

VLSI Implementation of Software Defined Radio

S. Sathish, J. Selvakumar

Assistant Professor, Department of ECE, Karpagam College of Engineering, Coimbatore, India

ABSTRACT: Software Defined Radio (SDR) is a flexible architecture which can be configured to adapt various wireless standards, waveforms, frequency bands, bandwidths, and modes of operations. It also presents prototype system for designing and testing of software defined radios in vhdl using altera and briefly discusses the salient functions of the prototype system for Cognitive Radio.

The philosophy behind the prototype is to do all waveform-specific processing such as channel coding, modulation, filtering etc. on a host (PC) and general purpose high-speed operations like digital up and down conversion, decimation and interpolation etc. inside FPGA. This implementation allows building software-defined and cognitive radio to explore various spectrum sensing, prediction and management technique.

KEYWORDS: VHDL; MATLAB simulink, software defined radio; cognitive radio; modulation; radio frequency (RF).

I. INTRODUCTION

Software Defined Radio is a flexible architecture that is applicable to many radio standards. Joseph Mitola coined the term software radio, to signal the shift from digital radio to multiband multimode software-defined radios. In an introduction of reconfigurable logic and the coining of the term SDR, the dominant implementation architecture used for RF Front-Ends (FEs) was the super-heterodyne architecture. The SDR is a radio communication system, which provides software control for a variety of modulation method, filtering, wideband or narrowband operations, spread spectrum techniques and waveform requirements etc. The frequency bands are still constrained at the RF Front-Ends. An SDR allows an implementation of signal processing functionality in software instead of dedicated hardware circuitry. So, the most obvious benefit is loading an appropriate program instead of having to build extra circuitry for different types of radio signals and communication protocols. The development of an SDR system implies to achieve two main goals.

To move the border between the analog and digital world (in TX and RX Paths) as much as possible toward radio frequency (RF) by adopting analog digital (A/D) and digital analog (D/A) conversion as near as possible to the antenna.

To replace the application specific integrated circuits (ASICs) dedicated hardware, with the re-configurable computing (FPGA) for baseband signal processing.

The FPGAs are mainly used in SDR RF Front-Ends (FEs) to improve the performance of DSP-chip-based systems. There is currently a wide range of FPGA products being offered by many semiconductor vendors; Xilinx, Altera, Atmel and AT&T etc. The architectural approaches used in these FPGAs are as diverse as their manufacturers. The obvious benefits of wireless transmission have led to a number of radio systems. The family contains different types of access networks namely; wireless personal area network (WPAN) IEEE-802.15.1 called Bluetooth, IEEE-802.15.4 is ZigBee, wireless local area network (WLAN) the IEEE-802.11 and wireless metropolitan area network (WMAN) the WiMax/IEEE-802.16. These standards apart from the channel bandwidth and the transmit power have different modulation techniques and transmission mechanisms, which can be implemented easily in a software; the RF Front-Ends may require for different frequency bands. The development of the IEEE 802.22 WRAN standard was aimed to use the cognitive radio techniques to allow sharing of geographically unused spectrum allocated to the television broadcast service. An IEEE-1900.1 is a first standard that defines the concepts for dynamic spectrum access, terminologies relating to emerging wireless networks, system functionality and spectrum management.

The paper presents a detailed survey of the existing software radio platforms and an experimental prototype system that is built in MATLAB/Simulink. The MATLAB has a rich family of signal processing block, unless we want to do a something fancy; the mex (MATLAB executable) can be used to build an executable functions for C++ code for stand-alone MATLAB engine. The prototype system is low priced solution to realize the software-defined radios techniques.

II. BACKGROUND & RELATED WORK

There are various hardware (HW) platforms and the software (SW) architectures that are used for defining the software radios. Many individuals, various research-groups and different companies have been working on software defined radio concept for years, pushing it to make a reality. This section presents a survey of the current SDR hardware platforms followed by the software architectures. The military solutions are not in the scope of the paper.

