EXPERIMENTAL EVALUATION OF SECONDARY NC-OFDM TRANSCEIVER FOR COEXISTENCE WITH PRIMARY TRANSMITTER

BY ABHISHEK BINDIGANAVILE

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Prof. Predrag Spasojevic

and approved by

_____

_____

_____

New Brunswick, New Jersey

May, 2012

ABSTRACT OF THE THESIS

Experimental Evaluation of Secondary NC-OFDM Transceiver for Coexistence with Primary Transmitter

by Abhishek Bindiganavile

Thesis Director: Prof. Predrag Spasojevic

RF Spectrum scarcity inhibits development of newer wireless communication technologies in a big way. With the current spectrum policies, there is heavy under utilization of available spectrum. This calls for more efficient spectrum allocation. One of the solutions to the problem is dynamic spectrum allocation algorithms. This can be achieved with the use of Cognitive Radio Technology. Cognitive radios allow for dynamic changing of transmission parameters depending on the environment and available spectrum bands. The definition of Cognitive radios terms it an intelligent radio which alters its transmission/reception parameters in accordance with the radio environment and the network state to utilize spectrum efficiently.

Cognitive radio technology allows for use of unlicensed secondary devices to utilize spectrum unused by licensed primary devices. The secondary devices can utilize the spectrum and transmit in bands not used by the primary devices in order to avoid any detrimental interference to the licensed primary users. This calls for stringent control of the secondary transceiver. In this thesis, we focus on designing an LTE standard based Non Contiguous Orthogonal Frequency Division Multiplexing Secondary Transceiver. The major focus has been on hardware design of the secondary LTE transceiver and a system level implementation on a National Instruments based Real Time Embedded

Controller platform, with relevant design blocks developed on FPGA hardware for efficient implementation. The various functionalities and tradeoffs involving hardware design of a transceiver have been studied. In addition, the effect of employing FPGA based design with a view of resource utilization and latency of the whole transceiver system, has been considered. Dynamic Spectrum Access setup was evaluated with the NI based secondary transceiver platform by evaluating the secondary receiver's performance in the presence of a primary user in the notch. This allowed for efficient utilization of spectrum for the unlicensed secondary user without causing detrimental interference to the primary user's signal.

List of Figures

List of Tables

Acknowledgements

I would like to thank, firstly, my family for their unwavering support and constant encouragement through my graduate study. I would like to thank my advisor Prof. Predrag Spasojevic, for his invaluable guidance and advice while working on my research. His inspiration was a driving force in many cases. The numerous discussions held with him have given me a deeper understanding of a lot of topics involving both communication theory and system level architecture and countless other areas as well. I would like to express my gratitude to Prof. Jeffrey Walling and Prof. David Daut for being on my thesis committee.

I am grateful to all my friends, faculty and staff at Winlab and the ECE Dept. as well, without whose invaluable help this experience would not have been the same. In particular, I would like to thank Srinivas Pinagapany, Samson Sequeira, Ajay Iyer, Swapnil Mhaske, Ashwin Revo and Chris Smith who were a part of this research project. I would also like to thank the Winlab staff for access to facilities and their help in academic and non academic matters.

I would also like to thank Masayuki Ariyoshi and Yasunori Futatsugi of System Platforms Research Labs, NEC Corporation, Japan for funding this research project and their valuable guidance throughout the project. I would also like to express my gratitude to Mr. Ahsan Aziz, National Instruments, Austin, Texas, for his valuable direction and help during the numerous discussions during the course of this project.

Dedication

To my parents, grandparents, my teachers and friends

Table of Contents

Chapter 1

Introduction

1.1 Background

Radio spectrum is a valuable commodity which is becoming scarce every passing day. Due to this scarcity, soon, we may be faced with the unavailability of this valuable resource thereby hindering newer and better wireless technologies. With the current spectrum allocation policies, there is severe under utilization of available spectrum. This calls for tighter spectrum control and more efficient spectrum allocation . Opportunistic spectrum access of licensed spectrum by unlicensed devices could be viewed as a potential solution to the problem of scarce spectrum availability. This solution advocates the secondary users to actively monitor the occupancy of licensed spectrum. This requires the secondary users to continually scan the spectrum and occupy the spectrum when it becomes available. One of the constraints in such a system would be to avoid harmful interference caused by unlicensed secondary users. In such a case, the optimizing the performance of the secondary system becomes significant, since secondary users are not pre-allocated dedicated frequency bands or time slots. The secondary systems also have to be robust in the sense that they have to overcome interference from possible primary users. The secondary receivers have to be efficient enough to detect secondary transmissions of low power in the presence of interfering primary users. The scenario is shown in Fig. 1.1.

In this thesis, the focus is on designing a secondary transceiver system and its performance characterized by performing error analysis. The secondary transceiver is an Long Term Evolution (LTE) [1] based Orthogonal Frequency Division Multiplexing (OFDM) system. The secondary system was originally developed on the USRP2/GNU Radio

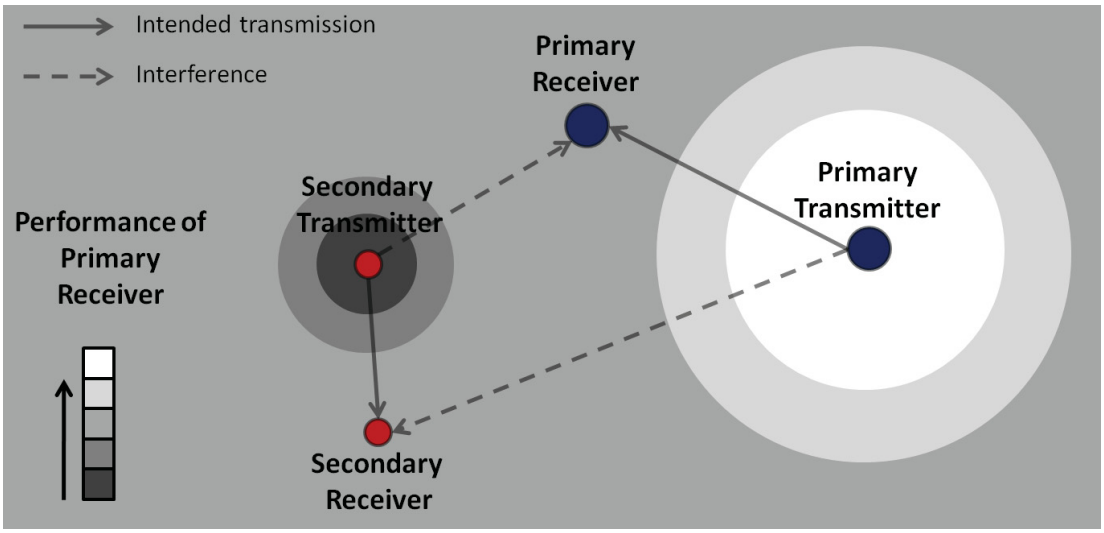


Figure 1.1: Co-existing Primary and Secondary users.

[2] based platform. An alternate implementation of the secondary system was considered on a National Instruments based LabView RT/FPGA platform [3]. A Dynamic Spectrum Access system was developed as a part of the WINLAB-NEC Collaborative Cognitive Radio Project. This implementation consists of adaptive spectrum sensing mechanism followed by dynamic secondary spectrum access using Non Contiguous Orthogonal Frequency Division Multiplexing (NC-OFDM)[4][5]. Also, an interference suppression technique was implemented to safeguard primary systems from interference due to the secondary system. The secondary system was implemented in the following two different platforms:

- USRP2/GNU Radio based platform on the Orbit Wireless testbed at WINLAB, Rutgers University.
- National Instruments based LabView Real Time/ FPGA Controller.

The system was evaluated to study the performance characteristics of the primary and secondary systems, and primarily the secondary transceiver's performance was characterized in the presence of a primary interferer.

1.2 Cognitive Radio Technology

Cognitive Radio (CR) is defined as "A paradigm for wireless communication in which either the network or the wireless node itself changes particular transmission or reception parameters to execute its tasks efficiently without interfering with licensed users". This means that the cognitive radio system must be aware of the Radio Frequency spectra occupied as well as the current state of the network in order to function in an optimal way. A cognitive radio system possesses the ability to perform opportunistic spectrum access, it is being toted as a likely solution to the spectrum scarcity problem. Since its first conception by Joseph Mitola [6][7][8] in 1998, significant research has been performed on making a cognitive radio system a practicality. The fundamental core functioning of a cognitive radio is a Software Defined Radio. The cognition function is added by various algorithms, which basically cause the radio to be a "smart" software defined radio.

1.3 Software Defined Radio

A software defined radio is a communication device where components which are typically implemented on hardware are realized using software and programmable hardware. This implementation can be performed on general purpose computers, as well as dedicated embedded systems. Typical characteristics of an SDR are wideband RF converters, Digital to Analog Converters (DAC), Analog to Digital Converters (ADC), multiband frontend antennas and a general purpose processor, which handles the fundamental signal processing. Such a platform becomes convenient for implementing dynamic spectrum access since it can be used to realize various modulation schemes and can operate on a wide range of frequencies owing to its wideband RF frontends. Algorithms can be designed such that various transmission characteristics can be varied on-the-fly based on the current RF occupancy and current state of the network. The DACs and the ADCs handle the baseband equivalent of the operations that can be performed by the SDR.

1.3.1 Functions of a Software Defined Radio

Some key functionalities of a Software Defined Radio can be listed as follows:

- Transmission
  - Determine channel parameters to verify channel availability
  - Evaluate the channel parameters
  - Dynamically set the transmission frequency and power
  - Adaptive channel modulation
- Reception
  - Diagnose the corresponding set channel and the neighboring channels.
  - Identify the channel modulation type
  - Adaptively estimate channel impairments and equalize them
  - Robust error detection and correction

Chapter 2

Foundations and Need for Dynamic Spectrum Access

2.1 Signal Model

The received signal *y(t)* is the sum of the transmitted signal *x(t)* and the Additive White Gaussian Noise (AWGN) *w(t)*. If N is the number of complex samples received at the receiver, then the received samples can be represented as

$$y(n) = x(n) + w(n) \tag{2.1}$$

2.2 Need for advanced Dynamic Spectrum Access

Traditional spectrum licences have static rights to a fixed amount of spectrum. With the advent of broadband wireless devices, spectrum scarcity has become a problem of national importance. The concept of DSA (Dynamic Spectrum Access) in which an unlicensed device opportunistically accesses the spectrum in the licensed frequency bands, can be used to alleviate the problem of scarce spectrum usage. The licensed devices are called the primary users (PU) and the unlicensed devices are called the secondary users (SU). The Primary User must be protected from detrimental interference from the opportunistic secondary users and this performance evaluation has been observed in [4][5].

2.2.1 Unlicensed Devices

Unlicensed radio frequency devices are classified as follows:

- Intentional radiators: These are devices that intentionally generate and emit RF energy by radiation or induction. Examples of intentional radiators are cordless

telephones, Remote Control Toys (RC) and other low power transmitting devices.

