DWT/IDWT processor for power line communication system

Article · September 2012

2 authors:

Thamizharasan Viswanathan
Erode Sengunthar Engineering College
21 PUBLICATIONS **100** CITATIONS
SEE PROFILE

Kalaivani Ramanathan
Erode Sengunthar Engineering College
2 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Available online at www.elixirpublishers.com (Elixir International Journal)

Power Electronics Engineering

Elixir Power Elec. Engg. 50 (2012) 10434-10439

ISSN: 2229-712X

DWT/IDWT processor for power line communication system

R.Kalaivani and V.Thamizharasan
Erode Sengunthar Enginnering College,Tamil Nadu, India.

ARTICLE INFO

Article history:
Received: 20 March 2012;
Received in revised form:
5 September 2012;
Accepted: 12 September 2012;

Keywords
DWT,
Distributive algorithm,
Low power OFDM.

ABSTRACT

In this paper, a modified distributive arithmetic algorithm is designed and implemented on FPGA for OFDM system. In order to reduce the number of storage elements required to store the partial products, input register is split into two sections. Bottom section of register is used to access the partial products and the top section of input register is used as select line of multiplexer structure. The addition of partial products is executed in parallel to reduce the latency of the DWT/IDWT computation. The modified DA algorithm is modeled using HDL and is implemented on Spartan 3E devices. The design operates at 60 MHz and consumes power less than 0.036 W and is thus suitable for low power applications. The developed DWT/IDWT model can be integrated with QAM modulator and form a OFDM system used for power line applications.

Introduction

As the common man is being educated about latest trends in information technology, use of internet is spreading rapidly. This has created demand in many new mobile and PC based applications. Home networks are one of the major application in which through internet and communication technology various electronic equipment in a home is connected. One of the methods for a low cost and reliable home network is through PLC (Power Line Communication). PLC is a method to communicate by using an existing power lines that are already installed to supply electricity. Power Line (PL) is used as transmission channel, and high-speed communication can be realized by using the high frequency band of 2-30MHz. PLC is an excellent system from the aspect of economy and convenience, because every home have been connected to the electric cables and thus new transmission line to be installed is not needed. From the main power line an additional connection can be drawn connection connecting PLC modem to outlet (Plug & Play) or PC[1][2]. In such a background, many empirical experiments with PLC modem by are being adopted; new techniques and block diagrams have been proposed and implemented.

There are many radio systems using the high frequency band such as radio equipment for a plane and ship, ham radio, which emit strong radio wave, and so it is possible for PLC system to be interfered from their systems. Orthogonal frequency division multiplexing system is one of the most promising technologies for current and future wireless communications. It is a form of multicarrier modulation technologies where data bits are encoded to multiple sub-carriers, while being sent simultaneously. Each sub-carrier in an Orthogonal Frequency Division Multiplexing (OFDM) [3-8] system is modulated in amplitude and phase by the data bits. Modulation techniques typically used are binary phase shift keying, Quadrature Phase Shift Keying (QPSK), Quadrature Amplitude Modulation (QAM), 16-QAM, 64-QAM etc., OFDM, which is often used for high-speed PLC, is one type of digital modulation technique in which baseband signals are modulated by QAM (quadrature amplitude modulation), and signals called “sub carriers” are converted by FFT (Fast Fourier Transform) or Discrete Wavelet Transform (DWT).

