To calibrate Unwanted SSB, use the **IQ Corrector** controls in the **TxTSP** tab. Change **I ch. gain** or **Q ch. gain** followed by **Phase correction** to reduce the Unwanted SSB.

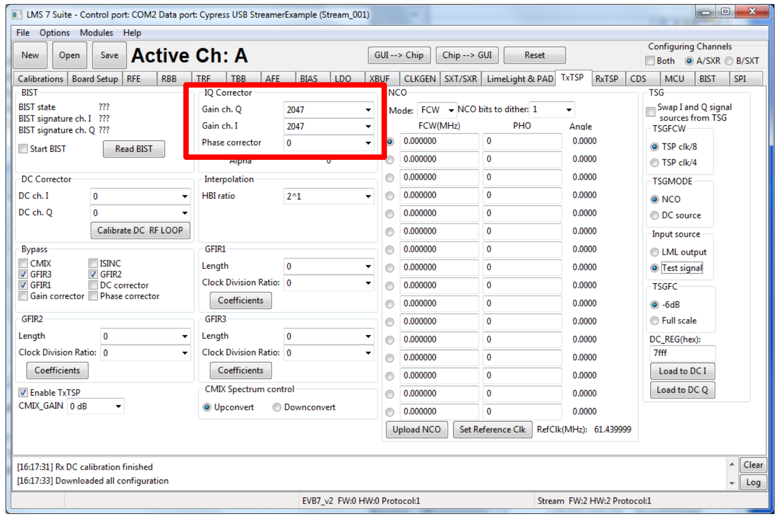

Figure 17 IQ Corrector block control

Calibrated Transceiver TX output should look like in the *Figure 18*.

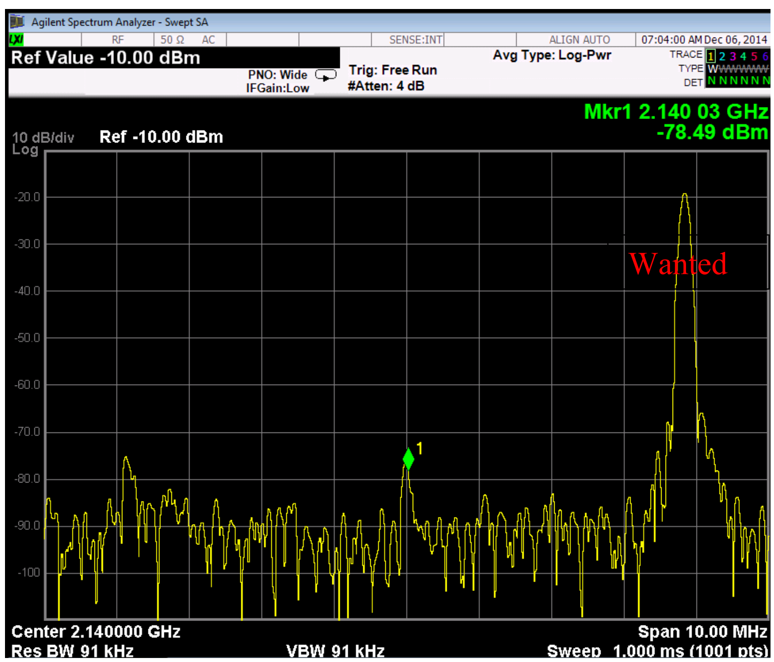

Figure 18 Calibrated Tx output

Once TX is calibrated the settings can be saved and can be recalled after chip power cycle. After calibration is complete and configure Tx path to accept data from Stream board; go to **TxTSP** and select **LML output** under Tx **Input Source** has to be selected to in TxTSP tab. See *Figure 14*.

NOTE: The Tx IQ and LO leakage calibration procedure can be done using auto calibration routines. The routines are accessed from **Calibration** tab in the GUI.

4.2.5 Load waveform for Tx Path

The programed FPGA is acting as waveform player for LMS7002M transceiver. In order to load the waveform, select **Modules** from top menu, then **FPGA Controls** from the drop down menu. See *Figure 19*.

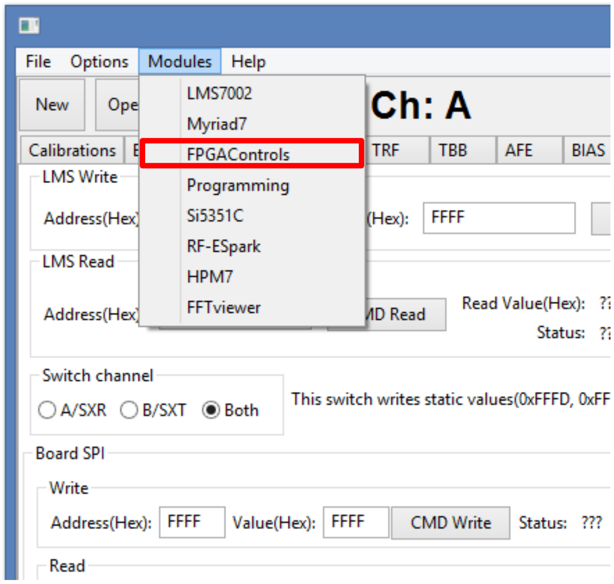

Figure 19 Select FPGA Control window

New window will appear in the bottom of the GUI, offering you to load supplied waveforms or custom waveforms. Please select to load WCDMA waveform by clocking on **W-CDMA button**. See *Figure 20*.

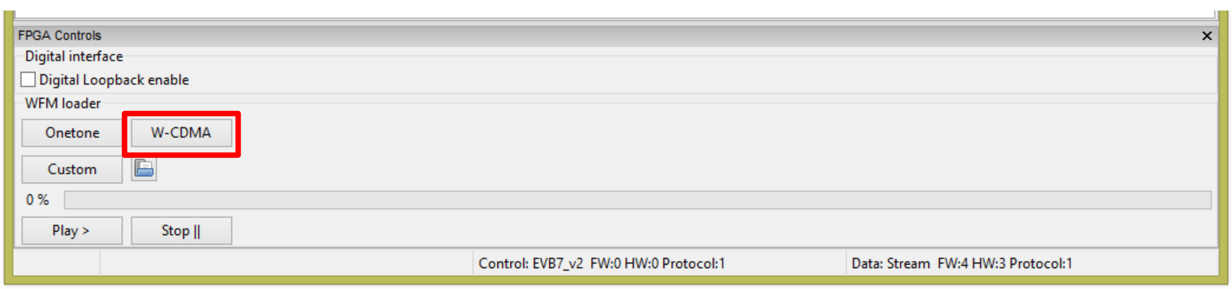

Figure 20 Waveform selection

The file loading process to the FPGA is shown by indication bar, see *Figure 21*.

