ISSN 2047-3338

Software-Defined Radio Architecture for Broadband OFDM Transceivers

Hafiz Muhammad Fahad Noman[1], Muhammad Fuzail[2] and Junaid Arshad[3]

[1,2,3]Department of Computer Science & Engineering, University of Engineering & Technology, Lahore, Pakistan

[1]engr.fahadnoman@gmail.com, [2]m.fuzail@ymail.com, [3]junaidarshad@uet.edu.pk

***Abstract*— This paper presents a software-defined radio (SDR) system with reconfigurable architecture for OFDM transceivers. DSP-based architecture provides interoperability and adaptability among operational modes of the OFDM system. But limitations due to memory, bus speed and data type compatibility in DSP motivates SDR architecture over USRP boards. This model is initial evaluation of the suitability of OFDM transmission over USRP for an upcoming large scale CR testbed. SDR Architecture supporting 3GPP-LTE microcomputer object (MCO) system is designed for both flexibility and extensibility. It explains new simulation models in MATLAB that lead toward the concept of the virtual system sample. In this framework, we completely merge hardware, software, algorithms, into one radio link system level objected oriented model. Extensibility concepts, virtual abstraction, and acceleration hardware are important architectural features. On the other hand, OFDM is a promising technique for high-speed data transmission over multipath fading channels and has been considered to be the best candidate for 4G mobile networks.**

***Index Terms*— SDR, OFDM, FFT, USRP, FPGA, LTE and CR**

I. INTRODUCTION

THE concept of software defined radio (SDR) is early efforts of Joseph Mitola who defined SDR as “Software Radio is a radio that is defined in software and whose physical layer behavior can be significantly altered through changes to its software”. Recently, groups such as the Software Radio Forum [4] have modified the original definition to include “concepts, technologies and standards for wireless communications systems and devices, to support the needs of all user domains including consumer, commercial, public safety, and military markets, and stakeholders such as regulatory authorities….and a collection of interfaces and standards.”

Recently wideband wireless OFDM communication systems have gained in popularity because of OFDM’s spectral efficiency and capability to transmit high data rates over broadband radio channels with frequency selective fading [1] and jamming. OFDM also provides the potential of low cost wireless services because of its high spectral efficiency [3]. Additionally, the complex task of channel estimation is substantially reduced if differential modulation is used [1]. The problem of inter symbol interference (ISI) in very dispersive environments can largely be removed in OFDM communication systems by the addition of a cyclic extension. Because of its robustness, OFDM has been adopted by a variety of standards which include the European Telecommunications Standards Institute (ETSI) for Terrestrial TV and the IEEE 802.11 standard for wireless LANs operating at bit rates up to 30 Mb/s at 5 GHz [3].

Moreover, The SDR solution provides a high flexibility to adapt the system to meet different needs of different customers. The use of software to create and control most of the functions in base stations and customer premise equipment allows for new levels of flexibility and low-cost system upgrades [1].

II. MOTIVATION AND PROBLEM DEFINTION

As semiconductor devices are shrinking, the rate of new services introduced will soon exceed the rate of miniaturization in electronic packaging. What is needed is a flexible, universal radio platform for receive and transmit, which can be programmed to steer to any band, tune to a channel of any bandwidth, and receive any modulation—all within reasonable physical constraints, including size, weight, power consumption, and more important, cost.

On the other hands, implementation of new generation wireless transceivers resulted in a significant complexity increase with respect to traditional modems. An example is the IEEE 802.11n with a maximum data rate of 600 Mbit/s. In the past, multiple digital signal processors (DSPs) were required to implement the baseband operation in a single transceiver.

With the rapid increase in transistor density, it has become feasible to keep the functionality entirely in the programmable DSP engines, greatly improving time to market and allowing faster changes and upgrades. The combination of advances in semiconductor technology in hardware devices such as field-programmable gate arrays (FPGAs) and DSPs, and the need for rapid upgrades (software) has transformed wireless transceivers into programmable devices, allowing quick upgrades and backward compatibility with old standards.

