Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 1, March 2016 , p.p(75-91)
ISSN 1992 – 0849

FPGA Implementation of Wavelet Filters for DWMT Systems

Omar M. Salih[1] , **Abdulrahman I. Siddiq**[2]

[1]College of Technology / Kirkuk , Iraq - Software Engineering Dep.

[1]oms@tcokirkuk.org

[2]College of Technology / Kirkuk , Iraq Electronic and Control Engineering Dep.

[2]dr.a.siddiq@ieee.org

Received date : 4 / 1 / 2015 **Accepted date : 22 / 10 / 2015**

ABSTRACT

Discrete Wavelet Multi-Tone (DWMT) systems acquired attention due to their high spectral efficiency and high data rates with respect to FFT-based multitone transmission systems. The complexity of the overall system is directly related to that of the elemental building block. In the literature, wavelet filters are designed subject to constraints for minimum interference. The structure of a Minimum Interference Wavelet Filter (MIWF) is very simple even for high filter orders. In this paper, DWMT systems using a two-branch wavelet filter bank in the transmitter and its inverse at the receiver are implemented using the Spartan XC3S1200E FPGA. The details of system implementation are presented for MIWF, Daubechies, and Coiflet wavelet filters. The tests show that, with respect to the other tested systems, the MIWF-based system is simpler, faster and capable to preserve its full precision BER performance even when the filter coefficients word size is reduced to 5 bits.

Keywords: *Discrete Wavelet Multi-Tone (DWMT), Field Programmable Gate Array (FPGA), Wavelet filters, Minimum Interference Wavelet Filter (MIWF)*

تنفيذ مرشحات المويجة لأنظمة الارسال متعددة الترددات باستخدام FPGA

باستخدام

عمر محمد صالح[١] ، عبد الرحمن اكرم صديق[٢]

[1]الكلية التقنية / كركوك
[1]oms@tcokirkuk.org
[2]الكلية التقنية / كركوك
[2]dr.a.siddiq@ieee.org

تاريخ استلام البحث: ٤ / ١ / ٢٠١٥ تاريخ قبول البحث: ٢٢ / ١٠ / ٢٠١٥

الملخص

اكتسبت انظمة الارسال متعددة الترددات المعتمدة على مرشحات المويجة (DWMT) اهميتها بسبب تميزها بكفاءة استخدام الطيف الترددي و بمعدلات الارسال العالية مقارنة بالانظمة المماثلة التي تستخدم محول فورير. بطبيعة الحال، فان درجة تعقيد انظمة DWMT تعتمد على درجة تعقيد مرشحات المويجة المستخدمة فيها. ان مرشحات المويجة المصممة لتحقيق متطلبات تقليل التداخل بين القنوات المتجاورة (MIWF) تتميز ببساطة تركيبها. في هذا البحث تم تنفيذ منظومة DWMT تستخدم مرشحات MIWF بواسطة مصفوفة البوابات القابلة للبرمجة حقليا (FPGA) من نوع Spartan XC3S1200E. كما تم تنفيذ منظومات DWMT تستخدم مرشحات من انواع اخرى مثل Daubechies و Coiflets لغرض مقارنة درجة تعقيد الانظمة و اداءها. اختبرت هذه الانظمة المنفذة تحت ظروف متعددة لقناة الاتصال و باطوال مختلفة للتمثيل العددي لمعاملات المرشحات. اظهرت الاختبارات بأن منظومة DWMT المعتمدة على مرشحات MIWF هي اقل تعقيد و اسرع وبامكانها المحافظة على نفس مستوى اداءها في حالة الدقة الكاملة حتى عندما يكون تمثيل معاملات المرشحات ٥ بت فقط، بينما ينخفض مستوى اداء المنظومات الاخرى المختبرة تحت نفس الظروف.

الكلمات الدالة: انظمة الارسال متعددة الترددات لمرشحات المويجة (DWMT)، مرشحات المويجة (MIWF)، مصفوفة البوابات القابلة للبرمجة حقليا (FPGA)

1. Introduction

The Discrete Wavelet Transform (DWT) has been successfully used in many applications such as digital signal and image processing and digital communications including the Discrete Wavelet Multi-Tone (DWMT) systems [1]-[5]. The implementation of DWT filters on Field Programmable Gate Array (FPGA) has been increasingly developed due to advances in technology and decreasing costs [4]. Generally, the computation of the DWT is complex and it is occasionally difficult to meet the requirements of real time operation [6]. Many FPGA architectures of the DWT have been proposed in the literature to optimize the speed and resource consumption [2],[5]-[8].