- Unintentional radiators: These are devices that generate and use RF energy within themselves but aren't intended to emit RF energy by radiation or conduction. Examples of these are personal computers, printers and other devices that possess internal clocks or other timing circuitry.

- Incidental radiators. These are devices that generate RF energy during the course of their operation but are not intentionally designed to generate or emit that energy.Examples include motors and mechanical light switches.

According to the rules of the FCC [1] the operator of an unlicensed RF device will need to cease operating upon notification that the device is causing harmful interference. Operation shall not resume until the harmful interference has been corrected. This constraint calls for stricter usage policies of the secondary user where a constant monitoring of the spectrum is required in order to prevent the detrimental interference caused by the secondary users on licensed primary users. There are multiple methods to have such a control structure, one of which is the usage of DSA algorithms. DSA is a spectrum allocation approach. A convenient modulation technique that can be used in such DSA communication systems is Orthogonal Frequency Division Multiplexing (OFDM), and a variant of the same, in particular, called Non-Contiguous Orthogonal Frequency Division Multiplexing (NC-OFDM) can be used to transmit a wide-band signal without interfering with narrow band primary users.

2.3 Types of DSA Techniques

There are three ways of employing DSA in a communication system. The secondary transmitter can alternately be an OFDM transmission or an NC-OFDM transmission. In either of the cases, it is important to have an efficient transmission of the secondary in order to not cause errors to the primary transmission.

- DSA using OFDM secondary Transmitter

- DSA using NC-OFDM secondary Transmitter

- DSA using IA-PFT secondary Transmitter.

2.3.1 DSA using OFDM Secondary Transceiver

The most basic technique is to transmit a secondary when there is no primary user. This would cause minimal interference to the primary user, both to the primary user and the secondary user. This technique, however, is inefficient because if the primary signals is narrow band, it would be inefficient usage of spectrum to not transmit a secondary signal.

2.3.2 DSA using NC-OFDM Secondary Transmitter

In order to solve the problem of inefficient spectrum utilization, a technique called NC-OFDM can be employed. A secondary signal can be transmitted with those subcarriers turned off corresponding to a primary transmission. In order to minimize detrimental interference to the primary user, a few adjacent subcarriers from the primary are also turned off. However, in spite of such preventive measures, spectral leakage causes harmful interference to both the primary and the secondary users.

2.3.3 DSA using IA-PFT Secondary Transmitter

This technique is a crucial part of our setup. Interference Avoidance using Partial Frequency and Time Windowing [9] is an advanced interference avoidance technique. The problem of spectral leakage is curbed by the use of time windowing and carrier cancellation schemes which result in maximum side lobe suppression.

2.4 System Model

In this section, we describe the system model used for the experimental evaluation.

2.4.1 Primary System

The primary transmitter used in our system is a GMSK modulated narrowband signal. This signal mimics the wireless microphone signal with a maximum bandwidth of

200kHz as specified by the FCC in part 74 of the Code of Federal Regulations. The primary signal is generated and transmitted over the air.

2.4.2 Secondary System Configuration

The secondary transmission configuration was derived from an LTE-Advanced standard. LTE-Advanced Standard use Orthogonal Frequency Division Multiple Access with a subcarrier spacing of 15kHz. To facilitate interference avoidance based dynamic spectrum access the bandwidth of interest is divided into resource blocks. A resource block is a specified number of OFDM sub-carriers that are allocated to a user for a predetermined amount of time. The spectral mask from the sensor is transformed into a format that specifies whether a resource block is occupied or vacant. This spectral mask is received by the secondary transmitter to decide which resource blocks can be used for transmission. To protect the primary user present in any one of the resource blocks from interference, we do not transmit in the two adjacent resource blocks as well.

NC-OFDM Transceiver

An NC-OFDM transceiver can dynamically adapt its operating frequency, bandwidth and sub-carrier allocation based on the observed RF spectrum environment in order to avoid any harmful interference to other primary/secondary users. The transmitter generates bits based on a preset pseudorandom seed to the modulator block where the bitstream is QPSK modulated and is converted to blocks of N_{FFT} parallel symbols by the serial-to-parallel conversion block. The subcarrier mapper deactivates the respective tones required for the notch and neighboring it. Further processing of the symbols is done by the OFDM modulator which performs the IFFT, parallel to serial conversion and Cyclic Prefix (CP) addition operations. The complex stream is then I/Q modulated and sent to the Digital-to-Analog Converter (DAC).

IA-PFT Transceiver

The IA-PFT technique [9] achieves higher suppression gain for a wider interference avoidance band. To achieve more suppression, we process the information symbols in

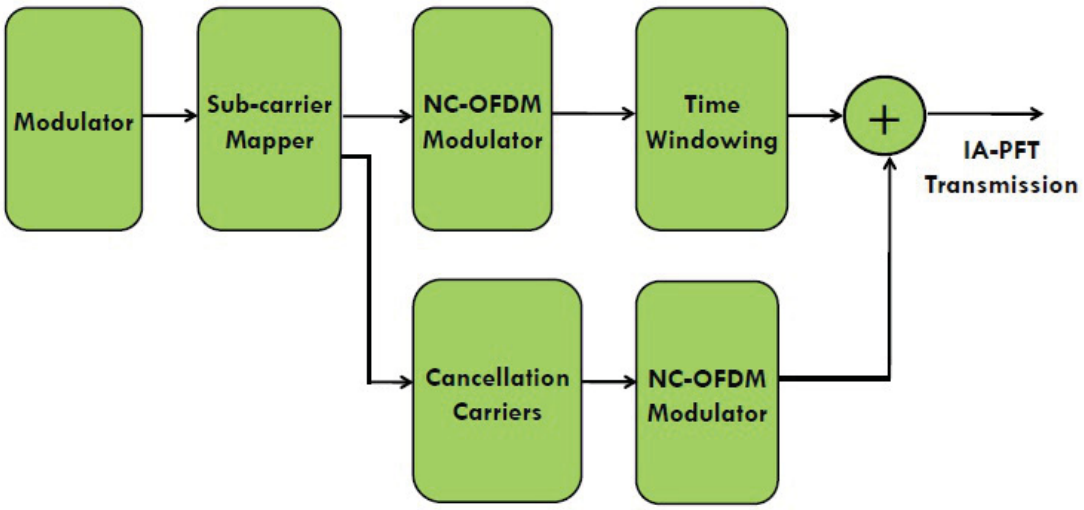


Figure 2.1: High level Block diagram of IA-PFT based NC-OFDM transmitter.

two streams, TW and CC as shown in the Fig. 2.1.

These two streams are generated by TW mapper and CC mapper blocks which know the position of the notch beforehand. Once the notch is known, the TW mapper deactivates 10 subcarriers on either side of the notch. The CC mapper generates a symbol stream with complementary sub carrier map having only 10 sub carriers on the outer edges of the notch. These two streams are then processed separately in parallel with complementary sets of data.

Time Windowing involves pulse shaping the NC-OFDM symbols with a raised cosine filter. Before shaping, an IFFT operation is performed on the modulated symbols. A Cyclic Prefix and a tail from the current symbol are added to the cyclic prefix of the next symbol as shown in Fig. 2.2 The tail of the current OFDM symbol is then discarded to generate the time window processed symbol stream as shown in Fig. 2.3

Carrier Cancelation Two CC tones are added to the extreme ends of the notch as shown in figure to cancel the interference due to the 10 active sub-carriers next to the edges of the notch. To calculate the strength of the CC tones we first generate the Inter-Carrier Interference matrix $\mathbf{W}$. The matrix $\mathbf{W}$ is precomputed using the size of

Figure 2.2: Tail of current symbol added to CP of next symbol.

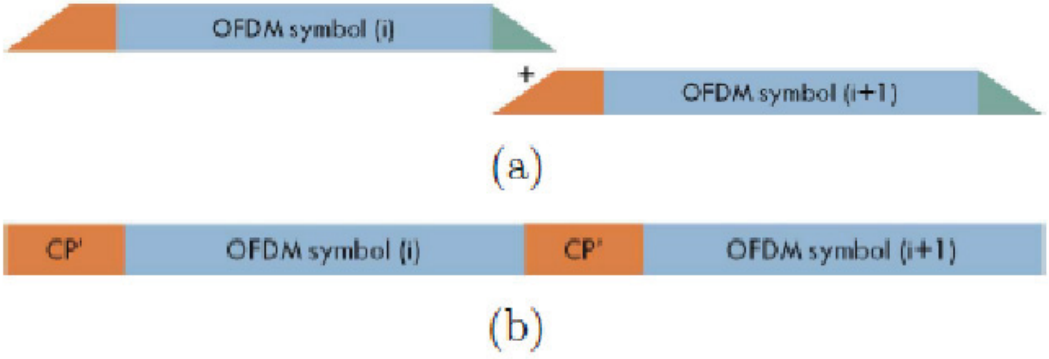


Figure 2.3: Time windowed symbol generation

the OFDM symbols (N_{FFT}). Depending on the position of the notch, a submatrix of $\mathbf{W}$ called $\tilde{\mathbf{W}}$ is selected to calculate the CC tones to be inserted. The CC tone vector h' is given by

$$\mathbf{h}' = \tilde{\mathbf{W}}'\mathbf{X} \tag{2.2}$$

where,

$\tilde{\mathbf{W}}'$ = Sub-matrix of $\mathbf{W}'$

$\mathbf{X}$ = Input symbol vector used for calculating CC tones.

An IFFT operation is performed over the CC symbols to generate the lower NC-OFDM symbols stream as shown in Fig. 2.4 In this stream, only the CC tones and the sub-carriers needed to calculate CC are active. As a result, the TW and CC streams are complementary in nature. The CC processed NC-OFDM symbols are zero padded to match the length of the TW processed symbols.

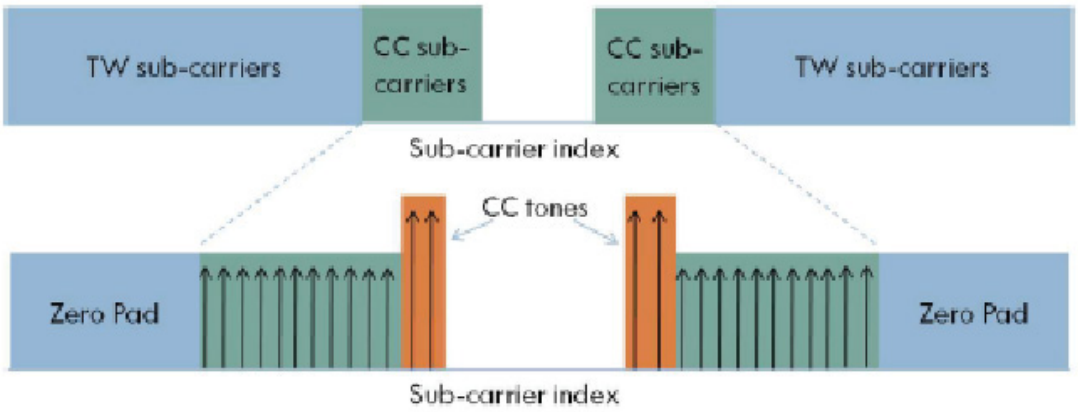


Figure 2.4: CC Stream generated.

