American J. of Engineering and Applied Sciences 3 (4): 670-677, 2010
ISSN 1941-7020

VHDL Modeling, Simulation and Prototyping of a Novel Arbitrary Signal Generation System

S.A. Abbasi, A.R.M. Alamoud and J.M. Al Shahrani
Department of Electrical Engineering, King Saud University,
Riyadh-11421, Saudi Arabia

Abstract: Problem statement: Arbitrary signal generators play an important role in many applications. Several different techniques utilizing both analog and digital approaches are being used for the generation of periodic signals. However, all of them suffer from many drawbacks. In this study, we present modeling, simulation and prototyping of a novel periodic arbitrary signal generation system using FPGAs. The proposed system utilizes orthogonal functions to generate a variety of periodic arbitrary signals. **Approach:** A new approach for designing arbitrary signals utilizing Walsh and Rademacher functions had been used. The design had been done using state-of-the-art high level design techniques and has been targeted to the latest available FPGA chips from Xilinx and Altera. **Results:** The simulation results demonstrated both the digital and analog versions were presented. It was found that all the signals generated showed precisely zero error and the signal generated was exactly the same as the desired one. **Conclusion:** Excellent accuracy with zero error is achieved. The designed and implemented Arbitrary signal Generation System is stand-alone and doesn't require the support of any computer hardware or software, as was needed in earlier attempts It has been concluded that virtually any periodic signal can be generated using the technique developed.

Key words: Field programmable gate arrays, hardware design languages, signal generators, Walsh analyze

INTRODUCTION

Generation of periodic arbitrary signals is a common problem faced by design and test engineers. The ability to generate arbitrary signals is central to many commercial and military applications. Arbitrary signals are used in a wide range of applications in many commercial and military fields such as radar applications, communication systems, simulation and testing, pulse generation, high-speed, low-jitter data and clock source, mixed-signal design and test, magnetic storage devices and telemetry satellite (Tie-Liang and Yu-Lin, 2001). By using arbitrary signals, engineers and scientists are able to generate unique signals that are specific to their applications. Most often, arbitrary signals are designed to simulate "real world" signals. It is possible to integrate glitches, drift, noise and other anomalies on an arbitrary signal that a device under test will encounter when it leaves the lab or manufacturing floor.

A number of techniques utilizing both analog and digital approaches are being used for the generation of arbitrary signals. These techniques range from Phase-Locked-Loop (PLL)-based techniques for very high-frequency synthesis, to dynamic programming of Digital-to-Analog Converter (DAC) outputs to generate arbitrary signals at lower frequencies. The Direct Digital Frequency Synthesizers (DDS or DDFS) are also widely used in modern communications and measurement systems.

It has been demonstrated that orthogonal functions, especially, the Rademacher and Walsh functions, in principle, may be used to synthesize any periodic signal efficiently (Qasim and Abbasi, 2006; Abbasi *et al*., 2006). This concept can be practically implemented using the state-of-the-art Field Programmable Gate Array (FPGA) technology. The current high-level design methodology may be utilized which offers many advantages including high accuracy, better speed and flexibility.

In high-level design technique, models are used to describe circuits. A model of a circuit is an abstraction that shows relevant features without associated details. In recent years there has been a trend towards using Hardware Description Language (HDL) for modeling and simulation of digital circuits/systems. Different types of HDLs such as Very High Speed Integrated

Corresponding Author: S.A. Abbasi, Department of Electrical Engineering, King Saud University, Riyadh-11421, Saudi Arabia

Circuit HDL (VHDL) and Virology exist, with different features and goals.

In this study, we present modeling, simulation and hardware implementation of a novel arbitrary signal generation system using orthogonal functions. Top-down methodology based on VHDL is adopted for the design.

MATERIALS AND METHODS

The arbitrary signal generation: All of the currently available techniques of arbitrary signal generation suffer from many serious drawbacks such as low output frequencies and a large set of the spurious signals. Further all of the techniques used so far are analog based techniques which typically require disproportionately large amounts of hardware as the signal complexity increases. Most of the techniques make use of Look Up Tables (LUTs) which are inherently noisy. None of the available techniques is thus considered entirely satisfactory (Kroupa *et al*., 2000; Cheng *et al*., 2004).

A new approach for generation of certain functions using FPGAs and orthogonal functions has been demonstrated earlier (Bin Ateeq *et al*., 2002). The orthogonal functions, in particular, the Rademacher and Walsh functions are a set of discrete valued functions that can be reduced to modulo-2 addition. These functions have received increasing attention in recent years in a variety of engineering areas such as communication, signal processing, system analyze and control (Bin Ateeq *et al*., 2002). Rademacher and Walsh functions are easy to generate and control using relatively simple hardware, taking only two amplitude values, +1 and -1. Thus they can be represented as binary signals. The Walsh functions, in turn, may be used to generate any given function/signal.