Referring to DWMT systems, wavelet filters are designed in [1] to meet the requirement of minimum Inter-Carrier Interference (ICI) and Inter-Symbol Interference (ISI) in these systems. The most interesting feature of the designed Minimum Interference Wavelet Filters (MIWF) is their simple structure. That is, an (N-1)th order MIWF consists of only two nonzero components separated by (N-2) zeros. Therefore, the hardware of DWMT systems based on the MIWFs will be simpler than those employing other types of wavelet filters.

In this paper, equivalent DWMT systems using the MIWF, Daubechies, and Coiflet filters are implemented using FPGA. The BER performance of these systems is evaluated under different test conditions. Then, they are compared to each other in terms of hardware complexity and real-time applicability related parameters to show the superiority of the MIWFs over the other tested wavelet filters. The rest of the paper is organized as follow: A brief overview on DWMT systems is given in Section II. The details of FPGA implementation of the MIWF, Daubechies and Coiflet filters are presented in Section III. The BER performance and hardware complexity of the implemented systems are presented in Sections IV and V, respectively. Finally, conclusion remarks are given in Section VI.

2. DWMT Systems

A DWMT system is the counterpart of the well-known orthogonal Frequency Division Multiplexing (OFDM), where the DWT is used instead of the DFT. This gives the DWMT advantages over OFDM including better spectral efficiency and higher data rates [3],[9],[10]. The core of a DWMT system is the M-band Inverse DWT (IDWT) and DWT blocks in the modulator and demodulator, respectively. The system performance is directly related to the ability of the used wavelet filters to resist ICI and ISI and stay orthogonal [1]. Generally, the wavelet modulator is constructed by iterating an elemental two-branch filter bank block (consisting of low pass and high pass filters) as a binary tree to achieve the division of the spectrum into M orthogonal bands. This is known as Wavelet Packet Modulation (WPM) [1],[5],[9]-[12]. This structure provides a significant design property which is that the overall system design is reduced to the design of the two-branch filter bank [1],[10].

The implementation of conventional wavelet filters may be subjected to the limitations of the available hardware, such as limited data size, processor speed and storage. These together with additional parameters related to the architecture of the filters, lead in turn to degradation in the performance of the DWMT system. Whereas, the MIWFs presented in [1] are promising alternatives which are capable of neglecting most of the implementation difficulties.

3. FPGA IMPLEMENTATION

The SpartanXC3S1200Eplatform is used to implement three DWMT systems using the MIWF, Daubechies, and Coiflet wavelet filters. As shown in Fig.(1) , the system comprises the transmitter (Tx), receiver (Rx), communication channel and in addition to memory blocks. The memory blocks, namely input ROM, noise ROM and output RAM are designed to store input data, noise and recovered data, respectively. The transmitter and receiver blocks are the point of comparison since their implementation depends on the specific employed wavelet filter. In order to evaluate the performance of the implemented systems, a communication channel used with various parameters to provide different test conditions for fair system qualification. The system controller is used to control the blocks of the system and provides synchronization.

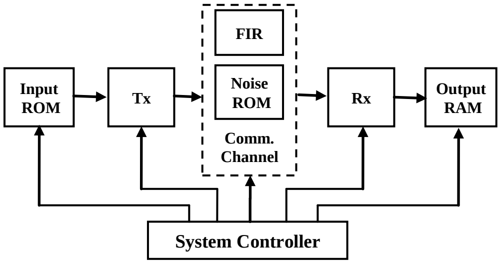


Fig. (1): Implemented DWMT block diagram

A. I/O Memory

The input data used in this work is taken to be a BPSK signal, therefore, two bits are sufficient to represent its samples in fixed point arithmetic. Due to the limited size of the built in memory of the used FPGA, the huge amount of data used for simulation (10^6 samples×2 bits for input data, 10^6 samples×16 bits for noise and 10^6 samples×2 bits for recovered data) are stored in text files on an external storage.

The input ROM is designed to operate as a dual port memory that reads data from the auxiliary storage file. In Fig.2(a) , it outputs two consecutive samples through the data ports D_A and D_B simultaneously in order to feed even and odd indexed samples to the transmitter's LPF and HPF, respectively. In order to implement the cases where the input data is a sequence of zeros such as the insertion of zeros required for oversampling or for signal padding, the Synchronous Set/Reset (SSR) signal is used. This signal controls the output buffer value of the ROM (in this case zero) to be transferred rather than the data associated with the current address. The used SSR technique neglects the need to implement hardware modules for these operations. The opposite function of the input ROM is the output RAM, shown in Fig.2(b) . It is used to store the recovered data in a file to be compared later with the original signal stored in the input ROM.