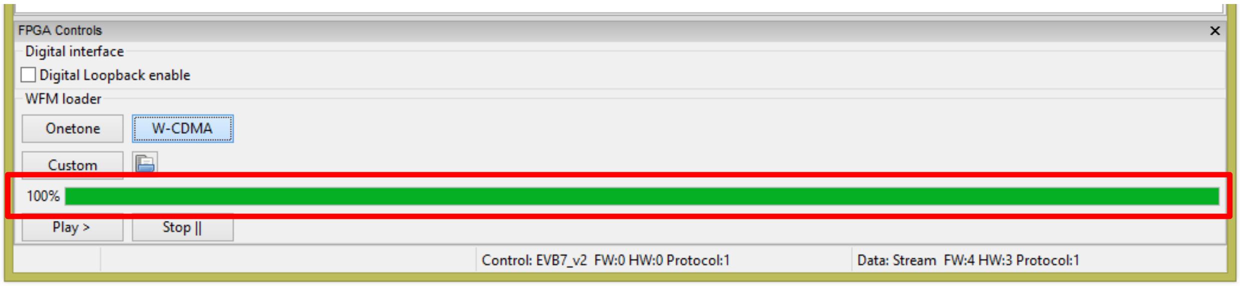

Figure 21 Loaded waveform indication

4.2.6 Digital Loopback Enable

On this FPGA there is also implemented option to receive data from LMS7002M receiver and stream back on to LMS7002M transmitter. In order to enable this option, click on the 'Digital Loopback enable' check box in the 'FPGA Control' Module Figure 22.

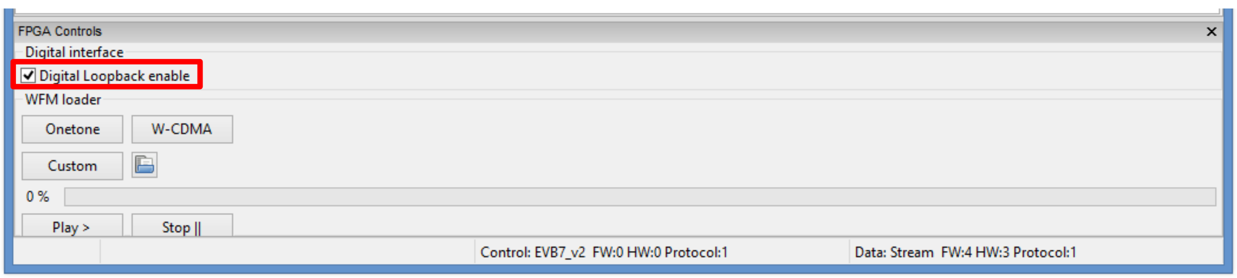

Figure 22 Select Digital Loopback enable

4.2.7 Measure Tx Path EVM

When W-CDAM waveform is loaded, the system start transmitting it instantly. The Tx output should look like in *Figure 23* and *Figure 24*.

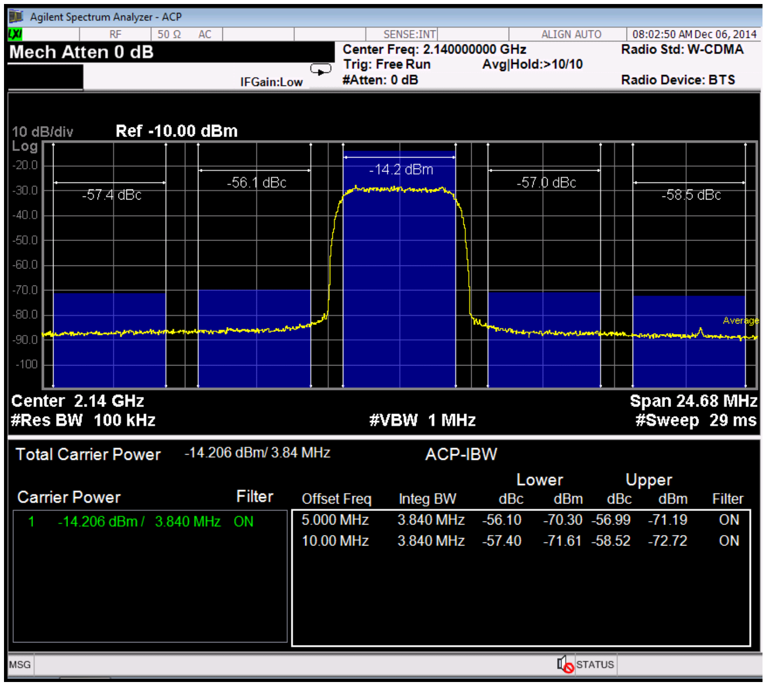

Figure 23 LMS7002M transmitter ACPR with WCDMA modulation

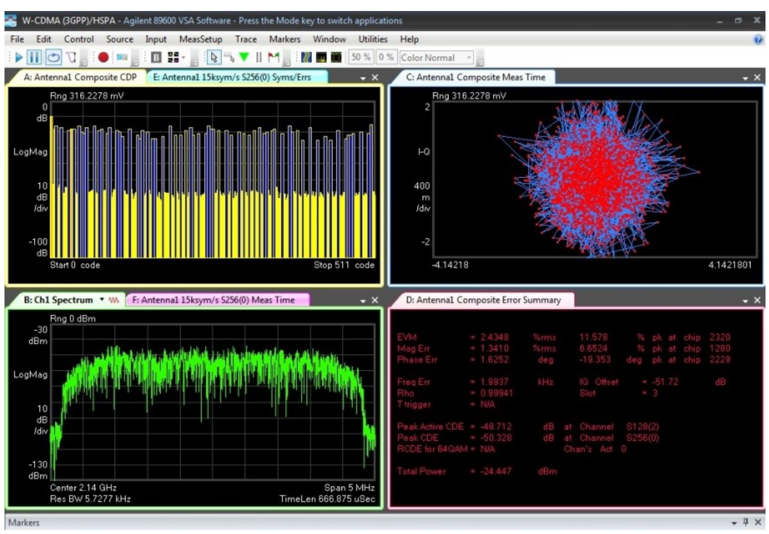

Figure 24 LMS7002M transmitter EVM (2.4 %) with WCDMA modulation

4.2.8 Run FTTviewer to analyze receiver spectrum

FFTviewer module is a part of lms7suite software. To run FFTviewer, go to top menu, select **Modules** and choose **FFTviewer**. See *Figure 25.*

