Software Defined Radio Transceiver Implementation

Gerry Corley, Magdalena Sanchez Mora, Dr. Ronan Farrell
Centre for Telecommunications Value-Chain Research (CTVR),
Institute of Microelectronics and Wireless Systems,
National University of Ireland Maynooth,
Ireland.

***Abstract*-This document presents the design and implementation of a low cost reconfigurable radio transceiver platform. The platform will be used as a research tool in the investigation of software defined radio techniques. The hardware presented is an evolution of work presented at the 2006 RIA colloquium.**

The platform consists of four hardware elements, namely a radio transmitter, a radio receiver, a baseband interface and a PC to perform signal processing and configuration. Data and control communication is performed via a USB 2.0 interface between the transceiver and a laptop PC. The platform development included a substantial software element to configure the hardware and to receive and transmit data between the PC and the transceiver. The technical choices, design and realization of the prototype are discussed.

I. INTRODUCTION

There is a multitude of telecommunications standards in use today, ranging from mobile communication standards such as GSM, PHS, and UMTS, to wireless LAN standards such as WiMAX and IEEE802.11x as well as future standards based on UWB technology. Traditional transceiver technology requires users to have separate equipment for each standard however SDR offers the possibility of using one terminal to receive many standards through the use of wideband reconfigurable transceivers and software signal processing. The main challenge is to optimize the tradeoffs between performance, power consumption and cost [1]. The NUIM platform is specifically designed to meet high RF standards while maintaining a low unit cost. This is achieved through the use of readily available low cost mobile communications components and a standard laptop PC for data processing and configuration [2].

The SDR hardware and its associated driver software have been entirely developed at the Institute of Microelectronics and Wireless Systems at the National University of Ireland Maynooth as part of a Centre for Telecommunications Value-Chain Research (CTVR) initiative.

Several modules have been developed, these include a radio transmitter, a radio receiver, a transmit baseband board, and a receive baseband board. A complete transceiver, Figure 1, consists of transmitter and receiver modules and their associated baseband modules. In the following sections the hardware modules and the system software will be briefly described.

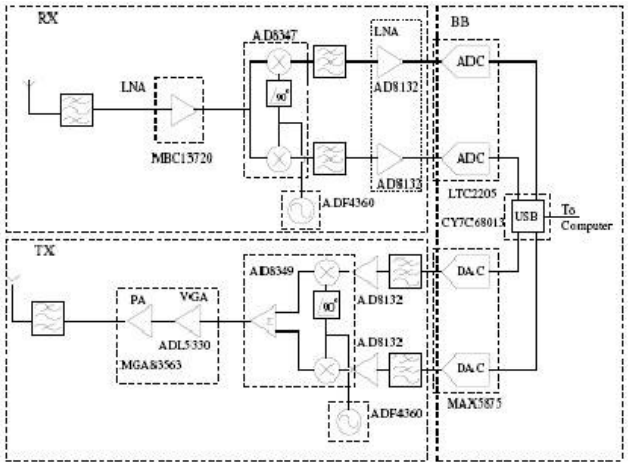

Figure 1: Block diagram of SDR hardware

II. TRANSMITTER SECTION

For maximum configurability it was decided to use an IQ direct conversion transmitter. Direct conversion allows large baseband bandwidth, avoids the use of fixed response analogue filters and image rejection filters and has a lower component count than superhetrodyne or low IF architectures. The I and Q signals are provided by the baseband board. The carrier frequency is set by a software controlled local oscillator. The output from the modulator is fed into a variable gain amplifier before being applied to a RF power amplifier. The current design operates between 2.1 and 2.45GHz, with a maximum output power of 24dBm and 256 levels of power control all under full software control.

III. RECEIVER SECTION

As with the transmitter, it was decided to use a direct conversion architecture for maximum flexibility. The receiver

consists of an LNA followed by an IQ demodulator with built in automatic gain control (AGC). The IQ outputs feed a pair of low pass anti-aliasing filters implemented using Analog Devices AD8132 differential op-amps. Both the receiver frequency and gain are under full software control from the system PC. The differential IQ signals are output for further processing by the receive baseband board. The complete receiver has a noise figure of less than 5dB, bandwidth of 40MHz and 48dB of gain control.

IV. BASEBAND SECTION

The baseband section comprises two boards, the transmitter baseband board and the receiver baseband board. The transmitter baseband board consists of a USB 2.0 interface to the host computer, a dual high-speed DAC with filtering, and a clock generator chip to control the conversion rate. The maximum sample conversion rate is 200MSps at 16bits resolution with a maximum analogue bandwidth of 40MHz on both I and Q channels. The USB 2.0 interface permits a maximum data transfer rate of 480Mbps. The radio transmitter board plugs into this board in 'piggyback' fashion through a series of SMB connectors, Figure 7, this makes for a low loss robust connection.