A. SDR Hardware Platforms

The hardware aspects of a SDR platform consist of the radio-frequency (RF) parts, communications links to the software-based signal processing elements (mostly a Host-PC). The rest may consist one or more of the following;

- ASICs (application-specific integrated circuits).
- FPGAs (field-programmable gate arrays).
- DSPs (digital signal processors).
- GPPs (general-purpose processors).

The ASICs are non-reprogrammable that contradicts the principle of SDR, but still used as a part for special characteristics. The FPGAs provide high computing power due to quasi-parallel processing nature while the DSP and GPPs are essentially serial in operation. The main strengths of DSPs and GPPs are their flexibility and easy configurability. The various SDR hardware solutions (RF Front-End) are available in commercial and academic area, providing potential opportunities in the radio communication industry.

Universal Software Radio Peripheral 2 (USRP2):

It is a brainchild of Matt Ettus (Ettus Research LLC). The USRP family of products has been nominated "Technology of the Year" award from the Wireless Innovation Forum, 2010. The USRP2 is a second generation of Universal Software Radio Peripheral, its platform consist Xilinx Spartan-III FPGA and general purpose AeMB processor.

Rice Wireless Open-Access Research Platform (WARP):

The wireless open-access research platform of Rice University is a scalable and extensible programmable platform, built for prototyping advanced wireless networks. The Xilinx Virtex-4 FX100 FPGA is used to enable programmability of both physical and network layer protocols on a single platform.

Berkeley Emulation Engine 3 (BEE3):

BEE3 is new generation of Berkeley Emulation Engine-2. It is jointly developed by Microsoft Research, UC Berkeley and BEEcube Inc. The platform contains four Virtex-5FPGAs, emulates over 64 RISC processor cores concurrently at a 100 MHz rate. It is useful for most computationally intensive real-time applications, a high-speed multiple FPGA and validation solution. BEE3 is suited as a real-time, real-world prototyping and development platform.

Kansas University Agile Radio (KUAR):

The KUAR hardware employs a Xilinx Virtex II Pro P30 FPGA along with 1.4 GHz Pentium M processor. It has been promoted through the defense advanced research projects agency (DARPA) next generation (XG) program. The complete system was developed in Simulink, implemented in Xilinx VHDL, by generating the VHDL code from Simulink model(s) using a Modelsim of Mentor Graphics. *Small Form Factor Software*

Defined Radio (SSF-SDR):

The Xilinx Inc. in collaboration with Lyrtech and Texas Instruments incorporated a SFF-SDR development platform for developing the handheld and mobile radios. The Xilinx Virtex-IV FPGA, TI DSP TMX320 and TI MSP430 MCU are used in addition with ARM926 embedded processor.

Intelligent Transport System (ITS);

National Institute of Information and Communications Technology (NICT) of Japan, developed a software-defined radio platform so-called NISTITS. It is specially designed for mobile communication, wireless LAN and digital terrestrial TV. The platform contains Xilinx Virtex-4 FPGA, and two general purposes process (GPP) 430 MIPS (240 MHz). The USRP2 is a cheaper and fast enough; the Xilinx Spartan-3 FPGA (XC3S2000) contains; 2M system-gates, 46080 equivalent logic cells. The larger FPGA and general purpose AeMB processor allows the USRP2 to be used as a standalone system without a host computer in many cases; the USRP2 is used in the prototype.

B. SDR Software Platforms

GNU Radio:

It is an open source software development tool that provides the signal processing runtime and processing blocks to implement software radios. The radio applications are written in Python, while the performance-critical signal processing components, implemented in C++ using processor floating point extensions where available. SWIG glues C++ classes into Python. In GNU Radio Python (*gr.flow_graph*) library of signal processing blocks is used to tie together the signal processing blocks of the waveforms. GNU Radio Companion (GRC) is a graphical tool for creating signal flow graphs and generating flow-graph source code.

Open-Source SCA Implementation - Embedded (OSSIE):

It is a Virginia Tech's open source, the core framework is based on the JTRS software communications architecture (SCA); the CORBA based communication model for SDR. The OSSIE is an object-oriented SCA operating environment, where signal processing components are written in C++. The operating environment, often referred to as the core framework, implements the management, configuration, and control of the radio system. Every OSSIE's component is considered having two parts: one part realizing the signal processing and another managing the SCA infrastructure. The OSSIE waveforms are described in an XML that is used to describe component properties and interconnections between components in a waveform.