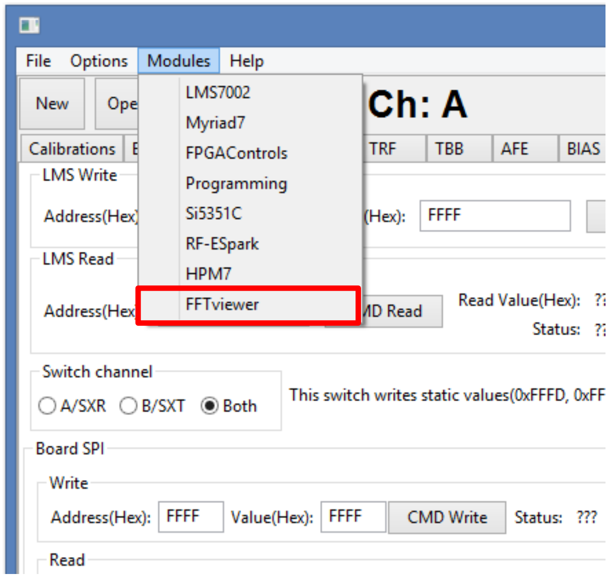

Figure 25 lms7suite module menu to select FFTviewer

FFTviewer control window will appear. Before start capturing data, set the **Sampling frequency**, select windowing function and press **Start**. See *Figure 26.*

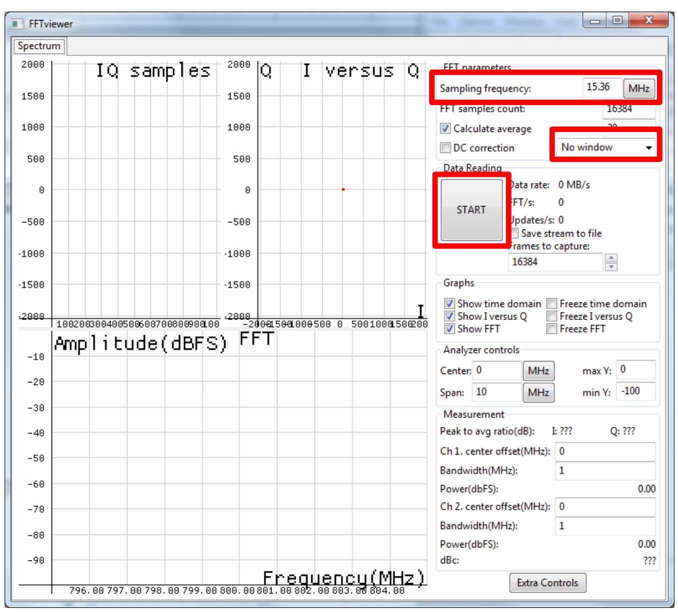

Figure 26 FFTviewer Controls

At this point, the FFTviewer start capturing data. Connect the generator to selected UNITE7002 receiver path. In the *Figure 27* showed the FFTviewer data capture with 1 MHz CW signal offset from LO.

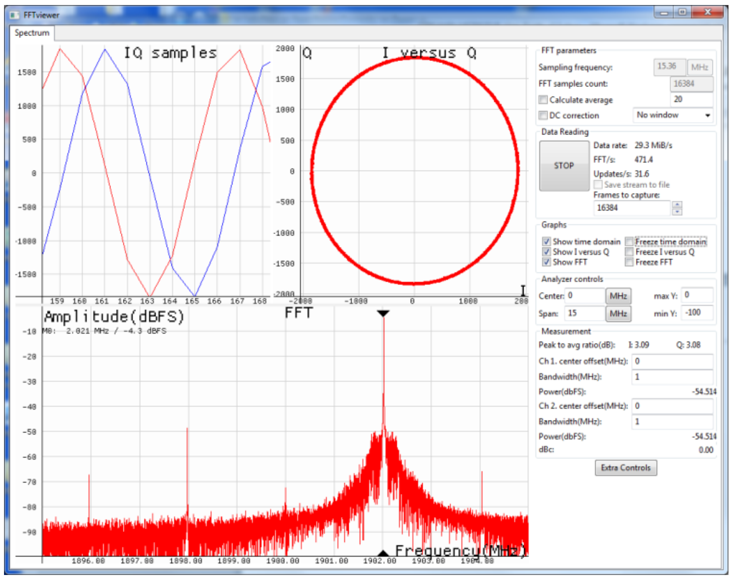

Figure 27 FFTviewer window in operation

4.2.9 Calibrate Rx path

Rx DC offset and Rx Unwanted SSB calibration routines have to be executed to calibrate receiver path. The Rx DC offset calibration split in two parts; Analog DC Offset calibration and digital DC offset removal procedure.

To execute Analog DC Offset calibration, select the **RFE** tab in the main GUI window. Make sure that you have selected channel A. In the **DC** box, change **Mixer LO signal** to **0.621 V** and click on **Calibrate DC** button. See *Figure 28* below.

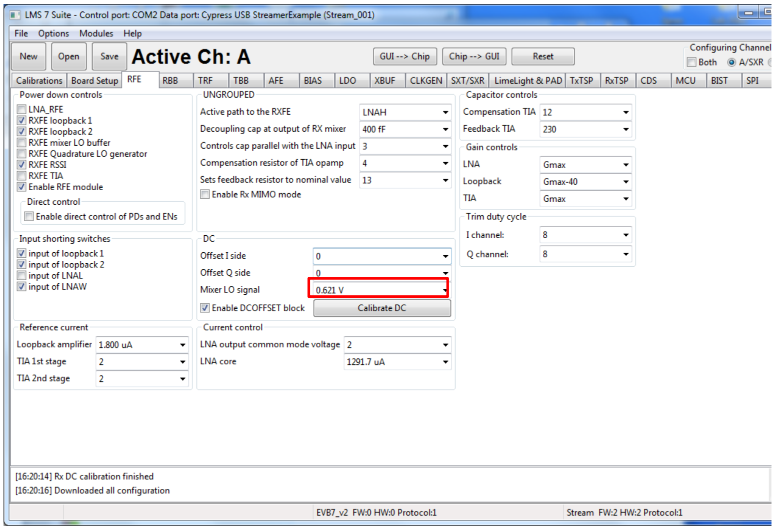

Figure 28 Calibrate RX DC offset

New window will pop-up indicating that calibration is completed. This also updates the DC offset calibration values in the **DC** box.
For residual DC offset calibration you need to enable the **DC corrector** in **RxTSP** tab. See *Figure* 29. It should be enabled by default.

