Using Red Pitaya for radio applications (from LF to HF)

Pavel Demin

January 31, 2016
Introduction
My journey into radio

– May 2009:
  Started to play with FPGA (Altera Cyclone III) and fast ADC (AD9228)

– September 2014:
  Discovered RTLSDR, GNU Radio and Gqrx thanks to Hackable Magazine №2

– December 2014:
  Started to play with Red Pitaya

– March 2015:
  Released Red Pitaya SDR receiver

– September 2015:
  Released Red Pitaya SDR transceiver

– November 2015:
  Joined Radio & Electronics Engineering Club ASBL (REEC)
Initial goals

- Listen to the radio with ADC and FPGA
- Get familiar with new Xilinx chips and tools
- Keep the number of lines of code low
- Make use of the existing libraries and programs
- Keep the budget low
Red Pitaya
Red Pitaya overview

– A single-board computer designed in Slovenia
– Cost ≈ 250 € (tax included)
– Key features:

  open-source-software measurement and control tool
  stand-alone GNU/Linux platform
  Xilinx Zynq All Programmable System-on-Chip (ARMv7-A CPU + FPGA)
  fast analog inputs and outputs
  wired and wireless (via a USB adapter) network connectivity

Fast analog inputs
(2ch. @ 125 MSPS, 14 bits)

Fast analog outputs
(2ch. @ 125 MSPS, 14 bits)
Red Pitaya application marketplace

– Applications can be installed through the application marketplace:

http://bazaar.redpitaya.com
Availability

– Its network of distributors makes Red Pitaya easily available worldwide:

LIST OF RED PITAYA DISTRIBUTORS:

Global Authorized Distributor:

MOUSER ELECTRONICS

EU:

North America:

India:

China:

Asia & Pacific:

South Korea:

January 31, 2016 Pavel Demin
Fast analog inputs and outputs

− RF inputs:

  Bandwidth: 50 MHz (3 dB)
  Input impedance: 1 MΩ // 10 pF
  Full scale voltage: 2 Vpp
  Linear Technology LTC2145-14 ADC:
  − two channels, 14 bits, 125 MSPS
  − 90 dB SFDR, 73.1 dB SNR, 11.9 ENOB

− RF outputs:

  Bandwidth: 50 MHz (3 dB)
  Load impedance: 50 Ω
  Full scale power: > 9 dBm
  NXP DAC1401D125 DAC:
  − two channels, 14 bits, 125 MSPS
  − 92 dB SFDR
**CPU, FPGA and DRAM**

− Xilinx Zynq Z-7010 All Programmable System-on-Chip (AP SoC):
  
  CPU: Dual-core ARM Cortex-A9, 667 MHz
  
  FPGA: 2200 logic blocks (CLB), 80 DSP blocks, 60 RAM blocks
  
− On-board DRAM: 512 MB, DDR3, 1066 MHz, 16-bit wide
FPGA components

- 2200 configurable logic block (CLB):
  - 8× 6-input look-up tables (LUT)
  - 16× flip-flops

- 80 DSP block:
  - 18 × 25 signed multiply
  - 48-bit adder/accumulator
  - 25-bit pre-adder

- 60 RAM block:
  - dual-port
  - 36 Kb
FPGA tools

- Xilinx Vivado Design Suite provides the following tools:
  Hardware description languages (Verilog and VHDL)
  Rich library of IP cores (DSP, math, video, imaging, etc)
  IP Integrator (supports graphical and Tcl-based design flows)
  High-Level Synthesis for C, C++ and SystemC

- For my projects, I’m currently using:
  **Verilog** to write custom IP cores
  **Tcl** to glue IP cores together
Some thoughts about cost

- ADC, AP SoC and SMA connectors are relatively expensive parts:

<table>
<thead>
<tr>
<th>Image</th>
<th>Part Number</th>
<th>Description</th>
<th>Customer Reference</th>
<th>Available Quantity</th>
<th>Backorder Quantity</th>
<th>Unit Price</th>
<th>Extended Price</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="LTC2145CUP-14#PBF-ND" /></td>
<td>LTC2145CUP-14#PBF-ND</td>
<td>IC ADC DUAL 14BIT 125MSPS 64-QFN</td>
<td>LTC2145-14 ADC</td>
<td>1 Immediate</td>
<td>0</td>
<td>68,91000</td>
<td>€ 68,91</td>
</tr>
<tr>
<td><img src="image.png" alt="122-1854-ND" /></td>
<td>122-1854-ND</td>
<td>IC SOC CORTEX-A9 ARTIX-7 400BGA</td>
<td>Zynq Z-7010 AP SoC</td>
<td>0 Immediate</td>
<td>1 Lead Time</td>
<td>57,43000</td>
<td>€ 57,43</td>
</tr>
<tr>
<td><img src="image.png" alt="J502-ND" /></td>
<td>J502-ND</td>
<td>CONN SMA JACK 50 OHM EDGE MNT</td>
<td>SMA connectors</td>
<td>4 Immediate</td>
<td>0</td>
<td>4,68000</td>
<td>€ 18,64</td>
</tr>
<tr>
<td><img src="image.png" alt="1450-1095-ND" /></td>
<td>1450-1095-ND</td>
<td>IC SDRAM DDR3 256M X 16 96-FBGA</td>
<td>512 MB DDR3 DRAM</td>
<td>1 Immediate</td>
<td>0</td>
<td>8,68000</td>
<td>€ 8,68</td>
</tr>
</tbody>
</table>

**Subtotal** € 153,66

- Components in small quantities would cost more than the assembled board
## Minimal kit for radio applications

