Thesis
Responses to Reviewers' Comments

September 7, 2015

Reviewer 1

> *There are some minor typo errors and some suggestions to improve some phrasings, indicated in various parts of the report, for the candidate to revise his thesis.*

Thank you for reading this thesis so carefully and providing these constructive review comments, which we have accepted and amended upon in full, as described below.

> *1. In chapter 2, to include a constellation diagram of an ideal OFDM received signals, such that it can be used to comparatively illustrate the effects of STO and CFO shown in Figure 2.7, 2.9, 2.10 and 2.11.*

The mentioned Figures are amended to include the constellation diagram of an ideal OFDM received signals.

> *2. In chapter 4, how are the frame detection threshold values for various channels derived/ determined?*

The threshold value depends on operating channel and needs to be determined empirically by simulation (in common with other systems such as [36]). Assuming the set of various channels are known, the threshold for these channels can be pre-determined by simulation. The performance of synchronisation is evaluated with increased threshold within a given channel model. The threshold value corresponding to the best performance is selected for the channel model. The determined values of threshold are stored in a look-up table. Based upon the current channel model and conditions, the corresponding threshold has been selected from the look-up table. The discussion on determining the frame detection threshold is amended in the subsection 4.3.1.

> *3. In chapter 6, figures comparing the different spectrums (e.g. 6.2, 6.3, 6.4) should be presented in different colors for clarity in comparing their characteristics.*

These mentioned Figures in chapter 6 are improved with using different colors.

4. A page listing the various notations used in the thesis should be included for easy reference when reading the thesis.

The list of notations is already presented in the amended thesis at page xiv.

5. One issue that the candidate should further address in the thesis is to present results that demonstrate the practical feasibility of, as well as the real-time performance achievable by the proposed techniques when implanted on actual FPGAs. While there is no reason to doubt that functionality of the proposed techniques, it would be more convincing by developing a prototype on a FPGA device to demonstrate the correctness and real-time performance. In addition, this will also help to cross-check and compare against the results obtained through simulations, as well as provide an indication of the real-time performance achievable, which is importance for CR application. This investigation should at least be performed for the proposed receiver, and can be done based on Figure 7.1, where an 'transmitter' is synthesized to emit baseband digital data (with effects simulating the various conditions encountered under practical environment such as multipath, CFO, STO effects) that loopback to the receiver that contains the various modules. The required FPGA development resources are readily available in the research centre in NTU/SCE, and should be relatively easy to be done as the candidate had already used the relevant tools to analyse and validate individually the various proposed modules.

In chapter 7, we analyse the ability of a proposed architecture applied the proposed method for a multiple standard application with regard to the adaptive time when system switches from one standard to another. As can be seen in Fig. 7.1, building up a prototype for demonstration requires a lot of hard engineering works. Developing the prototype of CRs for practical real-time measurements requires many related researches beyond the scope of thesis such as CR engine, RF Front-end, and MCMM. The works for a prototype can be listed for future direction. The main contributions are to propose the novel techniques to overcome the issue of OFDM for multiple standard CR application such as synchronisation and spectral leakage. The proposed methods are implemented by using Verilog HDL. The verification for the implementation is intensively done by the test-benches in which the output and intermediate values are compared to that of Matlab simulation to evaluate the correctness. Further works for system verification framework are described in a new sub-section 7.5 in Chapter 7. The sub-section presents comprehensive system verification with test-benches performed on Xilinx tools and the performance of golden model simulated on MATLAB.

Reviewer 2

While this work relates to cognitive radio applications, the decision making aspects of cognitive radio are not addressed in this dissertation, which is fine. Perhaps this should be made clear in the introduction.

