

## FPGA Realizations of Walsh Transforms for Different Transform and Word Lengths into Xilinx and Altera Chips

Zulfikar<sup>1</sup>, Shuja A. Abbasi<sup>2</sup>, Abdulrahman M. Alamoud<sup>3</sup>

<sup>1</sup>Department of Electrical and Computer Engineering, Syiah Kuala University, Indonesia

<sup>2,3</sup>Department of Electrical Engineering, King Saud University, Saudi Arabia

---

### Article Info

#### Article history:

Received Jan 25, 2018

Revised Jun 19, 2018

Accepted Jul 29, 2018

---

#### Keyword:

Hadamard transforms

Quartus

Rademacher functions

word lengths

Xilinx ISE

---

### ABSTRACT

This paper presents FPGA realizations of Walsh transforms. The realizations are targeted for the system of arbitrary waveform generation, addition/subtraction, multiplication, and processing of several signals based on Walsh transforms which is defined in term products of Rademacher functions. Input signals are passing through the system in serial, the output either signals or coefficients are also passing out in serial. To minimize the area utilization when the systems are realized in FPGA chips, the word lengths of every processing step have been designed carefully. Based on this, FPGA realizations of those various applications into Xilinx and Altera chips have been done. In Xilinx realizations, Xilinx ISE was used to display the results and to extract some critical parameters such as speed and static power. Meanwhile, the realizations into Altera chips have been conducted using Quartus. Comparisons of speed and power among Xilinx and Altera chip realizations are presented here even though this is not an apple to apple comparison. Finally, it can be concluded that Walsh transforms can be realized not only for the applications that have been done here, but it is potential can be used for other applications.

Copyright © 2018 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Zulfikar,

Department of Electrical and Computer Engineering,

Syiah Kuala University,

Jl Syech Abdul Rauf No. 7, Darussalam, Banda Aceh-23111, Indonesia.

Email: zulfikarsafrina@unsyiah.ac.id

---

## 1. INTRODUCTION

Discrete Fourier Transforms (DFT) technique for analyzing periodic digital signals already exist. However, the method is very complicated resulting in many problems during hardware implementation, and its use is to justify only with the complex systems. Walsh transforms (WT) based on Walsh functions may also be utilized to analyze the signal in the frequency domain for a particular case. It has been shown that, basically, a periodic digital signal also may be represented as a series of Walsh functions. An attempt has been made to use the concept to form a spectrum of digital signals.

Fino et al. initially proposed how to realize Walsh transforms based on addition and subtraction technique [1]. This idea attracts many scientists for developing how the Walsh transforms can be implemented in hardware. However, the method has a disadvantage such as it requires addition and subtraction of samples in word level. Later, a method of bit-level systolic arrays is developed to increase the speed of Walsh transforms [2]. Later then, Nayak et al. proposed a fully pipelined two-dimensional (2D) bit-level systolic architecture for achieving a more efficient realization [3].

Amira et al. proposed the new way of Walsh transforms realization based on Hadamard matrices that are called Fast Hadamard Transforms (FHT) [4]. A more intense works have been carried out during last two decades. For instance, the method of how Walsh functions are generated in four different orderings has also been introduced [5]. Later, Chandrasekaran et al. proposed the power analysis of Walsh transforms [6].

Then, a technique of the efficient architecture type of Walsh transforms was also developed in 2008 by Meher et al. [7]. Besides lots of other designs that have been published later.

The application technique of Walsh transforms for addition and multiplication of two digital signals was proposed earlier [8], [9]. More intensive research also has been published after that. The majority of the scientists and researchers are focussing their works on developing Walsh transforms only. However, even less, the technique for inverse Walsh transforms also have been elaborated [10]. The hardware implementation has also conducted recently for proofing the addition concept using Walsh transforms, and inverse Walsh transforms [11]. The primitive Spartan 3 has been used in the implementation, and the results were captured using a logic analyzer at 20 MHz.

Alternatively, scientists also developed algorithms of Fourier transforms by combining it with the Walsh transforms [12]-[14]. This concept is based on the simple calculation of Walsh transforms that seem to be ignored in the previous works. This algorithm such as Walsh transforms was adopted through a factorization of the intermediate transforms  $T$  for the coefficients calculation of DFT [12]. Monir et al. also then proposed the effective combination of the DFT and Walsh computations. The technique is used to perform what it called Fast Walsh Hadamard Transforms (FWHT). It was achieved by utilizing Radix-4 method [13]. Next, an efficient computing algorithm of both the Walsh transforms and the DFT transforms using the well-known Radix-2 also proposed [14].

The analysis and synthesis of periodic digital signals, after obtaining a spectrum has therefore been demonstrated. Multiple signals are also conveniently generated. Further, manipulations and processing of multiple signals from their digital spectrum have been shown [15], [16]. Therefore, there is a need to explore more on Walsh transforms realization. This paper presents several previous works of Walsh transforms realizations and some new results for a complete and comprehensive design. The realization of Walsh transforms targeted to state-of-the-art FPGAs from Xilinx and Altera. A comparative design of FPGA realization to Xilinx and Altera has been presented. The design is undergoing by exploring the properties of Walsh transforms base on products of Rademacher functions.

This paper presents the complete realization of Walsh transforms for arbitrary waveform generation (AWG), signal addition/ subtraction, multiplication of two signals and processing more than two signals. In the next section, some fundamental theories of Walsh transforms, and Walsh functions are presented. In section 3, a short and precise design of how Walsh transforms is used for realizations. The implementation of the design into FPGA is covered in section 4. Some discussions and comparisons of various results regarding the speed and static power dissipation are described here. Finally, some conclusions regarding the results are mentioned at the end of this paper.

## 2. DESIGN OF WALSH TRANSFORMS FOR FPGA REALIZATION

As has been described in the introduction, the Walsh transforms may be realized directly and it may also be implemented in terms product of Rademacher functions. The design of Walsh transforms application here is based on the second method since it more conveniently for hardware.

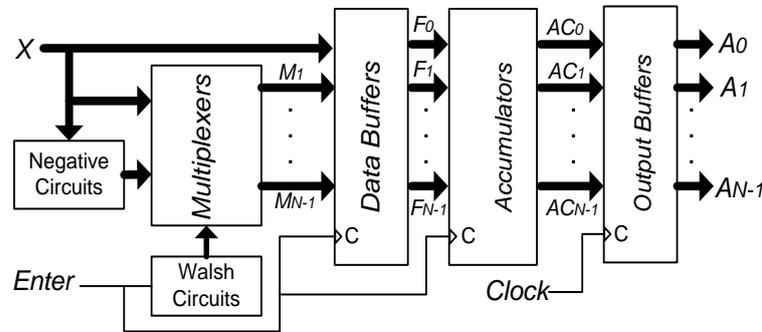
### 2.1. Design of WT and IWT

Walsh transforms conceived in terms product of Rademacher functions. Figure 1(a) shows the previously proposed of WT for transform lengths  $N$  [10]. Input data  $X$  is passed to the circuit serially, and they are controlled by Enter signal. Meanwhile, the outputs transformed coefficients  $Y$  are produced in parallel. Walsh circuit works based upon the product of Rademacher functions are used to control data buffers and accumulators. Figure 1(b) shows the proposed of Inverse Walsh transforms (IWT) for transform lengths  $N$  [10].  $N$  inputs (coefficients)  $C$  are passed into the circuit in parallel controlled Enter. Meanwhile, the outputs of  $H$  are produced in serial. Every time Enter goes high,  $C_n$  or  $-C_n$  (negative value of  $C_n$ ) will be passed to data buffers through multiplexers. At the same time, the data inside data buffers are passed to the output buffer. The multiplexers select  $C_n$  or  $-C_n$  based on output signals of the Walsh circuit. Walsh circuit control data buffers and accumulators.