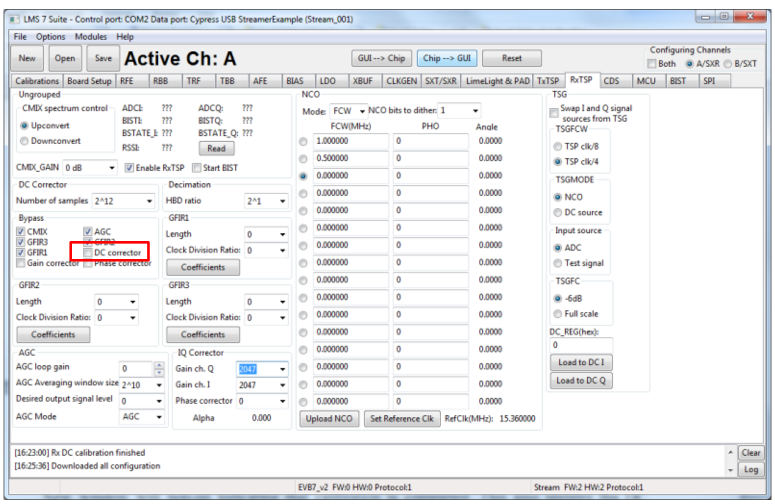

Figure 29 Enable DC corrector in RxTSP

The unwanted SSB can be seen on FFTviewer window by applying signal to one of the transceiver inputs. See *Figure 30*.

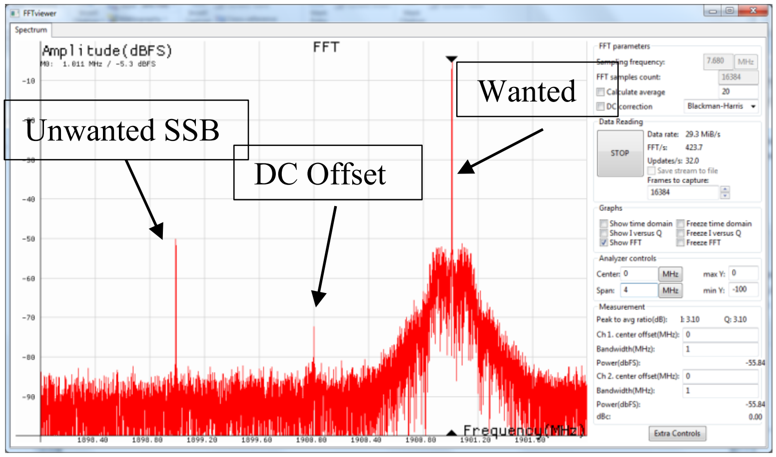

Figure 30 Receiver spectrum with un-calibrated IQ imbalance

To calibrate RX IQ imbalance go to **RxTSP** tab on *lms7suite* GUI. On **IQ Correction** box adjust **Gain ch**. **I** or **Gain ch. Q** followed by **Phase correction** to reduce the Unwanted SSB. See *Figure* 31.

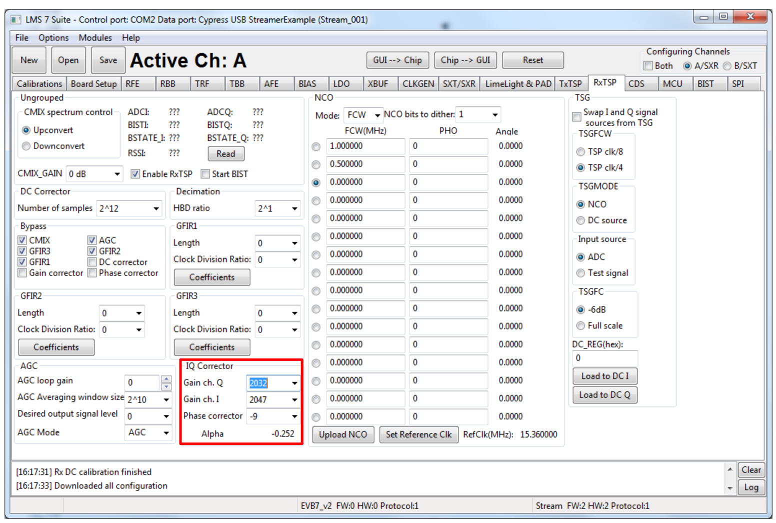

Figure 31 Rx IQ correction block

Calibrated receiver spectrum should look like in the *Figure 32*.

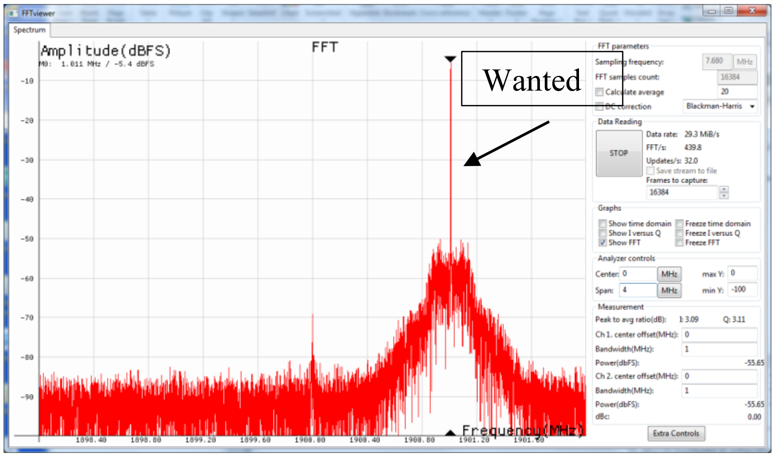

Figure 32 Calibrated Rx Spectrum

5 Appendix I

This chapter guides through the USB3 interface installation for the Stream board and Windows

5.1 FX3 microcontroller drivers installation

The communication between Stream board and PC (Lime7Suite) is done via the USB3 interface. Initially, Stream board comes with preprogramed drivers and ready to use. If new drivers require to be installed or firmware update, the steps have to be taken:

- Install windows drivers. Follow chapter "*5.2 USB3 Windows driver installation* procedure*"*.
- Install USB3 microcontroller drivers. Follow chapter *"5.3 Firmware installation for USB microcontroller"*.