In this work, we refer to the DWT-based OFDM as “DWT-OFDM”. DWT based OFDM model gives better performance compared with FFT based OFDM module. For real time implementation it is required to realize OFDM system on FPGA which support reconfigurabilty. Building blocks of OFDM are modulators (QAM) and DWT/IDWT. DWT forms one of the complex signal processing units that consumes more power, space and also has maximum delay. In this paper, efficient DWT architecture based on DA logic is designed and implemented on FPGA. Several architectures have been proposed for DWT such as convolution based, lifting based and distributive arithmetic [DA] [9-13]. Recently the lifting based architecture has been adopted by many researchers to compute DWT, as it consumes less number of multipliers and memory. For FPGA implementation, DA based DWT architectures have been adopted, as they eliminate use of multipliers. Haw-Jing Lo et al proposed a reusable Distributed Arithmetic Architecture for FIR Filtering, with 16-bit precision and 16-tap FIR filter [10]. But this design could not address the complexity of higher order filter implementation on FPGA. Chen Jing et al proposed an efficient wavelet transform on FPGA based on DA with 8 bit input and 28 words Look up Table (LUT) based implementation [12]. Memory occupied by this architecture is very large and thus is not suitable for high speed applications such as 3D-DWT. Wang Sen et al proposed DA based FIR filter design on FPGA, with 70-tap low pass filter, operating at 40MHz sampling frequency with 12–13 bit precision and this architecture was implemented on FPGA with minimum storage but latency and throughput of the design were affected [11]. Patrick Longa et al proposed an area efficient FIR filter design on FPGA's using Distributed Arithmetic, with 4 inputs LUT based FPGA implementation of DA-FIR [13].

This architecture is hardware efficient and better throughput performance and is useful for handheld applications that require low-power consumption. Most of the DA architectures strive for

Tele:
E-mail addresses: kalaitss72@yahoo.co.in

reduced memory and do not concentrate on speed and power. In this paper, we propose a modified DA based DWT OFDM unit for PLCs. Section 2 presents theoretical background of OFDM system for PLC and DWT architecture using DA algorithm. Section 3 presents the modified DA-algorithm for DWT. Section 4 presents the DWT architecture based on modified DA algorithm. Section 5 presents the results of FPGA implementation. Finally, conclusion is presented in section 6.

OFDM FOR PLCS

Figure 1 shows the block diagram of a PLC transmitter and receiver. Input data (voice or video) is scrambled and encoded. The encoded data is modulated using OFDM technique and is converted to analog data. The analog data is modulated to higher frequency and is amplified and transmitted on power line. At the receiver, the analog data is digitized using an ADC, the message is demodulated using OFDM and is decoded and unscrambled. The unscrambled data forms the message or information.

The OFDM block consists of a QAM modulator and IDWT at the transmitter and QAM demodulator and DWT at the receiver. Figure 2 shows the simulink model for OFDM transmitter and receiver. An input sine wave at low frequency (KHz) is modulated using QAM modulator with a carrier frequency (MHz). The modulated data is upsampled and is passed through the IDWT processor. The output of IDWT is transmitted through the channel and is demodulated at the receiver using the QAM demodulator and DWT. DWT/IDWT forms one of the major signal processing unit in the OFDM modulator. Next section discusses the basics of wavelet transforms.