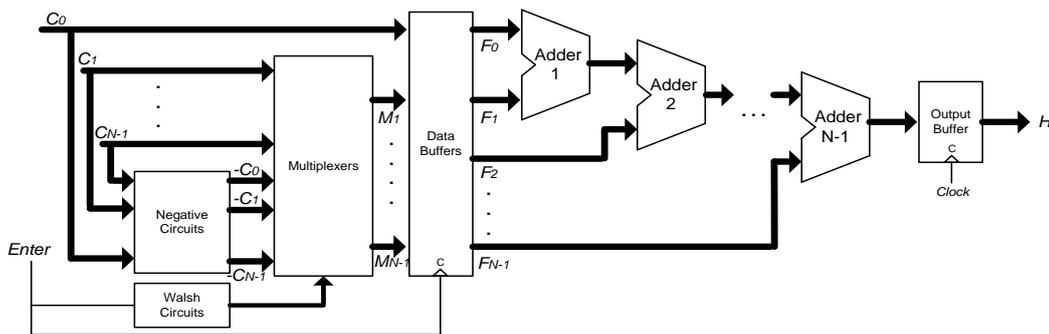
### 2.2. Walsh Transforms Applications

Walsh transforms can be applied for AWG, addition/ subtraction, processing of several signals, and multiplication system. The AWG system realized by combining WT and IWT, output results of WT becomes input for IWT. Therefore, the system can generate signal continuously [17]. The addition or subtraction system convert both input signals into the frequency domain using WT. These values are called coefficients of input signals. Then, both of the transformed signals (coefficients) are added or subtracted from each other. The result (another coefficient) is then converted back again to the time domain and consider as the output of addition or subtraction process using IWT [10].

Similarly, the application for multiplication of two signals is also performed by transforming the signals to the frequency domain (called coefficients) and by transforming back to time domain after processing. Coefficients of the first signal are multiplied by the coefficients of the second signal, resulted in another coefficient. The last coefficients are then transforming back to the time domain and consider as output.



(a)



(b)

Figure 1. Design for transform lengths  $N$  of [10]; (a) Walsh transforms; (b) Inverse Walsh transforms

### 2.3. Word Lengths Design

To reduce the circuit usage, it is required particular attention for choosing the suitable word lengths. Word lengths of the input signal are noted  $WI$ , word lengths for representing the output of Walsh transforms is denoted  $WO$  as can be calculated based on (1). Since inputs of IWT in the realizations is the output of WT, therefore the word lengths of input IWT is noted  $WIC$ . The word lengths of output inverse Walsh transforms labeled  $WOC$  on (2). This word length is to differentiate with word lengths of the input signal because, in some applications, word lengths of input and output signal are equal [10].

$$WO = WI + \log_2(N) \tag{1}$$

$$WOC = WIC - \log_2(N) \tag{2}$$

In the AWG design, the word lengths will be equal to WT and IWT since AWG is the combination of them. Word lengths of transformed signal  $WO=WIC$ , because the processed signal will be retransformed again. The word lengths of output AWG is equal to the input, so it is labeled  $WI$ . In other applications, such as addition, subtraction and multiplication, all word lengths are labeled same as the AWG application. The word length of addition or subtraction results  $WOO$  is as formulated in (3). The word lengths of multiplication result are according to (4), and its coefficient based on calculation as shown in (5).

$$WOO = WI + 1 \tag{3}$$

$$WOO = 2WI - 1 \tag{4}$$

$$WIC = \log_2(N) + 2\{(WI - 1 + \log_2(N))\} + 1 \tag{5}$$

Table 1 summarizes all word lengths required for design systems of transform lengths N and input word lengths WI or WIC (only for IWT). Those word lengths have been calculated in detail for minimizing circuit usage. A detail calculation of those formulas have been discussed explicitly, and the optimize word lengths are obtained by analyzing the behavior of word lengths characteristic using MATLAB [10].

Table 1. Word lengths design for transform lengths N and input word lengths WI or WIC [10]

System	WO	WIC = WOC	WOO
WT	WI+log2(N)	-	-
IWT	-	WIC-log2(N)	-
AWG	WI+log2(N)	WIC=WO	WI
Addition/ Subtraction	WI+log2(N)	WO+1	WI+1
Multiplication	WI+log2(N)	2{(WI-1+log2(N))+1}	2WI-1

### 3. FPGA REALIZATIONS

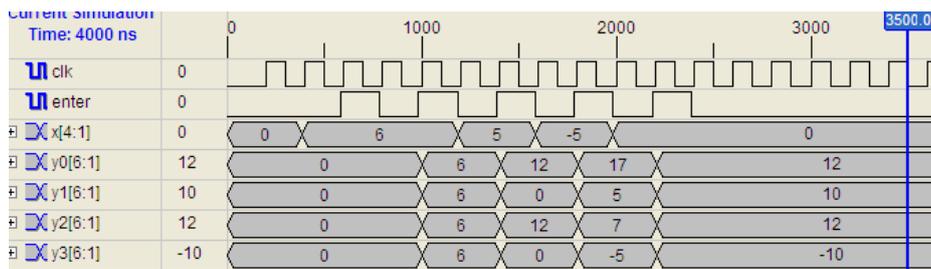
The realization will be performed and displayed for Walsh transforms, inverse Walsh transforms, arbitrary waveform generation, signal addition, signal subtraction, signal multiplication and processing several signals. The FPGA implementations are targeted to Xilinx and Altera chips. Xilinx ISE is used to simulate either behavior or timing, synthesize and estimate static power consumptions of Xilinx chips. Meanwhile, Quartus is used to simulate the design with the help of Modelsim for implementation into Altera chips.