5.2 USB3 Windows driver installation procedure

Once Stream board is connected, follow the installation procedure below.

1. Press *"Start Menu"* and right click on *"Computer"*, select *"Properties*" and *"Device Manager*".

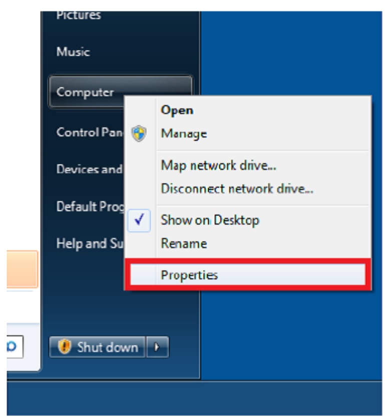

Figure 33 Open computer properties

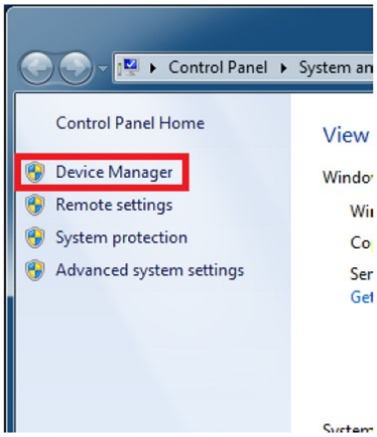

Figure 34 Open device manager

2. When Stream board is plugged in, on *"Device Manager*" menu it appears as *"WestBridge"* under *"Other devices"*. Right click on the *"WestBridge*" and select *"Update Driver Software"*.

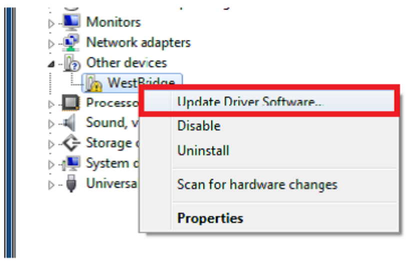

Figure 35 Update driver software

3. Select driver installation manually and choose driver from the following location: ..\ ***Design Kit Software Package\Windows drivers\Stream_Win\bin***.

Choose the driver which is suitable for the operating system running:

- Windows XP (wxp)
- Windows Vista (vista)
- Windows 7 (win7)
- Windows 8 (win8)

CPU type:

- x86(32bit-i386)
- x64(64bit-amd64)

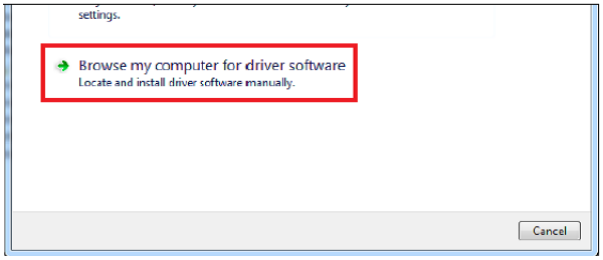

Figure 36 Browse for driver software

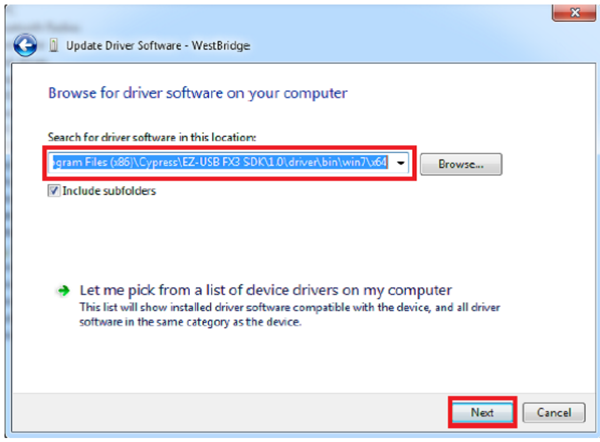

Figure 37 Select driver location

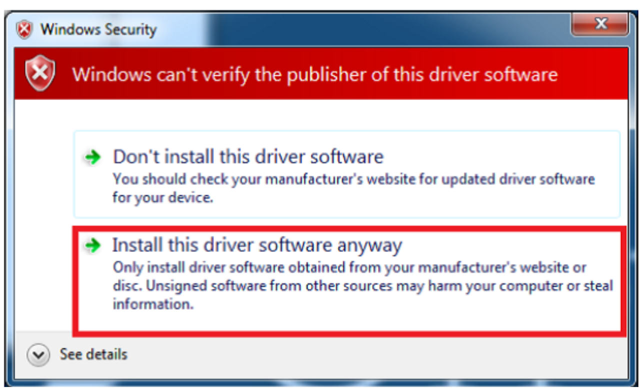

Figure 38 Confirm installation of unsigned driver

4. After successful installation *"Cypress USB BootLoader"* will appear under USB controller devices.

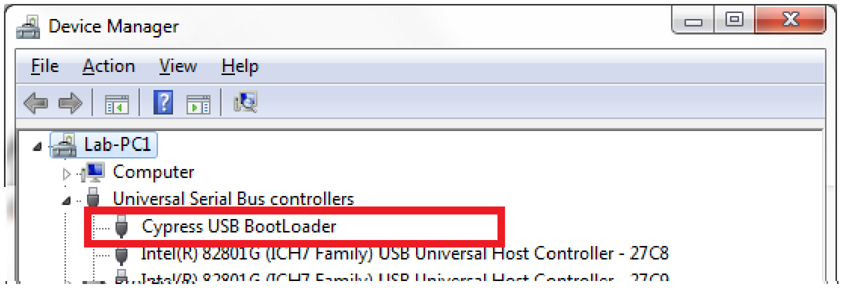

Figure 39 Device manager window after installation

NOTE: If you are using Windows 7 64 bit OS, you must disable **Driver Signature Enforcement.** To do this, Restart you PC, press F8 at startup and choose **Disable Driver Signature Enforcement**. This step is required to done once.

