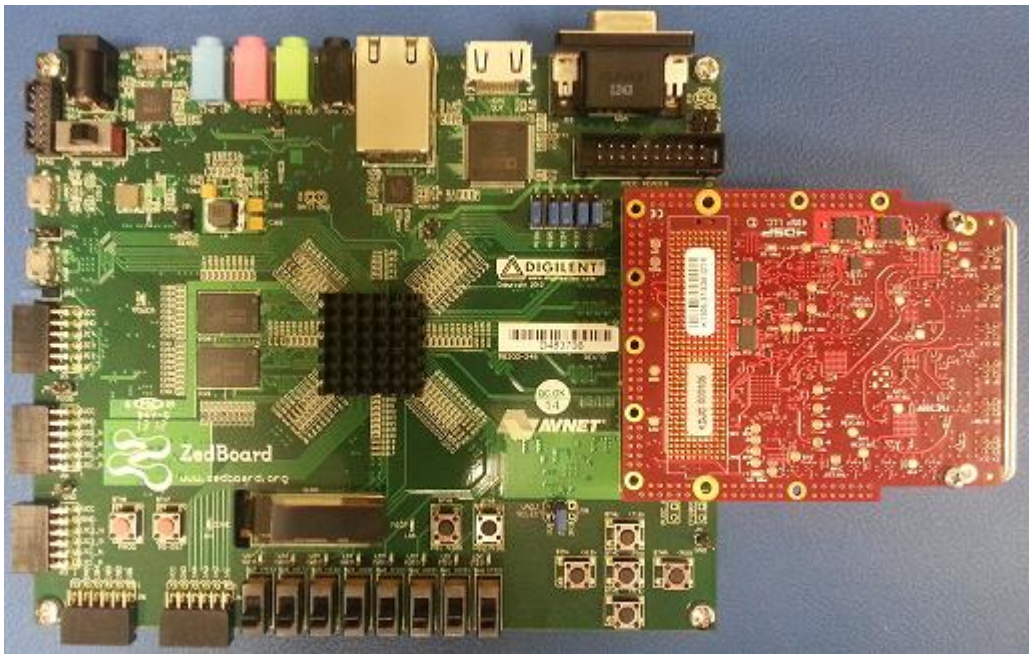
Zedboard-FMC30RF
User Manual



4DSP LLC, USA

Email: support@4dsp.com

Revision History

Date	History	Revision
2013-09-13	Initial release	1.0

Table of Contents

1 Related documents

1.1 Related Documents

- Zedboard Rev D. User Manual, Digilent
- UM015 r1.1 (FMC30RF).pdf User Manual, 4DSP
- SD001(sip_cid).pdf Star Documentation, 4DSP
- SD005(sip_i2c_master).pdf Star Documentation, 4DSP
- SD024(sip_cmd12_mux).pdf Star Documentation, 4DSP
- SD111(sip_fmc30rf).pdf Star Documentation, 4DSP
- SD179(sip_zedb_host_if).pdf Star Documentation, 4DSP
- CD316 (ZEDB_FMC30RF).pdf Constellation Documentation, 4DSP

2 Requirements

2.1 Requirements

- Windows 7 (64 bit preferred) installed on the host computer.
- Xilinx ISE 14.6 installed on the host computer.
- Vivado 2013.2 installed on the host computer.
- Host computer with a 1Gbps Ethernet interface.

3 General description

The Zebboard-FMC30RF is a software defined radio (SDR) development platform. The FMC30RF is an RF transceiver mezzanine board and the Zedboard is an FPGA development mother board. The development environment includes a graphical user interface able to transmit data and receive the same data in real time.

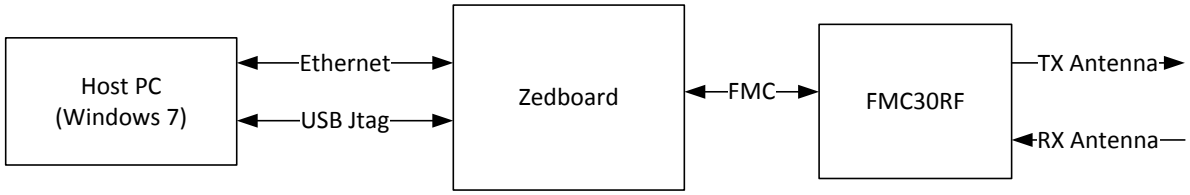


Figure 1: Zedboard-FMC30RF Environment

The host PC running Windows 7 is first configuring the FMC30RF through the Zedboard via Ethernet. Then a buffer containing transmit data is uploaded to the FMC30RF waveform memory. Finally the host PC is offloading and displaying samples from the FMC30RF in real time. The GUI application is also performing FFT processing in order to display the frequency domain instead of the time domain.

