NANYANG
TECHNOLOGICAL
UNIVERSITY

Power Optimisation for Adaptive Embedded Wideband Radios

A Thesis submitted to
the School of Computer Engineering
of Nanyang Technological University

by

Pham Hung Thinh

in fulfillment of the requirement for
the Degree of Doctor of Philosophy of
for Computer Engineering

February 9, 2015

Contents

List of Figures

List of Tables

Chapter 1

Research Introduction

Chapter 2

Background Literature

Chapter 3

Multiplierless Correlator Design for low-power systems

3.1 Introduction

Correlation operators are commonly used to perform synchronisation in OFDM systems. Auto-correlation based techniques are preferred for implementing OFDM synchronisation on FPGA because of their lower hardware costs. Dick and Harris [2] reported on the FPGA implementation of an OFDM transceiver. Their paper shows that FPGAs, with their highly parallel architecture, are suitable for the implementation of OFDM transceivers. Wang et al. [3] also presented an FPGA implementation of an OFDM-WLAN synchronizer. In [3], the timing synchronisation is obtained by double auto-correlation based on short training symbols that allows a reduction in the hardware cost on FPGA. Fort et al. [4] compared the performance and complexity of FPGA implementation of auto-correlation algorithms and cross-correlation algorithms. Results showed that the accuracy of cross-correlation algorithms is better than that of auto-correlation algorithms. However, the accuracy of cross-correlation comes at significant hardware cost. Despite proposing a new cross-correlator implementation to reduce hardware cost compared to a classic cross-correlation approach, it is still at least 5 times more complex to implement than auto-correlation, due to the fact that several multipliers are required.

Cross-correlation between received samples and a known preamble can achieve highly accurate timing synchronisation, however, this requires significant resources. Multiplierless correlators for timing synchronisation were introduced in [5], designed for IEEE 802.11a OFDM frames, based on expressing the correlator coefficients as sums of powers of two that only require shift and add operations. The authors identified a correlator that eliminates the need for

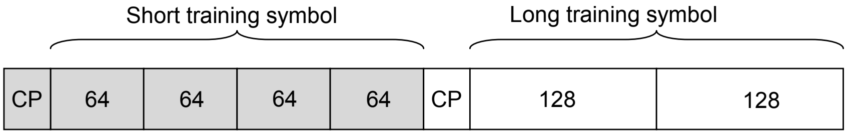


Figure 3.1: Downlink preamble symbols for IEEE802.16.

multiplication, requiring only 26 additions/subtractions per output while maintaining similar synchronisation accuracy to a multiplier-based implementation.

Modern FPGAs contain various resources that can be used to implement cross-correlation. This chapter presents the design of several correlators for timing synchronisation with preamble symbols based upon IEEE 802.16d. We compare designs using specialised DSP Slices to a multiplierless approach on Xilinx Virtex-6 and Spartan-6 FPGA devices. Attempting to implement correlation on FPGAs without considering and designing for the underlying architecture is likely to result in an inefficient implementation. In this chapter, we show optimised designs, built to fit the FPGA architecture, and evaluate performance, timing synchronisation accuracy, resource utilisation, and power consumption, to understand whether a multiplier-based mapping is beneficial when using modern devices.

3.2 Implementation of correlators

The downlink preamble in IEEE 802.16d [6] contains two consecutive OFDM symbols as shown in Fig. 3.1. The short symbol consists of 4 identical 64-sample fragments in time, preceded by a cyclic prefix (CP). This is followed by the long symbol which contains two repetitions of a 128 sample fragment and a CP [6].

The 64 samples in the short symbol are used to perform cross-correlation with received samples for timing synchronisation. Therefore, the correlators are designed to compute cross-correlation with 64 constant coefficients. In this chapter, we explore two approaches to implementing such correlators. The first is based on Xilinx Virtex-6 FPGA DSP48E1 Slices, the standard approach to such an implementation. The second is using multiplierless correlation implemented on both a Xilinx Virtex-6 and a low power Xilinx Spartan-6 device. Both are designed to receive real and imaginary 16-bit samples in Q1.15 fixed point format. The output is the sum of 64 coefficient products, each being smaller than unity. So, the complex output words are in 21-bit fixed point Q6.15 format.