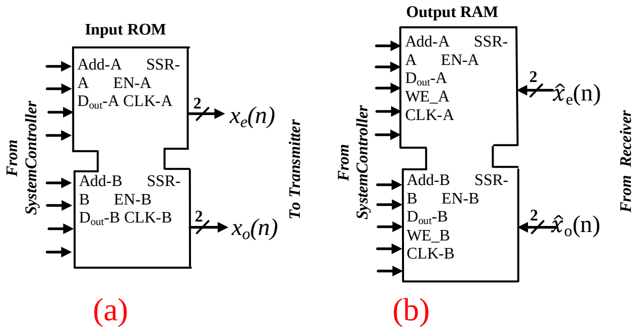


Fig. (2): Memory circuits for (a) transmitter (b) receiver

The data is written when the signal Write Enable (WE) for each port is enabled. The down sampling process is performed implicitly via the output RAM by clocking the WE signal by half the frequency speed of the system clock.

B. Transmitter

The transmitter consists of 6-tap FIR LPF and HPF filters. It is implemented for the MIWF, Daubechies and Coiflet wavelet filters. The transmitter uses fixed point arithmetic with 2’s complement number representation.

1) MIWF

The sampled impulse responses of the 6-tap MIWF filters are given as [1]

$$g[n] = \left[\tfrac{1}{\sqrt{2}}\ 0\ 0\ 0\ 0\ \tfrac{1}{\sqrt{2}}\right] \qquad (1)$$

And

$$h[n] = \left[\tfrac{1}{\sqrt{2}}\ 0\ 0\ 0\ 0\ \tfrac{-1}{\sqrt{2}}\right] \qquad (2)$$

where$g[n]$ and $h[n]$ are the LPF and HPF, respectively. However, FIR filters can be implemented in various architectures, the inverted structure is used in this work. Mathematically, the LPF’s output, $y_g[n]$, can be calculated by adding the values of the current sample with the fifth past sample and multiplying the result by $\frac{1}{\sqrt{2}}$. Similarly, the output of the HPF, $y_h[n]$, can be calculated by subtracting the value of the fifth past sample from the current sample then multiplying by $\frac{1}{\sqrt{2}}$, that is

$$y_g[n] = \frac{1}{\sqrt{2}}(x_e[n+5] + x_e[n]) \quad (3)$$

And

$$y_h[n] = \frac{1}{\sqrt{2}}(x_o[n+5] - x_o[n]) \quad (4)$$

where$x_e[n]$and $x_o[n]$are the even and odd samples of the input data, respectively. The final output of the transmitter becomes

$$y[n] = y_g[n] + y_h[n] \quad (5)$$

Recall that the input data is BPSK and the filter coefficients are known constants, then all of the possible multiplications can be pre-computed and stored in a 4-input Look-Up Table (LUT) rather than using the available 18×18 slower and more power consuming multiplier. The coefficient $\frac{1}{\sqrt{2}}$ is represented by 9 bits in the form of (1.8) meaning that 1 bit for the integer part and 8 bits for fractional part. The LUT contents are calculated and by neglecting any common redundant sign extended MSB or zero LSB, the transmitter output is represented by 10 bits in the form of (3.7). The detailed structure of the MIWF-based transmitter is shown in Fig.(3) .

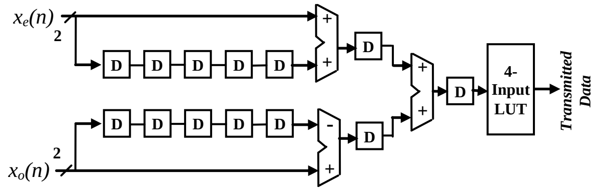


Fig. (3): Implemented MIWF transmitter filters

2) Daubechies and Coiflet Filters

The Daubechies (db3) and Coiflet (coif1) are used as the 6-tap wavelet filters. The coefficients of these filters are represented by 9 bits in the form of (1.8) to have the same accuracy as in the case of MIWFs. Transmitter architecture is illustrated in Fig.(4) . For the FIR realization, each multiplier can be replaced by a 2-input LUT as shown in Table (1) .

Table (1): db3 and coif1 LUT structure

Input	LUT Output
01	Filter coefficient
11	Complement of Filter coefficient
Others	Zero

The output of both LPF and HPF is bounded to 12 bits in the form of (4.8) which is sufficient to represent the addition of all possible intermediate product terms. The MSB is extended by 1 bit to avoid possible over/under flows when obtaining the final transmitter’s output.