In other words, significant amounts of computations are done by software in a DSP, including communication protocols. Software Defined Radio (SDR) platforms are foreseen to replace traditional radio systems where the main transceiver functionalities are still performed in hardware [6]. For instance, Universal Software Radio Peripheral (USRP) platform developed by Ettus research moves the whole baseband processing onto the host computer, while the hardware is only dedicated to high speed operations like digital up and down conversion and to the Radio Frequency (RF) frontend [7].

MIMO-OFDM is currently being considered for a number of developing wireless standards such as IEEE 802.16e-2005 and has been suggested for use in beyond 3G (B3G) and 4G wireless communications; consequently, the study of MIMO-OFDM in realistic environments is of great importance.

III. SOLUTION APPROACHES

A. Reconfigurable Architecture of SDR using DSP

Fig. 1 depicts a SDR transmitter and receiver with a reconfigurable architecture. At the transmitter, all baseband operations inside the dashed box are software-based processing modules on a given hardware platform as depicted in Fig. 1(a). The DSP software performs source encoding, channel coding, and data stream multiplexing and modulation as needed. Different modulation requirements are implemented with different software modules, including preamble sequences and hand-shaking protocols for transmitting. Similarly, at the receiver, all reversed baseband operations, such as de-multiplexing, de-modulation, channel decoding, source decoding are performed by software-based processing units on the given hardware platform as depicted in the dashed box of Fig. 1(b).

The flexibility of DSP-based solutions in the SDR systems is due to the programmability. It provides many benefits. For example, algorithms can continually be updated and improved as computational methods advance. For standards-based modules, such as IEEE 802.11[9], a programmable implementation allows modules to remain compliant as the standards upgraded. A software upgrade is usually all that is needed while the backward compatibility still promised in the existing modules.

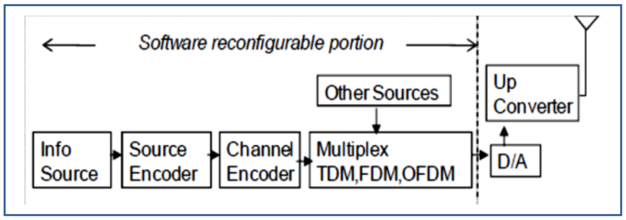

Fig. 1(a): Software reconfigurable transmitter block diagram

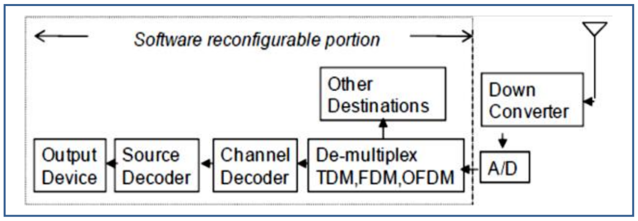

Fig. 1(b): Software reconfigurable receiver block diagram

While programmable devices serve as the heart of the hardware platform within SDR systems, the software-based transceiver enjoys its flexibility and adaptability for many applications. For example, certified software components compliant with 802.11 Wi-Fi standards are guaranteed to be interoperable with other modules. Note that the 802.11 standard is part of a family of IEEE standards devoted to characterize local and wide area networks. Since the wireless channel varies with time, link adaptation is recommended in order to support reliable communications and maximize the throughput. For example, IEEE 802.16 and 802.11n standards define a large set of modulation and coding schemes to facilitate this goal.

On the other hand, in the new deployment with cognitive radio techniques, adaptive modulation may be used to optimize spectrum usage, and minimize annual frequency lease costs. As a result, efficient and practical link adaptive techniques are needed for wireless networks.

The software modulation and demodulation modules of DSP based architecture can be efficiently updated and switched to meet this new design requirement. For this reason, we prefer, M-ary phase shift keying (PSK), and fast Fourier transform (FFT) modules of orthogonal frequency division multiplexing (OFDM) systems. For experimental implementation, TMS320C5409 DSP is employed. It is a 16-bit fixed-point computing engine. A 100 MHz clock implies to an instruction cycle of 10 ns. The development tool, Code Composer Studio (CCS), is used to simulate the DSP in a software sense so that it is not necessary to download and implement algorithms in an actual hardware base. As shown in Fig. 2, the coherent BPSK- and QPSK-OFDM systems have the same BER and agreed with the theoretical bit error rate (BER). Note that the QPSK-OFDM system employs I- and Q-channels, named as dI and dQ for BER testing as depicted in Fig. 2.