III. PACKET FORMULATION MODEL

The packet consists of three entities: the access code, the header, and the payload as shown Fig. 2. These three entities are processed separately and concatenated together to build a raw packet. The access code is 72 bits and used to detect the presence of a packet. The header is 54 bits that contains information associated with the packet and the link control. The payload ranges from zero to a maximum of 2745 bits.

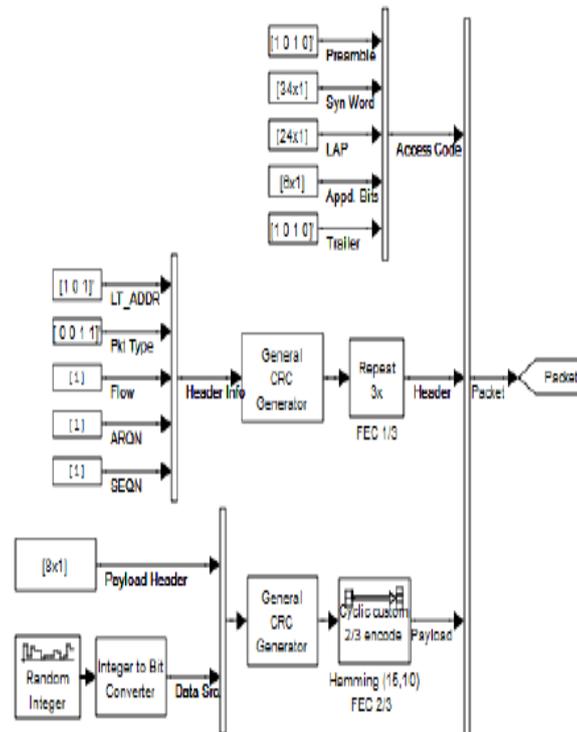


Fig : 1 Packet/Payload formulation model

In packet access code, the preamble is a fixed zero-one pattern of 4 symbols used to detect edges of received data. It should be 010 or 0101, the synchronization (SYNC) word is used for time synchronization, while the trailer is combination of 6-append-bits and 4-trailer-bits should be 1100101010 or 0011010101. In packet heard; the LT_ADDR is 3-bit active member address that is used to distinguish active members (slaves) in a piconet, the 4-bit packet type filed is used to distinguish between 16 different types of packets. The 1-bit flow used to control the flow over logical link, the flag is asserted when device is unable to receive any more data due to full receiver buffer. The 1-bit acknowledgement indication of success of transfer of the packet, the SEQN bit provides a sequential numbering scheme to order the data packet stream. The header error check (HEC) is an 8-bit word, used to check the header integrity. In payload encoding block the 136 bits are input from the source is overlapped by 8 bit payload header which is protected by CRC -16-CCITT, the resulting 160 bits sent to the encoding block where the 2/3 forward error correction (FEC) is applied to produce a total of 240 encoded bits. The 10-bit header info is protected by the 8 bit CRC which is then encoded by 1/3 FEC results in total of 54 bits. The header information is then concatenated with the 72 bits access code and the 240 payload encoded bits resulting in completion of 366 bits packet. The 72 bits access code is generated by lower address part of the master .

The SDR system development contains three sections for simplicity and reusability. The system uses a frequency hopping spread spectrum, using a pseudorandom sequence known to both transmitter and receiver. The transmitter and the receiver are properly synchronized so that they can hop together from channel to channel. The hop sequence is derived from the device address of the master; the phase in the hopping sequence is determined by its clock. The prototype system hops over 79 frequencies, each 1MHz wide, the bandwidth of the spectrum over which the hopping occurs is called the total hopping bandwidth that ranges 2.402-2.480 GHz.