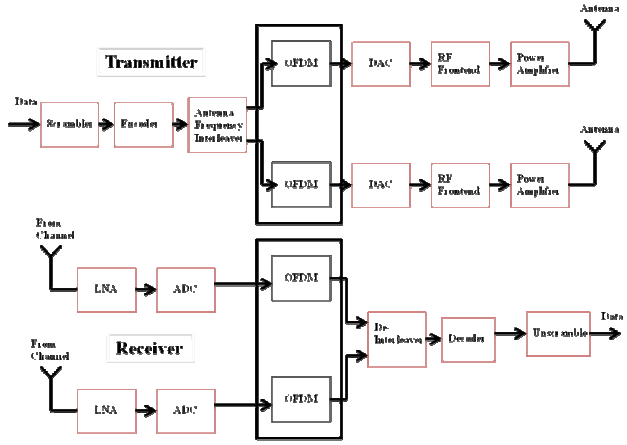

Figure 1 OFDM transmitter and receiver block diagram for PLCs

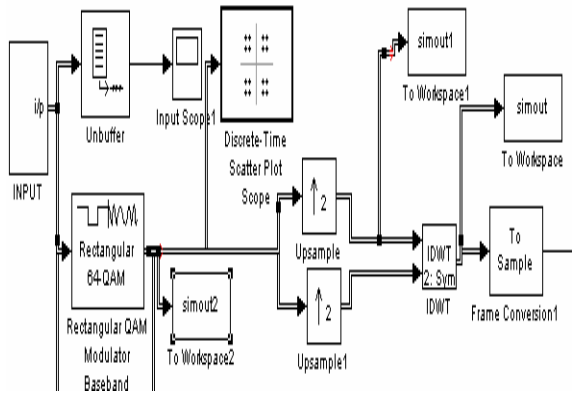

Figure 2 OFDM with QAM and DWT

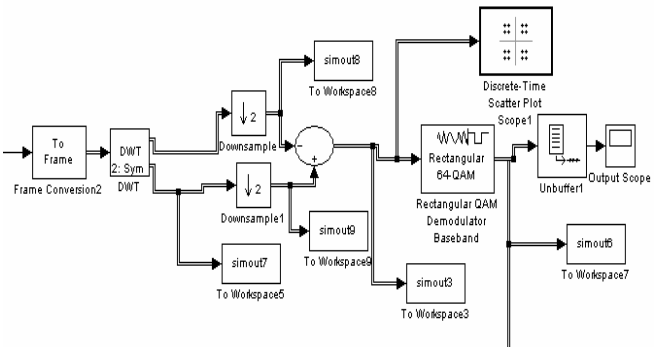

Figure 2 OFDM with QAM and DWT

DWT and IDWT

The DWT can be looked at as the multiresolution decomposition of a sequence. It takes a length N sequence z (n) as input and generates a length N sequence as the output. The output can be viewed as the multiresolution representation of z(n), and has N/2 values at the highest resolution and N/4 values at the next resolution and so on. That is the frequency resolution is low at the high frequencies and high at the low frequencies, while the time resolution is high at the higher frequencies and low at the lower frequencies. It essentially consists of multiplying the input sequence by translates and dilates of the wavelet (though this will not be obvious from the equation shown below). Let N = 2j and let the number of frequencies or resolutions, be J, considering J octaves. Therefore the frequency index varies as, 1, 2, . ., J corresponding to the scales 21, 22, . , 2j . The DWT/IDWT is given by Equation (1):

$$W(n,j) = \sum_{m=0}^{2n} W(n,j-1)\, w(2n-m)$$

$$W_H(n,j) = \sum_{m=0}^{2n} W(n,j-1)\, h(2n-m) \qquad (1)$$

w(n) and h(n) are Quadrature Mirror Filters derived from the wavelet. DWT module consist of a HPF and LPF followed by down sampling unit as shown in Figure 3 below to commute the approximation and detail coefficients of the input. The input sequence X(n) is filter using high pass and low pass filters, the output Yd(n) and Ya(n) represent the approximation and detail samples of the input signal X(n). The filter coefficients for the high pass and low pass filter of image is shown in Table 1 as recommended by OFDM system.

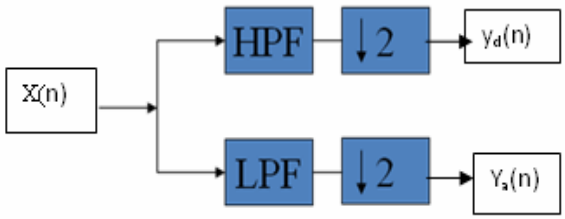

Figure 3 1D-DWT Filtering

The basic building blocks shown in Figure 3 is used as a macro that computes the high pass and low pass outputs based on the filter coefficients shown in Table 1. The basic DA-DWT architecture with 8 filter coefficients is discussed in detail in the next section.