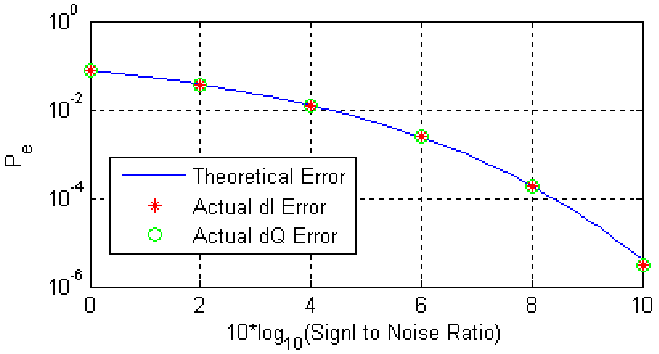

Fig. 2: BER of the BPSK- and QPSK- OFDM system

B. SDR using USRP boards for OFDM

USRP platform has been originally designed for supporting GNU radio. This is a software tool for runtime signal processing including a wide set of blocks which can be linked with the aim of building a transceiver chain. USRP hardware can however also be interfaced with customized C++ code by using Universal Peripheral Driver (UHD) , thus overcoming the flexibility limitations as well as the lack of temporization features in the GNU radio libraries.

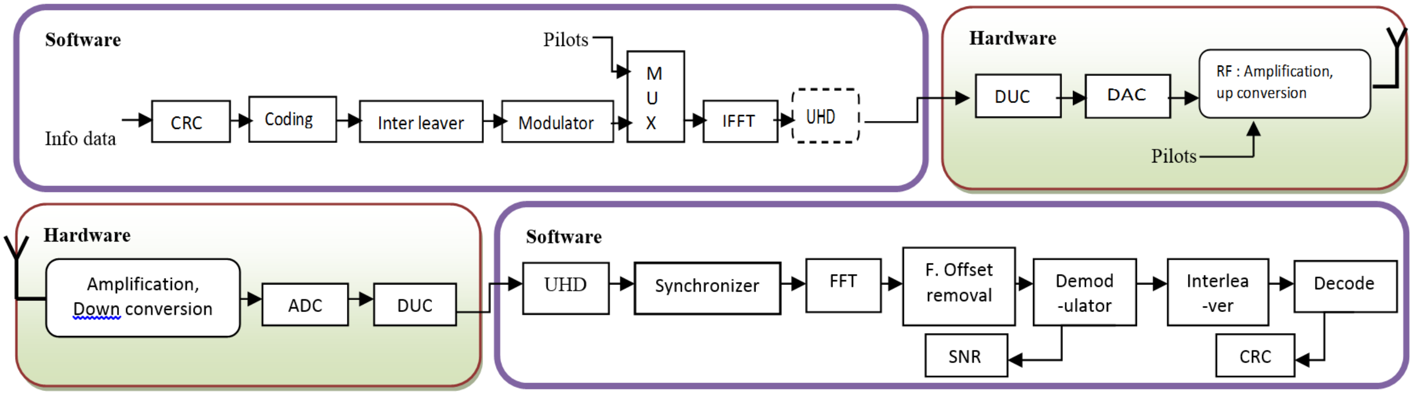

Fig. 3: Simplified structure of the OFDM

The USRP2 board is an evolution of a previous USRP computer-hosted hardware for software radio and it has been released in September 2008. It features a Xilinx Spartan 3-2000 Field Programmable Gate Array (FPGA), 2 Analog-to-Digital Converters (ADCs) at 100 MS/s, 14 bits, and 2 Digital to- Analog Converters (DACs) at 400 MS/s, 16 bits. It also features a Secure Digital (SD) card reader for the firmware. The internal clock is a 100 MHz Voltage Controlled Oscillator (VCO) with nominal accuracy of 10 ppm, but the board also provides connection to an external reference clock as well as a Pulse Per Second (PPS) port. USRP2 can be interfaced with the host PC through Gigabit Ethernet.