IV. FREQUENCY HOPPING SPREAD SPECTRUM MODEL

Spread spectrums techniques are signal structuring techniques that employ a transmission bandwidth which is several orders of magnitude greater than the minimum required signal bandwidth. These techniques are used for multiple access and/or multiple functions. The frequency hopping involves a periodic change of transmission frequency, deterministically at a rate of 1600Hz; the hopping is derived from the master device address and its clock. The set of possible carrier frequencies is called the hopset that are 79, while the bandwidth of a channel used in the hoeset is called the instantaneous bandwidth that is 1 MHz. According to the section 15.247 (a) of the FCC regulations for FHSS devices requires devices to hop over at least 75 channels and limit the maximum bandwidth of each hopping channel to 1MHz. The hop selection kernel as shown in Fig. 2, addresses a register containing the RF channel indices those refer to 79 RF channels ranging 2.402GHz - 2.480GHz. The inputs A to D

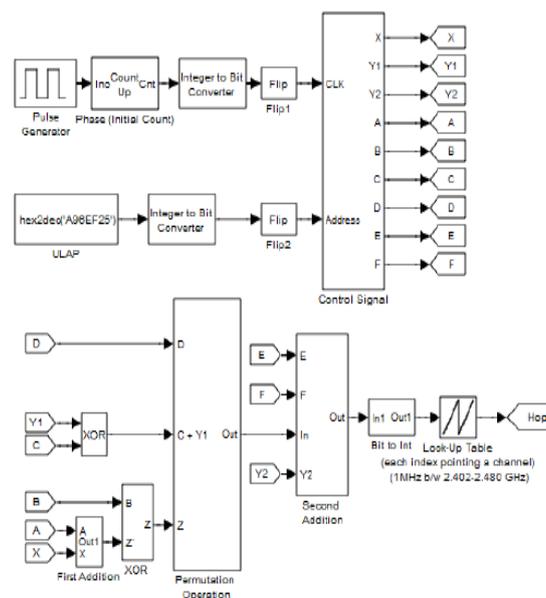


Fig. 2. Frequency hopping spread spectrum model

determines the ordering within the segment, the inputs E and F determine the mapping onto the hop frequencies. The X input determines the phase in the 32-hop segment, whereas Y1 and Y2 select the time slots between master-to-slave and slave-to-master for time division duplex.

V. MODULATION & TRANSMISSION MODEL

Modulation is a process by which some characteristics of a carrier wave, is varied in accordance with an information bearing signal. The information will be less vulnerable to noise or interference and it also permits the use of multi-access techniques. The basic aim of digital modulation is to transfer a digital bit stream over an analog Bandpass channel. The system uses a Gaussian frequency shift keying (GFSK) modulation, before the baseband pulses (-1, 1) go into the FSK modulator, it is passed through a Gaussian filter to make the pulse smoother so to limit its spectral width. The frequency shift keying involves binary signaling by using the two frequencies separated by Δf Hz, where Δf is a frequency deviation that is smaller as compared to the carrier frequency f_c . The modulation index μ is defined as;

$$\mu = \Delta f T \tag{1}$$

Where T is a symbol time (that is the inverse of the datarate for binary schemes), the frequency deviation (Δf) is the maximum frequency shift with respect to the carrier frequency, if a '0' or '1' is being transmitted, and it can vary between 140 KHz and 175 KHz. In GFSK modulation, the symbol '0' shall be modulated on $-\Delta f$ and symbol '1' shall be modulated on $+\Delta f$.

The interpolation factor in the transmit path and decimation factor in the receive path must be properly configured. Interpolation (also known as up sampling) increases the sampling rate and requires that we somehow produce values between the samples of the signal. The interpolation by a factor q is accomplished by inserting $q-1$ zeros in between each sample of $x[n]$ and then discrete time low-pass filtering. The sampling process is critical for radio receivers that digitize signals. Sampling an analog signal at IF or RF results in replicas of the signal's spectrum repeated at uniform intervals. The choice of the sampling rate of such signals is dependent on the signal's bandwidth and the IF or RF center frequency. The sampling frequency (Nyquist frequency) or sampling rate f_s is defined as the number of samples obtained in one second; $f_s = 1/T$. The time interval between successive samples is referred to as the sampling interval ($T = 1/f_s$), the Nyquist sampling rate or Nyquist rate can be calculated by the following equation 2.

$$\text{Sample Rate} = \text{ADC Rate}/\text{decimation rate} \tag{2}$$

The Nyquist rate can be used to calculate the maximum RF bandwidth as, the uniform sampling rate f_s (in samples per unit time) should be at least twice the maximum bandwidth.