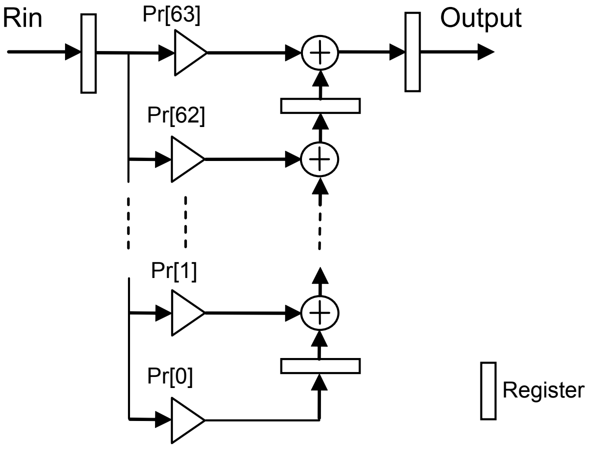
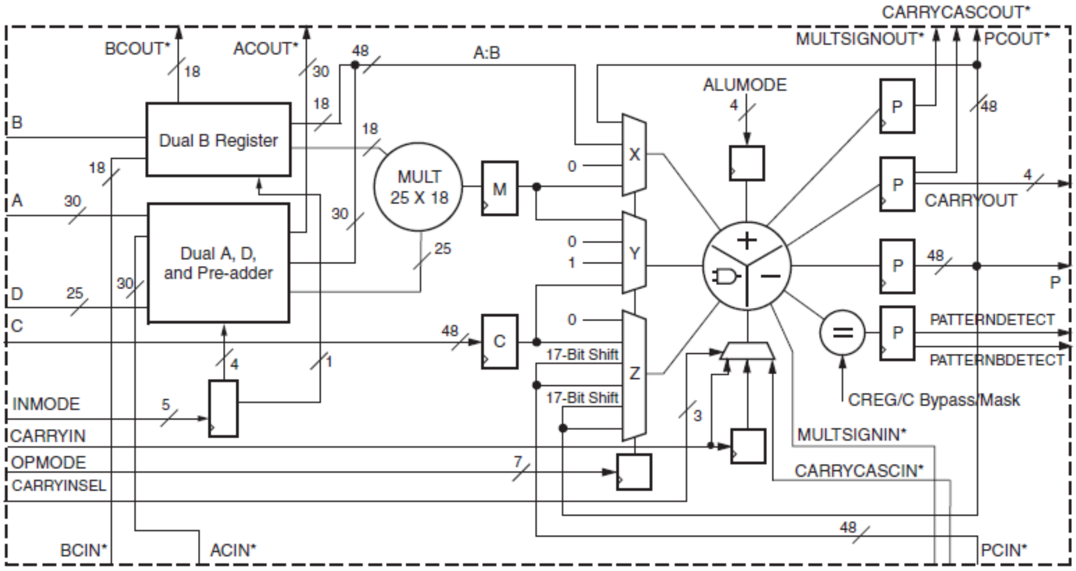


Figure 3.2: Transpose Direct Form of Correlator.

If such a design was implemented blindly, with no consideration for the FPGA architecture, the synthesis tools would infer the use of embedded DSP blocks for multiplication, but would likely achieve poor timing due to an inability to optimise the use of the DSP block. The DSP48E1 primitives on the Virtex-6 FPGAs have additional circuitry within them that enables the design of optimised datapaths, but this must be done manually, through writing code in a particular style. Otherwise, the synthesis tools cannot always infer the most efficient structure. In our design, we have taken into account the internal structure of the DSP block, and made the design as lean as possible. The multiplierless design is specified entirely manually, and thus cannot be inferred by the tools.

3.2.1 Design of DSP48E1 Based Correlator

The DSP48E1 Slice inside the Virtex-6 shown in Fig. 3.3 contains a multiplier followed by a configurable arithmetic unit to provide many independent functions, e.g., multiply, multiply accumulate, multiply add, three-input add, and more [1]. It also allows the datapath to be configured for various imput combinations and register stages; a three-stage pipeline offers maximum performance. Since the DSP Slice is designed to mirror the structure of an FIR filter tap, it is ideally suited to implement correlation, and would hence be the method of choice for this application. Our first design uses non-pipelined DSP48E1 Slices in transpose direct form, as shown in Fig. 3.2, with 64 coefficients corresponding to the 64 samples in the preamble. The coefficients are pre-computed according to the IEEE802.16d standard. The second



*These signals are dedicated routing paths internal to the DSP48E1 column. They are not accessible via fabric routing resources.