NOTE: If you are using Windows 8 or later, to disable driver signature enforcement manual can be found in this [link].

5.3 Firmware installation for USB microcontroller

For CYUSB3013 USB microcontroller firmware installation, please Cypress Control center utility *"USB Control Center"*, which can be found in this [link].

Cypress FX3 USB microcontroller has an integrated boot loader, which starts automatically after power-up or reset.

If FLASH memory is empty or connector J3 (on Stream board) is open, USB3 microcontroller boots-up with factory firmware. Run the *"USB Control Center"* application and in the menu select *"Cypress USB BootLoader"* line as shown in *Figure 40.*

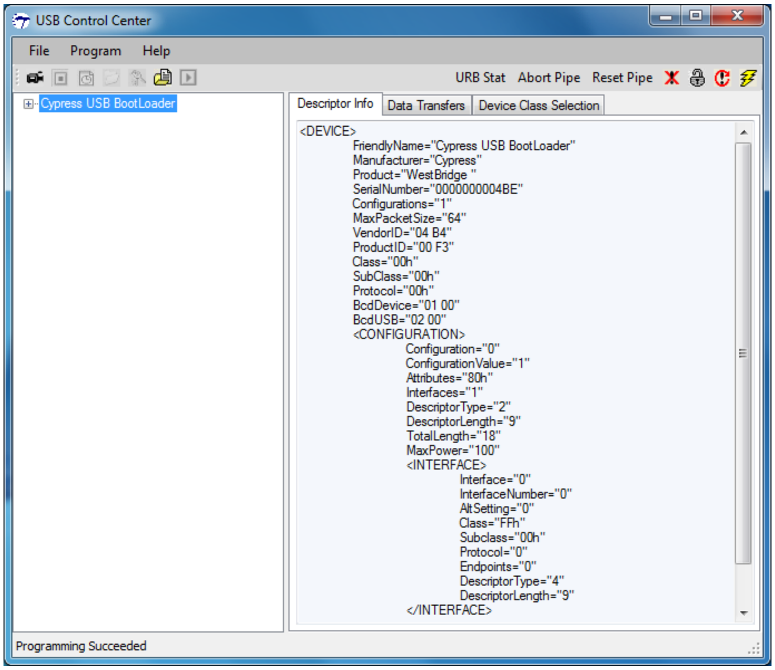

Figure 40 Default FX3 firmware, supplied by internal logic

There are two ways of uploading the firmware to USB3 microcontroller:

- Program internal RAM memory. Follow procedure described in chapter *"5.3.1 Uploading firmware to the FX3 RAM"*. The memory will be cleared after first power cycle.
- Program external FLASH memory connected to USB3 controller. Follow procedure described in chapter *"5.3.2 Uploading firmware to empty FLASH"*. The USB3 microcontroller will boot from FLASH memory after every power-on.

5.3.1 Uploading firmware to the FX3 RAM

Start *"CyControl.exe*" application and select **Cypress USB BootLoader** as shown in *Figure 40*. Choose menu command **Program → FX3 → RAM**. In the new pop-up window, select ***stream (v2r2 FW4) 2015 07 07.img*** file provided (..***Design Kit Software Package\MCU Firmware\Stream_FW*** folder) and press **Open**. Status bar of the **USB Control Center** application will indicate **Programming RAM**. This message will change to the **Programming succeeded** after programming is done.

If you expand **Cypress USB StreamerExample** line in **USB Control Center** application now, you will see different USB configuration as shown in *Figure 41*.

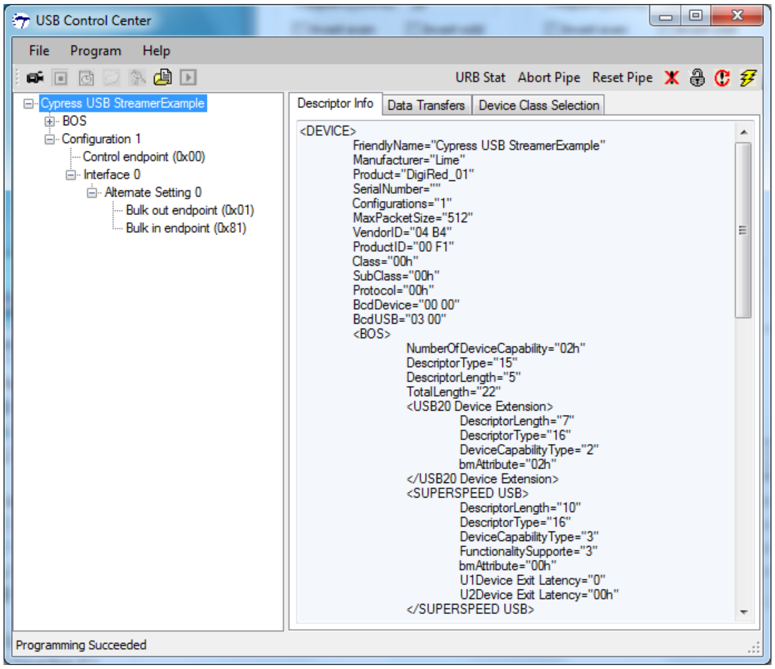

Figure 41 FX3 after custom firmware is downloaded

5.3.2 Uploading firmware to empty FLASH

If external FLASH is empty, short the jumper J3 and connect Stream board to the PC. Start *"CyControl.exe"* application and select **Cypress USB BootLoader** as shown in *Figure 40*. Choose menu command **Program → FX3 → SPI FLASH**. In the status bar you will see **Waiting for Cypress Boot Programmer device to enumerate....** and after some time window will appear. Select provided ***stream (v2r2 FW4) 2015 07 07.img*** file (..***Design Kit Software Package\MCU Firmware\Stream_FW*** folder) and press **Open**. Status bar of the **USB Control Center** application will indicate **Programming of SPI FLASH in Progress….** This message will change to the **Programming succeeded** after FLASH programming is done.

<u>NOTE</u>: USB3 microcontroller mi will boot firmware uploaded to FLASH each time after power-on if jumper J3 is shorted.