Distributive arithmetic algorithm

FPGA architectures have LUTs for implementation of complex logic applications. Also 75% of the resources on FPGAs being LUTs, it is required to utilize the LUTs efficiently

to realize DWT architecture. DA logic is adopted for realizing FIR filters that occupy LUTs on FPGA. DWT based on DA approach have been extensively adopted for FPGA implementation. The relation between input x and output y in a FIR filter can be expressed as sum of product

$$y = \sum_{k=1}^{K} A_k x_k \qquad (2)$$

Where xk is a 2's-complement binary number scaled such that | xk |<1, Ak is fixed filter coefficients and yk output of filter in 2's-complement binary number. The input xk : {bk0, bk1, bk2……, bk(N-1) }, is represented using word length=N and bk0 is the sign bit. Thus input can be expressed as

$$x_k = -b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \qquad (3)$$

Substituting 3 in 2,

$$y = \sum_{k=1}^{K} A_k \left[-b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \right] \qquad (4)$$

Simplifying 3, gives rise to

$$y = \sum_{n=1}^{N-1} \left[\sum_{k=1}^{K} A_k b_{kn} \right] 2^{-n} + \sum_{k=1}^{K} A_k (-b_{k0}) \qquad (5)$$

Where K=Number of taps (inputs) and N is the word length of data.From (4) it is observed that the term $\left[\sum_{k=1}^{K} A_k b_{kn} \right]$ has only 2K possible values and the term $\sum_{k=1}^{K} A_k (-b_{k0})$ has only 2K possible values, thus the partial products can be stored in LUT or ROM of size 2*2K .

Figure 4 shows the hardware architecture for DA based filter design. Inputs x are used as addresses of ROM and the partial products that are computed are accessed and accumulated at the output. The partial products stored in the memory are shown in Table 2.

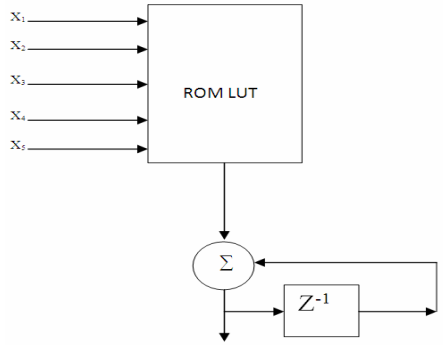

Figure 4 Hardware for DA based filter

FPGA architectures have LUTs for implementation of complex logic applications. Also 75% of the resources on FPGAs being LUTs, it is required to utilize the LUTs efficiently to realize DWT architecture. DA logic is adopted for realizing FIR filters that occupy LUTs on FPGA. DWT based on DA approach have been extensively adopted for FPGA implementation. The relation between input x and output y in a FIR filter can be expressed as sum of product

$$y = \sum_{k=1}^{K} A_k x_k \qquad (2)$$

Where x_k is a 2's-complement binary number scaled such that | x_k |<1, A_k is fixed filter coefficients and y_k output of filter in 2's-complement binary number. The input x_k : {b_{k0}, b_{k1}, b_{k2}……, $b_{k(N-1)}$ }, is represented using word length=N and b_{k0} is the sign bit. Thus input can be expressed as

$$x_k = -b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \qquad (3)$$

Substituting 3 in 2,

$$y = \sum_{k=1}^{K} A_k \left[-b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \right]$$

Design of DA based DWT

The basic DA architecture is as shown in Figure 5. With 8 input registers forming the address of the memory, 256 partial products are computed and stored in ROM. The data stored in input registers [W, V, U, T, S, R, Q, and P] each of 16 bits are serially loaded into the SISO registers. To load the set of 8 registers it requires 16x8 clock cycles. During this phase the input registers are configured as SISO. Once the data is loaded into the registers, the LSB of all the 8 registers are connected to the address bus of the LUT. The LSBs that are used as addresses enable the corresponding memory location. The data available at that location is read out and is accumulated in the adder/subtractor unit. The output obtained at every clock cycle is shifted right and is stored into the accumulator. The contents of input registers are shifted serially out, this requires 16 clock cycles. After 16 clock cycles the contents of the accumulator will consist of the final output Y(n) and the contents of SISO registers are reloaded. To compute the output sample Y(n+1), new set of input is loaded into the SISO register, this requires another 16 clock cycles. Once the new set of data is loaded the output sample Y(n+1) is computed in 16 clock cycles. Thus the latency of the network is (16*8 + 16) clock cycles and throughput is 32 clock cycles. The basic FPGA architecture consists of configurable logic blocks (CLB), each CLB consists of 4 LUTs, thus can be configured as 16x4 ROM, in order to store data of size 256x8, it required to configure 32 LUTs or 8 CLBs. Thus the basic DA architecture eliminates multipliers required to compute filter outputs, thus replacing them by ROM.