#### 3.1. Walsh Transforms

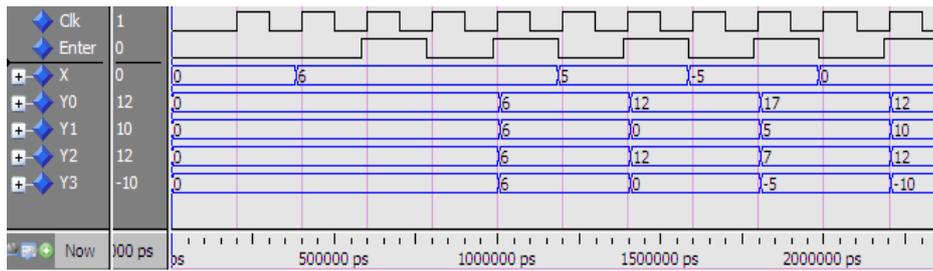
The Walsh transforms designed in section 2 has been implemented on Xilinx and Altera chips for transform lengths N=4, N=16 and input word lengths WI=4, WI=8. The input signal is passing through the system serially, but the results are arranged in parallel. Figure 2 shows Xilinx and Altera timing simulations of WT for N=4 and WI=4. In the figure, x[4:1]={6,6,5,-5} represents input signal in 4 bit sign number. Meanwhile, the output y0[6:1]=12, y1[6:1]=10, y2[6:1]=12, and y3[6:1]= -10 are results of transformation represented in 6 bit sign number. The figure also shows step by step process of updating the transformation results. For instance, the result of y1[6:1] initially is 0 before the input signal available.

Soon after the first input value available which is x[4:1]=6, the result is updated to be y1[6:1]=0+6=6. Then after the second input value x[4:1]=6 available, the result becomes y1[6:1]=0+6-6=0. After the third input x[4:1]=5 come into the system, the result will be y1[6:1]=0+6-6+5=5. Then after the last input x[4:1]=-5 entered the system, the final output is y1[6:1]=0+6-6+5-(-5)=10. These updating processes are triggered by rising edge of Enter.

Figure 3 displayed Xilinx and Altera close simulation results. Figure 3(a) views a close examination when the third input value already in the system. There is a delay (call clock to pad delay in Xilinx) about 6.4 ns from rising edge of signal Enter to output changes. Figure 3(b) views a close examination for the delay of about 8.5 ns in Altera (Quartus) implementation.

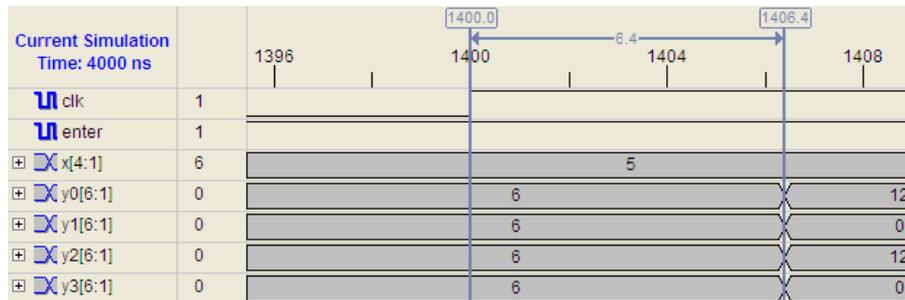


(a)

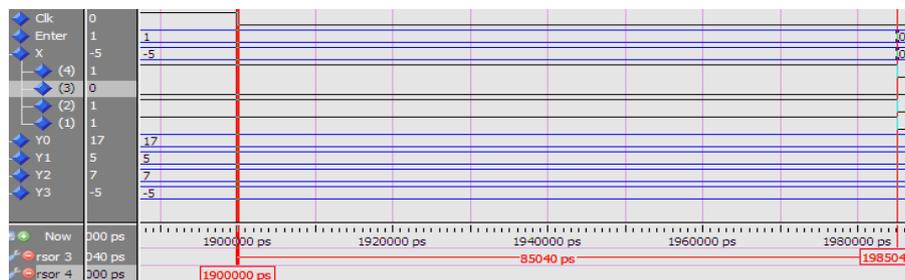


(b)

Figure 2. Timing simulation of WT for N=4 and WI=4; (a) Xilinx; (b) Altera



(a)



(b)

Figure 3. Close simulation of WT for N=4 and WI=4; (a) Xilinx; (b) Altera

Figure 4 shows Xilinx behavior simulation of WT for N=16 and WI=8 and Figure 5 views Altera timing simulation of WT for N=16 and WI=8. Those figures display the simulation of input signal  $s[8:1]$  (Xilinx) and  $x[8:1]$  (Altera) for transform lengths N=16. The input signal is represented in 8-bit sign number so that the output has to be represented at least in 12-bit sign number format according to (1).