The UHD provides a wide set of Application Programming Interfaces (APIs) for interfacing the host computer with the USRP2 board [10]. Moreover, it is possible through the APIs to set the main RF parameters such as carrier frequency and gain of transmitter/receiver amplifiers, as well as transmit and receive sample rate. An interesting feature which differentiates the UHD from GNU radio libraries is the metadata which is attached to the data transferred between host PC and USRP2. The metadata specifies time-related settings which enable a precise control on the transmission and reception of time samples. This will enable network synchronization of the multiple cognitive nodes in our upcoming testbed.

(1) Transmitter baseband processing

In the transmitter a Cyclic Redundancy Code (CRC) is appended to a vector of information bits, which is then fed to a Viterbi encoder with rate 1/3 [11]. The choice of the coding scheme is motivated by a trade-off between expected performance and decoding latency. The coded bits are interleaved and QPSK modulated. Then, the resultant data symbols are divided onto groups of cardinality N_{sub} and mapped over the transmission subcarriers. An OFDM time symbol for each of these groups is thus obtained by Inverse Fast Fourier Transform (IFFT) with size N_{FFT} plus Cyclic

Prefix (CP) insertion by appending the last N_{CP} samples at the beginning of the symbol. Note that, while GNU radio uses by default the preamble for both synchronization and channel estimation [6], we decided to allocate different resources for the two tasks. This is because the same preamble can be used by multiple nodes, while different frequency patterns are needed for discriminating the channel frequency responses of multiple nodes.

(2) Receiver baseband processing

In the receiver a wide set of samples is retrieved from the hardware through UHD and fed to the synchronizer block. The synchronizer exploits two consecutive tasks. Firstly, the symbol synchronization is acquired by using the maximum likelihood (ML) estimation algorithm. This technique exploits the redundancy information given by the CP by self-correlating the vector of the received samples with its delayed version. Furthermore, it also estimates the frequency offset due to the difference in the transmitter and receiver oscillators as a fraction of the subcarrier spacing. It is therefore possible to simply estimate the SNR as S/N. The retrieved bit stream is then de-interleaved and Viterbi decoded. Finally, the CRC is computed to check whether the data transmission has been successful.

(3) SNR vs. input power level

SNR performance as a function of the input level at the receiver as well as the receiver gain can be measured in order to estimate the efficiency of the system for reliable communication. Transmitter and receiver boards are connected via cables and 25 dB attenuators. The transmit power is swapped in order to obtain different input power levels at the receiver. For each combination of input level and receiver gain, 2000 samples were gathered and the SNR estimated. The gain value set in the receiver through UHD is actually split into a Low Noise Amplifier (LNA) gain and a base-band gain. The LNA gain can be set to 0, 15 or 30dB. The remaining gain is achieved at baseband. The split of the gain value is done with priority given to the LNA gain, in order to maximize the sensitivity. Results are shown in Figure 3. It can be seen that the base-band gain can only marginally improve the SNR. Note that in practice, the exact value of the input signal level is not known beforehand

C. Micro Computer Object based SDR Architecture supporting 3GPP-LTE

SDR system architecture supporting 3GPP-LTE description is illustrated in Figure 5. It focuses on broadband wireless protocols such as 3GPP-LTE [12] as a target application. The