The receiver baseband section is of similar construction. It consists of a USB interface to the host computer, a multiplexer, two high-speed ADCs, a clock control chip and two anti-aliasing filters. This board is capable of digitizing the I and Q channels at 80MSps with 16 bits of resolution.

Although the ADC's and DAC's are capable of very high conversion rates, currently the maximum operating speed of the system is limited by the USB interface and the radio software running on the PC.

V. SOFTWARE

The software can be divided into three sections [3]:

- The API or user interface software which allows the user to run third party software programs
- A USB driver for the Linux operating system that enables communication between the PC and the SDR hardware.
- The embedded code running on the USB microcontroller (CY7C68013A), which allows the PC to control the SDR hardware.

A diagrammatic representation of this software structure is shown in Figure 2. This approach produces a software system that hides hardware specific details and provides a consistent and high-level base from which to develop user applications for controlling our SDR hardware prototype. Existing software radio platforms such as OSSIE from Virginia Tech and IRIS from Trinity College Dublin [4] are two examples of applications for rapid development of SDR components and waveforms. These software radio platforms can be easily integrated with our software system.

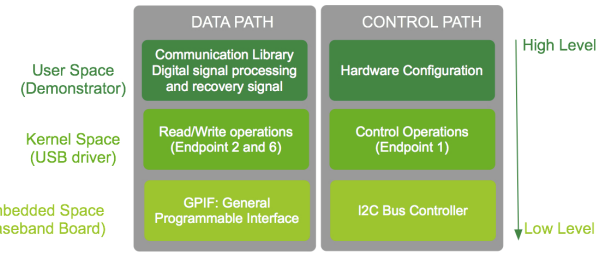

Figure 2: Software modules

VI. MEASUREMENTS

In order to verify that the SDR platform including hardware and software was fully functional, a simple communications link was set up at 2.35GHz. The system was used to transmit data between two laptop computers at 256kbps using BPSK with 8 samples per symbol. The received data was processed with Matlab to extract eye and constellation diagrams, the Matlab Simulink block diagram is shown in Figure 3.

Figure 4 shows the spectrum at the transmitter output for BPSK modulated with a 256kbps PBRS-23 data stream. The output is a clean sinc function with a power level of about 10dBm when integrated over a 5MHz bandwidth.

Figure 5 and Figure 6 show the eye and constellation diagrams at the receiver laptop. There is very little noise present as the attenuation between transmitter and receiver

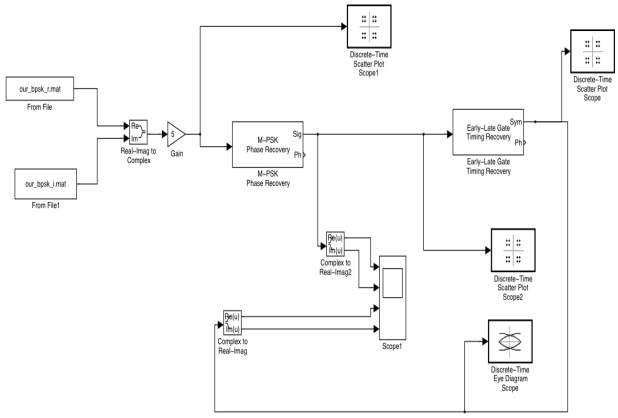

Figure 3: Matlab implementation

was only 50dB which guaranteed a high CNR. There was an undesirable DC offset on the recovered data as can be seen on the eye and constellation diagrams however this problem will be rectified in the near future.