<table>
<thead>
<tr>
<th>Item</th>
<th>Quantity</th>
<th>Price (€)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Pitaya Open Source Instrument</td>
<td>1</td>
<td>250</td>
</tr>
<tr>
<td>Fan, 30 × 30 × 15 mm, 5 V</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>Power supply, micro USB, 5 V, 2 A</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>SMA tee adapter, SMA plug, SMA jack, SMA jack</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>SMA terminator, 50 Ω</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>SMA-BNC adapter, SMA plug, BNC jack</td>
<td>4</td>
<td>25</td>
</tr>
<tr>
<td>SMA cable, SMA jack, SMA plug, RG-174, 15 cm</td>
<td>4</td>
<td>15</td>
</tr>
</tbody>
</table>

---

Total: **360 €** (tax included)
Software defined radio (SDR)
Software defined radio (SDR)

- Superheterodyne receiver:

- SDR receiver:

- LO, mixer, filter and demodulator are done by digital signal processing (DSP)
Localized and distributed DSP

DSP can be localized (e.g. GNU Radio running on the on-board CPU):

-or distributed:

ADC samples (125 MSPS, 16 bits) 2000 Mb/s
Ethernet 500 Mb/s
Wi-Fi 50 Mb/s
4G LTE 5 Mb/s
Audio (48 kSPS, 16 bits) 1 Mb/s

January 31, 2016 Pavel Demin
Digital down-converter (DDC)

- ADC samples are processed by a digital down-converter (DDC) running on the Red Pitaya’s FPGA:

![Diagram](https://via.placeholder.com/150)

Xilinx IP cores

Custom IP cores

ADC clock domain

CPU clock domain

ADC interface

FIFO

FIFO config register

CPU status register

ADC interface

125 MSPS, 14 bits

FIFO

125 MSPS, 14 bits

Complex multiplier

125 MSPS, 24 bits

CIC

50-3125 complex multiplier

DDS

cos - sin

20-1250 kSPS, 24 bits

FIFO

125 MSPS, 14 bits

125 MSPS, 24 bits

40-2500 kSPS, 24 bits

FIR

↓ 2

20-1250 kSPS, 24 bits

20-1250 kSPS, 32 bits

ADC samples are processed by a digital down-converter (DDC) running on the Red Pitaya’s FPGA:
CIC and FIR filters

- Calculated frequency response (decimation by a factor of 50):

- Measured frequency response:
Digital up-converter (DUC)

- Digital up-converter (DUC) consists of the similar blocks but arranged in an opposite order:
Putting it all together

- Two SDR transceiver applications are available from the Red Pitaya application marketplace:

- These applications configure FPGA and start TCP or UDP servers that communicate with SDR programs running on a remote PC:
**SDR programs**

- SDR programs provide:
  - graphical user interface
  - spectrum display
  - modulation/demodulation

- Red Pitaya SDR transceiver applications work with the following programs:

<table>
<thead>
<tr>
<th>plug-ins/libraries/protocols</th>
<th>SDR programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtIO_RedPitaya_TRX plug-in (only RX)</td>
<td>HDSDR</td>
</tr>
<tr>
<td></td>
<td>SDR# (≤ 1.0.0.1361)</td>
</tr>
<tr>
<td>gr-osmosdr/lib/redpitaya</td>
<td>GNU Radio and GNU Radio Companion</td>
</tr>
<tr>
<td></td>
<td>Gqrx</td>
</tr>
<tr>
<td>SoapySDR/SoapyRedPitaya</td>
<td>Pothos</td>
</tr>
<tr>
<td></td>
<td>CubicSDR</td>
</tr>
<tr>
<td>HPSDR/Metis communication protocol</td>
<td>PowerSDR mRX PS</td>
</tr>
<tr>
<td></td>
<td>QUISK</td>
</tr>
<tr>
<td></td>
<td>ghpssdr3-alex</td>
</tr>
<tr>
<td></td>
<td>openHPSDR Android Application</td>
</tr>
<tr>
<td></td>
<td>Ham VNA vector network analyzer</td>
</tr>
</tbody>
</table>
Feedback from radio amateurs
Red Pitaya SDR with 5W power amplifier

– Wolfgang Kiefer (DH1AKF) published pictures of his 5W station:

http://www.mikrocontroller.net/topic/385102
Comparison with Flex-6500

— Ger Metselaar (PAØAER) compared Red Pitaya SDR with Flex-6500:

http://www.pa0aer.com/projecten/red-pitaya

<table>
<thead>
<tr>
<th></th>
<th>Flex-6500</th>
<th>Red Pitaya SDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise floor level</td>
<td>-130 dBm</td>
<td>-120 dBm</td>
</tr>
<tr>
<td>Suppression of the intermodulation products</td>
<td>97 dB</td>
<td>75 dB</td>
</tr>
</tbody>
</table>
Red Pitaya SDR in the news

– Johan van Dijk (PA3ANG) published an article about Red Pitaya SDR in the January, 2016 issue of DKARS Magazine:

Concluding remarks
Summary

− Red Pitaya is a very interesting platform for building various measurement and control systems, experimenting with FPGA and DSP algorithms, sharing knowledge and experiences.

− It is a nice SDR building block thanks to the excellent open-source SDR tools.
Where is the source code?

The source code and more details about my projects can be found at:

http://pavel-demin.github.io/red-pitaya-notes
Interesting links

– Red Pitaya
  http://redpitaya.com

– The Scientist and Engineer’s Guide to Digital Signal Processing
  http://www.dsptool.com

– dspGuru: Digital Signal Processing Articles
  http://www.dsptool.com/dsp/articles

– ARRL: Software Defined Radio
  http://www.arrl.org/software-defined-radio

– GNU Radio: Suggested Reading
  http://gnuradio.org/redmine/projects/gnuradio/wiki/SuggestedReading