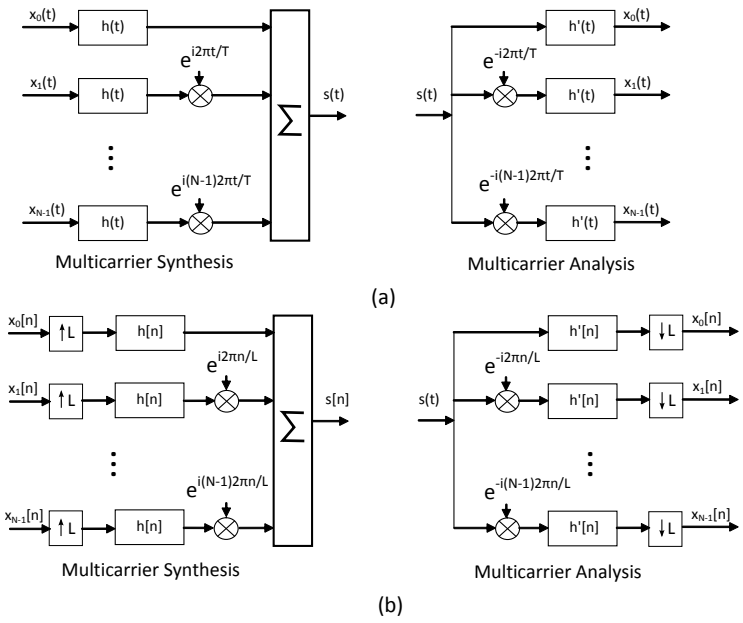

Figure 1: Block diagram of a multicarrier modulated system, (a) in the countinuous-time and (b) in discrete-time

First of all, thank you for your review. I have amended the text exactly as requested in the comments. This point is clearly revised in the introduction.

> *1. In Section 2.1, Mr. Pham states that OFDM system implementation is simple, low cost, and can be more effectively parameterized than FBMC systems. Further justification of this assertion is needed. It may be helpful to provide a more comprehensive comparison between OFDM and FBMC techniques.*

This point was amended in the subsection 2.1.1 to provide more comprehensive comparison between OFDM and FBMC described as follows: In order to understand FBMC and its distinctness from OFDM, it is best to study multicarrier systems, in which the output signal can be expressed in the continuous time domain as Equ.1. This can be a unified formulation for both OFDM and FBMC.

$$s(t) = \sum_{n} \sum_{k=0}^{N-1} x_k[n] h(t - nT) e^{i2\pi(t-nT)f_k}, \tag{1}$$

where $x_k[n]$ denotes the sample of the k_{th} subcarrier data symbol in n_{th} symbol of continuous multicarrier symbols, f_k is the k_{th} subcarrier in a set of N used subcarriers, T is the multicarrier symbol duration, and $h(t)$ is a prototype filter. This transceiver of multicarrier system can be modelled as a block diagram, shown in Fig. 1. As can be seen in the discrete-time domain, N data symbols at synthesis are up sampled by a

factor of L, which is calculated by $\frac{T}{T_S}$, T_S denotes sample period of output sequence $s[n]$, and then filtered by a prototype filter $h[n]$. The output of each data stream will be modulated by frequency of multi-carriers and the summed for transmission. The signal in the receiver is demodulated and then filtered by a bank of matched filters $h'[n]$, and down sampled by a factor of L. When critical sampling applies $L = N$ and the prototype filter $h[n]$ is selected as a rectangular pulse in the time domain, i.e. a $sinc$ pulse in the frequency, this multicarrier system becomes a conventional OFDM system.

FBMC is different from OFDM in the selection of the prototype filters $h[n]$, and matched filters $h'[n]$. Also, the $h[n]$ and $h'[n]$ filters are chosen and designed depending on the adopted FBMC modulation technique. By using the well-designed filters for each subcarrier, FBMC could be a more effective solution in comparison to OFDM in term of ICI cancelation and spectral leakage suppression because nonadjacent subcarriers are almost completely separated by a bank of filters.