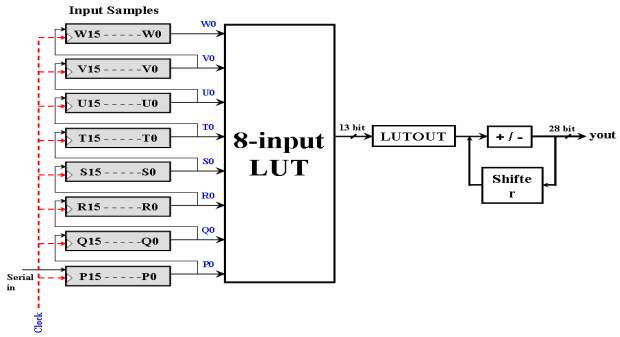

Figure 5 Basic DA architecture

The DA-DWT architecture is built using the structure shown in Figure 5. As there are 9 filter coefficients in the low pass it requires a ROM of size 512x8 (filter coefficients represented by 8 bits), and for the high pass a memory of size 128x8. The latency in computing low pass filter output is 160 (16 bit input register) clock cycles and through put of 32 clock cycles and for the high pass output, latency is 128 clock cycles and through put of 32 clock cycles. The limitations in this basic architecture are that the architecture has higher latency and also occupies more memory space (LUT). In order to reduce the

latency and increase throughput, a modified architecture is proposed.

MODIFIED DA BASED DWT

There are 9 filter coefficients for low pass and 7 filter coefficients for high pass during the analysis phase. For reconstruction, there are 7 coefficients for low pass and 9 coefficients for high pass, thus 9/7 is bi-orthogonal and is symmetric. To represent the fraction numbers shown in table 2, it requires 14 bit numbers, thus for FPGA implementation, it is required to represent the filter coefficient using fixed point or floating point number. In this work, we have used fixed point number representation. The filter coefficients are first scaled using a scaling factor of 1024 for low pass filter and a scaling factor of 256 for the high pass filter. The scaled values are rounded to the nearest integer value. The scaled and rounded number is represented using twos complement number, thus can be used to represent both signed and unsigned numbers. Table 3 presents the modified filter coefficients.

DESIGN OF MODIFIED DA-DWT ARCHITECTURE

In this examples input X = [3 6 1 7] and the filter coefficients H = [2 4 5 6]. The DA architecture for the above set of inputs is given in Figure 6.

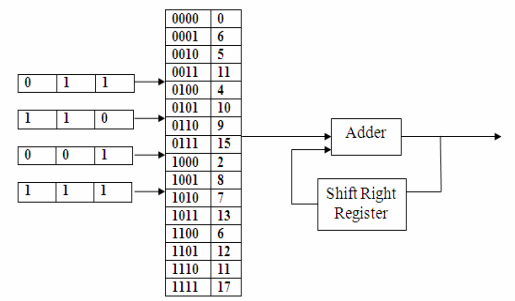

Figure 6 Basic DA algorithm architecture

SISO register is loaded with input data in 12 clock cycles, and three clock cycles is required to compute the first output sample. Thus the latency is 15 clock cycles, throughput is 6 clock cycles and ROM size is 16 x 5. In order to reduce memory size and increase throughput, the modified architecture is presented in Figure 7.