The sampling rate of the device is 100 MHz, so an interpolation rate of 512 results in a Simulink sample time of $512/100e6$.

$$B = f_s/2 \tag{3}$$

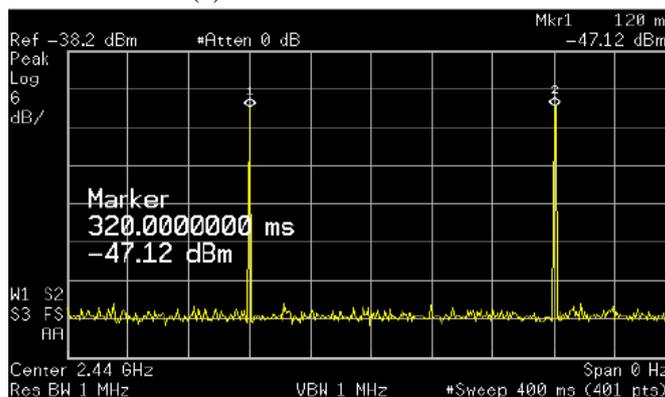


Fig. 3. Transmission spectrum view

There are still many open issues; a flexibility of an SDR as the possibility to load and install new software on an SDR unit (i.e. mobile phones) and also the security issues. The major challenge with SDR is how to achieve sufficient computational capacity, in particular for small handheld units where a number of operations may be carried out in parallel and the deterministic behavior allows low-level optimization of implementations. The IEEE standards committee on next generation radio and spectrum management and the ETSI reconfigurable radio systems technical committee have made the principles for the software defined radios and especially for cognitive radios.

VI. COGNITION FOR INTERFERENCE ESTIMATION

The cognitive radio (CR) is an emerging new technology, which is currently far from mature in terms of real applications. The CR is considered an intelligent wireless communication system that should be aware of its surrounding environment. It mainly consists of three modules as shown in Fig. 6; spectrum sensing, spectrum predicting, and spectrum management. A survey of spectrum sensing algorithms for cognitive radio applications is presented..

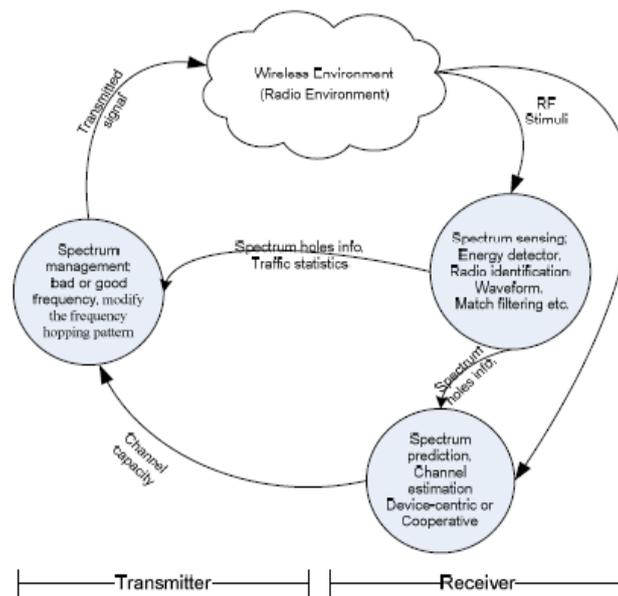


Fig. 4 Basic cognitive cycle model

For re-configurability a cognitive radio looks naturally to software- defined radio to perform its task. The prototype system has some intelligence to allow a radio terminal to automatically sense, recognize, and make wise use of available radio frequency spectrum at a given time within the frequency band. The method for interference estimation consists of measuring the bit error rate and percentage of packets loss per channel (frequency).The prototype system uses an interference mitigation technique with the ability to detect the presence of other systems operating in the same band (ISM). The master device controls all packet transmission. The measurements collected by a slave device are sent to the master to avoid data transmission on a “bad” channel. The statistics are based on the packet lost measurements per-channel. The channel is listed as bad, when the packet lost is below the threshold. The master device modifies the frequency hopping pattern that is called an adaptive frequency hopping. It is used for determining whether there are other devices present in the ISM band or not and to avoid them.