4 Getting started

4.1 Connect the hardware

- Connect one of the supplied antennas to the “RF IN” connector on the front panel.
- Connect one of the supplied antennas to the “RF I/O” connector on the front panel.
- Connect an Ethernet cable between the host PC and the Zedboard.
- Connect an USB cable between the host PC and the Zedboard “PROG” connector. This cable will be used to program the FPGA from the host PC.

4.2 Configure TCP/IP in the host

Open the Network and Sharing Center on Windows 7. Click “Change adapter settings.” Open Properties on Local Area Connection. Click “Internet Protocol Version 4 (TCP/IPv4)” and open Properties. TCP/IPv4 must set as the following picture.

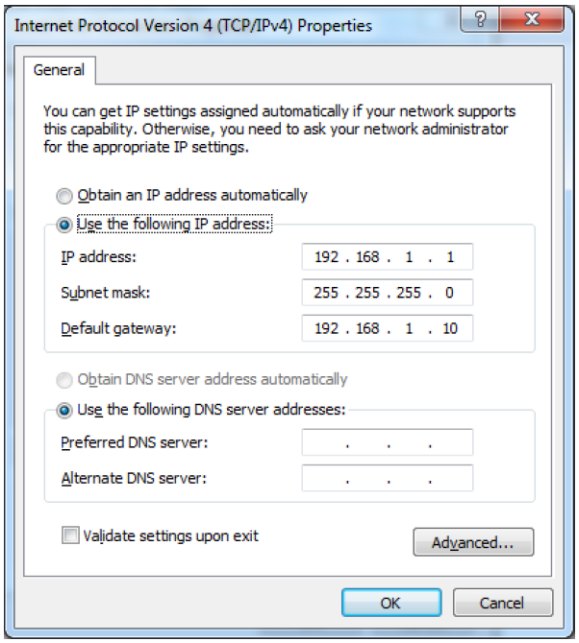


Figure 2: TCP/IP Settings

4.3 Upload firmware and Zynq software to the Zedboard

- Start an ISE command prompt (ISE Design Suite 64/32 Bit Command Prompt) from the Windows Start Menu.
- Change to the directory “Ready” :
  cd C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Ready.
- Execute the download script “Go” using Zedboard firmware as argument :
  go 316_zedb_fmc30rf.bit
- It takes about 45 seconds in order to upload the firmware and the Zynq software. The last portion of the console output should look like the following:

  Download Progress..10.20.30.40.50.60.70.80.90.Done
  Setting PC with Program Start Address 0x00100000
  Processor started. Type "stop" to stop processor

  RUNNING> Disconnected from Target 64

  Disconnected from Target 352

4.4 Run the FMC30RF Demo GUI

The FMC30RF Demo graphical user interface is launched from the Windows Start menu, group "4DSP" and subgroup "Zedboard-FMC30RF". The GUI screen that you will see is depicted in the following image.

Several settings files come preinstalled and you can select these settings files from the "Setting file" drop down list on the bottom left of the GUI. Selecting one of the settings file will make the GUI to prompt for a transmit buffer. Three transmit buffers come preinstalled and you can simply browse to one of these three.

4.5 Experiment with the FMC30RF Demo GUI

Settings files as well as transmit buffers can be changed in real time without to restart the graphical user interface. This allows changing of test condition easily. In addition, some basic settings can be changed in real time also:

- "Attenuation" directly modifies the chip configuration to add attenuation in the RX path.
- "Enable Second RF Amplifier" enables the second low noise amplifier (LNA) on the board.
- "Digital Loopback" directly modifies the chip configuration to not use the analog front/backend.

4.6 Creating new buffers/settings file

The initial settings files are located at C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Software\Profiles. These profiles are only defining configuration of two chips on the FMC30RF; The TRF3765 and the TRF3720. One should be able to create different profiles by referring to these initial settings in conjunction with the chip manufacture's (Texas Instruments) literature. The settings files list is refreshed each time the graphical user application is started.

The initial buffer files are located at C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Software\Buffers. It is also possible to use any user created buffer file instead the one provided with the software. The format of the buffer is complex data, signed integers.