Figure 3.3: Structure of DSP48E1 slice inside the Virtex-6 [1]

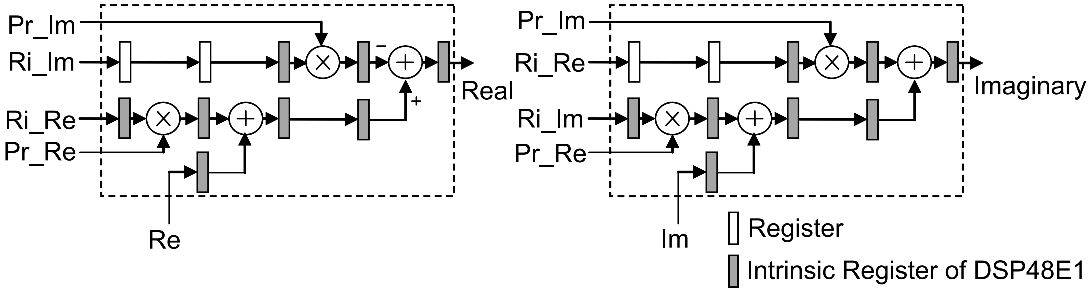


Figure 3.4: Pipeline structure of the complex number multiply-add.

design spreads the complex multiply-adds in a five-stage pipeline, shown in Fig. 3.4, consisting of DSP48E1 Slices configured for three stage internal pipelining. Ri_Re, Ri_Im are the real and imaginary parts of received sample, respectively. Pr_Re, Pr_Im similarly represent known preamble. The pipeline registers for the Pr_Re, Pr_Im are eliminated because they are considered to be of constant value. Re, Im are the real and imaginary parts of the previous multiply-add. Fig. 3.5 presents the pipeline structure of the correlator. The additional pipeline registers are required for handling the received sample. Adding pipeline registers should increase performance significantly.

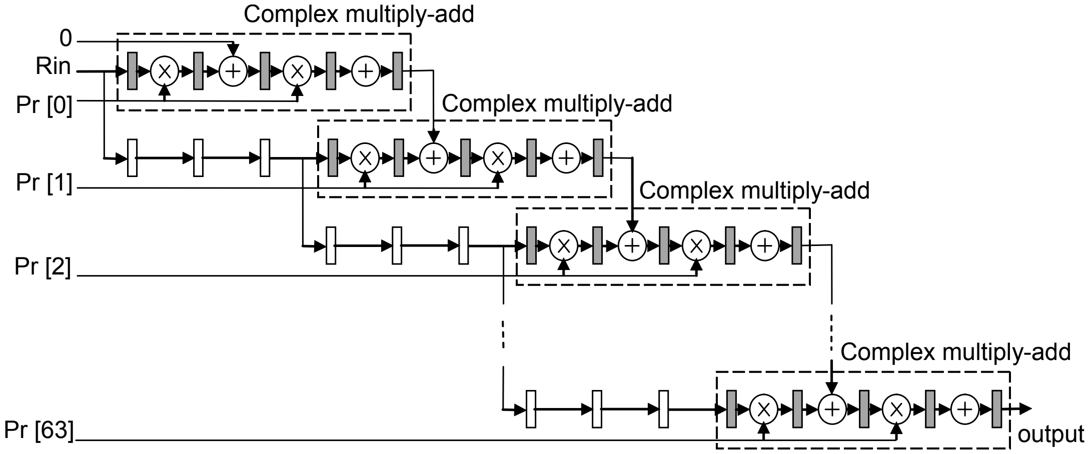


Figure 3.5: Pipeline structure of correlator using DSP48E1 Slices.