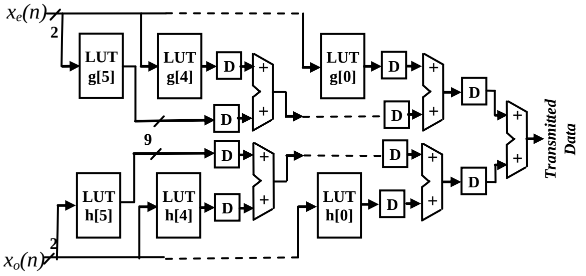


Fig. (4): Implemented db3 and coif1 transmitter filters

C. Test Channel

In order to evaluate the BER performance for the three implemented wavelet systems, a 4-tap FIR test channel given by *a*[*n*]=[1 0.25 0.125 0.0625] in addition to AWGN with different SNR values (0 to 30 dB) is used. The coefficients of **a** can be easily implemented using shift/add operations since they are integer powers of 2. The implemented test channel architecture is illustrated in Fig.(5) .

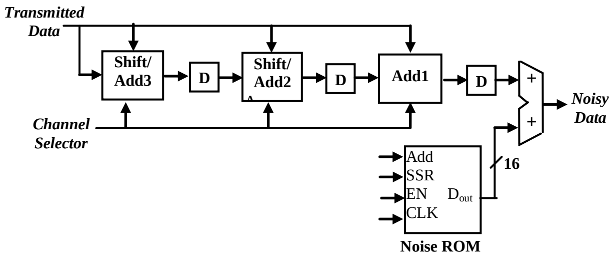


Fig. (5): Implemented channel architecture

To simulate the system with or without the effect of the channel FIR filter (with AWGN only), a control signal is used for this purpose, namely ‘channel-selector’. Having the test channel is minimum phase, a(0)=1, then if the channel-selector is asserted, add/shift2 and add/shift1 are disabled, then adder1 adds the current sample with zero yielding the same value. The samples of the AWGN are represented by 16 bits in the form of (4.12) and stored in a single port ROM.

D. Receiver

1) MIWF

The receiver filters are implemented using the same reduction explained in the transmitter, except that the shared multiplier is implemented before the addition performed in LPF or the subtraction in HPF, as shown in Fig.(6) .

Since the values of noisy data cannot be predicted, a dedicated 18x18 multiplier must be used rather than the LUT as in transmitter filters. However, more reduction can be gained by implementing only one shared delay chain for the LPF and HPF. The final output for both filters is then thresholded by assigning +1 (=01_2) for positive numbers and -1 (=11_2) for negative numbers.

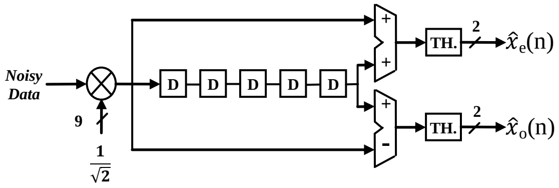


Fig. (6): Implemented MIWF receiver filters

2) Daubechies and Coiflet Filters

There is no possible reduction for both db3 and coif1 receiver filters due to the unpredictable input. The implemented receiver structure for these filters is illustrated in Fig.(7) .

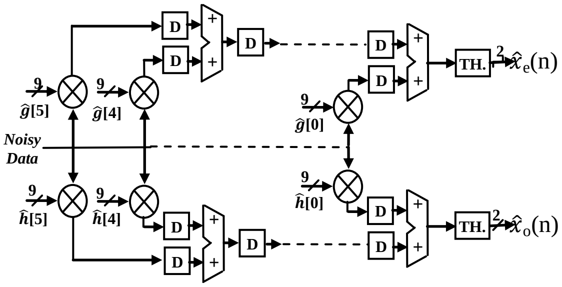
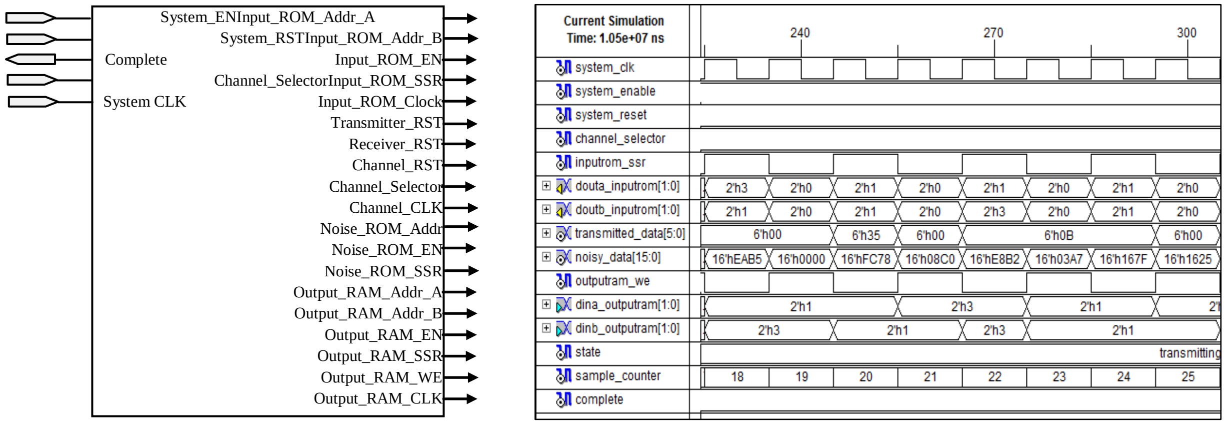