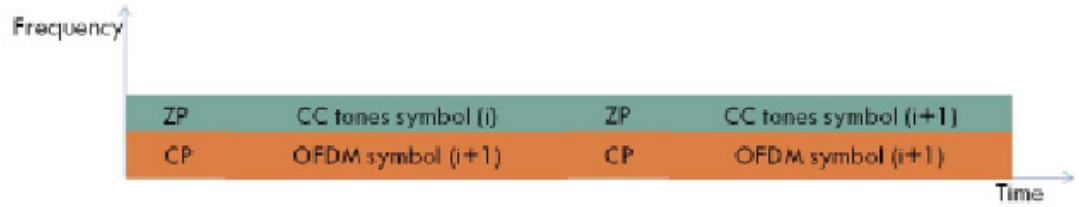


Figure 2.5: Combination of TW and CC sequences.

Combination Step The CC symbols and the TW symbols are combined by the adder, as shown in Fig. 2.5 The output of the adder is the IA-PFT processed NC-OFDM symbols, which has additional interference suppression gain compared to NC-OFDM.

Figure 2.6 summarizes the selection of tones for CC and TW carriers. The combined tones of Carrier Cancellation and Time Windowing collectively act on suppressing interference in the notch. Because of the choice of the tones to generate the CC and TW signals, they reduce interference in the edges of the notch and the center of the notch respectively.

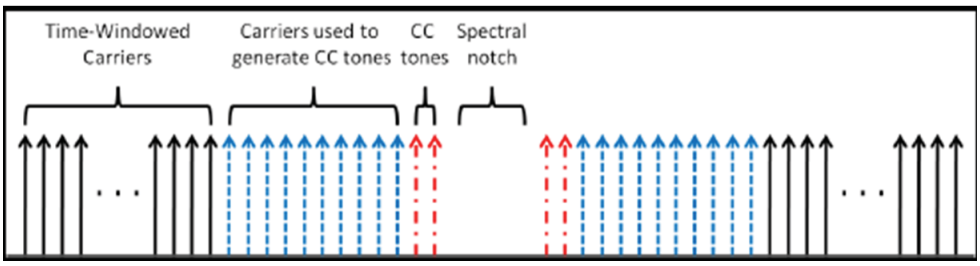


Figure 2.6: Tone selection for Interference suppression.

Chapter 3

Design and Implementation of the System

The overall system was developed on National Instruments (NI) Real Time (RT) Controller platform working on a LabView FPGA platform. Parts of the design were designed on the Real Time Controller and some performance intensive components were designed on the FPGA. In order to design an LTE based Secondary NC-OFDM Transceiver, we employed the National Instruments Real Time Controller based LabView platform. This was an alternate implementation as compared to the already existing USRP2 based GNURadio platform, upon which, some experimentation was also performed. The advantage of using a NI based LabView platform is that LabView provides a graphical programming environment, which enables easier transition from design to implementation. Using its extensive math and RF Communication and Signal Processing libraries, an entire communication system can be designed. The Real Time Embedded Controller enables a design with the constraints of a real world communication system and provides stricter control of the execution of the overall design. The following few sections describe the overall platform and the design decisions and tradeoffs made in order to accommodate an accurate implementation of a real-world communication system.

3.1 Overall Platform Description

The secondary transmitter and receiver applications were implemented in a single physical unit. The primary transmitter was implemented in a separate physical unit. The primary transmitter comprised of a Linux-based host computer and a commercially available hardware unit called the Universal Software Radio Peripheral version. 2 (USRP2) along with RF daughter boards which could operate in different frequency

bands.

The transmitter application was a part of an open source software radio package called GNURadio [10] software package installed on the host computer that communicates with the USRP2 via Gigabit Ethernet. The GNURadio software development toolkit provides the necessary signal processing blocks to implement software radios using readily available external RF Hardware such as USRP2.

The secondary transmitter and receiver applications were developed on a more complex and robust piece of hardware platform based on the National Instruments Real Time Embedded Controller FlexRIO (Reconfigurable Input Output) platform.

The NI RT Controller platform is housed in a PXIe Chassis and it hosts two Virtex-5 FPGAs called the NI FlexRIO FPGA Modules for PXIe [11]. The whole chassis houses an embedded LabView RT Controller as well, apart from the FPGAs. These FPGAs are interfaced to the analog interface to a baseband transceiver. The baseband transceiver is connected to two RF daughter boards which can function in various frequency bands via an interposer board which acts as a hardware interface from the baseband side to the RF end.

The RT Controller is connected to a host Windows based machine via Gigabit Ethernet and the host machine runs the National Instruments based LabView application. All of the design implementation is done using the LabView Graphical programming language. The RT Embedded FlexRIO controller communicates with the baseband Transceiver at a rate of 800MB/sec. The baseband transceiver communicates with the RF end via a transposer at a rate of 100 MS/sec with two channels of differential IQ. The RF end comprises of XCVR-2450 Transceivers capable of functioning in the 2.4-2.5GHz band and the 4.9-5GHz bands. The XCVRs are controlled by the drivers specified for them via the LabView interface.

Because of the presence of a Real Time Controller in the platform, the design can be implemented in such a way that it mimics real world systems, wherein, certain blocks of the system are expected to complete faster than others in order to preserve synchronization of data flowing through the system. This enables a more complex but robust design, which is guaranteed to execute in a stipulated time.

A Real Time system is defined as a system which reliably responds to an event, without fail, or to perform an operation within a guaranteed time period. In earlier implementations of this system, e.g, the USRP2 based Secondary Transceivers, this was not an option as the software platform GNURadio did not allow for such a degree of control regarding the timing of certain processes over others. A Real Time system is characterized by the following parameters:

- Loop Cycle Time: This defines the execution time of a block within a single loop.
- Jitter: The jitter in a system characterizes the variation in the execution time as opposed to the desired time.
- Time Critical Code: A piece of code that is constrained to work in a stipulated time period.
- Priority: This parameter defines the execution of a specific piece of code relative to the other blocks in the system.
- Determinism: This defines the consistency of the system to respond to external events controlling it.

The LabView software platform allows for a graphical programming environment, where various pieces of code/implementation of design are developed using a visual graphical unit called Virtual Instrument (VI). A VI basically contains a piece of code which makes use of readily available visual units for various structures.

3.1.1 Need for a Real Time Implementation

Using an Embedded RT Controller enables a complete design of a system, as opposed to a simulation or an emulation. A simulation of a transceiver system would mostly be done on a host machine using any of the available simulation tools/applications. However, using a host machine to develop a system simulation severely constrains an in-depth analysis of the system in the sense that, on a host machine, the processor time is shared between programs. High priority codes (VI) can be preempted as the

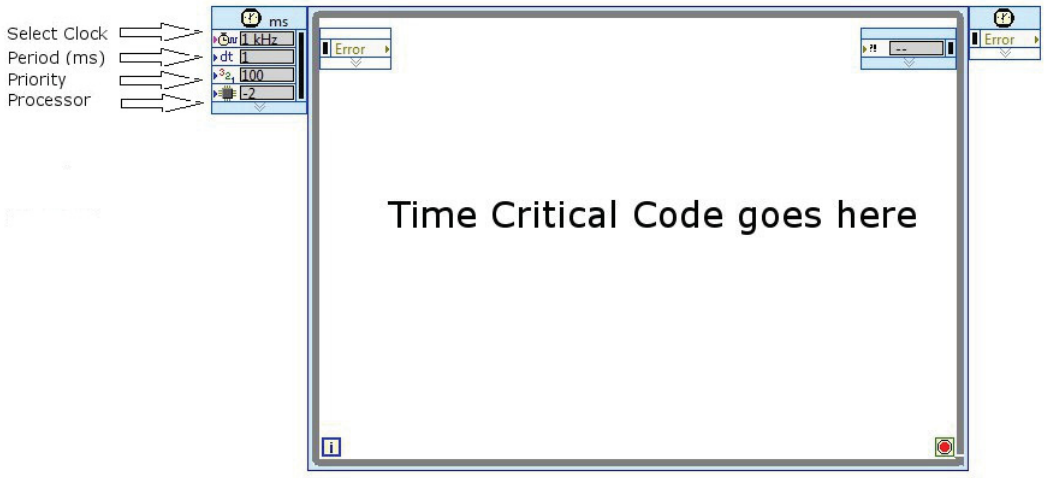


Figure 3.1: Sample Time Critical Loop.

processor needs to service keyboard and mouse interrupts, etc. A host processor cannot guarantee determinism as a result and there would be a high amount of jitter.

In comparison, a Real Time system ensures that high priority tasks are executed first and they can be designed in such a way that they don't require user input from the peripheral devices.

The Real Time module on the LabView RT Controller platform enables the usage of timed structures in the form of timed loops. These timed loops can be configured to function within a certain stipulated time and the priority of the tasks within these timed loops can be set as well. Fig. 3.1 shows a sample timed loop VI in the LabView programming environment. The clock frequency, priority and the processor can be set prior to execution. The period parameter as shown in the figure sets the constraints for the execution to be completed in the stipulated time set by the period control parameter, which in the case of this example is shown to be 1 ms.

One of the advantages provided by the NI-RT based platform is that performance intensive operations/blocks can be designed on the in-house FPGA, which ensures in-time, quick and efficient execution of those blocks that require it. The platform contains two Virtex-5 Xilinx FPGAs and the LabView software base provides a bunch of fixed point libraries for FPGA implementation as well which enables a design with the most time constrained components of the system to be designed on the FPGA for quick

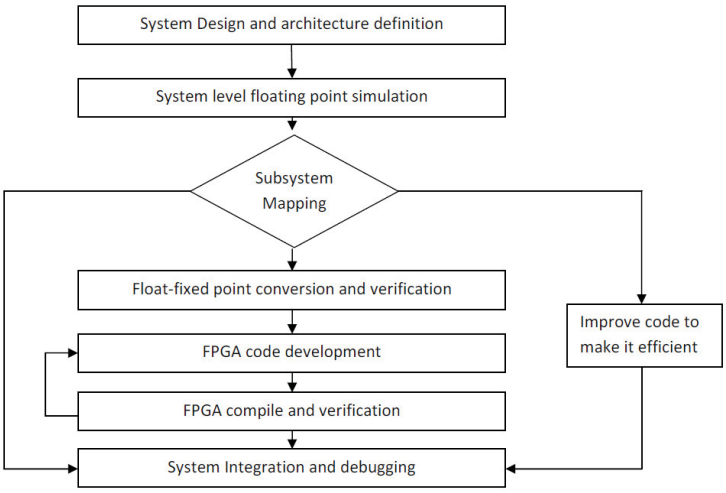


Figure 3.2: Design flow for FPGA based architecture.