5.3.3 Uploading firmware to non-empty external FLASH memory

To update external FLASH memory with new firmware, follow these steps:

1. Disconnect Stream from USB port.
2. Make sure that jumper J3 is open.
3. Connect Stream board to USB port.
4. Short jumper J3.
5. Do the steps described in section *5.3.2.*

6 Appendix II

This section describes the USB2 driver installation procedure for UNITE7002 board.

6.1 USB2 Windows driver installation procedure

The steps to install windows drivers are as follows (please note that these steps may vary based on the specific version of Windows software being used and you may need to be logged in as Administrator to accomplish them):

1. Connect UNITE7002 board to your PC via the USB cable.

2. Go to **Control Panel > System > Device Manager**

3. Locate **USB to LMS7002M** under **Other devices** and press right click to select **Properties,** see *Figure 42*.

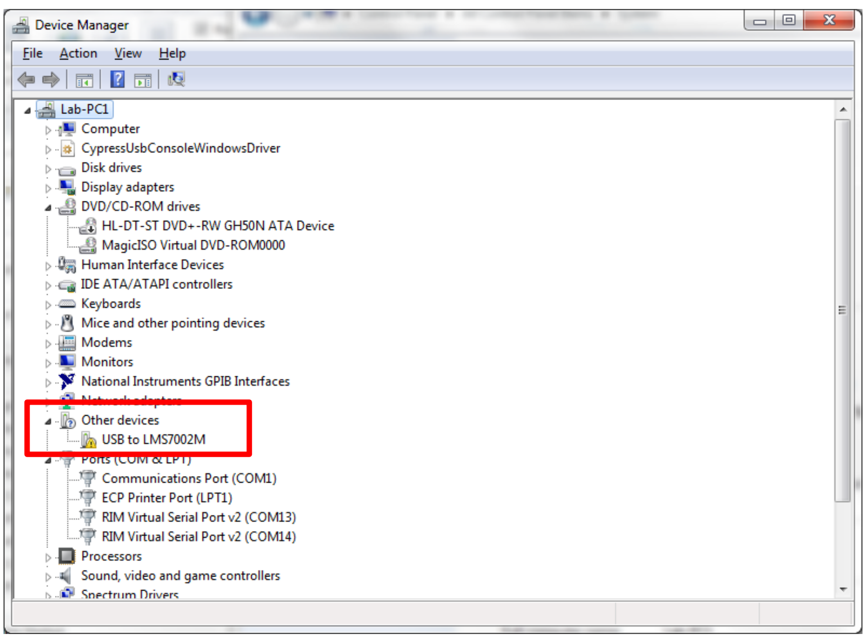

Figure 42 Device Manager content

4. When a new window pops-up press **Update Driver**, see *Figure 43*.

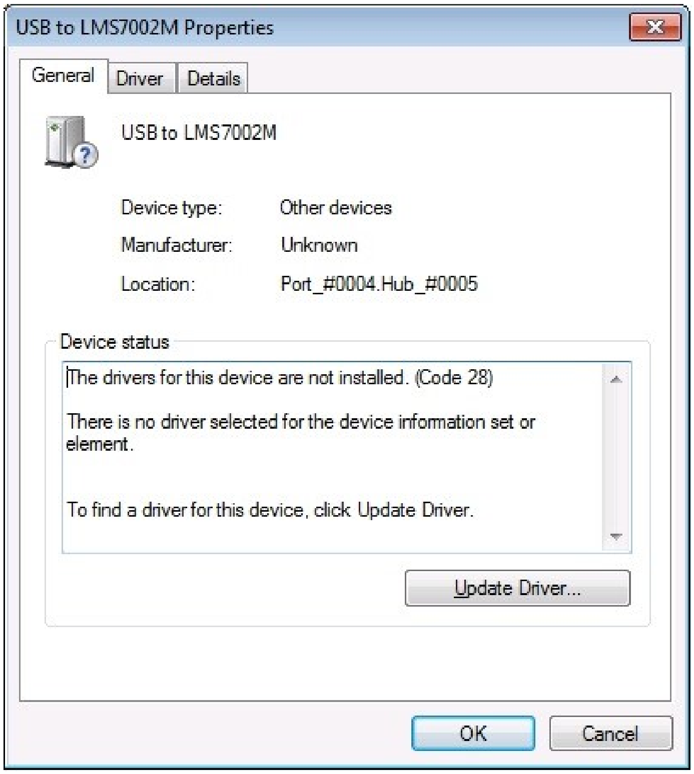

Figure 43 Device properties

5. Select **Browse my computer for driver software,** locate the driver provided with the kit and press **Next**, see *Figure 44*.

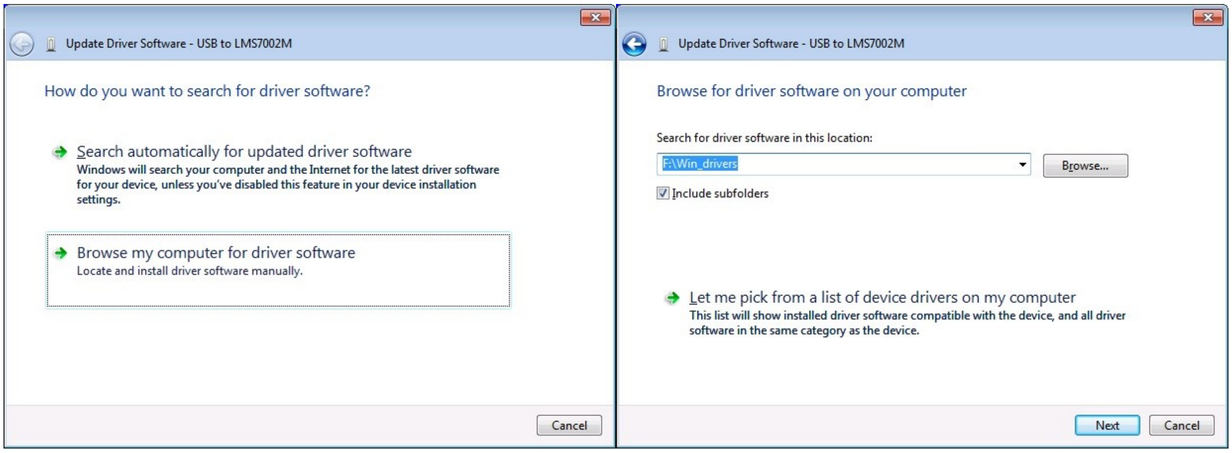

Figure 44 Locate USB driver.