Fig. (7): Implemented db3 and coif1 receiver filters

E. System Controller

It is a four states FSM circuit used to control the simulation of the overall implemented system through the states (disable, reset, transmitting/ receiving and complete) using the control signals shown in fig.8.



(a) (b)

Fig. (8): (a) System Controller (b) Simplified timing waveforms

4. BER PERFORMANCE

The BER performance of the implemented DWMT systems is evaluated by testing them under various filter coefficient representation sizes and channel conditions. A BPSK signal consisting of 10^6 statistically independent bits is transmitted and the BER is computed by comparing the contents of the input ROM and the output RAM.

As a reference BER performance, the implemented systems are simulated using MATLAB with full precision under the same channel conditions. The simulation results for an AWGN channel are shown in Fig.(9) , where the tested systems show equivalent performance.

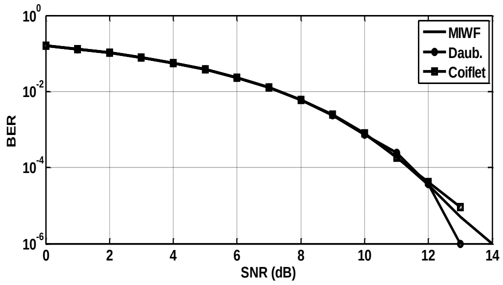
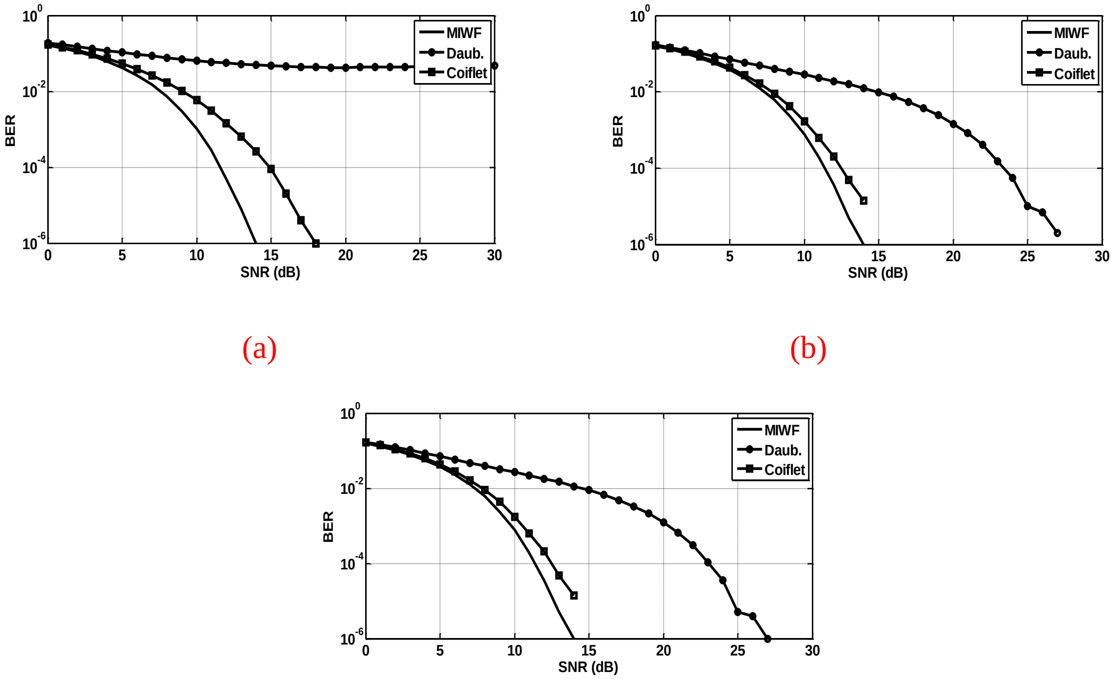


Fig. (9): BER performance of MATLAB simulated systems over AWGN channel

Next, in order to investigate the effect of filter coefficient truncation, the BER of the FPGA implemented systems is evaluated and plotted in Fig.(10) for filter coefficients represented by 5, 9, and 15 bits, with the noise samples being represented by 16 bits.