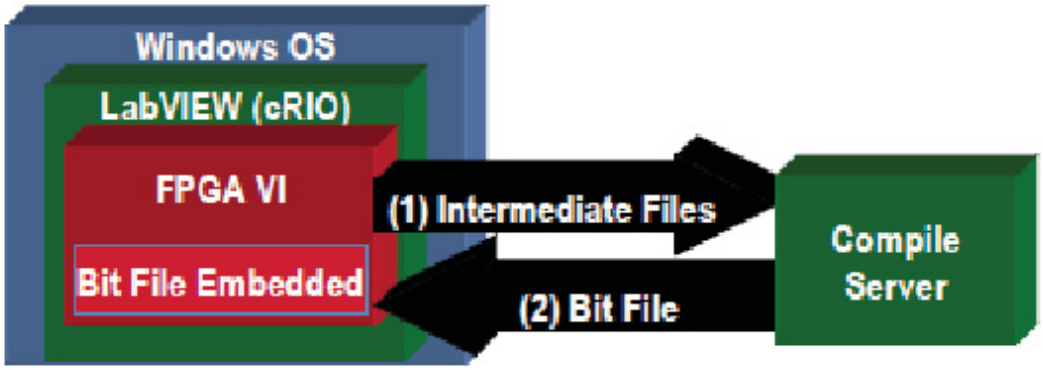


Figure 3.3: Compiling the FPGA Code (VI). (Courtesy National Instruments LabVIEW FPGA Module Tutorial

execution. The overall design flow is shown in Fig. 3.2

In order to design blocks in FPGA, the LabView FPGA Module platform is used, where LabView code is converted to hardware circuitry and then implemented on the FPGA. The LabView FPGA module comes built in with a Xilinx ISE Compiler, which turns LabView code to executable code. Fig. 3.3 shows the compilation process.

The FPGA module first compiles the VI, and this graphical code is translated to VHDL code. The Xilinx ISE Compiler compiles this VHDL code and creates the circuit. The compiler then works on optimizing the implementation. After compilation, a bitstream is generated and this bitstream is loaded on the system at run-time and

the whole system is executed with the crucial components loaded onto the FPGA for faster and efficient execution.

One of the most important design choices that needs to be made is regarding the components to be placed in the FPGA and those in the RT Controller. Depending on timing constraints of the application, certain tradeoffs need to be made in order to ensure efficient execution and intelligent design. The fundamental hitch that comes with FPGA designs is the resources to be utilized on the FPGA board. An FPGA has a limited number of slices and can handle a limited amount of memory, with respect to space constraints. Another issue with the FPGA is that because of the rapid speed in computation possible on the FPGAs, the amount of data transferred to the FPGA needs to be equally quick in order to avoid timing violations and delays due to non reception of data for computation on the FPGA.

The LabView FPGA VI and host VIs are inherently asynchronous and hence, run independent of each other. This holds true for a VI designed on the RT Controller as well. However, when data is transferred from one target to another, for instance, from a host VI to an RT VI or an RT VI to an FPGA VI, sufficient handshaking must be ensured to maintain synchronization of the VIs, failing which, data can either reach the FPGA VI too slow, which would cause the FPGA to throw a timing exception or the FPGA would perform computations on garbage data. The Data transfer synchronization needs to be implemented based on the application's needs and data buffering is required for tight synchronization and transferring large chunks of data for processing and computation. This becomes crucial when one VI writes data faster than the subsequent VI can read/process it.

3.1.2 Platform Dependent Issues and Tradeoffs

As discussed in the previous section, some strict control needs to be established when designing a system with multiple targets communicating with each other. An effective way to ensure efficient and strict on-time data transfer between targets is to use Direct Memory Access (DMA) data transfer methods. In order to transfer large amounts of data at high rates between the target and the host, as would be the case in a

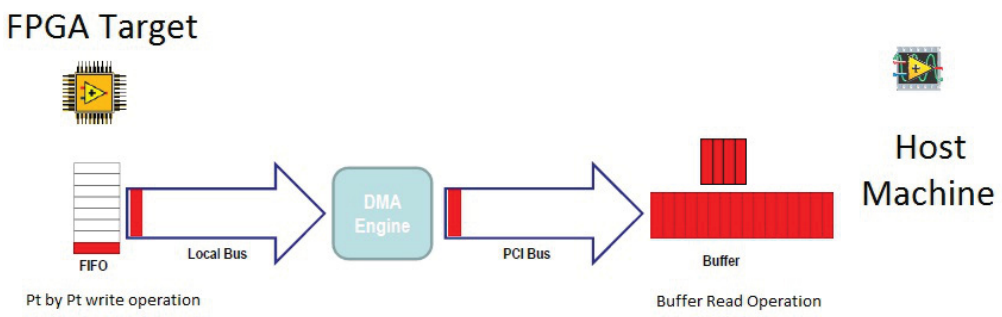


Figure 3.4: DMA data transfer method. (Courtesy National Instruments LabVIEW Real Time Module Tutorial

communication system, is to buffer the data and the best way to buffer data is to use DMA. A DMA data transfer method consists of two parts, one on the FPGA side and the other on the host/RT side. A DMA streams large amounts of data between computer memory and the FPGA. A simple block shown in Fig. 3.4 demonstrates the usage of DMA for Target to Host Transfer of data.

Designing a system on the FPGA can involve designing IP from scratch, for functionalities that are required. However, LabView's integration with Xilinx provides efficient Xilinx Coregen IP which can be integrated into the design. For instance, an FFT block can be designed from scratch on the FPGA, but, an even more robust and efficient implementation in the form of a Xilinx Coregen IP for the FFT can be configured to suit the system's needs, which would ensure efficient execution.

Depending on the platform that we used, we had to make decisions on what blocks of the design are put on the RT Controller and what blocks were to be designed on the FPGA. One of the tradeoffs made was with respect to designing blocks between the RT and FPGA, which essentially communicated with each other and hence required strict handshaking between them. In order to preserve space on the FPGA, large look up tables were avoided where necessary and these were designed on the RT, which was abundant in space. An RT DMA FIFO was designed to handle the data transfer from the RT to the FPGA.

3.1.3 Hardware Design: Space v/s Timing Tradeoffs

One of the major criteria when designing FPGA based system is the tradeoff between space and time constraints. An efficient FPGA module can counter space constraints. Efficient designing of the system where large look-up tables are avoided can help greatly in optimal and low space occupancy on the FPGA. Large look-up tables can be avoided by preallocating data arrays outside of the FPGA and using FIFOs.

The timing constraints on an FPGA stem mainly from complex math operations or, poor response to clock stimuli by the design. There are many ways of countering timing delays on an FPGA design, including:

- Using High Throughput Arithmetic blocks.
- Employing pipelining based design. Pipelining allows for performing multiple operations per clock cycle.

Pipelining results in efficient FPGA code design. It increases the throughput of an FPGA code. Any sequential block can be implemented in a pipeline based design to take advantage of the parallel processing capabilities. When using pipelining, there needs to be stricter control between sub-blocks and better handshaking protocols are required between the sub-blocks. Failure in using a robust handshaking protocol causes tremendous delays and hampers smooth execution of the block.

Fig. 3.5 refers to the pipelining functionalities between subblocks in an FPGA VI. As opposed to a single sub-VI functioning every clock cycle, there are three sub-VIs executing in every clock cycle.

Handshaking protocols are paramount for efficient execution of a pipeline based design. A simple technique called Four-Wire Protocol in LabView FPGA module allows for ease of controlling mechanism for various blocks. Usually, in an FPGA block, certain sub-blocks require multiple clock cycles to finish execution, and these subblocks do not return valid outputs at every clock cycle. To ensure numerical accuracy of computation on the FPGA, blocks must be notified about the validity of inputs and outputs.

The four-wire protocol uses four such control signals which basically notify the

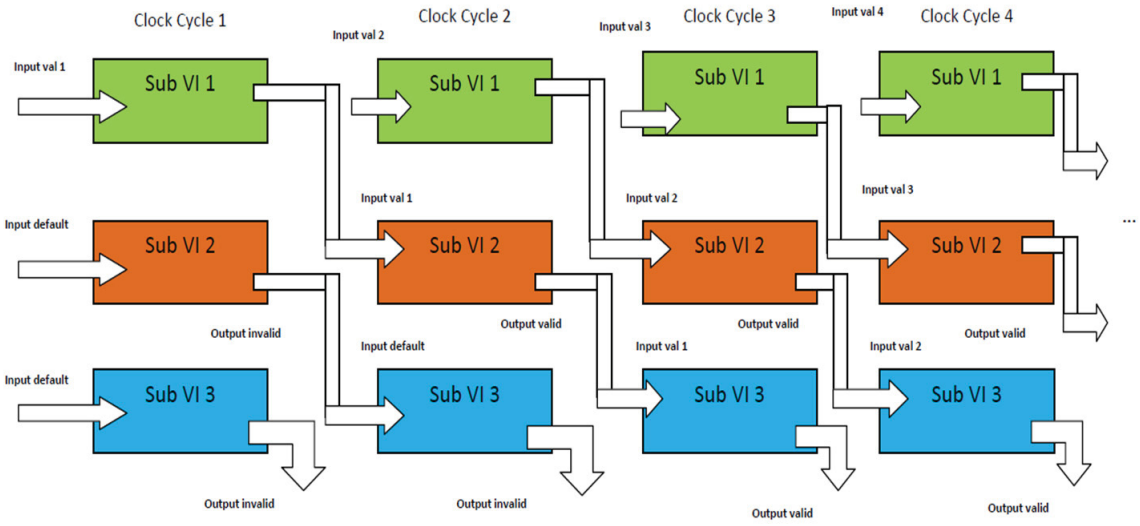


Figure 3.5: Pipelining for efficient FPGA design.

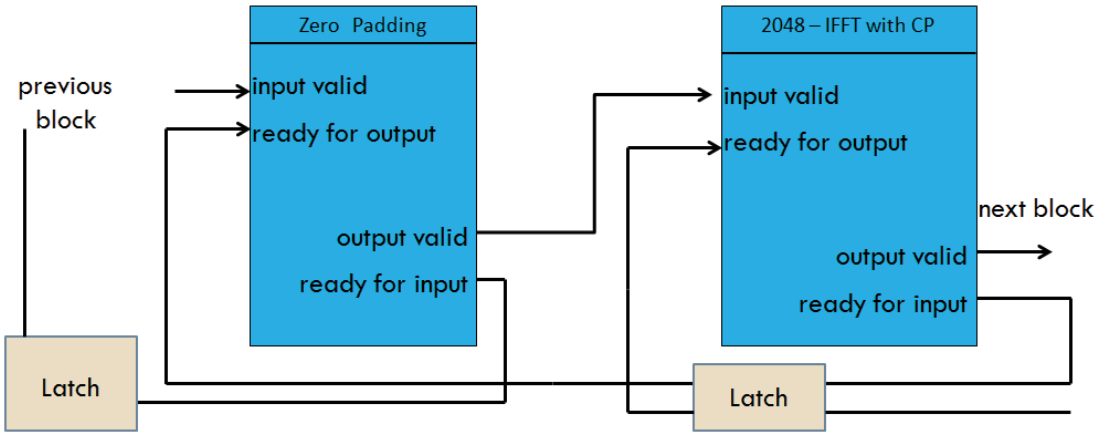


Figure 3.6: Handshaking of control signals: Four Wire Protocol.

previous or subsequent blocks about inputs or outputs being valid or invalid. Fig. 3.6 shows a sample example of the four wire protocol.