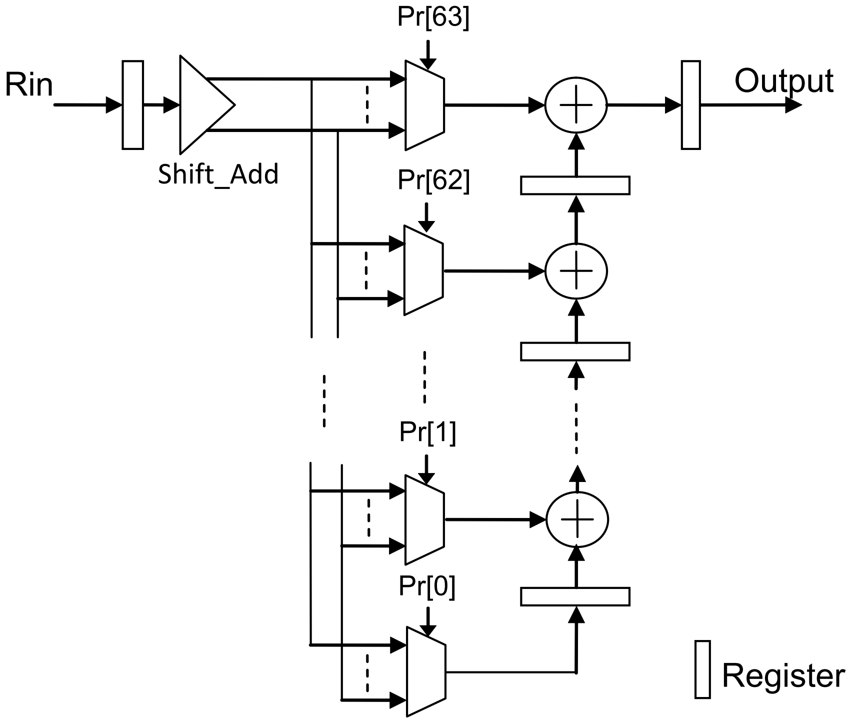


Figure 3.6: Structure of multiplierless correlators.

3.2.2 Design of Multiplierless Correlator

The principle of multiplierless correlators is to represent the coefficients and round them in the form of summed powers of two. Hence, a shift and add are performed instead of multiplying by coefficients. It is expected that multiplierless correlation is more efficient, but with embedded hard multipliers in modern FPGAs, it is unclear whether they should still be considered favourable. Furthermore, synchronisation accuracy must be considered. To explore this, four alternative multiplierless correlators are implemented using four coefficient sets with an increasing degree of rounding, to compare the cost and performance and evaluate against multiplier-based correlators. The coefficient sets are found by quantising the 64 normalised preamble samples with quantisations of 1, 0.5, 0.25, and 0.125.

The proposed structure for multiplierless correlators is shown in Fig. 3.6. This structure is based on the transpose-direct-form in Fig. 3.2. Instead of using multipliers to multiply input samples by coefficients, the $Shift\_Add$ block and multiplexers are used to perform the equivalent operation without an actual multiplication. But the $Shift\_Add$ block, multilexers and value of $Pr[n]$ are different depending upon the quantised coefficient set being used. The $Shift\_Add$ block performs shift and add on received samples according to the degree of quantisation that is applied. To optimise resources in the case of small numbers of bit quantisation, one common $Shift\_Add$ block is used for all 64 coefficients instead of 64 separate $Shift\_Add$ blocks. This common $Shift\_Add$ block calculates all possible values for 64 coefficients. The multiplexers are used to select the corresponding values from $Shift\_Add$ to accumulate in order to generate the correlator output. These are based on expressed coefficients $Pr[n]$ that are pre-computed based on quantizing the 64 preamble samples. Since the $Pr[n]$ values are constants, after synthesising the design, the multiplexer is optimised as hard-wired logic, and the preamble cannot be changed. To support different OFDM preambles, the $Pr[n]$ could be stored in a register, and a real multiplexer used instead of hard-wired logic. This results in increased resource utilisation but provides a more flexible solution.

3.2.3 Implementation Results

The designs presented were synthesised and fully implemented using Xilinx ISE 13.2, targetting Xilinx Virtex-6 (V6) and Spartan-6 (S6) devices. The results of implementation are

Table 3.1: Implementation Report

Design	Occupied Slices		DSP48E1s	Freq. (MHz)	
	V6	S6	V6	V6	S6
DSPc	742 (6%)	-	256 (88%)	119	-
DSP_pp	1,110 (9%)	-	256 (88%)	398	-
ML1	661 (5%)	762 (6%)	0 (0%)	309	174
ML2	983 (8%)	1,071 (9%)	0 (0%)	268	158
ML3	1,191 (10%)	1,257 (10%)	0 (0%)	234	136
ML4	1,496 (12%)	1,517 (13%)	0 (0%)	208	124

reported in terms of the number of occupied slices, DSP48E1s Slices and maximum frequency as summarised in Table 3.1.

$DSPc$, DSP_pp are correlator designs using DSP Slices, in non-pipelined and pipelined structures, respectively. $ML1$, $ML2$, $ML3$, $ML4$ are multiplierless correlators with coefficient quantisations of 1, 0.5, 0.25, 0.125, respectively.