(a)

(b)

(c)

Fig. (10): BER performance of FPGA implemented systems over AWGN channel with filter coefficient size (a) 5 bits (b) 9 bits (c) 15 bits

It is interesting to note that the MIWF-based system preserves its performance even in the case of sever coefficient truncation (5 bits) whereas the Coiflet-based system requires more SNR to achieve the same level of BER. The Daubechies-based system requires about 10 dB increase in SNR to achieve a BER of 10^{-6} in the cases of 9 and 15 bits filter coefficients with respect to its full precision performance, and it fails to operate properly in the case of 5 bits. This advantage of the MIWF-based system is due to the simple internal arithmetic operations with respect to db3 and coif1-based systems. That is, in an n^{th} order MIWF filter there are only two non-zero coefficients where as in db3 and coif1 filters there are n+1 non-zero coefficients, and the truncation in their binary representation will cause more error accumulation.

However, in practical systems, the communication channel introduces impairments related to multipath fading and frequency selectivity in addition to the AWGN. Therefore, the implemented systems are tested over a test channel characterized by the sampled impulse response $a[n]$=[1 0.25 0.125 0.0625], which is implemented as an FIR filter as described in section 3. This channel introduces a peak distortion, defined as $\frac{1}{|a_0|}\sum_{i=1}^{3} a_i$, of 0.4375 to the transmitted signal, which is considered as a sever channel condition. The MATLAB simulation results for this channel (**a** with AWGN) are shown in Fig.(11) . The three tested systems converge and reach the 10^{-6} BER level but at greater SNR with respect to the case of AWGN channel, with the MIWF-based system is affected by the channel distortion more than the other tested systems.

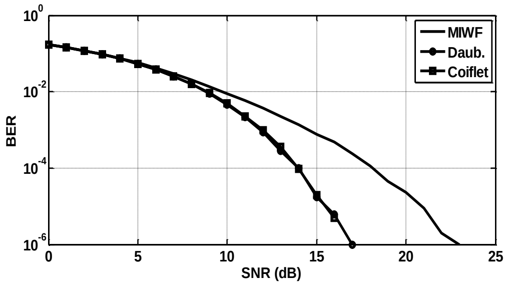
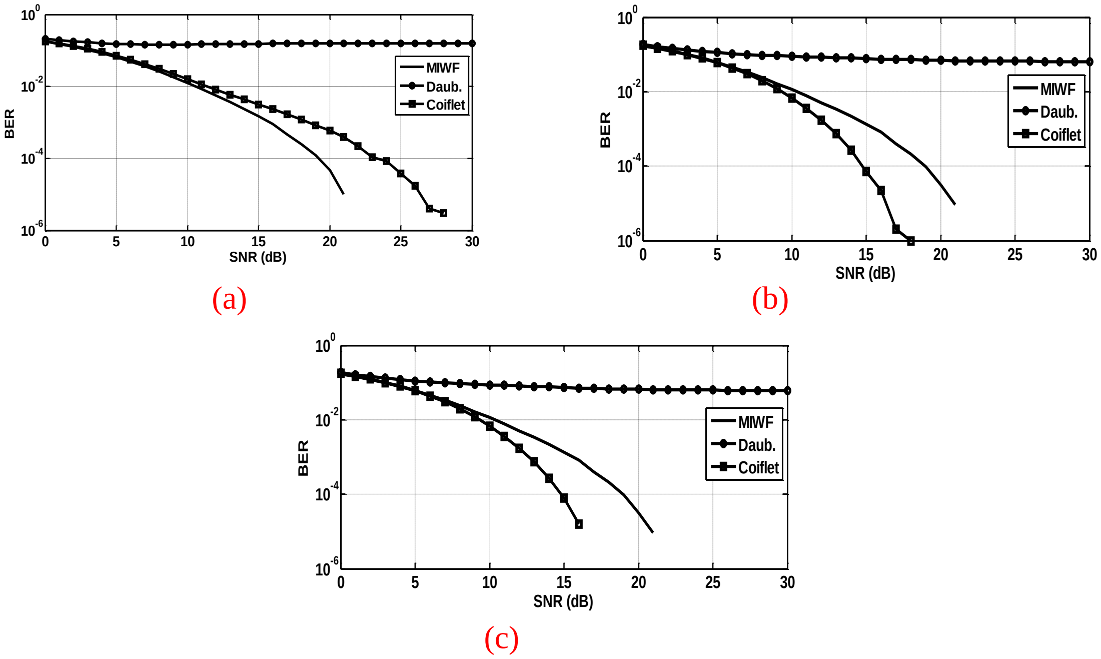


Fig. (11): BER performance of MATLAB simulated systems over (**a** with AWGN)

For the same channel, the BER performance of the FPGA implemented systems is shown in Fig.(12) for 5, 9, and 15 bits filter coefficient sizes, and 16-bit noise samples. Clearly, the Daubechies-based system fails to withstand the truncation, loses its orthogonality and becomes unable to achieve acceptable BER in any of the tested cases. Whereas, for the same test cases, the MIWF-based system preserves its performance and it is not affected by the truncation. The BER of the Coiflet-based system degrades for 5-bit filter coefficients and becomes better as the filter coefficient size is increased.