- Input valid: Specifies next data point arrival for processing.
- Output valid: Indicates current output data point is valid and can be used by future blocks.
- Ready for output: Specifies whether subsequent block can accept a new data point.

- Ready for input: Indicates whether current block can accept a new input data point at the next clock cycle.

3.2 System Design

We have designed an LTE based Secondary transceiver. A crucial component in an LTE setup is the OFDM transmitter. The LTE OFDM parameters are specified by the LTE standard and Fig. 3.7 lists the OFDM parameters as specified by the standard.

In our design, we employ the following parameters for experimentation:

- Transmission Bandwidth: 20MHz
- Subframe duration: 1 ms
- Sampling Frequency: 30.72MHz
- Number of occupied Subcarriers: 1201
- Number of OFDM symbols per subframe: 12
- Modulation: QPSK

We designed a hardware architecture of the LTE secondary system, wherein, most of the non time critical operations were performed on the RT Controller and the time critical operations were designed by combining a timed structure of the RT communicating with the FPGA, thereby ensuring in-time execution of the system.

Fig. 3.8 shows the overall block diagram of the secondary transceiver. As can be seen in the block, various blocks have been designed on various platforms as is dictated by the design. Most of the design was developed on the RT Controller, with the performance intensive FFT blocks ported to the FPGA for ease and efficiency in computation.

3.2.1 System Design Description

The System design is broken into two separate streams. One for the Transmitter end and the other for the Receiver end. Each has been described in detail here.

Tranemiesion BW (MHz)		1.4	3	5	10	15	20
Subframe duration		1.0 ms					
Subcarrier spacing		15 kHz					
Sampling frequency (MHz)		1.92	3.84	7.68	15.36	23.04	30.72
Number of occupied subcarriers		73	181	301	601	901	1201
Number of OFDM symbols per sub frame		14/12 (Normal/Extended CP)					
CP length (µs)	Normal	4.69 × 6, 5.21x1					
	Extended	16.67					

Figure 3.7: LTE OFDM parameters.

Secondary Transmitter System

The RF Communication libraries and LabView's Modulation Toolkit libraries provide an easy interface to design bit-stream generators and modulators. The data bit stream is generated using a Fibonacci pseudorandom sequence with a preset order. This block allows us to control the parameters and generate pseudorandom bits of different seeds and orders. Once the bits are generated, the data is QPSK modulated and the complex symbols are sent to the Reference/Data Multiplexer.

The Reference/Data Multiplexer is a crucial part of the design in the sense that, this block acts as the bridge between the RT Controller based code and the FPGA based code. As data is continually generated by the bitstream generator and the modulated data is continually supplied to the data/reference multiplexer, the multiplexer block inserts a reference symbol after every 5 data symbols. The reference/data multiplexer block is designed in a time critical loop on the RT Controller. This block accepts alternatively data and reference symbols and pushes them to the FPGA module, via a Real Time FIFO (RT-FIFO). There are two such FIFOs transferring two different

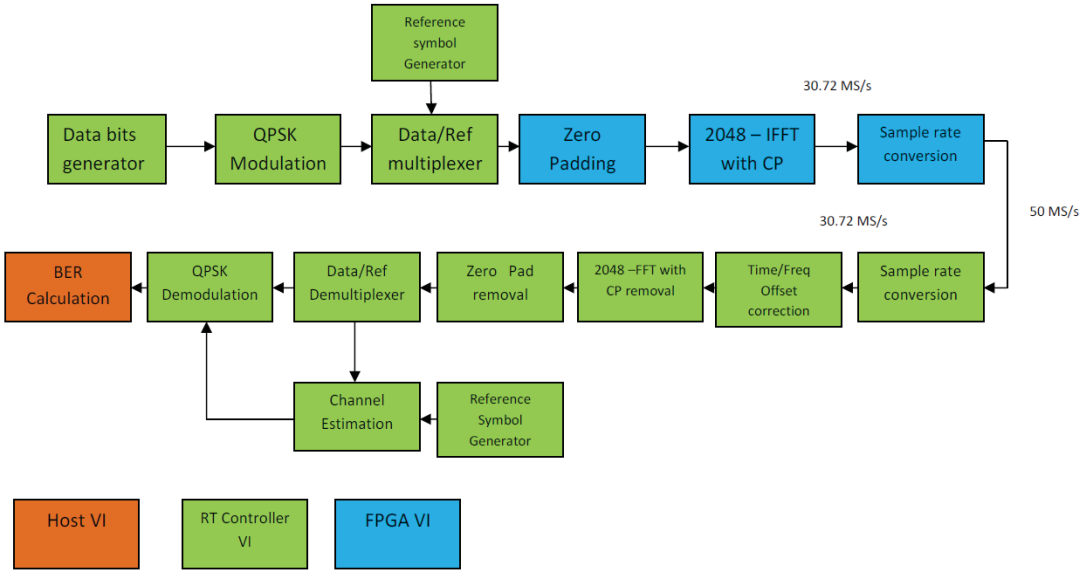


Figure 3.8: Block Diagram of Secondary LTE OFDM Transceiver.

chunks of data to the FPGA. One of them transfers the data symbols and the other FIFO transfers the Reference symbols, for further processing on the FPGA. Figure 3.9 describes a sample RT FIFO used on the actual system implementation.

Once the data is moved to the FPGA for processing, the FPGA employs a counting logic to keep track of the data and reference symbols, numbering 5 and 1 respectively and moves the data to the IFFT block, which performs a zero pad and 2048-pt IFFT. The data symbols are of length 1200 as per the LTE standard describes earlier. 1200 symbols correspond to 1000 data symbols and 200 reference symbols interspersed with one another. Zeros have to be padded to this symbol block in order to get the length to correspond to 2048 symbols, which is taken as input to the IFFT block. The IFFT block used in the design is the Xilinx Coregen IP IFFT block, which performs faster and more efficiently than an FFT block created on FPGA from scratch. After the IFFT processing, the sampling rate of the signal is the requisite 30.72 MS/s as was specified by the standard.

The FPGA code works in such a way that, once compiled, it generates a binary bitfile, which is loaded onto the Xilinx Virtex - 5 FPGAs housed in the hardware platform and this bitfile dictates the circuitry specified by the FPGA code.

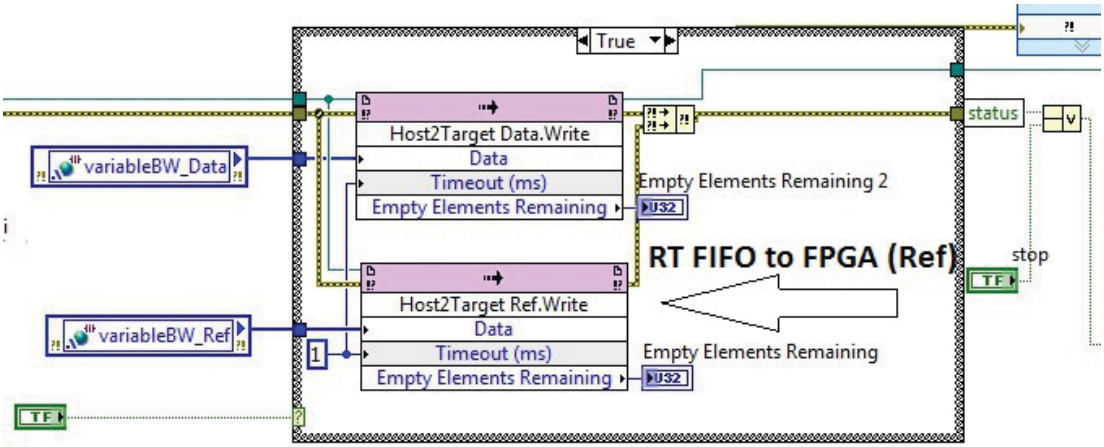


Figure 3.9: Sample VI for RT FIFO for Reference and Data symbols to the FPGA.

Table 3.1: OFDM Operation

Parameter	Value
Sampling Rate	30.72 MS/s
FFT Size	2048
CP Length	512
OFDM Symbols per subframe	12
Subframe Length	1ms

The FFT symbols now have to have a Cyclic Prefix added to them, in order for OFDM modulation. A Cyclic Prefix (CP) length of 512 was chosen to concur to the required sampling rate of 30.72 MS/s, which correspond to 2560 symbols (2048 IFFT Symbols + 512 CP). The CP operation is shown in Fig. 3.10

The RT Controller was interfaced to some peripheral devices, like the baseband transceiver and hence, the interposer and eventually the XCVR2450 RF daughter boards for the RF front end. The NI 5781 Baseband transceiver contains the ADC and the DAC, which perform the Digital to Analog and the reverse operations at baseband. The baseband transceiver works at a rate of 50 MS/s. However, the sampling rate achieved in our system was 30.72 MS/s as dictated by the standard, and hence, this sampling rate needed to be up converted, or, the samples needed a sampling frequency converter. This resampling operation was also performed on the FPGA, as the data obtained from the IFFT block could directly be sent to the resampling block, which

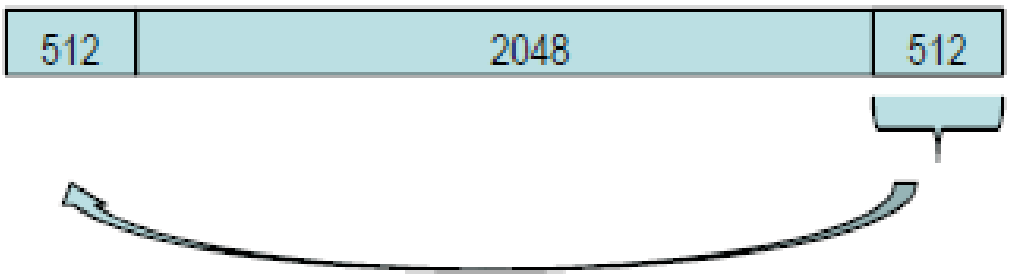


Figure 3.10: Cyclic Prefix addition to form OFDM symbol.