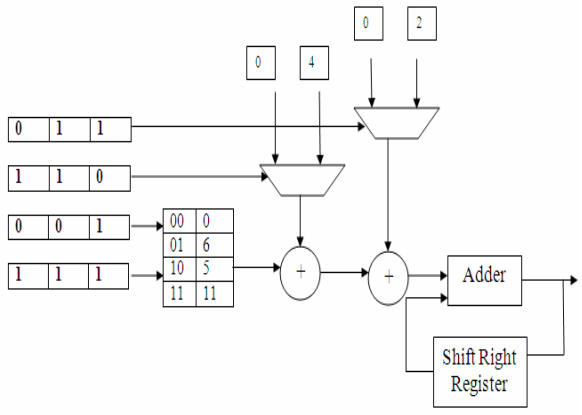

Figure 7 Modified DA algorithm architecture

The modified architecture consists of input register, LUT (4 locations), multiplexers, adders and shift register. The basic DA architecture is divided into two stages, the last two registers are used to address the LUT and the first two registers are used as select line of the multiplexer. The output of multiplexer and the ROM contents are accumulated. The final output of the two stage adder is accumulated as in the basic DA register. The latency of the modified architecture is 15 clock cycles, throughput is 6 clock cycles and ROM size is reduced to 4x5 from 16x5 (75% of reduction in memory space). The modified DA architecture based on above discussion is presented in Figure 8.

RESULTS

Table 4 shows the comparison of basic DA-DWT architecture and the modified DA-DWT architecture. The modified DA-DWT algorithm has reduced area; the ROM sizes are reduced by 93.75% and 93% for the low pass and high pass respectively.

Based on the discussions presented, the first stage of 3D DWT is designed and modeled using HDL. The design is simulated using ModelSim and Implemented using Xilinx ISE targeting Spartan IIIE FPGA. The DA output computed using the modified DA algorithm is shown in Figure 9. A known set of input pixels are processed using the modified DA architecture, the results obtained are compared with theoretical values. Thus from the results obtained, the functionality of the modified DA-DWT is verified and is used in realizing DWT algorithm.

The functionally verified HDL model is synthesized using Spartan 3E Xilinx FPGA using ISE flow. The synthesized netlist is implemented on FPGA. The FPGA results obtained for the modified DA algorithm is compared with two other DA algorithms [14-15] implemented on Spartan3E. FPGA implementations of all the three architecture have been carried out and the results are tabulated in table 5.

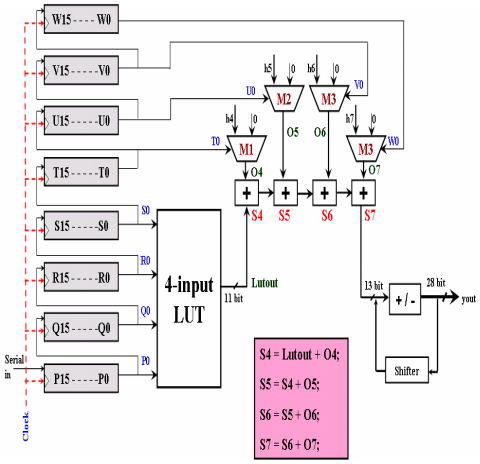

Figure 8 Modified complete DA architecture

Spartan3E (xc3s100e-5-vq100) FPGA device from Xilinx has been selected for implementation, Xilinx ISE flow is adopted for design and implementation. Form the results obtained it is found that the modified DA occupies 198 slices (No. of LUTs reduced by 38%) and power consumption is 0.036 W (Power reduced by 4%). But the delay of the proposed architecture is 16.259ns twice slower than reference design.