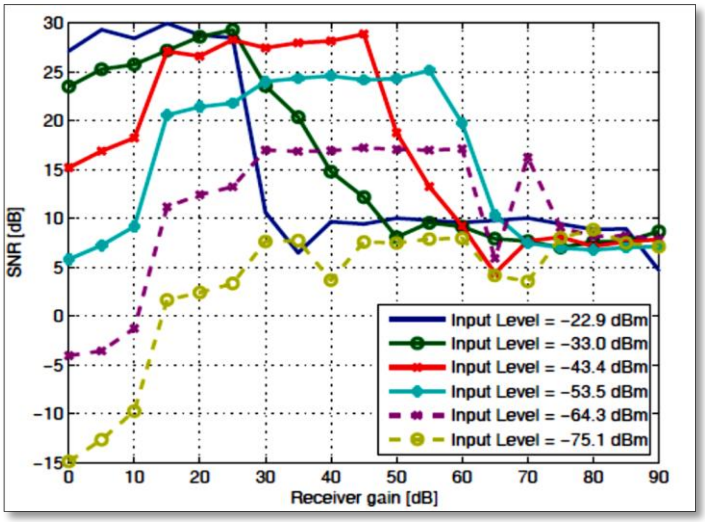

Fig. 4: The estimated SNR as a function of input level and receiver gain

downlink physical channel of 3GPP-LTE is a shared resource upon which multiple users simultaneously receive downlink transmission. The downlink utilizes QPSK, 16QAM and 64QAM multi-carrier orthogonal frequency division multiplexing with multiple access (MCOFDM-A). The 3GPP-LTE protocol supports both frequency division duplex (FDD) and time division duplex (TDD) signaling. In FDD mode, 15 KHz spaced sub-carriers are denoted as resource elements (REs). The normal cyclic prefix is 4.76μsec.

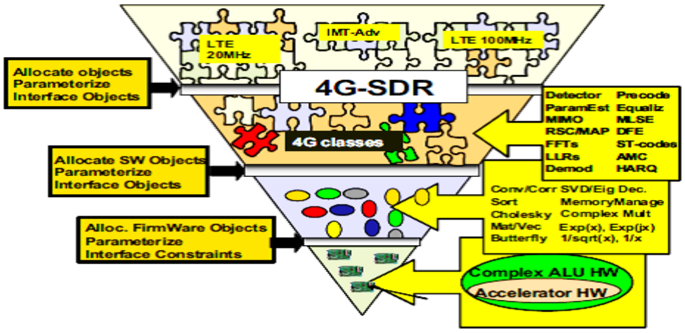

Fig. 5: SDR Hierarchal Architecture supporting 3GPP-LTE

Analysis of control and signaling overhead also lead the 3GPP-Radio Access Network (RAN) group to define a smallest resolvable time-frequency data structure, known as a resource block (RB). For a downlink transmission using a normal cyclic prefix, a resource block consists of 2-Dim grid of 12 REs in frequency versus a time grid of 14 OFDM symbols. Each OFDM symbol waveform is approximately 71.4μsec in duration. The actual duration depends on the size of the cycle prefix chosen. A sub-frame represents a 2 OFDM slots. A frame is 10msec, and corresponds to 10 sub-frames or 20 slots.

This SDR solution involves defining an ALU firmware systems with hardware acceleration complexes modeled as programmable, extensible, and invoke able micro level communication objects (MCOs). This implies that we have an object oriented class description of the MCO hardware unit. Sometimes we interact with the physical hardware unit and sometimes with its firmware model which can be instantiated as an object. In Fig. 5, MCOs are shown at the bottom of the pyramid.

A key aspect of MCOs is that the assemblers and C compilers are generated through correct by construction methodologies. The program can be changed quickly using high level language. Also, multi-cycle run-to-completion hardware accelerated FSM module instructions can be inserted into the instruction set, and external I/O ports are devoted to very-wide register files that contain late binding extensible instruction parameterizable after design completion. Multi-cycle instructions pass control of the MCO system state over to the finite state machine (FSM) in the control unit corresponding to the multicycle instruction. These modules can accelerate computation by using larger control words via an FSM. In addition they access unique complex arithmetic hardware in the ALU which remains dormant in normal single cycle instructions. MCOs are fixed point, but are enabled with quasi-floating point capability by means of extension scale registers that allow power of 2 shifting of data in ALU registers. This enables numerically sensitive operations to be performed with of nearly floating point precision, while fixed point operations can be performed for operations with low sensitivity to quantization noise associated with fixed point register affects.

In normal SDR control operation, the MCO performs single cycle operations. In accelerated instruction mode, the MCO controller changes from its normal system state to the k_{th} accelerator state. This enables the k_{th} accelerator's control stream to control the entire hardware regime of the MCO hardware and allows the accelerator to access specialized ALU logic that increases the effective latency in clock cycles of the accelerator operation. The overall Matlab model consists of an ALU, controller, program memory, multiple (N) data memories, instruction decoder, address generator unit, and the extensible instruction memory. In Matlab each MCO hardware module is defined via a class description. In turn, we create classes of high level primitive operations and protocol classes, each of which are allocate multiple MCOs.