On the other hand, OFDM has been the dominant technique adopted for broadband multicarrier communication. OFDM has many important and desirable features over the FBMC. OFDM was originally developed focusing on a low-complexity implementation. The low complexity of OFDM is achieved thanks to a fundamental assumption in which subcarriers of OFDM symbol are perfectly synchronized and orthogonal with the consecutive subcarriers. Thus the subcarriers are used for modulation at the transmitter using an IFFT block; inversely, they are separated by using an FFT block at the receiver. By contrast, FBMC may be more complex than OFDM. The demand for well-designed filters in FBMC results in increasing complexity and resource requirements. Moreover, while employing MIMO technique in OFDM to increase the system's capacity and resulting in high spectral efficiency is a straightforward work, unfortunately, the development of MIMO-FBMC systems is relatively more complex. OFDM modulation has been the dominant technique adopted for many wireless standards and has been investigated in terms of spectral sensing and carrier allocation for CRs. OFDM system implementation is simple, low cost, and can be more effectively parameterised. A single baseband implementation can be made to flexibly support multiple standards like 802.11[14], 802.16[15], and 802.22[16], as well as supporting future OFDM-based standards.

> *2. Chapter 3 presents the multiplierless correlator work. There are aspects of this chapter that may need further examination. I appreciate the significance of the problem this chapter is addressing, and the value in the demonstrated savings; however, the methods for performing the comparisons dont always seem fair. One note in resource usage: a DSP48 processing module is rated to operate at hundreds of MHz. If the desired operation speed is N times slower than the maximum operating speed, then a single DSP48 can cover N instances of the operation through time multiplexing. Therefore, when performing a resource estimate, the required number of units should be derated by the operational speed, otherwise it represents an underutilized system. In this case, the target operating frequency is 50 MHz while capacity of the DSP48s are over 300 MHz.*

The additional discussion on the aspect of operating speed is added as follows: The

multiplierless correlator and the DSP-based correlator can be performed at the speed of multiple times higher than desired operation speed of baseband modulation. Both of them can apply the structure of time multiplexing to obtain the trade-off between resource usage and operating speed. It should be noted that increased operating speed leads to significantly increased power. The power consumption is proportional to the square of operating speed. To have the fair comparison, both of the multiplierless correlator and the DSP-based correlator are investigated on the same transposed direction form. The structure of time multiplexing for correlator is beyond the scope of the investigation. The subsection 3.2.3 in chapter 3 is amended for this point.

> *3. While power comparisons are made between the multiplierless version and DSP48 version, it isnt clear if the same highly quantized coefficients are being used in the DSP48 case. If not, I would expect those values to drop significantly. To a degree, this discussion breaks down into a traditional precision analysis problem. If indeed 32 bits of precision are not needed in this correlator, then the DSP48 could be the wrong component to use.*

The use of DSP48 version for comparison to the multiplierless version is further discussed as follows: To obtain precise computation, DSP48 version is used for typical multiplication of fixed point format, Q1.15. Because the DSP48 is a hardcore, it is impossible to optimise the internal components of DSP48 for the case of reducing number of quantised bits. The static power of DSP based correlator is not drop significantly when reducing the number of quantised bits. The use of DSP48 for less number of quantised bits is wasteful and reduced precision. Hence, the DSP48 version for small number of quantised bits is not necessary to be evaluated. This explanation is added in the chapter 3 at the subsection 3.2.3.

> *4. It states that the multiplierless design must be specified manually, and cannot be inferred by the tools from higher level descriptions (I assume HDLs). After reviewing the structure presented in this chapter, I believe this assertion should be reexamnied.*

This assertion is revised to be clear as follows: The correlator designs must be specified manually, and cannot be inferred by the tools from higher level descriptions such as high level synthesis (HLS) or Simulink that allows designing system at high level abstraction. This amendment is made in the chapter 3 at the section 3.2.

> *5. Taking advantage of the periodic nature of the energy distribution is an excellent way of reducing the computational burden of gross frame synchronization. When compared to full cross-correlation techniques this reduces the needed computation to a much smaller viable amount.*

The computational reduction is further explained to show clearly how it achieves the significant decrease in term of resource usage as follows: The needed computation of proposed architecture for frame synchronization compared to the full cross-correlation techniques is minimised by not only taking advantage of the periodic nature of preamble. It should be noted that profiting from the real number computation of proposed

metrics, instead of complex number, and from using multiplierless correlator also obtain a significant computational cost. This amendment is made in the chapter 4 at the subsection 4.3.4.2.