CONCLUSION

OFDM based communication is finding more popularity in PLCs. OFDM system consisting of QAM and DWT forms the basic signal processing blocks. DWT/IDWT is computationally intensive and also consumes more power. Thus in this work a modified DA-algorithm is developed and implemented on FPGA optimizing power and area. Modified DA-algorithm consumes total power of 0.036 W, operates at a frequency of 61 MHz. HDL models for proposed DA algorithm is modeled using HDL, and implemented on Spartan 3E FPGA using Xilinx ISE. The developed algorithm is suitable for low power applications and

is very much suitable for power line communications. QAM modulators can be modeled using HDL and integrated with DWT module to further enhance the performance of the OFDM system. Reconfigurabilty of the OFDM module can be achieved by designing suitable architectures for FPGA based on available resources on FPGA.

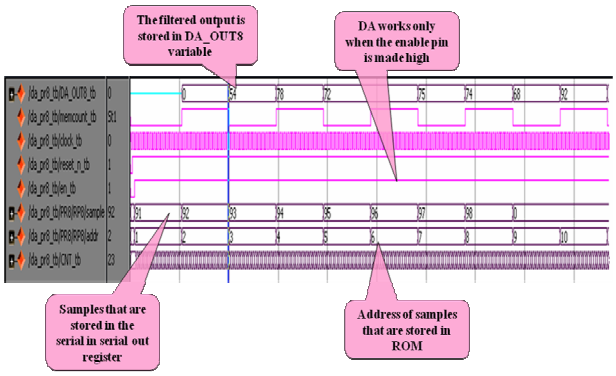

Figure 9 Simulation results of modified DA-DWT

REFERENCES

M. Tokuda: "Technical trends in high-speed power line communication (Invited)", IEICE Trans. Commun., Vol.E88-B, No.8, pp.3115-3120, 2005.

HPA (Home Plug Power Line Alliance) home page address: http://www.homeplug.org/home/

Goldberg: "Evaluation of Power Line Communication Systems", Proceedings 15th International Wroclaw Symposium on Electromagnetic Compatibility, pp.103-106, 2000.

D. Hansen: "Megabits Per Second on 50Hz Power Lines", Proceedings 15th International Wroclaw Symposium on EMC, pp.107-110, 2000.

J. C. Richards: "Characterization of Access Broadband Over Power Line(BPL) Systems by Measurements", 2005 IEEE International Symposium on EMC, Proceedings Vol.3, pp.982-987, 2005.

Jamin, and P. Mahonen, "Wavelet Packet Modulation for Wireless Communications", Wiley Wireless Communications and networking, Journal, vol. 5, no. 2, pp. 123-137, Mar. 2005.

M. K. Lakshmanan and H. Nikookar, "A Review of Wavelets for Digital Wireless Communication", Wireless Personal Communications Springer, 37: 387-420, Jan. 2006.

M. Guatier, J. Lienard, and M. Arndt, "Efficient Wavelet Packet Modulation for Wireless Communication", AICT'07 IEEE Computer Society, 2007.

W. Selesnick, "the Double Density Dual-Tree DWT", IEEE Transactions on Signal Processing, 52(5): 1304 – 1315, May 2004.

M. Lina,"Complex Daubechies Wavelets: Filter Design and Applications", ISAAC Conference, June 1997.

Haw-Jing Lo, Heejong Yoo , David V. Anderson, A reusable distributed arithmetic architecture for FIR filtering, IEEE Conference Proceedings of 51st Midwest Symposium on Circuits and Systems, pp.233-236, August 2008.

Wang Sen, Tang Bin, Zhu Jun, Distributed arithmetic for FIR filter design on FPGA, IEEE International Conference on Communications, Circuits and Systems, pp.620-623, July 2007.

Chen Jing, Hou Yuan Bin, Efficient wavelet transform on FPGA using advanced distributed arithmetic, The Eighth International Conference on Electronic Measurement and Instruments, pp.512 – 515, 2007.

Patrick Longa, Ali Miri, Area-efficient FIR filter design on FPGAs using distributed arithmetic, IEEE International Symposium on Signal Processing and Information Technology, pp.248-252, June 2006.