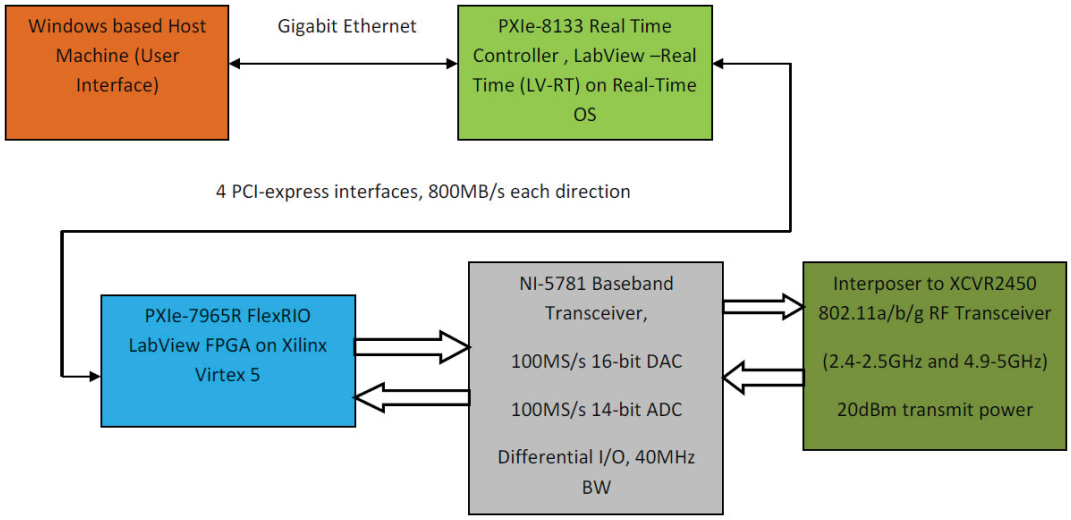


Figure 3.11: Hardware Block Diagram.

converted the sampling rate from 30.72 MS/s to 50 MS/s to be compatible to the baseband transceiver and DAC/ADC interface. This interface of the RT Controller to the hardware is shown in Fig. 3.11

The baseband transceiver transfers the data to the RF end, which is transmitted over the air via the XCVR daughter boards. The hardware design block diagram shows the complete hardware setup right till the RF transceiver stage. The time constraints placed on the design stem mostly from the link between the RT controller and the FPGA Module. The design specifies a strict timing requirement on a sampling rate of 30.72 MS/s in 1ms. Hence, a timed loop is used to control this module and ensure execution within the specified 1ms period. The RT-FPGA combined module, which

is the Ref/Data interleaver onwards has to complete execution within the 1ms stipulated period. This places a strict condition on the data transfers between the RT and the FPGA modules. Efficient handshaking mechanisms ensure the data transfer and computation occur in time and not allow for delay in data transmission.

3.2.2 FPGA Module Description for IA-PFT Based NC-OFDM Transmitter

The blocks in the FPGA are more severely constrained in time than the RT Module blocks. The FPGA, essentially works faster than the RT controller and handles higher data rates. This puts some strict control on the blocks designed on the FPGA. In our design, we decided to include the following blocks on the FPGA design of the NC-OFDM based Secondary Transmitter with IA-PFT:

- Zero Padding
- Tone selection for CC and TW
- IFFT with CP Insertion
- Time windowing scheme
- Arbitrary Resampler

The major constraint with respect to time is that, the FPGA interfaces with the DAC and hence, it has to be ensured that the DAC gets data at a rate of 50MHz or 50 Million samples in every second.

Each subblock in the FPGA design adds to the data rate and each block is progressively faster than the previous block.

Figure 3.12 shows the flow of the data in the FPGA and it can be observed that the data rates are increasing along the path of the FPGA. The units used for the data rate are Complex Samples/millisecond, where the millisecond denotes the subframe duration.

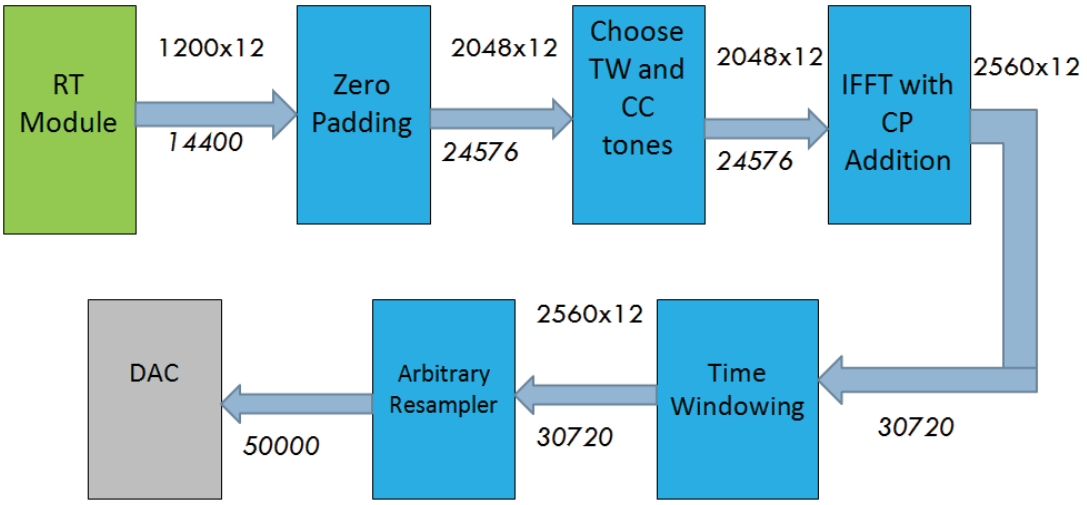


Figure 3.12: Data flow through the FPGA, Units in Complex samples/millisecond.

Secondary Receiver System

The Receiver application was completely designed on the Real Time Controller, for ease of implementation of design. The samples are acquired at the RF end by the XCVR daughter board operating on the Receive mode, and transfers received I/Q samples to the ADC on the NI-5781 Baseband Transceiver, which converts this RF signal into baseband spectrum. The Receiver application is then applied to this baseband equivalent of the received signal. The algorithms employed at the receiver include:

- For the Time/Frequency Offset estimation, we use the Maximum likelihood method described by [12]
- For Channel estimation, we use the Least Squares estimation technique.
- A 1-tap frequency domain equalizer performs the channel equalization.
- The detection/demodulation in the receiver application employs a maximum likelihood technique.

Upon receiving the received samples, the CP is stripped off and the FFT block performs a 2048 pt FFT. The reference symbols are generated again on the receiver end for synchronization purposes and these reference symbols are used to decode the symbols and finally a QPSK demodulation yields hard decoded bits, where we compute

the sign of each complex sample and make a decision to be either 0 or 1, depending on a negative or positive sign of the complex samples, respectively.

Secondary NCOFDM and IAPFT Transmitter

In order to employ this system to perform DSA in the presence of primary systems, the OFDM transmitter in the original secondary transmitter had to be altered to a Non Contiguous OFDM Transmission system, where, depending on the position of an interfering primary system, the secondary would null out those corresponding sub carriers. Nulling subcarriers is as simple is zeroing out the respective locations in the frequency domain. In order to develop an IA-PFT based secondary transmitter system, Carrier Cancellation and Time Windowing need to be performed on the NC-OFDM signal. The Carrier Cancellation step is applied to symbols, i.e, in the frequency domain, and hence, it can be performed right before the data is sent to the FPGA for the IFFT operation. The precalculated **W** is used to compute the carrier tones and these carrier tones are just replaced at the edges of the notch. The Time Windowing operation, however works on a parallel stream of data, which is in the time domain, and hence, the pulse shaping and addition of the tail bits of the current symbol to the CP of the next, are performed in the FPGA and the subsequent signals are both processed by IFFT blocks and combined together to generate a single stream of samples. Due to the presence of two parallel streams of data, working with different interference avoidance techniques each, the requirement was now increased to 2 IFFT blocks on the FPGA, as opposed to the single IFFT block in the case of a normal OFDM transmitter.

3.3 Advantages of Using a RT/FPGA Platform for Design and Implementation

A communication system can be designed and its working can be simulated over various simulation platforms. However, using a real-time hardware setup to design a transceiver allows for a real-world implementation of a communication system and regular over-the-air experimentation is enabled, which leads to a fairer evaluation of the system, in

terms of characterization of the receiver. The National Instruments setup provides such a platform for design of a communication system mimicking a real-world system. The advantages of using such a real-time/hardware platform could be enlisted as:

- Using the FPGAs on board results in faster and efficient execution of design.
- Timing constraints can be met in the communication system due to the use of timed loops in the RT Controller.
- An open design of the system allows for rapid debugging of issues because data can be tapped out of any stage and viewed at, to discern errors/synchronization issues.
- Simple arithmetic operations like additions and logic blocks in the design can be implemented in the FPGA for quicker execution

The biggest advantage of using a Real-Time based design over earlier implementations like the USRP2 based GNURadio implementation, is that timing delays can be avoided and due to the strict constraints placed on the execution time, it is ensured smooth and in-time execution. Also, as opposed to the GNURadio implementation, debugging on the LabView RT platform is easier because of the ability of the LabView based design to be open for testing and debugging at any stage and samples can be tapped out at any stage, starting from the bit generation stage to the time domain samples following the IFFT operation for OFDM.

Chapter 4

Experimentation and Results

4.1 Experimental Evaluation of DSA Using USRP2 based Implementation

Experimental evaluation of a DSA scheme was performed on a GNURadio based USRP2 setup in [13], [14]. On a USRP2 based setup, the primary system was a wireless microphone signal and the secondary system was based on a NC-OFDM transmitter designed on GNURadio based implementation and the primary receiver's performance was evaluated. The experimental setup was implemented at Orbit [15], Winlab, Rutgers University. The experimental setup was placed as shown in Fig. 4.1.

The NC-OFDM and IA-PFT based secondary transmitters were both developed using GNURadio.The spectrograms for the coexistence of secondary systems in the presence of the primary are shown in Fig. 4.2

The NC-OFDM and IAPFT transmitters were evaluated and the suppression of the spectrum within the notch was observed. With the usage of IA-PFT technique, a suppression of nearly 10-12dB was observed in the notch. Fig. 4.3 shows the interference suppression within the notch.

The primary receiver was evaluated by performing a series of experiments at Winlab, Rutgers University. The secondary NC-OFDM transmitter power was kept constant and the primary transmitter's power was varied to obtain the primary receiver's error rate against the primary transmitter with a constant secondary transmit amplitude. The same experiment was performed with the IA-PFT based secondary transmitter as well. Fig. 4.4 shows the primary receiver's evaluation in a coexisting primary and secondary system setup.