6. If the **Windows Security** window appears, select **Install this driver software anyway,** see *Figure 45*.

Figure 45 Hardware wizard. Install driver softwaer anyway

Windows should proceed to install drivers at this stage. Generally, the installation procedure of the USB drivers for UNITE7002 board, has to be done once.

<u>NOTE</u>: Before running the control software, unplug then plug your device back into your computer.

6.2 Determining Serial Port

After driver installation, Windows will assign a serial port to UNITE7002 board. To check your board serial port number, please follow these steps:

1. Go to **Control Panel > System > Device Manager**

2. Locate **USB Virtual Serial Port** under **Ports (COM & LPT)**

Note that in this system example it has enumerated as COM2 (*Figure 46*).

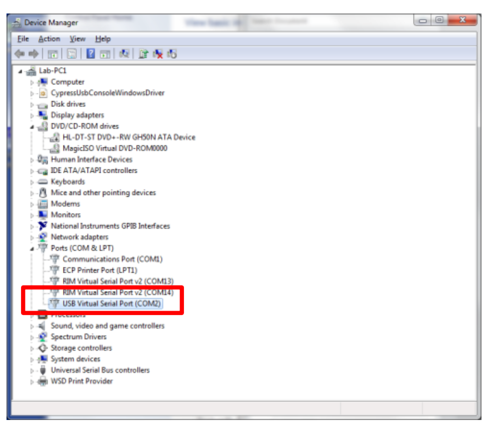

Figure 46 Check for new communication port

7 Appendix III

This section describes how to load custom bitstream to Stream board FPGA.

7.1 Load bitstream to FPGA

The Altera Cyclone IV FPGA which sits on the Stream board can be programmed using "*lms7suite"*software. To call FPGA programing function, go to **Modules** form top menu and select **Programing** form the drop down menu. See *Figure 47.*

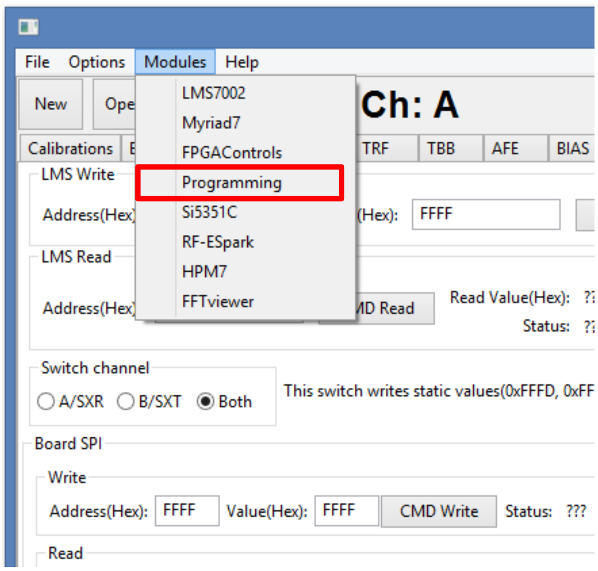

Figure 47 lms7suite module menu to select FPGA programing tool

New control section should appear in the bottom of the main window, as shown in the *Figure 48*.

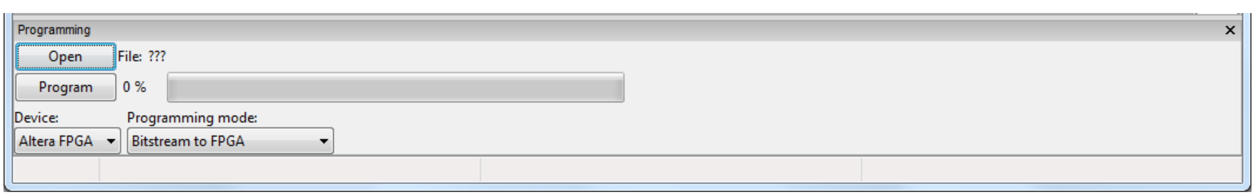

Figure 48 FPGA programing tool interface

Software loads raw binary files (*.rbf) [link] to FPGA and it offers couple options to do that, see *Figure 49.*

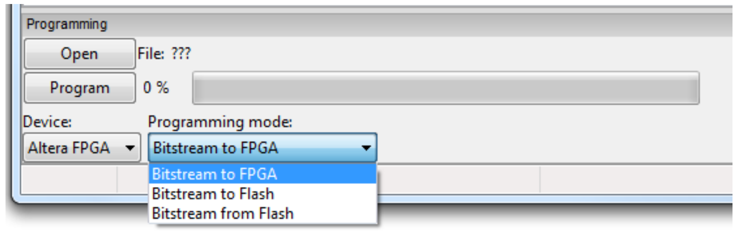

Figure 49 FPGA programing options

The programing functions are described below:

1. **Bitstream to FPGA –** this function loads selected *.rbf file from PC to FPGA. Select your wanted bitstream file by clicking **Open** and initiate FPGA programing by clicking on **Program.**
2. **Bitstream to FLASH –** this function loads selected *.rbf file from PC to external FPGA FLASH memory. Select your wanted bitstream file by clicking **Open** and initiate FLASH memory programing by clicking on **Program.**
3. **Bitstream from FLASH -** loads bitstream from external FPGA FLASH memory to FPGA. To initiate programing click on **Program** button.

The new massage will come up when the programing is finished. See Figure 50.

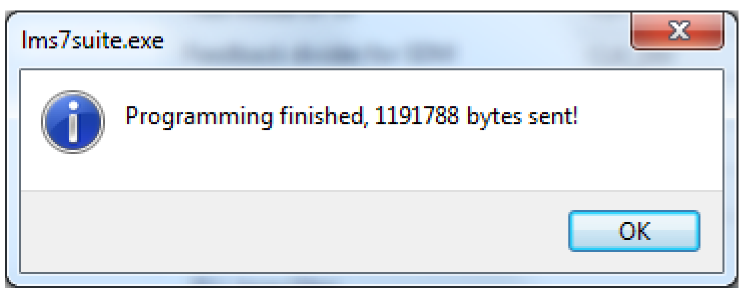

Figure 50 Successfully FPGA programing massage