Heejong Yoo, David V. Anderson, Hardware-efficient distributed arithmetic architecture for high-order digital filters, IEEE International Conference on Acoustics, Speech and Signal Processing, pp.18-23, March 2005.

M. Nagabushanam, Cyril Prasanna Raj P, S. Ramachandran, Design and Implementation of Parallel and Pipelined Distributive Arithmetic Based Discrete Wavelet Transform IP Core, European Journal of Scientific Research ISSN 1450-216X Vol.35 No.3 (2009), pp.378-392.

Cyril Prasanna Raj P, Design and FPGA implementation of modified DA algorithm for 2D DWT", SASTech journal, Vol. 5, pp. 22-27, 2009

Table 1. Daubechies 9/7-tap filter coefficients

Taps	Low pass filter	Taps	High pass filter
4	0.6029490183263579	3	1.1115087052456994
3,5	0.2668641184428723	2,4	-0.5912717631142470
2,6	-0.078223266528987	1,5	-0.0575435262284995
1,7	-0.016864118442874	0,6	0.09127176311424948
0,8	0.02674875741080976		

Table 2. Look-up Table

INPUT CODE b5 b4 b3 b2 b1 b0	Memory contents
0 0 0 0 0	0
0 0 0 0 1	A1
0 0 0 1 0	A2
0 0 0 1 1	A1+A2
0 0 1 0 0	A3
0 0 1 0 1	A1+A3
0 0 1 1 0	A1+A2
0 0 1 1 1	A1+A2+A3
0 1 0 0 0	A4
0 1 0 0 1	A4+A1
0 1 0 1 0	A4+A2
0 1 0 1 1	A4+A2+A1
0 1 1 0 0	A4+A3
0 1 1 0 1	A4+A3+A1
0 1 1 1 0	A4+A3+A2
0 1 1 1 1	A4+A3+A2+A1
1 0 0 0 0	A5
1 0 0 0 1	A5+A1
1 0 0 1 0	A5+A2
1 0 0 1 1	A5+A1+A2
1 0 1 0 0	A5+A3
1 0 1 0 1	A5+A1+A3
1 0 1 1 0	A5+A1+A2
1 0 1 1 1	A5+A1+A2+A3
1 1 0 0 0	A5+A4
1 1 0 0 1	A5+A4+A1
1 1 0 1 0	A5+A4+A2
1 1 0 1 1	A5+A4+A2+A1
1 1 1 0 0	A4+A3
1 1 1 0 1	A5+A4+A3+A1
1 1 1 1 0	A5+A4+A3+A2
1 1 1 1 1	A5+A4+A3+A2+A1

TABLE 3
Modified Daubechies 9/7-tap filter coefficients

TAPS	LOW PASS FILTER	TAPS	HIGH PASS FILTER
4	617	3	284
3,5	273	2,4	-151
2,6	-80	1,5	-15
1,7	-17	0,6	23
0,8	27	-	-

TABLE 4
COMPARISON OF DA-DWT AND MODIFIED DA-DWT

Basic DA-DWT						Modified DA-DWT					
Latency		Throughput		ROM Size		Latency		Throughput		ROM Size	
High pass	Low pass	High pass	Low pass	High pass	Low pass	High pass	Low pass	High pass	Low pass	High pass	Low pass
128	160	32	32	128x8	512x8	128	160	32	32	9x8	32x8

TABLE 5
COMPARISON OF DA-DWT ARCHITECTURES IMPLEMENTED ON SPARTAN3E FPGA

	Basic DA	Reference	Proposed
Spartan3E (xc3s100e-5-vq100)			
No. of 4 input LUT's	312	358	198
No. of Flip-flops	107	134	81
Delay	11.509ns	8.608ns	16.259ns
Total power (W)	0.03717W	0.04857W	0.036W
Total Quiescent power (W)	0.03362W	0.03370W	0.033W
Total Dynamic power (W)	0.00355W	0.01487W	0.0031W