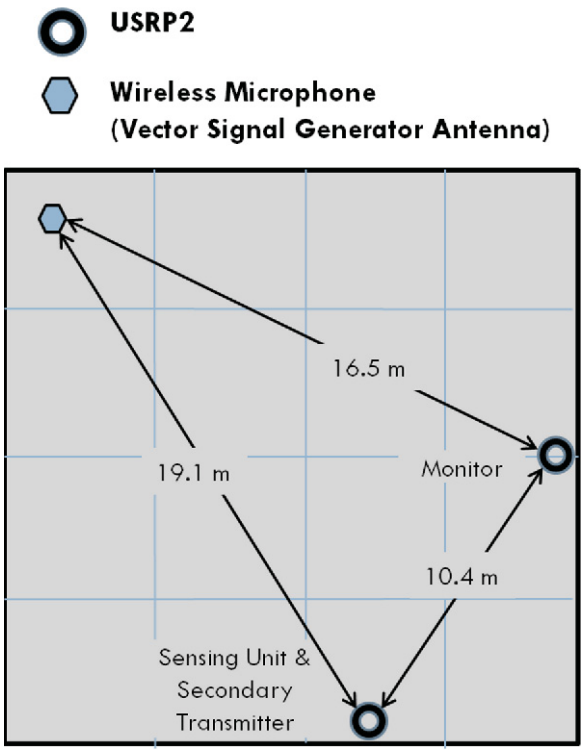


Figure 4.1: Experimental Setup using USRP2s.

4.2 Experimental Evaluation of DSA Using the NI Labview Real Time Controller Setup

The secondary receiver was evaluated on an alternate platform, by developing the NC-OFDM/IA-PFT based transceivers on the National Instruments Real Time Embedded controller platform. The primary system was implemented on the USRP2, which mimicked a wireless microphone signal of 180kHz bandwidth. The Secondary receiver was built for both configurations of NC-OFDM alone and IA-PFT based NC-OFDM. The secondary transceiver was designed in such a way to enable easier visual recognition, by plotting the power spectrum and the constellation of the signal, which enabled easier debugging of the system, in case of transmission errors.

The secondary NC-OFDM transceiver was employed in a DSA setup and the experiments were performed with the usage of IA-PFT offline. The IA-PFT algorithm resulted in significant suppression of spectral leakage within the notch. Fig. 4.5 shows the suppression within the notch. It can be observed that there was a suppression of

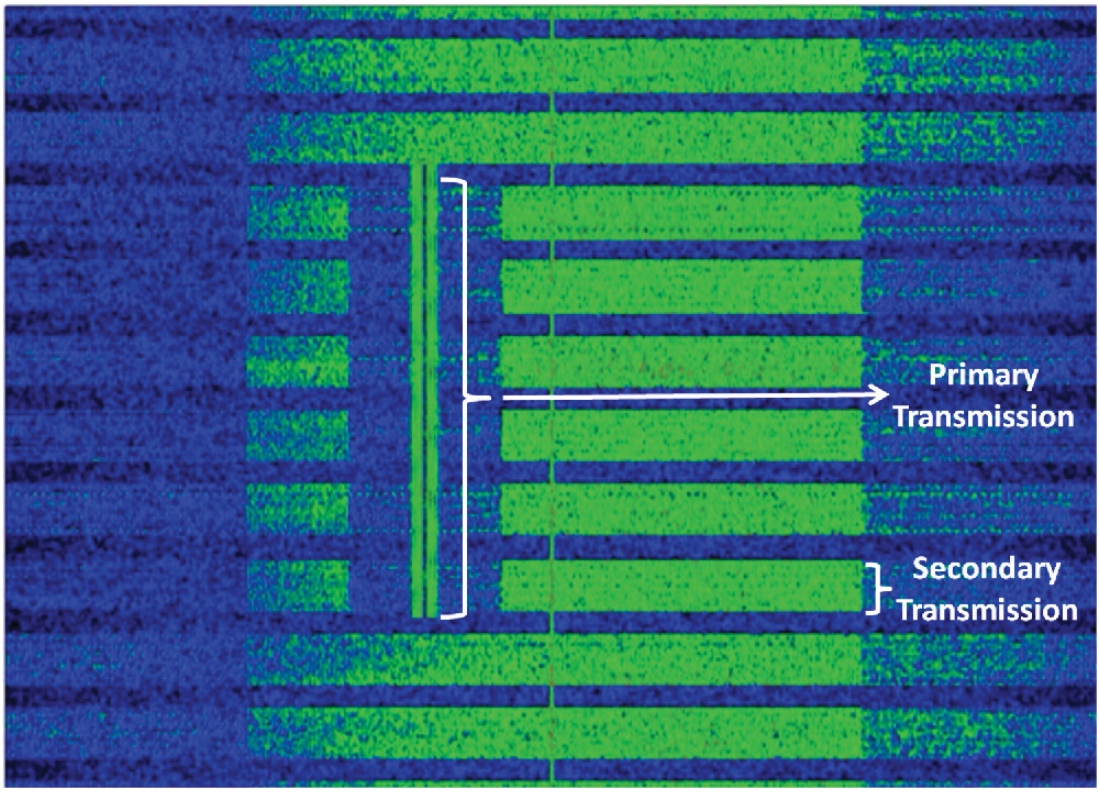


Figure 4.2: Spectrogram of secondary NC-OFDM signal in the presence of the primary.

about 10dB even in the case of the National Instruments based Real Time implementation of the secondary LTE based NC-OFDM transmitter module.

The programming platform allows for displaying power spectra and constellation diagrams of the subsystems as required. By placement of PSD and Constellation at the secondary receiver, we can view the response of the secondary receiver in the channel. Multiple experiments were conducted by keeping a constant secondary transmit power and varying the primary transmit power within the notch. Initially, we started with the absence of the primary transmitter.

Fig. **??** shows the placement of the notch on the power spectrum, in the absence of the primary signal. The plot was a screen capture from the LabView programming environment's front panel, where we display the power spectrum and the constellation for the secondary transceiver system.

The primary transmitter was then turned on and the spectrum and constellation was viewed at varied primary transmitter amplitudes. In the presence of the primary,

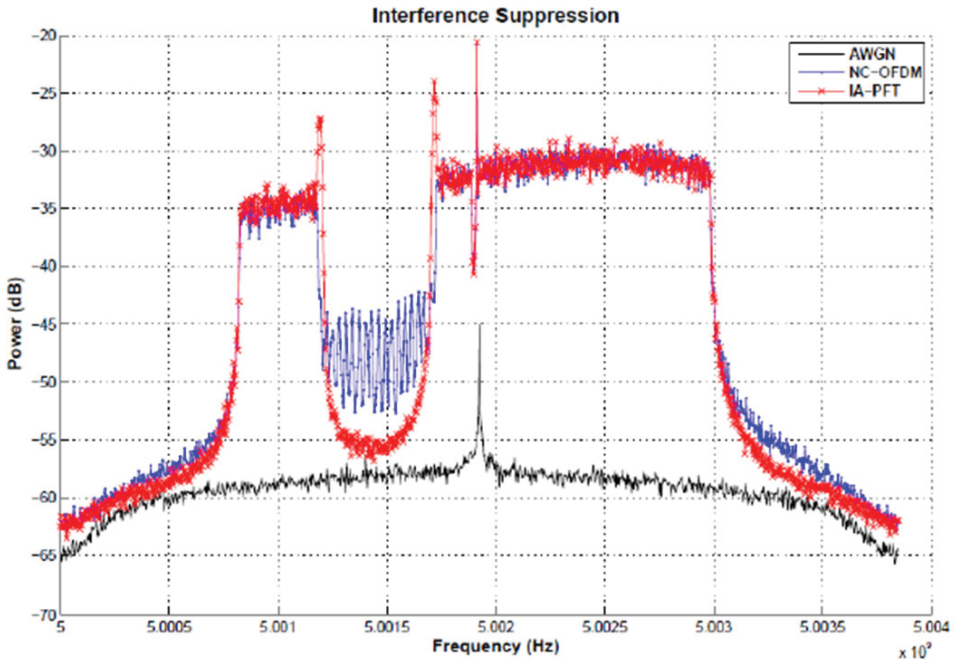


Figure 4.3: Interference Suppression in the notch due to IA-PFT based NC-OFDM Secondary Transmitter.

we can notice the power leaked into the notch raise because of poor interference suppression capabilities of the normal NC-OFDM secondary transmitter system. Fig. 4.5(c) shows the power spectrum of the secondary transmitter in the presence of a primary transmission within the notch.

The constellation for the secondary receiver system for an over-the-air transmission without the presence of the primary in the notch can be seen in Fig. 4.5(d)

The constellation for the secondary transmission, in the presence of the primary is shown in Fig. **??**

With increase in the transmit power of the primary transmitter, through Fig. 4.5(e) and Fig.4.5(f), we can see that the secondary receiver module fails to discern the signal because the spectrum because of the high primary transmit power and this causes errors in demodulation at the secondary receiver and the information regarding the primary transmitter as procured by the secondary transmitter is incorrect.

The secondary transceiver was used to perform an error analysis of the secondary system in the presence of an interfering primary signal.

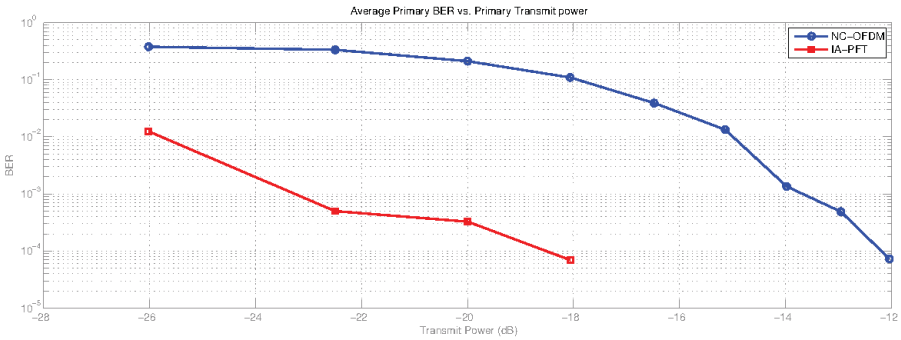
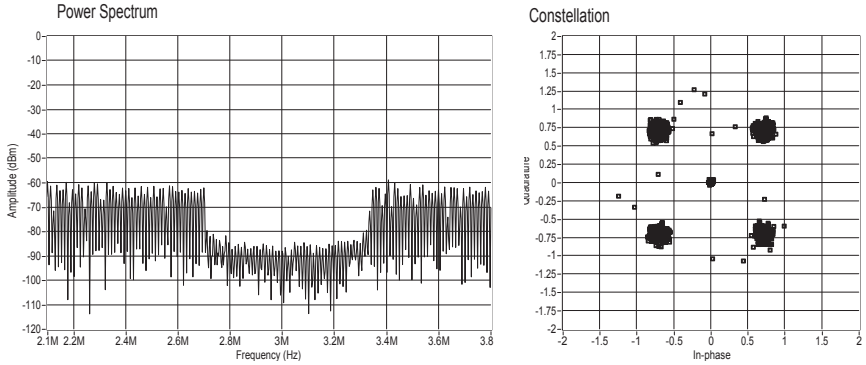


Figure 4.4: Primary Receiver BER v/s Primary Transmit Power.