(a)

(b)

(c)

Fig. (12): BER performance of FPGA implemented systems over (**a** with AWGN) channel with filter coefficient size (a) 5 bits (b) 9 bits (c) 15 bits

5. System Complexity

The hardware complexity of the implemented systems is compared in terms of the number of slices, flip-flops, LUTs, and multipliers consumed from the available hardware on the Spartan XC3S1200Echip. Comparisons are presented for the overall system (including the Tx, Rx, communication channel, and system controller), transmitter alone, and receiver alone, all when implemented using different filter coefficient sizes (5, 9, and 15 bits). Tables (2) through (4) present the available and consumed hardware for the tested systems.

Table (2): Hardware utilization, speed and dynamic power for 5-bit filter coefficients

Parameter	Available	MIWF			Daubechies			Coiflet		
		Sys.	Tx.	Rx.	Sys.	Tx.	Rx.	Sys.	Tx.	Rx.
Slices	8672	222	19	103	411	44	286	423	42	318
Flip-flops	17344	292	35	174	639	80	468	662	77	492
LUTs	17344	237	16	57	513	82	258	602	77	350
multipliers	28	1	0	1	7	0	7	6	0	6
Power (mW)		52	23	29	95	31	64	93	30	63
Speed (MHz)		141	337	246	141	264	230	141	263	230

Table (3): Hardware utilization, speed and dynamic power for 9-bit filter coefficients

Parameter	MIWF			Daubechies			Coiflet		
	Sys.	Tx.	Rx.	Sys.	Tx.	Rx.	Sys.	Tx.	Rx.
Slices	248	20	123	511	63	340	509	65	337
Flip-flops	336	36	206	829	121	606	828	125	602
LUTs	265	17	67	620	114	317	622	116	317
multipliers	1	0	1	12	0	12	12	0	12
Power (mW)	59	29	30	128	40	88	127	40	87
Speed (MHz)	141	337	234	141	250	220	141	250	220

Table (4): Hardware utilization, speed and dynamic power for 15-bit filter coefficients

Parameter	MIWF			Daubechies			Coiflet		
	Sys.	Tx.	Rx.	Sys.	Tx.	Rx.	Sys.	Tx.	Rx.
Slices	307	20	184	671	92	569	670	91	593
Flip-flops	443	36	294	1144	176	840	1139	175	836
LUTs	321	17	118	1023	165	657	1019	165	653
multipliers	2	0	2	24	0	24	24	0	24
Power (mW)	79	40	39	190	52	138	190	52	138
Speed (MHz)	118	337	207	110	240	204	110	239	204

Generally, the hardware utilization of the Daubechies and Coiflet-based systems is almost the same. This is expected since the structure of the transmitter and receiver of these filters is closely the same. The MIWF-based system requires less hardware utilization for the same test conditions. However, among the observed parameters, the most important is the number of multipliers, since they are complex in structure and the most time and power consuming elements in the system. The MIWF-based system requires only one multiplier for filter coefficient sizes of 5 and 9 bits, whereas, the Daubechies-based system requires 7 and 12 multipliers, respectively. For 15-bit filter coefficients, the MIWF-based system requires 2 multipliers and the Daubechies and Coiflet-based systems require 24 multipliers.

The simplicity in hardware results in faster operation. Tables (2) through (4) also show the speed of operation of the implemented systems in terms of the operating frequency in MHz. The speed of the channel and system controller modules, are the same for all tested systems. They are included in the speed of the whole system given in the tables. They may limit the speed of whole system when they are slower than the transmitter or the receiver. Therefore, the speeds presented for the transmitter and the receiver are more informative. Due to its simplicity, the speed of the MIWF-based Tx is greater than that of the other systems in all tested cases. In spite of being simpler, the speed of the MIWF-based Rx is equivalent to those of the Daubechies and Coiflet-based Rxs since their multipliers are implemented to operate in parallel. However, as expected, the consumed dynamic power by the MIWF-based system is always less than that consumed by the other tested systems, as shown in Tables (2) through (4) . Finally, the system complexity comparisons show that the MIWF wavelet filters are simpler, faster, less power consuming and efficient for DWMT systems.