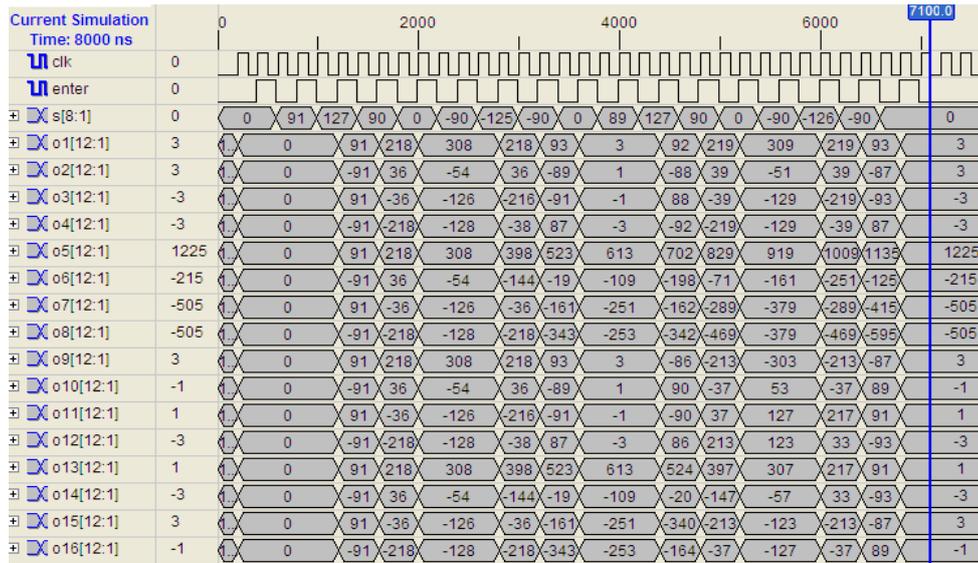


Figure 4. Xilinx behavior simulation of WT for N=16 and WI=8

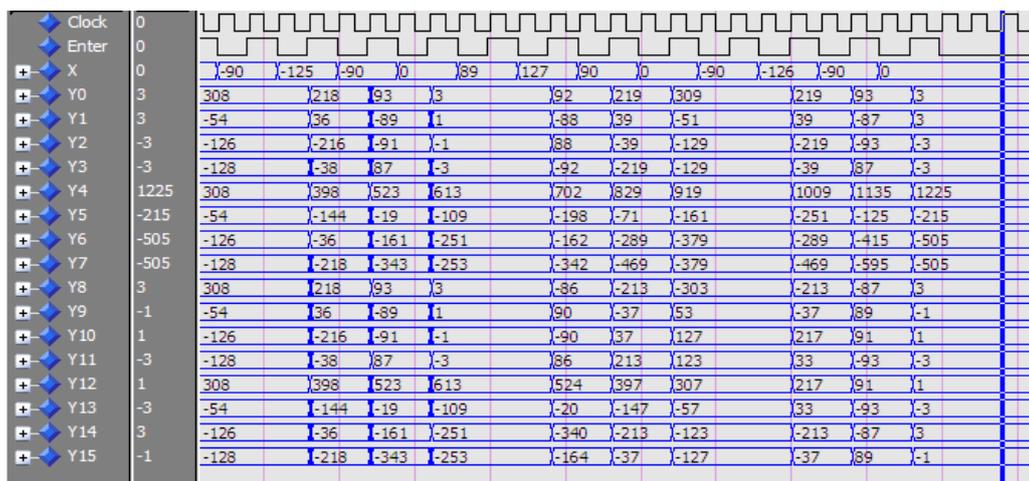
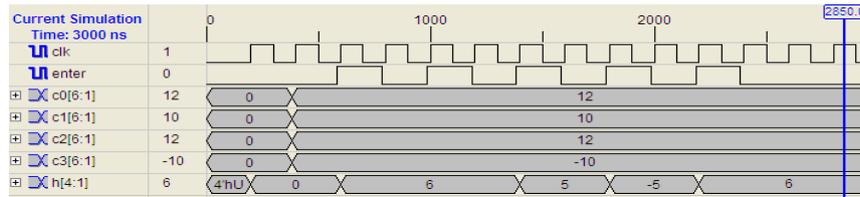


Figure 5. Altera timing simulation of WT for N=16 and WI=8

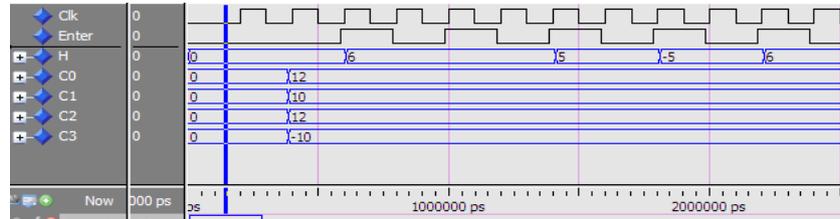
### 3.2. Inverse Walsh Transforms

Inverse Walsh transforms works opposite of the Walsh transforms. Therefore, in this realization, it will trade the output of WT as an input for IWT. Inverse Walsh transforms as designed in section 2 has been implemented on Xilinx and Altera chips for transform lengths N=4 and input word lengths WIC=6. The input signal is passing through the system in parallel, but the results are arranged in serial.

Figure 6(a) shows Xilinx behavior simulation of IWT for N=4 and WIC=6. In the figure, c0[6:1]=12, c1[6:1]=10, c2[6:1]=12, and c3[6:1]=-10 are the representation of input signal in 6 bit sign number. Meanwhile, the inverse output h[4:1]={6,6,5,-5} is represented in 4-bit sign number based on Equation (2). The figure also shows step by step process of resulting in the inverse transform results. For example, the result of h[4:1] initially is 0 (it is not considered as output result) before the input signals available and before Enter goes high. Soon after Enter goes high, the result is updated to be h[4:1]={6}. Then after the second Enter goes high, the result becomes h[4:1]={6,6}. After the third triggered Enter, the result will be h[4:1]={6,6,5}. Next time Enter triggered, the final output is h[4:1]={6,6,5,-5}. Similarly, Figure 6(b) shows the same process of inverse Walsh transforms for N=4 and WIC=6. The input value of C0, C1, C2, and C3 are passed in parallel, and the output H is gathered in serial.



(a)

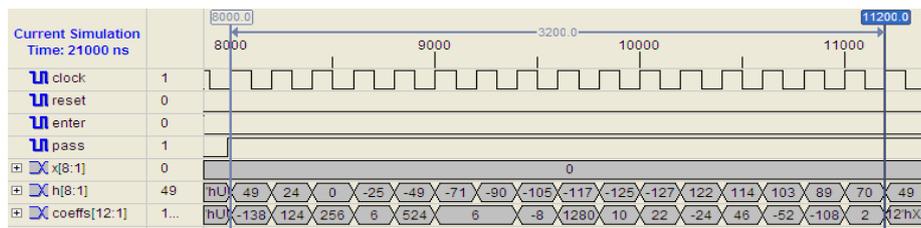


(b)

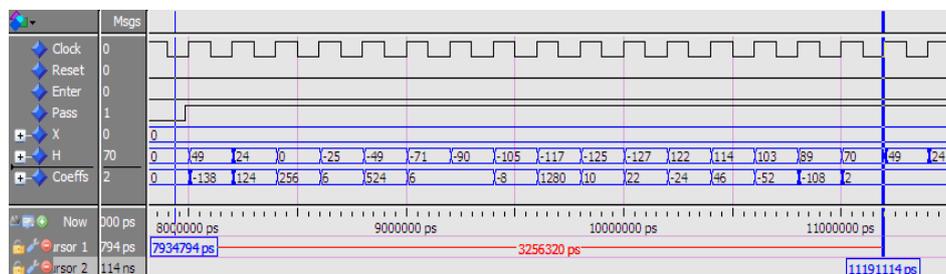
Figure 6. Simulation of IWT for N=4 and WIC=6; (a) Xilinx behavior; (b) Altera timing

### 3.3. Arbitrary Waveform Generation

Arbitrary Waveform Generation is designed by combining Walsh transforms, and inverse Walsh transforms [17]. AWG has been implemented on Xilinx and Altera chips for transform lengths N=16 and input word lengths WI=8. The input signal is passing into the system serially; the results are also in serial. Both of them are formatted to 8-bit sign number. Figure 7(a) shows Xilinx behavior simulation of AWG for N=16 and WI=8. In the figure, Reset is used to delete all previously stored values, and Pass is used for gathering output from the system. the value of input signal  $x[8:1]=\{ 49,24,0,-25,-49,-71,-90,-105,-117,-125,-127, 122,114,103,89,70\}$  is passing into the system one by one based upon the rise edge of Enter. The figure also views coefficients of the input or output signal (Coeffs[12:1]). Those values are calculated based on Equation (1). Figure 7(b) shows the similar result when it is implemented in the Altera chip.