Table 3.1 reveals that the DSP_pp uses more logic slices due to its pipeline structure. The slices in DSP48E1 based designs are used for registers and route-thrus while the slices in the multiplierless designs are mostly used as logic. The number of slices used in the multiplierless designs increases as the coefficient quantisation becomes finer. The DSP48E1-based designs use 256 DSP Slices, 4 for each complex multiply plus 6-9% of logic resources. The multiplierless designs use only logic to compute the cross correlation with 64 complex coefficients. The total logic area is a small fraction of the whole device: around 5-12% of total resources in the Virtex-6, and around 6-13% of total resources in the equivalent Spartan-6. While Spartan-6 devices do include DSP Slices, their number is insufficient to implement the full 64 sample complex cross correlation. This shows an ideal scenario where multiplierless correlation makes sense, and hence serves as a motivation for this study.

The maximum frequencies, reported after place and route, decrease for multiplierless designs according to the degree of coefficient quantisation. Meanwhile the non-pipelined DSP48E1 design is slower than the multiplierless designs. However, the pipelined DSP48E1 design can achieve higher frequency.

A post-place-and-route simulation in ModelSim was used to estimate the power consumption of the system using the Xilinx XPower tool. Table 3.2 shows the power dissipation of the designs running at 50 MHz. The DSP48E1-based correlators consume more power than the multiplierless correlators, but this is due primarily to increased dynamic power when using the DSP48E1s on the Virtex-6. The dynamic power of the non-pipelined DSP48E1-based correlator DSP_Cor is greatest at 846 mW, but pipelining reduces this by a factor of more than 2.5 times, due to reduced switching activity between the multiplier and adder. The dynamic power of the multiplierless designs increases from 133 mW to 203 mW on Virtex-6 and from 149 mW to 294 mW on Spartan-6 as finer coefficient quantisation is used. It is important to note that the quiescent power of the Spartan-6 is much lower by design. Hence, we can see that using this multiplierless technique allows us to implement synchronisation on a Spartan-6 device, where a multiplier-based design is not possible, saving significant power and sacrificing little in terms of accuracy.

Table 3.2: Power Consumption at 50MHz.

Correlators	Quiescent(mW)		Dynamic(mW)		Total(mW)	
	V-6	S-6	V-6	S-6	V-6	S-6
DSP_Cor	1312	-	846	-	2158	-
DSP_pp_Cor	1300	-	328	-	1628	-
ML1_Cor	1296	67	133	149	1429	216
ML2_Cor	1296	68	160	197	1456	265
ML3_Cor	1297	70	182	239	1479	309
ML4_Cor	1297	71	203	294	1500	365

We also investigated how total power consumption varies with frequency, as shown in Fig. 3.7. As frequency increases, the finer quantisations and DSP48E1-based designs begin to consume proportionally more power. Overall, multiplierless designs on the Spartan-6 consume 75% to 85% less power than the same designs on the Virtex-6, and a 0.25 quantisation design on the Spartan-6 consumes 81% to 85% less power then the DSP48E1-based design on a Virtex-6.

The *DPS_Cor* implementation represents how a “blind” design would be mapped. Our architecture-aware designs show significantly better performance, reduced area, and reduced power consumption.

3.3 Simulation and discussion

In order to validate our designs at the application level, we simulate them using ModelSim with an IEEE802.16 OFDM frame created using MATLAB, including the preamble symbols,