4.7 Graphical User Interface view

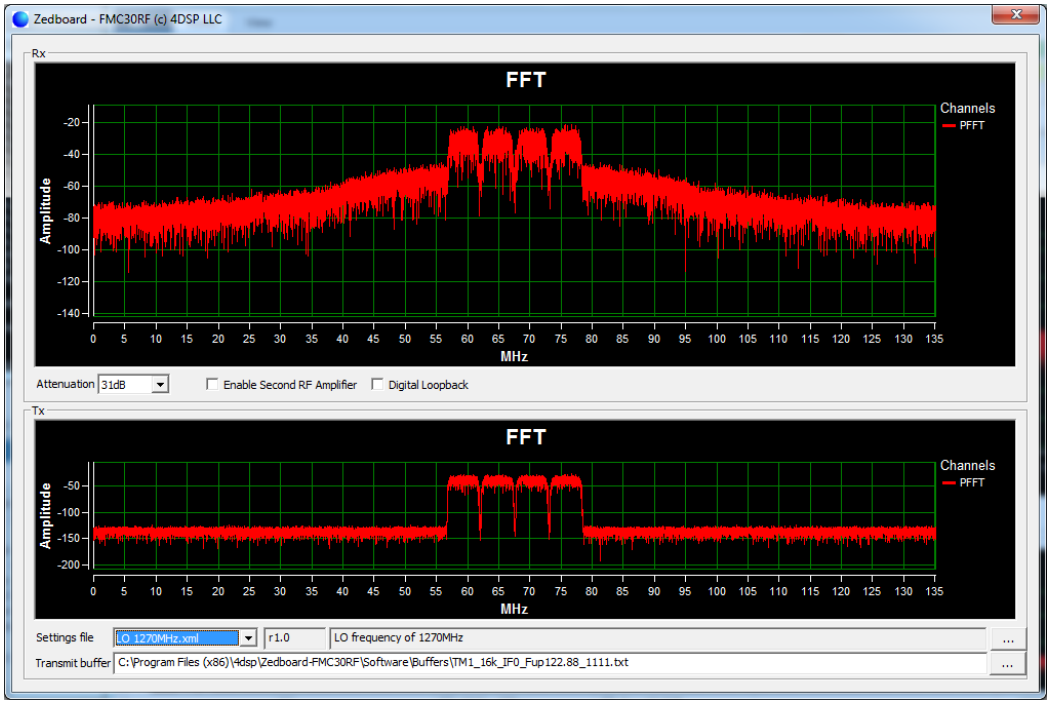


Figure 3: LO 1.270GHz profile

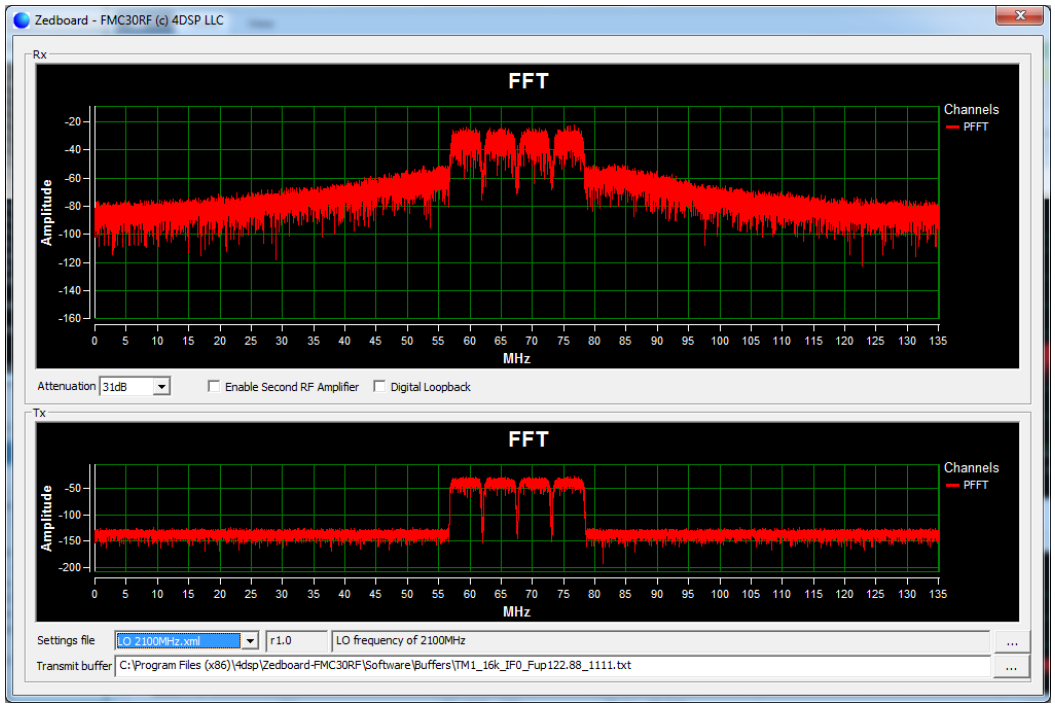


Figure 4: LO 2.1GHz profile

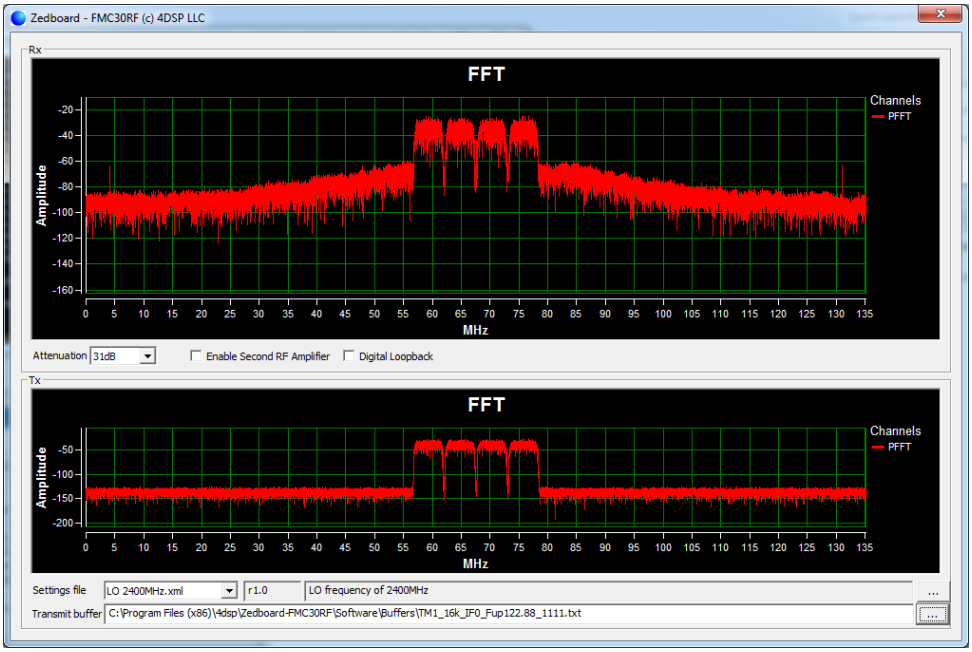


Figure 5: LO 2.4GHz profile

5 The FMC30RF Control DLL (CFMC30RFDll)

The FMC30RF Demo GUI is using this control DLL in order to control the FMC30RF through the Zedboard. The source code is provided to the customer and can be found under C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Software\Source Code.

Visual Studio 2012 is required in order to recompile this dll. As soon the DLL is loaded by your software you can call any function declared in "generic_interfaces.h" via function pointer.

5.1 Add the DLL in your project

- Include the "generic_interfaces.h" header to your software application.
- Add a typedef for the function pointer in your module:
  "`typedef IFmc30RF* (__cdecl *ifmc30rf_factory)();`

5.2 Code to load the DLL

```
// Try to load the FMC30RF control DLL
HINSTANCE dll_handle = ::LoadLibrary(TEXT("CFMC30RFDll.dll"));
if (!dll_handle) {
	AfxMessageBox("Could not find 'CFMC30RFDll.dll'");
	exit(-1);
}

// Get the function from the DLL
ifmc30rf_factory factory_func = reinterpret_cast<ifmc30rf_factory>(