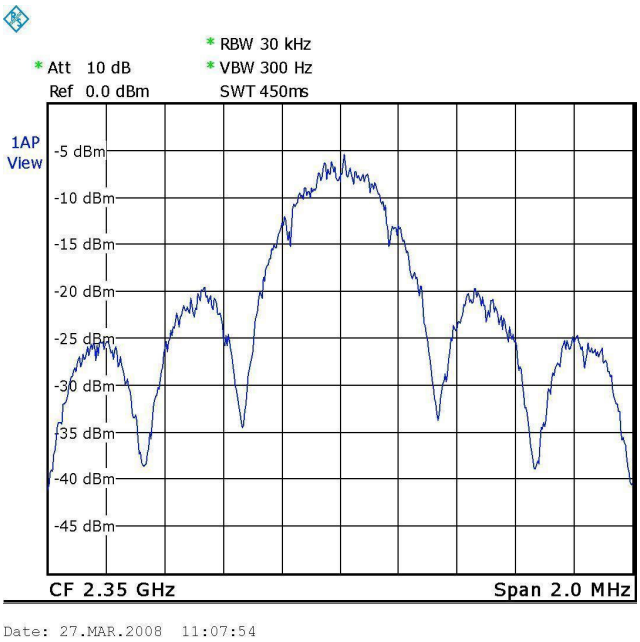

Figure 4: BPSK Spectrum at transmitter output

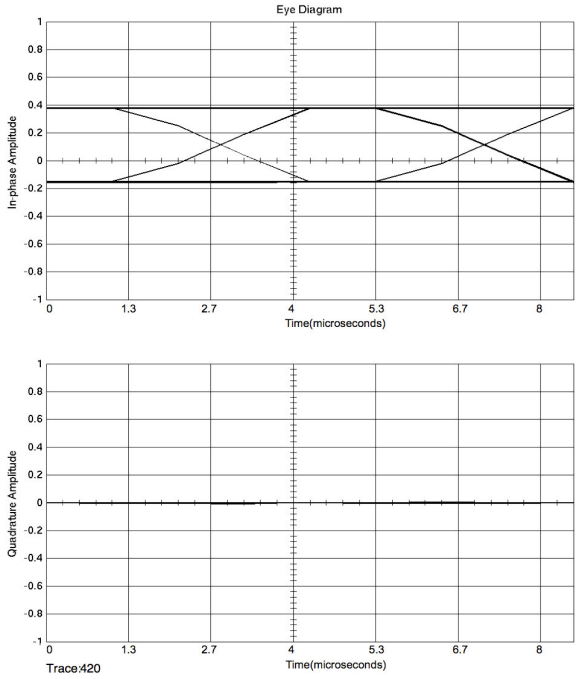

Figure 5: Eye diagram for received BPSK signal

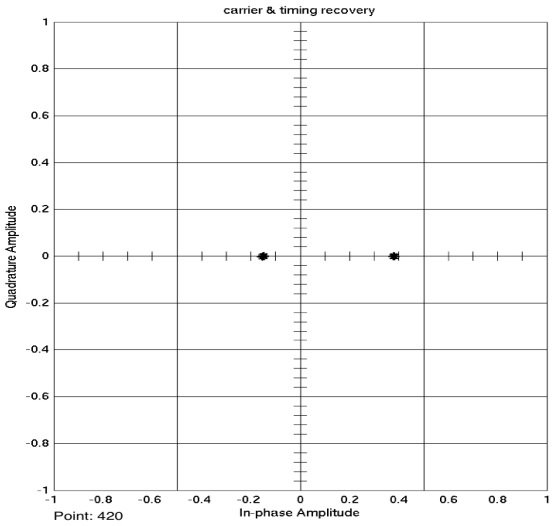

Figure 6: Constellation diagram for received BPSK signal

A photograph of the hardware elements of the SDR platform is shown in Figure 7.

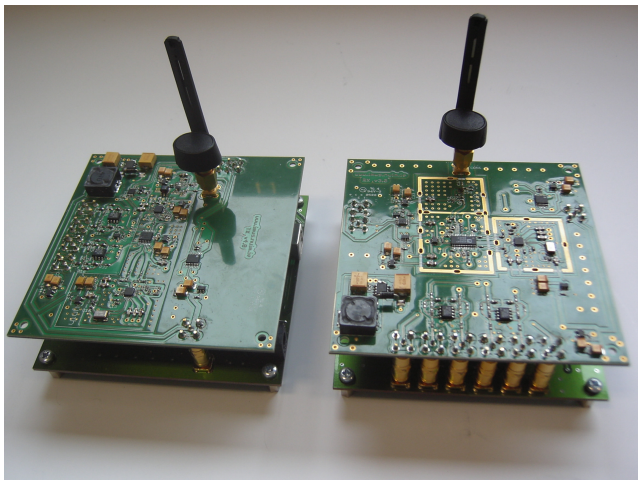

Figure 7: SDR hardware platform

VII. CONCLUSIONS

This paper presented a test platform for the exploration and development of SDR technology. The hardware uses off-the-shelf components in a wide bandwidth direct conversion transceiver architecture. The software allows easy configuration of the hardware and can be integrated with existing software radio platforms such as OSSIE from Virginia Tech or IRIS from Trinity College Dublin. A simple BPSK communications link was established between two laptop computers and measurements at different points on this link were presented.

ACKNOWLEDGMENT

The authors would like to thank the Centre for Telecommunications Value Chain Driven Research (CTVR) and the Science Foundation of Ireland for supporting this research.

REFERENCES

[1] W. Tuttlebee, Ed., *Software Defined Radio, Enabling Technologies*, John Wiley & Sons, Chichester, UK, 2002.

[2] G.Baldwin, L.Ruiz, L.Barrandon, R.Farrell, "Low cost experimental Software Defined Radio Systems", SDR Technical Forum 2007, 5-9 November, Denver Colorado

[3] M.Sanchez Mora, L.Ruiz, G.Baldwin, R.Farrell, "Software engine development for SDR", SDR Technical Forum 2007, 5-9 November, Denver Colorado.

[4] http://ossie.wireless.vt.edu/trac , http://ntrg.cs.tcd.ie/swradio.php