VII. CONCLUSION AND FUTURE WORK

In this paper firstly we presented a survey of SDR SW architectures and SDR HW platforms and secondly developed a prototype system (testbed) for experimenting and emulating the software defined radios. The motivation for a test bed is provided by the need to validate various coding schemes, modulation techniques, and spread spectrum algorithms to measure the quality of service. The platform independent SDR prototype system is developed in MATLAB/Simulink and interfaced with the field programmable gate array. The prototype system opens the doors to experiment different spread spectrum techniques, various coding and modulation schemes for software-defined radios to realize the cognitive radios. Future work would be practical implications of various statistical and artificial intelligence techniques for cognitive radios; The performance of the baseband transmitter is analyzed using constellation and eye diagrams for different modulation and compression techniques while considering an additive white Guassian noise channel. The performance of the receiver is analyzed by comparing the input and output waveforms.

REFERENCES

- [1]Akira Fujimaki et al, ‘Broad Band Sojvhare-Dejined Radio Receivers Based on Superconductive Devices’, IEEE Transaction on Applied Superconductivity, Vol. 11,No. 1,March2001.
- [2]Buracchini.E,“The software radio concept,” IEEE Communications Magazine, vol. 38, no. 9, Sept. 2000.



- [3]Chang,C., J. Wawrzynek, and R. W. Brodersen, "Bee2: a high-end reconfigurable computing system," in Proc. of IEEE Magazine, Design and Test of Computers, vol. 22, no. 2. April 2005.
- [4]Crochiere. R. E. and L. R. Rabiner, "Interpolation and decimation of digital signals - A tutorial review," in Proc. of the IEEE, vol. 69, no.3, March 1981.
- [5]Enrico Buracchini, "The software radio concept," IEEE Communications Magazine, vol. 38, no. 9, Sept. 2000.
- [6]González. C. R. A., "Open-source SCA-based core framework and rapid development tools enable software-defined radio education and research," IEEE Communication. Magazine., vol. 47, no. 10, pp. 48– 55, Oct. 2009.
- [7]Jerker.B. and V. Seppo, "Convergence of hardware and software in platforms for radio technologies," IEEE Communications Magazine, vol.44, no. 11, 2006.
- [8]Minden. G. J., et al, "Cognitive radios for dynamic spectrum access – an agile radio for wireless innovation," IEEE Communications Magazine, vol. 45, no. 5, Dublin, Ireland, May 2007.
- [9]Mitola J., "Software radio architecture: a mathematical perspective," IEEE Journal on Selected Areas in Communication. vol. 17, no. 4, April 1999.
- [9]Mitola J., "The software radio architecture," IEEE Communications Magazine, pp. 26–38, May 1995.
- [10]Newman,T.R S.M.S. Hasan, D. Depoy, T. Bose, and J.H. Reed "Designing and deploying a building-wide cognitive radio network testbed," IEEE Communications Magazine, vol. 48, no 9, Sept. 2010.
- [11]Oetting.J. D., "A comparison of modulations techniques for digital radios," IEEE Transactions on Communications. vol. 27, no.12, Dec.1979.
- [12]Rouphael.T. J.,RF and Digital Signal Processing for Software-Defined Radio – A Multi-Standard Multi-Mode Approach, Elsevier Inc, 2009.
- [13]Tim Hentschel and Gerhard Fettweis, "Sample Rate Conversion .for Sqr'iware Radio". IEEE Communications Magazine, pp. 142-1 50, August 2000.
- [14]Walden.R.H., "Analog-to-digital converter survey and analysis," IEEE Journal on Selected Areas in Communications, vol. 17, no. 4, April 1999.
- [15]WILLIG et al., "Wireless technologies in industrial network," IEEE Special Issue on Industrial Communication Systems, vol. 93, June 2005.
- [16] Yücek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," IEEE Communication Surveys & Tutorials, vol. 11, no. 1, April 2009.