(a)



(b)

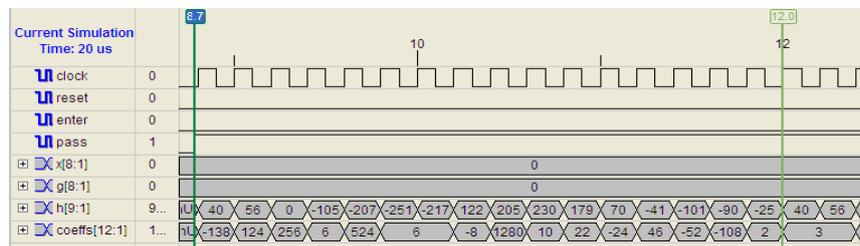
Figure 7. Simulation of AWG for N=16 and WI=8; (a) Xilinx behavior; (b) Altera timing

### 3.4. Addition System

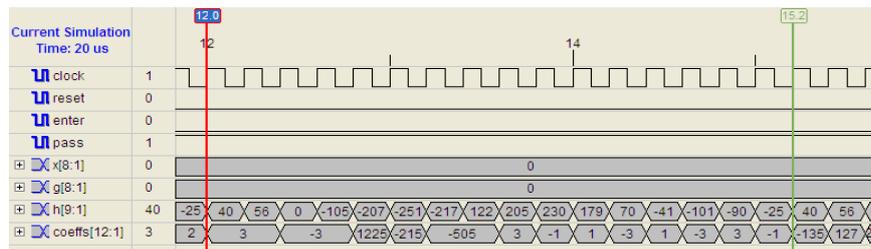
The design of addition system has been implemented on Xilinx and Altera chips for transform lengths  $N=16$  and input word lengths  $WI=8$ . Signal inputs  $x[8:1]$  and  $g[8:1]$  are passing into the system serially, the result signal  $h[9:1]$  and its coefficients are also in serial. Both of the input signals are formatted to 8-bit sign number, and the output addition result is formatted in 9-bit sign number based on Equation (3), and the coefficients of the output signal are formatted in 12-bit sign number based on calculation according to Equation (1).

Figure 8 shows Xilinx behavior simulation of addition for  $N=16$  and  $WI=8$ . Result signal of addition process  $h[8:1]$  and coefficients of signal  $x[8:1]$  are shown in Figure 8(a). Figure 8(b) shows output and the coefficients of signal  $g[8:1]$ . The coefficients of the output signal are shown in Figure 8(c). Detail values of input and output signals are listed below.

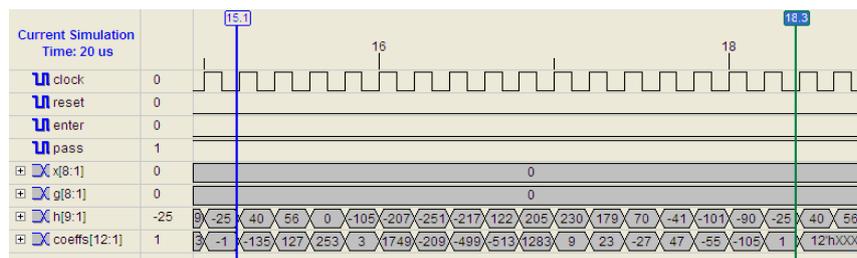
$x[8:1]=\{-49,-71,-90,-105,-117,-125,-127,122,114,103,89,70,49,24,0,-25\}$   
 $g[8:1]=\{89,127,90,0,-90,-126,-90,0,91,127,90,0,-90,-125,-90,0\}$   
 $h[9:1]=\{40,56,0,-105,-207,-251,-217,122,205,230,179,70,-41,-101,-90,-25,40,56\}$



(a)



(b)



(c)

Figure 8. Xilinx behavior simulation of addition for  $N=16$  and  $WI=8$ ; (a) outputs and coefficients of  $x[8:1]$  ; (b) outputs and coefficients of  $g[8:1]$ ; (c) outputs and coefficients of  $h[9:1]$

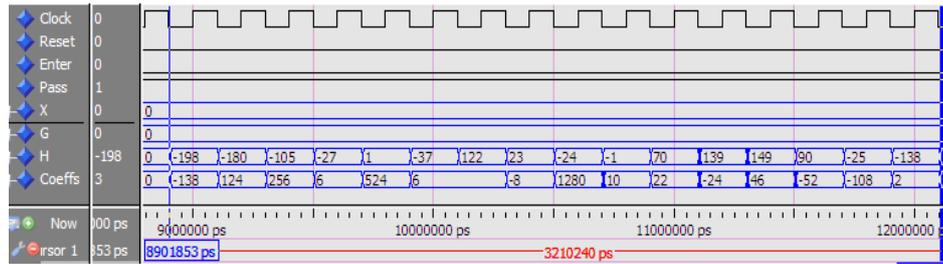
### 3.5. Subtraction System

The design of subtraction system has been implemented on Xilinx and Altera chips for transform lengths  $N=16$  and input word lengths  $WI=8$ . Input signals  $x$  and  $g$  are passing into the system serially, the result signal  $h$  and its coefficients are also passing in serial. Both of the input signals are formatted in 8-bit sign number, the output result of subtraction  $h$  is formatted in 9-bit sign number based on the calculation of

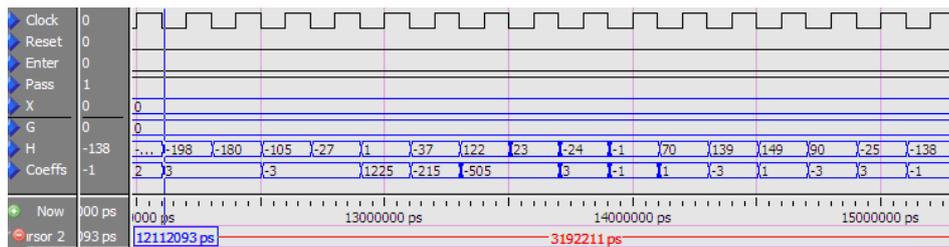
(3), and the coefficients of the output signal are formatted in 12-bit sign number based on the calculation of (1).

Figure 9 shows Altera timing simulation of subtraction system for  $N=16$  and  $WI=8$ . Result signal of subtraction  $h[8:1]$  and coefficients of signal  $x[8:1]$  are shown in Figure 9(a). Figure 9(b) shows output and the coefficients of signal  $g[8:1]$ . The coefficients of the output signal are shown in the Figure (9c). The result signal  $h$  is calculated by subtracting signal  $x$  with signal  $g$ . Detail values of input and output signals are listed below.

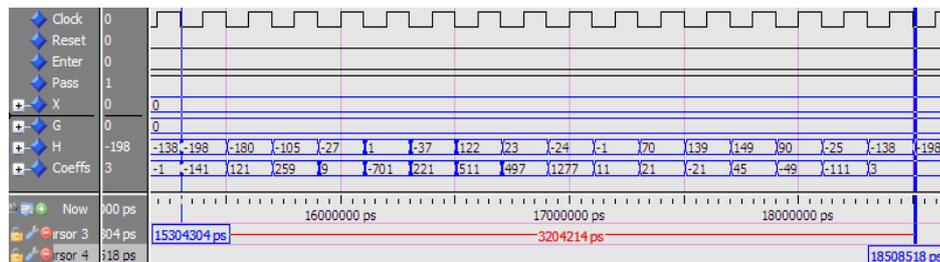
$x = \{-71, -90, -105, -117, -125, -127, 122, 114, 103, 89, 70, 49, 24, 0, -25, -49\}$   
 $g = \{127, 90, 0, -90, -126, -90, 0, 91, 127, 90, 0, -90, -125, -90, 0, 89\}$   
 $h = \{-198, -180, -105, -27, 1, -37, 122, 23, -24, -1, 70, 139, 149, 90, -25, -138, -138, -141, 121, 259, 9, -701, 221, 511, 497, 1277, 11, 21, -21, 45, -49, -111, 3, -198\}$