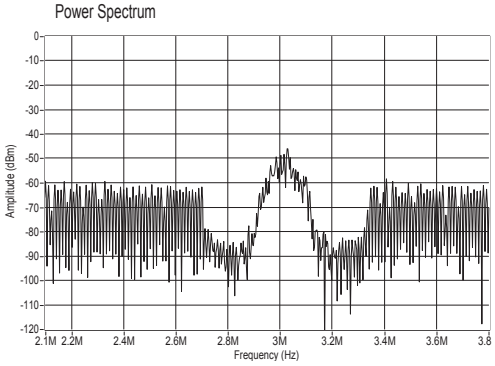
(a) Power Spectrum of the Secondary NC-OFDM signal without primary transmitter in the notch.

(b) Secondary NC-OFDM transmitter's constellation in the absence of the primary.

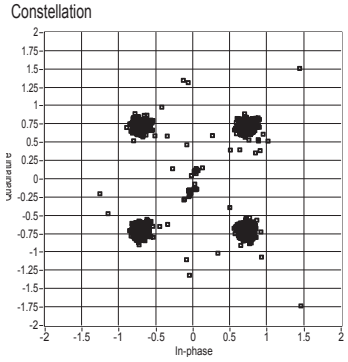
The secondary NC-OFDM Transceiver with IA-PFT was used in DSA experimentation and a suppression of nearly 10dB was observed in the notch. Fig. 4.5 shows that, as compared to normal NC-OFDM, the added IA-PFT technique performs well in suppressing leakage into the notch.

The secondary receiver was characterized by running an error analysis of the secondary receiver with varying primary transmit power within the notch and keeping a constant secondary transmit power. This way, the performance of the secondary receiver can be evaluated and an inference can be made on the effect that a primary transmitter has on a secondary NC-OFDM link. The BER of the secondary receiver was plotted against the varying primary transmit amplitude/power and Fig. 4.6 shows the BER for the secondary NC-OFDM receiver.

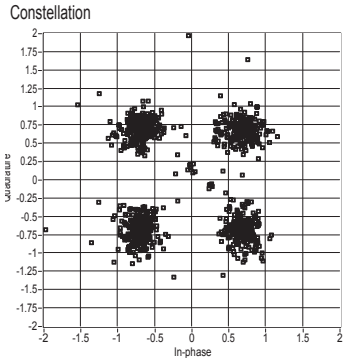
Fig. 4.6 shows that the performance of the secondary receiver degrades with increase



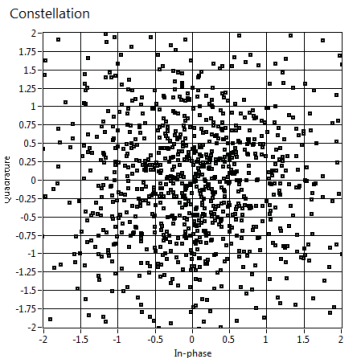
(c) Power Spectrum of the NC-OFDM secondary transmitter in the presence of a licensed primary user within the notch.



(d) Secondary Receiver Constellation in the presence of a primary user.



(e) Secondary Receiver Constellation in the presence of a primary transmit amplitude of 0.4.



(f) Secondary Receiver Constellation in the presence of a primary transmit amplitude of 0.6.

in primary transmit power. This evaluates the effect of the primary transmitter on the secondary NC-OFDM transceiver. With a high primary transmit power, the secondary receiver finds it increasingly difficult to discern the transmission on the system and this leads to demodulation and detection errors, which cause the communication system to fail. With use of an interference suppression in the notch, the error performance can be improved and the operating range for the secondary with respect to the primary can be increased.

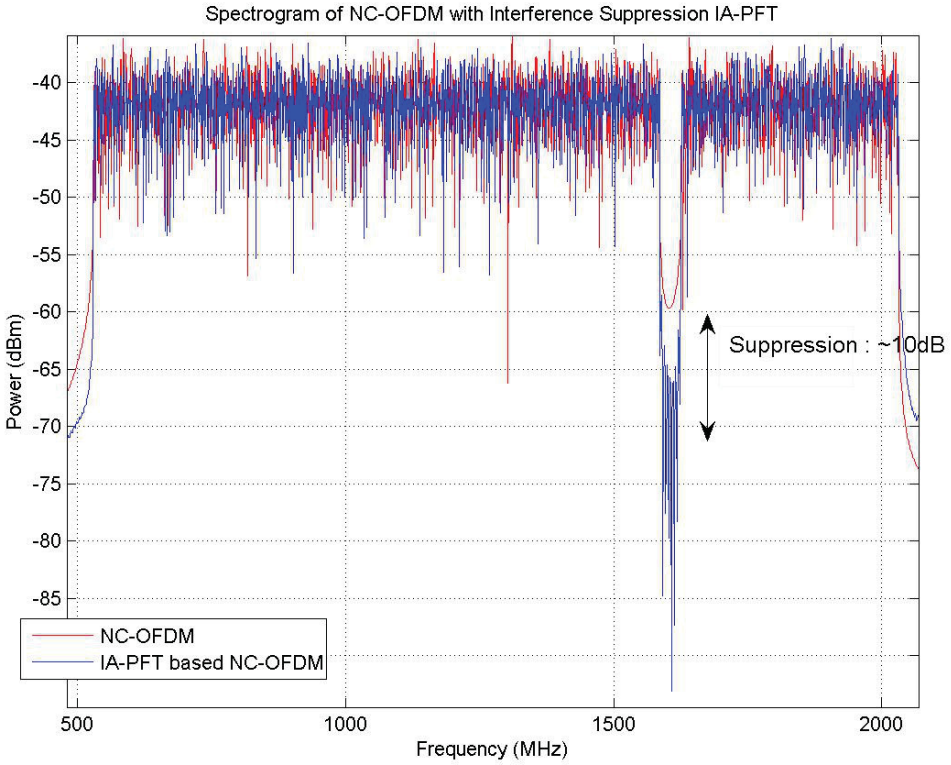


Figure 4.5: Interference Suppression in the notch using IA-PFT technique.

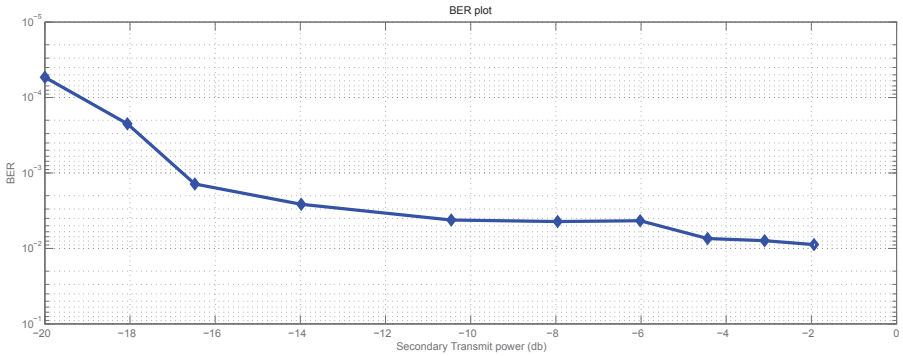


Figure 4.6: BER of the secondary NC-OFDM receiver against the primary transmit amplitude

Chapter 5

Conclusion and Future Direction

5.1 Conclusion

This thesis focused on a hardware FPGA platform based LTE transceiver design and experimental evaluations were performed on the same hardware setup. A hardware setup enabled greater control of the timing considerations of a real-world transceiver system and allowed for stringent performance criterion while maintaining efficient communication. This implementation contrasted earlier implementation of the transceiver on a software platform and resulted in better hardware accelerated performance with performance intensive blocks routed on circuitry on FPGAs present on the hardware setup. The timing considerations were kept in mind, during the design and implementation stages and a more efficient design of the transceiver was made. The secondary transceiver was used in a Dynamic Spectrum Access experimental setup and experimental evaluation of the secondary transmitter was made in the presence of an interfering primary. DSA was demonstrated by measuring the error rate of the secondary receiver in the presence of a varying amplitude primary transmitter in the notched band, where the secondary transmitter does not transmit.

The Hardware based design serves as a benchmark platform for an LTE Based NC-OFDM Transceiver module and experimentation on this platform can be made for various cases. One such case covered in this thesis, is the DSA experimental setup. With the use of advanced interference suppression schemes such as IA-PFT, the experimental results show a clear suppression of interference within the notch of about 10-12dB. Also, the BER analysis of both the primary and secondary receivers characterize each respectively and give more insight into the effect of the primary transmitter on the secondary receiver and vice versa.

5.2 Future work

The secondary transceiver can be used as a benchmark design as it very closely performs as an LTE transceiver and this design can be used as a reference to design complicated other complicated LTE standard based transmission schemes. Better FPGA designs can be made with the FPGA blocks already designed in this system and the system throughput can be increased.

References

[1] W. McCoy, "Overview of the 3GPP Long Term Evolution Physical Layer."

[2] "Universal Software Radio Peripheral," http://www.ettus.com.

[3] "National Instruments LabVIEW Real Time Controller Platform," http://www.ni.com/labview/realtime/.

[4] S. Pinagapany, S. Sequeira, Y. Futatsugi, M. Ariyoshi, and P. Spasojevic, "A Demonstration of Adaptive Spectrum Sensing and Interference Suppressed Opportunistic Secondary Transmission," in *CROWNCOM*, 2011.

[5] S. Pinagapany, S. Sequeira, A. Bindiganavile, Y. Futatsugi, M. Ariyoshi, and P. Spasojevic, "Experimental Evaluation of Interference Avoidance Opportunistic Secondary Transmission for Coexistence with Primary Systems," in *SDR'11 WinnComm*, 2011.

[6] J. Mitola III, "An integrated agent architecture for software defined radio," Ph.D. dissertation, Royal Institute of Technology (KTH), Sweden, 2000.

[7] ——, "The software radio concept," pp. 26–38, 1995.

[8] E. Buracchini, "The software radio concept," pp. 138–143, 2000.

[9] Y. Futatsugi and M. Ariyoshi, "A Study on Interference Suppression Technique by Partitioned Frequency- and Time-domain Processing for Dynamic Spectrum Access," in *IEICE Tech. Rep., vol. 110, no. 41, SR2010-9, pp. 49-55*, 2010.

[10] "GNU Radio," http://www.gnuradio.org.

[11] "NI PXIe RT Embedded Controller Setup," http://sine.ni.com/nips/cds/view/p/lang/en/nid/208834.

[12] J.-J. van de Beek, M. Sandell, and P. O. Brjesson, "Ml estimation of timing and frequency offset in multicarrier systems," 1996.

[13] S. Sequiera, "Energy based Spectrum Sensing for enabling Dynamic Spectrum Access in Cognitive Radios," Master's thesis, Rutgers, The State University of New Jersey, 2011.

[14] S. Pinagapany, "Decentralized Spectrum Allocation Schemes for Cognitive Radio Networking," Master's thesis, Rutgers, The State University of New Jersey, 2011.

[15] "Orbit, Winlab," http://www.orbit-lab.org.