	 ::GetProcAddress(dll_handle, "GetFmc30RF"));
if (!factory_func) {
	AfxMessageBox("Problem with 'CFMC30RFDll.dll'");
	::FreeLibrary(dll_handle);
	exit(-2);
}
```

5.3 Code to get function pointers

```
// Ask the factory for a new object
IFmc30RF* pCFmc30RF = factory_func();
```

5.4 Calling the Open function

```
// Open/init FMC30RF
if(m_pCFmc30RF->Open(lregtrf3765, lregtrf3720, bExclude2ndLNA, nAttenuation,
bDigitalLoopback)!=TRUE) {
	MessageBox("Could not open FMC30RF hardware", "oups", MB_OK);
	return;
}
```

6 The ZYNQ software

The Zynq software source code is located at C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Software\Source Code\Zynq\zedb_host_if.

The process of using this software is to first export the hardware specification to SDK, create a Xilinx BSP and finally importing the zedb_host_if.

Please refer to SD179 Zedboard Host Interface chapter 6 for more information on how to actually import the software. Please pay attention to the Xilinx SDK BSP modifications about the tcp_snd_buf size.

7 Vivado Firmware

An archive of the complete Vivado project is located at C:\Program Files (x86)\4dsp\Zedboard-FMC30RF\Firmware.

The firmware archive can be extracted to your host PC and the Vivado project can then be loaded and compiled.