Each MCO contains a wide-width register file of memory in which extensible instruction capability can be externally loaded prior to run time or dynamically during run time. When the MCO controller recognizes the specialized extensible instruction the decode bits are fetched from the extensible instruction memory, which has been preloaded with the correct sequence of decode bits.

IV. CONCLUSION

This paper presents a SDR based OFDM transceiver using 3 different approaches. It is explained by a software reconfigurable OFDM system using a programmable fixed-point DSP, customized SDR architecture for OFDM transmission over USRP boards and Micro Computer Object based SDR Architecture supporting 3GPP-LTE.

DSP-based architecture provides interoperability and adaptability among operational modes of the OFDM system. Adaptive modulation can be applied to this system which minimizes the antenna sizes, while still being able to provide high data rate. But limitations due to memory, bus speed and data type compatibility in DSP motivates to an alternate optimized solution.

SDR architecture for OFDM transmission over USRP boards is proposed for overcoming the detrimental effects due to the limited accuracy of the internal reference clock. Results in terms of SNR offer practical insights on how to set the receiver gain according to the input power level. This model is initial evaluation of the suitability of OFDM transmission over USRP for an upcoming large scale CR test bed.

SDR Architecture supporting 3GPP-LTE micro computer object (MCO) system is designed for both flexibility and extensibility. It explains new simulation models in MATLAB that lead toward the concept of the virtual system sample. In this framework, we completely merge hardware, software, algorithms, into one radio link system level objected oriented model. Extensibility concepts, virtual abstraction, and acceleration hardware are important architectural features.

V. FUTURE DEVELOPMENTS

Future research efforts are being carried in accommodating multiple Component Carriers (CCs) in the transmission spectrum in order to enable multiple nodes to dynamically share the same bandwidth. Moreover, a multi-threaded version of the SDR transceiver will be implemented with the aim of decreasing the processing latency and full filling the real time requirements of CR testbed which leads the concept of overlay CR to 4G cellular networks for orthogonal frequency- division multiplexing (OFDM)-based TV spectrum sharing.

REFERENCES

[1] Vijay K. Carg, Wireless Communications and Networking, First Ed.,Morgan Kaufmann Publishers, 2008

[2] Lee Pucker, « Finding MIMO : A Proposed Model for Incorporating Multiple Input Multiple Output Into Software Defined Radio”, Proceeding of the SDR 05 Technical Conference and ProductExposition;2005

[3] J. Chuang and N. Sollenberger, “Beyond 3G: wideband wireless data access based on OFDM and dynamic packet assignment,” in IEEE Communications Magazine, July 2000.

[4] SDR Forum, www.sdrforum.org.

[5] B. Kelley, “Software Defined Radio for Broadband OFDM Protocols,” Proc. IEEE Intern. Conf. Systems, Man, Cybernetics, San Antoniou, TX, Oct. 2009, pp. 2309-2314.

[6] M. Dillinger, K. Madani, and N. Alonistioti, Software Defined Radio:Architectures, Systems and Functions. John Wiley - IEEE Press, 2003.

[7] Ettus Research, http://www.ettus.com.

[8] J.E. Kleider, S. Chuprun, and S. Gifford, “Throughput Maximization for Survivable OFDM Subject to QOS Constraint,” in Proc. ARL Federated Laboratory 4th Annual Symposium, pp.105-110, 2000.

[9] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, “Measurements of a wireless link in an industrial environment using an IEEE 802.11 compliant physical layer,” IEEE Trans. Ind. Electron., vol. 49, no. 6, Dec. 2002, pp. 1265-1282.

[10] Universal Hardware Driver, http://ettusapps.sourcerepo.com/redmine/ettus/projects/uhd/wiki.

[11] J. Proakis, Digital Communications, 4th edition. Mc-Graw Hill, 2001.

[12] 3GPP TS 36.201: 3GPP TSG-RAN EUTRA LTE Physical Layer General Description, R8 WG1, September 2007.3GPP LTE