(a)



(b)



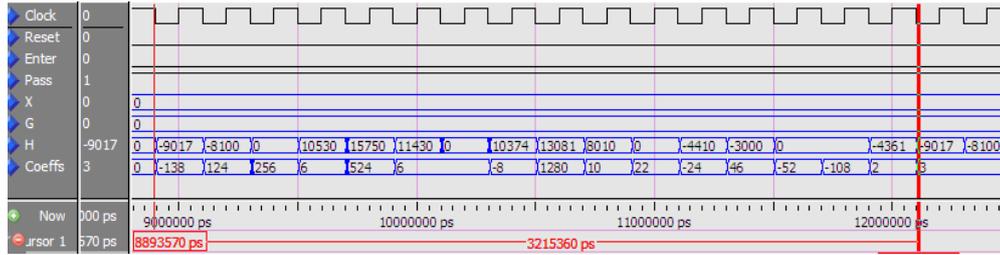
(c)

Figure 9. Altera timing simulation of subtraction for  $N=16$  and  $WI=8$ ; (a) outputs and coefficients of  $x$ ; (b) outputs and coefficients  $g$ ; (c) outputs and coefficients  $h$

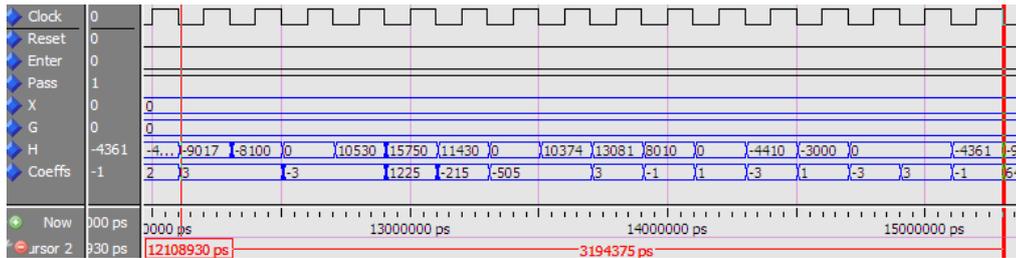
### 3.6. Multiplication System

The design multiplication system has been implemented on Xilinx and Altera chips for transform lengths  $N=16$  and input word lengths  $WI=8$ . Input signals  $x$  and  $g$  are passing into the system serially, the result signal  $h$  and its coefficients are also passing in serial. Both of the input signals are formatted in 8-bit sign number, the output result of multiplication system is formatted in 15-bit sign number according to (4), and the coefficients of the output signal are formatted in 23-bit sign number based on the calculation of (5).

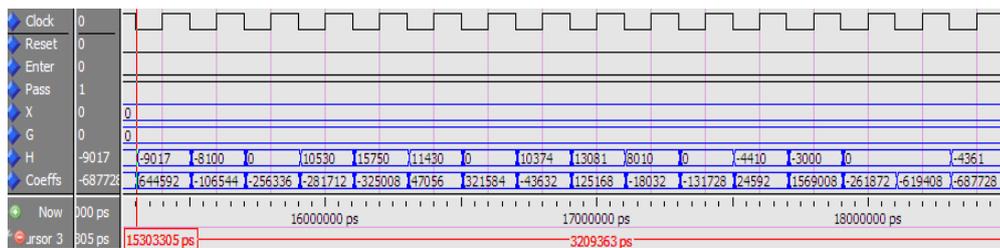
Figure 10 shows Altera timing simulation of multiplication system for  $N=16$  and  $WI=8$ . Result signal of multiplication  $h$  and coefficients of signal  $x$  are shown in Figure 10(a). Figure 10(b) shows output and coefficients of signal  $g$ . The coefficients of output signal  $h$  are shown in Figure 10(c). Detail of inputs, output and coefficient value are tabulated in Table 2.



(a)



(b)



(c)

Figure 10. Altera timing simulation of multiplication for N=16 and WI=8; (a) outputs and coefficients of x; (b) outputs and coefficients of g; (c) outputs and coefficients of h

Table 2. Signals and coefficients of multiplication system

No.	Signal			Coefficients of Signal		
	x	g	h	x	g	h
1	-71	127	-9017	-138	3	644592
2	-90	90	-8100	124	3	-106544
3	-105	0	0	256	-3	-256336
4	-117	-90	10530	6	-3	-281712
5	-125	-126	15750	524	1225	-325008
6	-127	-90	11430	6	-215	47056
7	122	0	0	6	-505	321584
8	114	91	10374	-8	-505	-43632
9	103	127	13081	1280	3	125168
10	89	90	8010	10	-1	-18032
11	70	0	0	22	1	-131728
12	49	-90	-4410	-24	-3	24592
13	24	-125	-3000	46	1	1569008
14	0	-90	0	-52	-3	-261872
15	-25	0	0	-108	3	-619408
16	-49	89	-4361	2	-1	-687728

### 3.7. Processing Several Signals

Realization of WT is also implemented for a system to process several signals. In this case, a system of  $h=x+g-j$  has been realized. Where h refers to the output signal and the rest refer to input signals. This process has been implemented into Xilinx and Altera chips for transform lengths N=4 and input word lengths

WI=4. Input signals are passing into the system serially, and the results are also passing out in serial. The word lengths of output signals will be WOO=8 bit. The system requires 2 bit more and 2 bit for processing three signals. This number is the maximum value to be preserved. However, based on the discussions in section 2 and analyzing of word lengths behavior, WOO=6 will be enough.

Figure 11 shows a realization of processing of  $h=x+g-j$ . All input signals are formatted in 4-bit sign number, the output signal  $h$  have to be at least in 6-bit format. Signal  $x[4:1]=\{-6,-2,3,7\}$ , signal  $g[4:1]=\{6,6,5,-5\}$ , and signal  $j[4:1]=\{-5,5,-7,1\}$  are passing into the system in serial based upon the rise edge of Enter. The output signal  $h[6:1]=\{5,-1,15,1\}$  will be available when Pass in high state. All of the coefficients are passing out at coeffs [6:1]. The first four numbers represent coefficients of signal  $x[4:1]$ ; the second four numbers are coefficients of signal  $g[4:1]$ ; and the last four numbers represent the coefficients of signal  $j[4:1]$ .

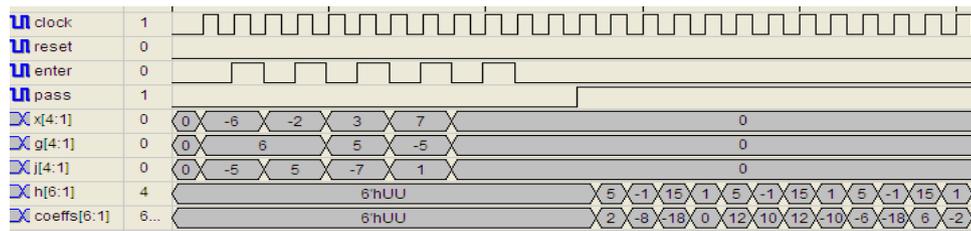


Figure 11. Xilinx behavior simulation of processing several signals  $x+g-j$  for  $N=4$  and  $WI=4$

### 3.8. Speed Comparisons

The realizations of Walsh transforms for the designed systems has been demonstrated in the previous sections into various Xilinx and Altera chips. Xilinx ISE and Quartus are the primary tools for those simulations, besides other software such as Modelsim for displaying the simulation results. To estimate the speed, the design has been synthesized for finding timing summary. For instance, the list of timing summary below is performed under Xilinx ISE using the fastest chip (speed grade: 5) of Spartan 3.