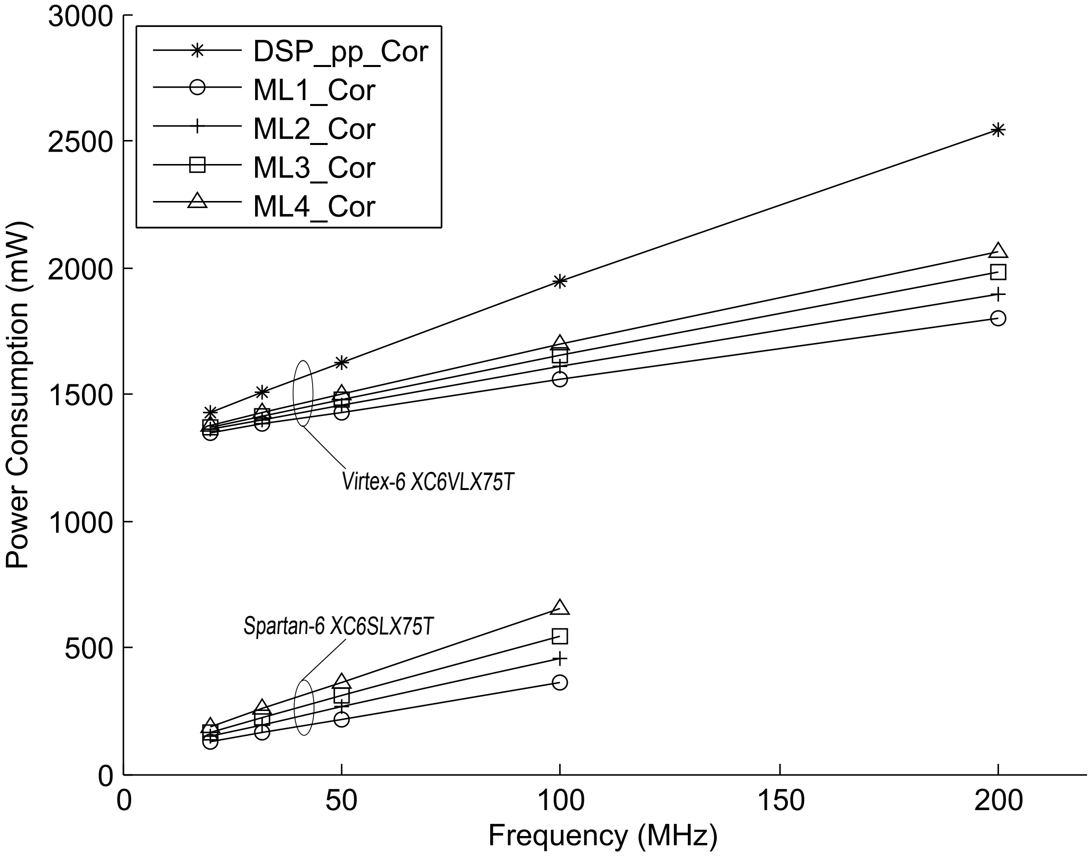


Figure 3.7: Power consumption of designs.

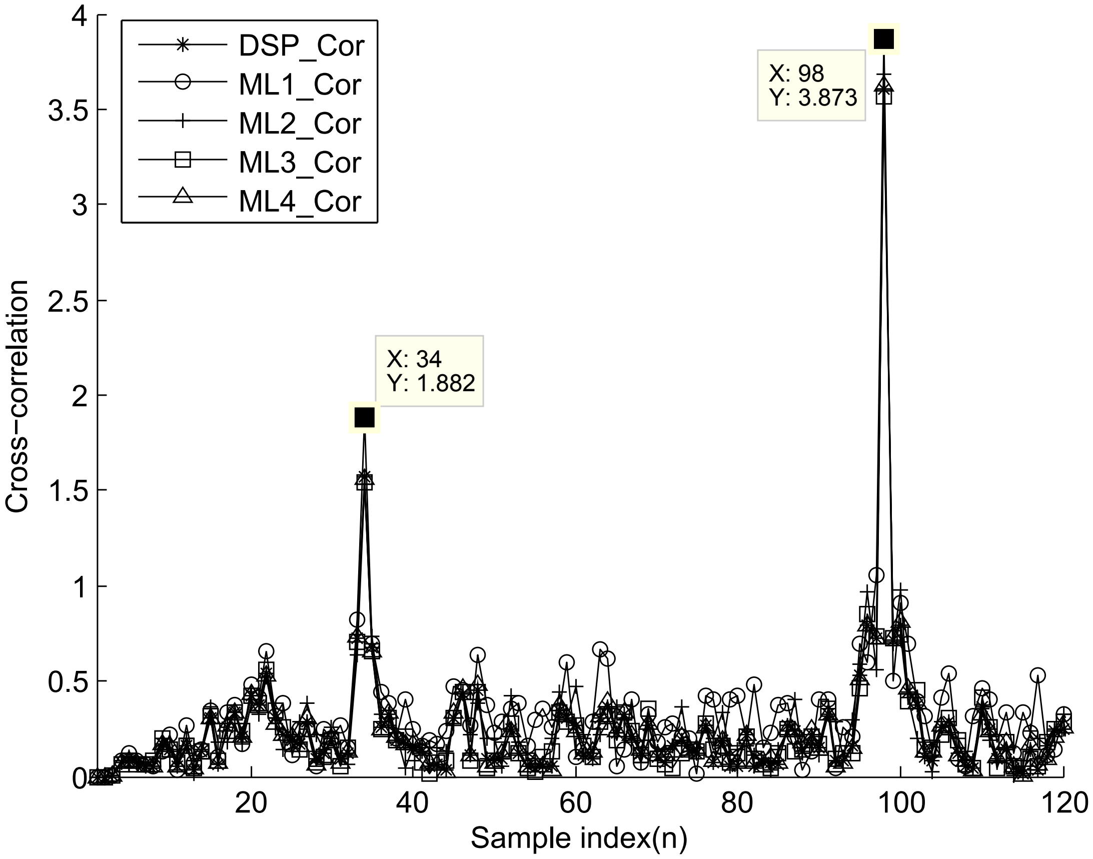


Figure 3.8: Correlator output with SNR = 10dB.