> *6. In Chapter 6, the performance of the proposed solution is very good, but seems to come at a significant computational cost due to the extension of the IFFT and increased sample rate. These costs should be quantified.*

The discussion on computational cost is extended by providing a additional complexity analysis in terms of hardware usage (i.e. FFs and DSPs) for dynamically sized IFFT blocks and the FIR filter. This amendment is made in the chapter 6 at the subsection 6.4.2.

> *7. In Chapter 7, the premise is that faster reconfiguration is better, yet very little is provided on the requirements for when it is fast enough. There is significant emphasis on buffering streams at the boundaries of PR modules. I can appreciate the object of not wanting drop any data; however, for the radio standards discussed in previous chapters, there are higher-level protocols that manage the retransmission of lost or dropped packet of data.*

This point is revised more details as follows: When the system adapts to a new condition (i.e., switching the operating standard), the reconfiguration operation is performed that may suspend the data processing in a duration. The received packet of data may be lost. The reconfiguration is required as fast as that the input data buffered during reconfiguration do not caused the buffer overflow that leads to the packet of data dropping. Even there are higher-level protocols that manage the retransmission of lost or dropped packet of data, but performing these mechanisms are wasteful and significant increase the cost in terms of computation and power consumption. This amendment is made in the chapter 7 at the section 7.1.

> *8. One other issue with this chapter is that Mr. Pham is allowing the limitations of the tool establish bounds on what is possible with this work, instead of examining the theoretical limits of FPGA re-programmability. Much of this is due to the acceptance of the slot-based partial reconfiguration (PR) module. This, in turn, skews some of the metrics used throughout the chapter. For example, Table 7.4 summarizes the resource usage of several radio components. Since the PR slot model is assumed, all of the resource in the slot are consumed by the component, regardless of whether they are used in the computation or not. The configuration time for a component of a given slot is constant, even if it small component (an impact on the reconfiguration latency a metric used in this chapter).*

The chapter 7 focuses for the OFDM-based CR architecture to reduce the reconfiguration time on applying PR technique and parameterised module when the system is adapted. The research on PR methods is out of scope. However, the aspect of PR methods is worth to be mentioned. This point is further discussed as follows: Because the

slot based PR is widely used and supported by Xilinx tool, we assume that slot-based PR is employed for performing PR. All of the resources in the slot are consumed by the component, regardless of whether they are used in the computation or not. The configuration time for a component of a given slot is constant, even if it small component. There are some researches [a1] to reduce the wasted resource from unused logic in PR slot for the small component that may reduce the reconfiguration time of small component. However, these approach are just for the limited FPGA devises and requires more expert work to implement the design. Furthermore, this improvement still does not help for overall reconfiguration latency because the overall latency have to take into account the worst case latency (i.e., the reconfiguration time of largest components) when the system adapts from a standard to another. This amendment is made in the chapter 7 at the section 7.4.2. [a1] Sohanghpurwala, A.A.; Athanas, P.; Frangieh, T.; Wood, A., "OpenPR: An Open-Source Partial-Reconfiguration Toolkit for Xilinx FPGAs," IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), May 2011

> *9. The strategy of over-clocking a component immediately after it has been reconfigured to clear the backlog, the turn to normal clocking speed has consequences: designing a module to operate at 2x the needed clock rate is likely going to consume more resources than a component optimized for 1x clock speed.*

The strategy of over-clocking is explained more clearly as follows: Increasing operating speed is done basically by controlling the MMCM module to adjust the speed of input clock of received modules, rx_clk. Because the received modules implemented on FPGA fabric are able to work with clock that is higher than 2X the operating clock, designing these modules supporting 2X the operating clock do not consume more resources compared to the case of 1x clock speed. This amendment is clarified in the chapter 7 at the subsection 7.3.1.

I would like to thank the reviewers for their insightful comments that have helped improve the thesis.