6. Conclusion

In this paper, we present an FPGA implementation of the elemental building block of WPM structured DWMT systems. Three types of wavelet filters are used, namely, the Daubechies, Coiflet, and the MIWF. The latter has a simple structure with respect to other wavelet filters. This simplicity enables significant reductions to the computational operations need to be implemented. The BER performance tests show the ability of the MIWF-based system to withstand sever channel and data truncation conditions, whereas, Daubechies and Coiflet-based systems encounter performance degradation. Although this relative superior performance, the MIWF-based system shows performance degradation when tested over sever channel conditions. The hardware complexity of the implemented systems is compared in terms of the number of slices, flip-flops, LUTs, and required multipliers. These comparisons show clearly the great simplicity of the MIWF with respect to the other tested wavelet filters.

Therefore, it can be concluded that a multi-band DWMT system based on the MIWF will be significantly simpler, faster, and less power consuming as compared with equivalent systems based on Daubechies and Coiflet filters. A possible future work may be the implementation of multi-band DWMT system employing the MIWF and comparing its performance and hardware complexity with DWMT systems employing other types of wavelet filters.

REFERENCES

[1] A. I. Siddiq and A. A. Kadhim, "***Design of Wavelet Filters for DWMT Systems***," in *Proc. of the International Conference on Future Communication Networks (ICFCN)*, 2012, pp. 7 – 11.

[2] A. Darji, S. Shukla, S. N. Merchant and A. N. Chandorkar, "***Hardware Efficient VLSI Architecture for 3-D Discrete Wavelet Transform***," *proc. of the 13th International Conference on VLSI Design and Embedded Systems*, 2014 , pp. 348 – 352.

[3] A. Khan and S. Baig, "***Channel Equalization for Discrete Wavelet Multitone Transceiver in Wireline Channels***," *Proc. of the 9th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, 2012, pp. 394 – 398.

[4] Mohammed Bahoura and Hassan Ezzaidi,"***FPGA-Implementation of Discrete Wavelet Transform with Application to Signal Denoising***," *Circuits, Systems, and Signal Processing*, Vol. 31, Issue:3, 2012, pp. 987-1015.

[5] Chao Wang and WoonSengGan,"***Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Packet Transform***," *IEEE Trans. on Circuitsand Systems II: Express Briefs*, Vol. 54, Issue: 5, 2007 , pp. 422 – 426.

[6] Jose Chilo and Thomas Lindblad, "***Hardware Implementation of 1D Wavelet Transform on an FPGA for Infrasound Signal Classification***," *IEEE Trans. on Nuclear Science*, Vol. 55, No. 1, Feb. 2008, pp. 9-13.

[7] N.H. Ja'afar, A. Ahmad and A. Amira, "***Distributed Arithmetic Architecture of Discrete Wavelet Transform (DWT) with Hybrid Method***," *Proc. of the IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS)*, 2013, pp. 501 – 507.

[8] K.G. Oweiss, A. Mason, Y. Suhail, A.M. Kamboh, K.E. Thomson, "***A Scalable Wavelet Transform VLSI Architecture for Real-Time Signal Processing in High-Density Intra-Cortical Implants,***" *IEEE Trans. Circuits Syst*. Vol. 54, No. 6**,** 2007, pp. 1266–1278.

[9] H. Hosseini, S.K.B.S. Yusof, N. Fisal, A. Farzamnia, " ***Wavelet Packet-Based Transceiver for Cognitive UWB applications***," *Canadian Journal of Electrical and Computer Engineering*, Vol. 37, Issue 2, 2014, pp. 59 – 64.

[10] D. Daly, C. Heneghan, A. Fagan, and M. Vetterli, "***Optimal Wavelet Packet Modulation Under Finite Complexity Constraint***,"*IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 3, 2002, pp. III-2789.

[11] M. Matthieu, M. Arndt, and JoëlLienard, "***Efficient Wavelet Packet Modulation for Wireless Communication,***" *the 3rd Advanced International Conference on Telecommunications, (AICT)*, 2007, pp. 19.

[12] A. Jamin and P. Mähönen, "***Wavelet packet modulation for wireless communications***," *Wireless Communications and Mobile Computing*, vol. 5, no. 2, 2005, pp.123-137.

AUTHOR

Omar M. Salih: received B.Sc. degree in Electronic and Control Engineering from Technical college of Kirkuk in 2005 and M.Sc. in Computer Engineering from Technical college of Mosul in 2010. He is a lecturer at the Software Engineering department at Technical college of Kirkuk since 2011.His research area of interest include digital signal and image processing and VLSI design and implementation of signal processing algorithms.