data symbols and effects of an AWGN channel. Cross-correlation results using the correlator designs are compared to corresponding results in MATLAB to verify the correctness of implementation. To evaluate the accuracy of timing synchronisation acheivable by these designs, the correlation outputs are plotted in Fig. 3.8 for random data frames at 10 dB SNR. The output of each correlator is slightly different because of rounding, but the timing synchronisation depends upon the location of the peaks being at the position of the preamble. All the correlator designs achieve this most of the time, as shown at indices 34 and 98 for the CP samples and the first preamble samples respectively, for a single frame.

In order to evaluate the synchronisation accuracy of these approaches, we simulate 10,000 correlation operations in an AWGN channel with a detection strategy as follows. First, find the first peak, $P1$, over 64 samples. Next, find the second peak, $P2$, in the next 64 samples from the first peak and compute average value, avg, of the samples between two peaks. If $(P1 - avg) \leq 0.75 * (P2 - avg)$, the start of frame is detected and the correctness of the position can be checked. It should be noted that in all cases, the peaks were known to be located within the two search regions and that the detection strategies described above are compatible with those of other authors such as [7] and [5]. Fig. 3.9 plots the results in terms of failure rate against AWGN SNR and show that the designs are able to accurately detect the start of frame even in low SNR conditions. The failure rate of $ML1\_Cor$ is the highest, as expected, due to the coarse quantisation. For SNRs above 4 dB, the failure rates of $ML2\_Cor$, $ML3\_Cor$, $ML4\_Cor$ and DSP_Cor differ less than 0.05% from each other. This suggests that sacrificing accuracy by using multiplierless cross-correlation is feasible and has negligible impact on synchronisation accuracy. Combined with the results in the previous section, we can be confident that low-power FPGAs, such as the Spartan-6, with insufficient resources for multiplier-based synchronisation correlation, are still feasible for implementing robust OFDM receivers.

3.4 Summary

The DSP48E1 Slices on modern Virtex-6 FPGA devices seem to offer the ideal resource for implementing correlation-based frame synchronisers. However, as we have discovered, in the context of synchronisation for IEEE802.16 OFDM systems, simplified multiplierless designs offer comparable synchornisation performance. While the DSP48E1-based correlators can obtain higher clock speeds, this is only possible through a detailed pipelined design. Furthermore, their power consumption and resource usage is considerably greater. Since low-power, low-cost devices such as the Xilinx Spartan-6 do not include sufficient DSP Slices, this suggests adopting multiplierless designs for low-power implementations. We have shown that while very low quantisation resolution does impact synchronisation performance, with a step size of just 0.5, synchronisation accuracy is on par with multiplier-based correlation. Multiplierless correlation on a Spartan-6 can save over 85% power compared to a DSP Slice design on a Virtex-6 FPGA.

This work has been submitted as a journal paper to IEEE transactions on Very Large Scale Integration (VLSI) systems.