#### Timing Summary:

```

-----
Speed Grade: -5
  Minimum period: 27.140ns (Maximum Frequency: 36.846MHz)
  Minimum input arrival time before clock: 7.917ns
  Maximum output required time after clock: 6.216ns
  Maximum combinational path delay: No path found
Timing Detail:
-----
All values displayed in nanoseconds (ns)
=====
Timing constraint: Default period analysis for Clock 'Enter'
  Clock period: 4.815ns (frequency: 207.693MHz)
  Total number of paths / destination ports: 3786 / 301
-----
Delay:                4.815ns (Levels of Logic = 2)
  Source:              R1_1 (FF)
  Destination:        F7_1 (FF)
  Source Clock:       Enter rising
  Destination Clock:  Enter rising
  Data Path: R1_1 to F7_1
                    Gate Net
Cell:in->out fanout Delay Delay Logical Name (Net Name)
-----
FDC:C->Q             33   0.626 1.875 R1_1 (R1_1)
LUT2:I0->O           15   0.479 1.180 Result<1>141 (Result<1>141)
LUT4:I1->O            1   0.479 0.000 F7_mux0001<2>>1 (F7_mux0001<2>)
FDE:D                 1   0.176         F7_1
-----
Total                4.815ns (1.760ns logic, 3.055ns route)
                    (36.6% logic, 63.4% route)
    
```

It can be seen that minimum period of Clock is 27.14 ns or maximum frequency will be 36.864 MHz, with minimum input arrival time is 7.917 ns and maximum output required time after is 6.216 ns. However, the clock period of Enter is 4.815 ns, or it might reach 207.693 MHz. Most of the delay is because of routing which is about 2/3 of the total delay. To make a fair comparison of the realizations, the designs have been implemented into the Virtex-4 chip using Xilinx ISE and Stratic IV using Quartus. Table 3 views

list of speeds of Virtex-4 realization. The fastest system of transform lengths  $N=16$  is inverse Walsh transforms following by Walsh transforms which about 547 MHz and 446 MHz, respectively. The slowest process is for multiplying of two signals which are only 16 MHz. The realizations of  $N=4$  are done only to show a more clear view of how the design system is working.

Table 3. Speed comparisons on Xilinx Virtex-4

System	N	WI or WIC	Speed (MHz)
WT	4	4	561
	16	8	446
IWT	4	6	716
	16	12	547
AWG	16	8	77
Addition	16	8	38
Subtraction	16	8	38
Multiplication	16	8	16
Several Signal	4	4	67

Table 4 shows a list of speeds of Stratic IV realizations. They are almost similar to the Xilinx implementations. The fastest system when the realization performed for transform lengths  $N=16$  is Walsh transforms following by inverse Walsh transforms which is about 293 MHz and 170 MHz, respectively. The slowest process is for the system of multiplication of two signals which is only 38 MHz. The comparison of this to other designed system has been made previously for Walsh transforms [10] and AWG system [17].

Table 4. Speed comparison on Altera Stratic IV

System	N	WI or WIC	Speed (MHz)
WT	4	4	636
	16	8	293
IWT	4	6	476
	16	12	170
AWG	16	8	78
Addition	16	8	48
Subtraction	16	8	48
Multiplication	16	8	38
Several Signal	4	4	72

### 3.9. Static Power Comparisons

The realizations also have been performed to estimate the static power consumption. The design system has been synthesized (Xilinx ISE) and power analyzed (Quartus) for finding the estimation of static power consumption. For instance, the list of power summary below is performed using Xilinx ISE of the fastest chip (speed grade: 5) of Spartan 3.

Power summary:	I (mA)	P (mW)
-----		
Total estimated power consumption:		37
---		
Vccint 1.20V:	10	12
Vccaux 2.50V:	10	25
Vcco25 2.50V:	0	0
---		
Clocks:	0	0
Inputs:	0	0
Logic:	0	0
Outputs:		
Vcco25	0	0
Signals:	0	0
---		
Quiescent Vccint 1.20V:	10	12
Quiescent Vccaux 2.50V:	10	25
-----		
Thermal summary:		
-----		
Estimated junction temperature:		26C
Ambient temp:	25C	
Case temp:	26C	
Theta J-A range:	31 - 32C/W	
-----		

It can be seen that power consumption is 37 mW. This power estimation is performed using  $V_{ccint}$  1.2 V,  $V_{ccaux}$  2.5 V, Quiescent  $V_{ccint}$  1.2 V and Quiescent  $V_{ccaux}$  2.5 V under 10 mA current. The estimate is assumed to be performed under junction temperature 26 °C, ambient temperature 25 °C, case temperature 26 °C and theta J-A range between 31 – 32 °C/W.

Quartus power analyzer has been used to analyze the static power consumptions of Altera chips. Table 5 views list of power dissipations of Spartan 3 and Cyclone IV GX chips when they are implemented using various transform lengths and word lengths. There is no significant difference in the power dissipation of various Walsh transforms in both chips. Power dissipation of Spartan 3 is 56 mW for WT (N=16, WI=8) and IWT (N=16, WIC=12) realizations; the rest systems are equal which is 37 mW. Meanwhile, the achievements in the Altera Cyclone chip require power from 80.9 mW up to 121.80 mW. Again, WT and IWT systems of N=16 realizations require consuming more power than other systems. Generally, unlike speed, Cyclone IV GX consumes power twice higher than Spartan 3. However, this is not an apple to apple comparison since both chips work on a different platform system.

Table 5. Power dissipation comparison among several realizations into Xilinx and Altera chips

System	N	WI or WIC	Power (mW)	
			Spartan 3	Cyclone IV GX
WT	4	4	37	80.90
	16	8	56	121.08
IWT	4	6	37	88.94
	16	12	56	120.94
AWG	16	8	37	80.92
Addition	16	8	37	89.01
Subtraction	16	8	37	89.01
Multiplication	16	8	-	121.80
Several Signal	4	4	37	80.90

#### 4. CONCLUSION

Realizations of Walsh transforms for demonstrating AWG, addition/ subtraction, multiplication, and processing several signals systems into various FPGA chips has been done successfully. Walsh transforms realized in term product of Rademacher functions. The realizations are performed using transform lengths N=4 and N=16; higher transform lengths will be conveniently conducted later. The real system nowadays is performed using word lengths of 32 bit or 64 bit. However, in this paper, the word lengths are chosen smaller for simplicity of simulations. Walsh transforms can be realized not only by the application that has been done here, but it is potential can be used for other applications.