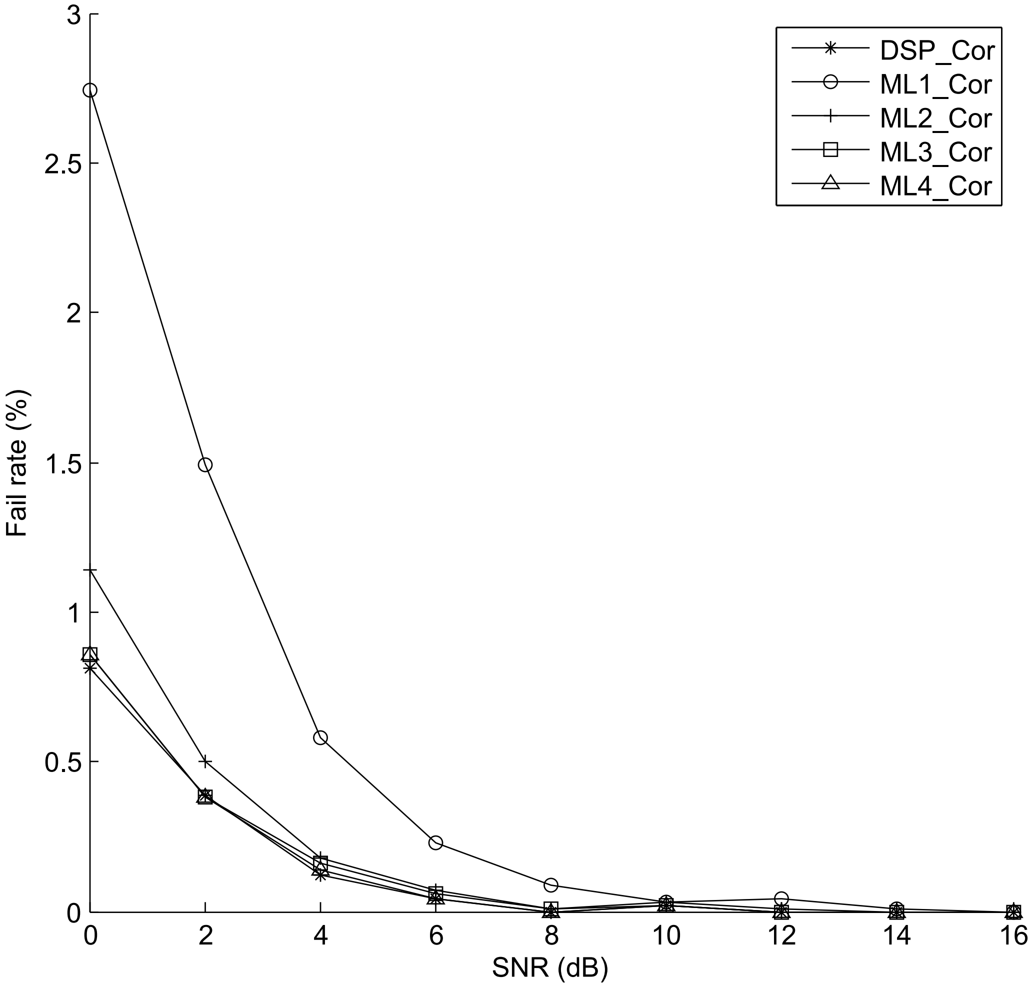


Figure 3.9: Detection failure rate.

Chapter 4

A Method for OFDM Timing Synchronisation

Chapter 5

A CFO Estimation Method for OFDM Synchronisation

Chapter 6

A Spectrum Efficient Shaping Method

Chapter 7

A Novel Architecture for Multiple Standard Cognitive Radios

Chapter 8

Conclusion and Future Work

References

[1] "Virtex-6 FPGA DSP48E1 Slice: User Guide." Xilinx, San Jose, CA. www.xilinx.com/support/documentation/user_guides/ug369.pdf, 2011.

[2] C. Dick and F. Harris, "FPGA implementation of an OFDM PHY," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 905 – 909, Nov. 2003.

[3] K. Wang, J. Singh, and M. Faulkner, "FPGA implementation of an OFDM-WLAN synchronizer," in *Second IEEE International Workshop on Electronic Design, Test and Applications, 2004. DELTA 2004*, pp. 89 – 94, Jan. 2004.

[4] A. Fort, J.-W. Weijers, V. Derudder, W. Eberle, and A. Bourdoux, "A performance and complexity comparison of auto-correlation and cross-correlation for OFDM burst synchronization," in *IEEE International Conference on Acoustics,Speech,and Signal Processing, 2003 (ICASSP'03)*, vol. 2, pp. II – 341–4, april 2003.

[5] K.-W. Yip, Y.-C. Wu, and T.-S. Ng, "Design of multiplierless correlators for timing synchronization in IEEE 802.11a wireless LANs," *IEEE Transactions on Consumer Electronics*, vol. 49, pp. 107 – 114, Feb. 2003.

[6] IEEE std.802.16-2009, *IEEE Standard for Local and Metropolitan Area Networks Part16: Air Interface for Fixed Broadband Wireless Access Systems.*

[7] C. N. Kishore and V. U. Reddy, "A frame synchronization and frequency offset estimation algorithm for OFDM system and its analysis," *EURASIP Journal on Wireless Communications and Networking*, vol. 2006, pp. 1–16, 2006.