#### REFERENCES

- [1] B.J. Fino and V.R. Algazi, "Unified matrix treatment of the fast Walsh-Hadamard transform", *IEEE Transactions on Computers*, vol. 25, pp. 1142-1146, 1976.
- [2] L.W. Chang and M.C. Wu, "A bit level systolic array for Walsh-Hadamard transforms", *Signal Processing*, vol. 31, pp. 341-347, 1993.
- [3] S. Nayak and P. Meher, "High throughput VLSI implementation of discrete orthogonal transforms using bit-level vector-matrix multiplier", *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, pp. 655-658, 1999.
- [4] A. Amira, A. Bouridane, and P. Milligan, "A novel architecture for Walsh Hadamard transforms using distributed arithmetic principles", in *Electronics, Circuits and Systems, 2000. ICECS 2000. The 7th IEEE International Conference on*, 2000, pp. 182-185.
- [5] B. Falkowski and T. Sasao, "Unified algorithm to generate Walsh functions in four different orderings and its programmable hardware implementations", *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, pp. 819-826, 2005.
- [6] A. Amira and S. Chandrasekaran, "Power modeling and efficient FPGA implementation of FHT for signal processing", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, pp. 286-295, 2007.
- [7] P.K. Meher and J.C. Patra, "Fully-pipelined efficient architectures for FPGA realization of discrete Hadamard transform", in *Application-Specific Systems, Architectures and Processors, 2008. ASAP 2008. International Conference on*, 2008, pp. 43-48.
- [8] Z.M. Yusuf, S.A. Abbasi, and A. Alamoud, "FPGA Based Analysis and Multiplication of Digital Signals", in *Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference on*, 2010, pp. 32-36.
- [9] S.A. Abbasi and A. Alamoud, "FPGA based processing of digital signals using Walsh analysis", in *Electrical, Control and Computer Engineering (INECCE), 2011 International Conference on*, 2011, pp. 440-444.

- [10] S.A. Abbasi and A. Alamoud, "FPGA based Walsh and inverse Walsh transforms for signal processing", *Elektronika ir Elektrotechnika*, vol. 18, pp. 3-8, 2012.
- [11] Z. Zulfikar, S.A. Abbasi, and A.R.M. Alamoud, "FPGA Hardware Realization: Addition of Two Digital Signals Based on Walsh Transforms", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, pp. 2688-2697, 2016.
- [12] S. Boussakta and A. Holt, "Fast algorithm for calculation of both Walsh-Hadamard and Fourier transforms (FWFTs)", *Electronics Letters*, vol. 25, pp. 1352-1354, 1989.
- [13] M.T. Hamood and S. Boussakta, "Fast walsh-hadamard-fourier transform algorithm", *IEEE Transactions on Signal Processing*, vol. 59, pp. 5627-5631, 2011.
- [14] T. Su and F. Yu, "A family of fast hadamard-fourier transform algorithms", *IEEE Signal Processing Letters*, vol. 19, pp. 583-586, 2012.
- [15] H. Walidainy, "A novel 4-point discrete fourier transforms circuit based on product of Rademacher functions", in *Electrical Engineering and Informatics (ICEEI), 2015 International Conference on*, 2015, pp. 132-137.
- [16] Z. Zulfikar and H. Walidainy, "Design of 8-point DFT based on Rademacher Functions", *International Journal of Electrical and Computer Engineering*, vol. 6, p. 1551, 2016.
- [17] S. Zulfikar, S. Abbasi, and A. Alamoud, "Design and implementation of an improved arbitrary waveform generator based on Walsh functions", *International Journal of Physical Sciences*, vol. 7, pp. 1554-1563, 2012.

## BIOGRAPHIES OF AUTHORS



Zulfikar, he was born in Beureunuen, Aceh, Indonesia, in 1975. He received his B.E. degree in Electrical Engineering from North Sumatera University, Medan, Indonesia, the M. Sc. Degree in Electrical Engineering from King Saud University, Riyadh, Saudi Arabia, in 1999 and 2011, respectively. Currently, he is studying the Ph.D. program at the University of Malaya, Malaysia. He joined as teaching staff in the Department of Electronics at Politeknik Caltex Riau, Pekanbaru, Indonesia in 2003. He served as head of Industrial Control Laboratory, Politeknik Caltex Riau from 2003 to 2006. In 2006, he joined the Electrical Engineering Department, Syiah Kuala University. He has been appointed as head of Digital Laboratory for two successive years. His current research interests include VLSI design, System on Chips (SoC) and System for gathering renewable energy.



Shuja A. Abbasi, he was born in Amroha, India in 1950. He obtained the degrees of B.Sc. Engineering and M.Sc. Engineering in Electrical Engineering in 1970 and 1972 respectively from Aligarh Muslim University (AMU), Aligarh, India with the first position in the University. He did Ph.D. from University of Southampton, England in 1980 in Microelectronics. He joined as Assistant Professor in the Department of Electrical Engineering at Aligarh Muslim University, Aligarh, India in 1971, was promoted to the positions of Associate Professor and Professor in 1982 and 1986 respectively. He shifted to the newly created Department of Electronics Engineering at AMU as Professor in 1988. He served as Chairman, Department of Electronics Engineering, AMU from 1996 to 1999. He held many Academic/Administrative positions in the past at AMU and outside. He joined as Professor of Electronics Engineering at College of Engineering, King Saud University, Riyadh, Saudi Arabia in 1999 and is continuing there since then. He has more than 100 research publications to his credit so far. He has completed many client funded projects from various organizations. His current interests include VLSI design and technology.



Abdulrahman A. Alamoud, he was born in Onaizah, Saudi Arabia on Sept. 21, 1946. He earned his B.Sc. degree in Electrical Engineering, College of Engineering (COE) from the University of Riyadh (renamed later as KSU). He earned his M.Sc., in Microelectronics, and Ph.D., in photovoltaic solar cells, from West Virginia University, Morgantown, W.V., USA in 1974 and 1984 respectively. In June 1984, he joined the Department of Electrical Engineering, KSU and was promoted to the rank of Professor in 1999. In 1991 he took a one year leave of absence from KSU and joined the Advanced Electronics Company (AEC), Riyadh, Saudi Arabia as the Special Projects Director. In 1992 he was appointed as Director, Research Center, COE, KSU for a two-term period in June 1996. In the academic year June 1996- Sept 1997 he was a Visiting Research Associate Professor, National Renewable Energy Laboratory, Golden, Colorado, USA (July 15-Dec. 15, 97) where he worked on the development of thin films CdTe Solar Cells and characterization of materials (such as semiconductors thin films and Saudi white sand rocks) and a Visiting Research Associate Professor, VLSI Research Group, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. Worked on the design of VLSI circuits using Cadence (Mar. 9-Aug. 22, 97). He was chosen to be the Vice Dean for Administrative Affairs, COE, KSU during the period of June 1999- June 2005. His research interests are in both microelectronics, Solar Cells and Materials, and Photovoltaic Systems.