This concept can be practically implemented using the state-of-the-art Field Programmable Gate Array (FPGA) technology. The FPGAs provide a new approach to Application Specific Integrated Circuit (ASIC) implementations that feature both large scale integration and user reprogrammability. Short turn around time and low manufacturing costs has made FPGA technology popular for rapid system prototyping. The current high-level design methodology may be utilized which offers many advantages.

The theory and applications of Walsh functions and Walsh series were described in detail by Golubov *et al*. (1991). An attempt at hardware realization of Rademacher functions and Walsh functions and the generation of digital and analog sinusoidal signals was described by Bin Ateeq *et al*. (2002). The design and implementation of some more digital periodic waves using orthogonal functions was described by Qasim and Abbasi (2006).

For hardware implementation, it is first necessary to calculate expansion coefficients for the given desired signal. This required the use of computers along with software like MATLAB. In the present study, this problem is also addressed. The generation of expansion coefficients has been included in the VHDL design itself. This makes the system stand-alone without the need of any other hardware. A single prototype is thus capable of generating any arbitrary periodic signal.

The orthogonal functions: The selection and use of any orthogonal set of functions depends primarily on the type of problem under study. For instance, while some sets render rather simple and useful solutions to a certain problem, other sets give complicated and less useful forms of solution. Orthogonal functions consisting of Rademacher and Walsh functions are a set of discrete valued functions. These functions and their transforms are important analytical tools for signal processing and have wide applications in digital communication, digital image processing, statistical analyze and generation of signals. The most important factor responsible for the increased use of Walsh functions is their digital nature. As the Walsh transform matrix is purely real, with entries {-1, 1}, fast operations require fewer operations than the comparable Fast Fourier Transform (FFT). This in turn implies saving in processing time and storage allocations when using a digital signal processor. These functions are easy to generate and control using relatively simple hardware and thus readily lend itself for real time signal generation ideal for the synthesis of different periodic signals (Beauchamp, 1975).

Rademacher and Walsh functions: A Rademacher function of the nth order is defined by (Beauchamp, 1975):

$$\varphi(n+1, x) = \text{Sgn}(\sin 2\pi 2^n x),\ n = 0,1,2,\ldots 0 \le x<1 \quad (1)$$

where, $\varphi(0, x) = 1$ and the signum function Sgn(y) is defined by (Beauchamp, 1975):

$$\text{Sgn}(y) = \begin{cases} +1, & y \ge 0 \\ -1, & y < 0 \end{cases} \quad (2)$$

The definition of Rademacher functions may be extended over the whole non-negative real line by the periodicity property (Beauchamp, 1975):

$$\varphi(0, x+1) = \varphi(0, x) \quad (3)$$

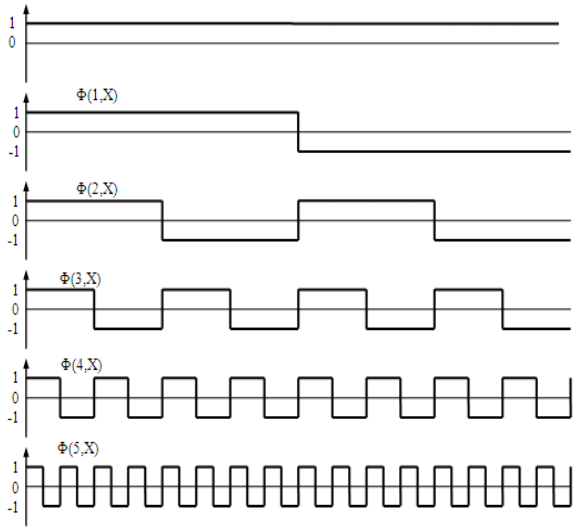

Fig. 1: The first six Rademacher functions

The first six Rademacher functions are given in Fig. 1.

Walsh functions are defined as a set of orthogonal functions that is complete over the normalized interval [0,1), each function taking only values of +1 or −1, except at a finite number of discontinuity points, where it takes the value zero (Beauchamp, 1975). Thus Walsh functions are orthogonal, normalized and complete. For orthogonality, if we multiply any two distinct functions and integrate them over the interval, the result must be zero. For normality, if two functions are one and the same, the integral of their product must be unity. Similarly, for the functions to be complete, the set of orthogonal functions can be used to approximate any given function within the defined interval as a linear combination thereof, such that its mean square error in the interval tends to zero as the number of orthogonal functions increases.

Walsh functions are defined in a number of ways. We selected a definition which results in easier implementation on FPGAs. We denote the Walsh function by ψ(n, x) defined on the interval [0,1) such that Beauchamp (1975):

$$\psi(0, x) = 1, \quad 0 \le x < 1 \tag{4}$$

and

$$\psi(n,x) = \prod_{i=0}^{N} \left[\varphi(i+1,x)\right]^{n_i}, \quad n_i \in \{0,1\} \tag{5}$$

where, the integer n is assumed to have the dyadic (binary) representation given as:

$$n = \sum_{i=0}^{N} 2^i n_i \tag{6}$$

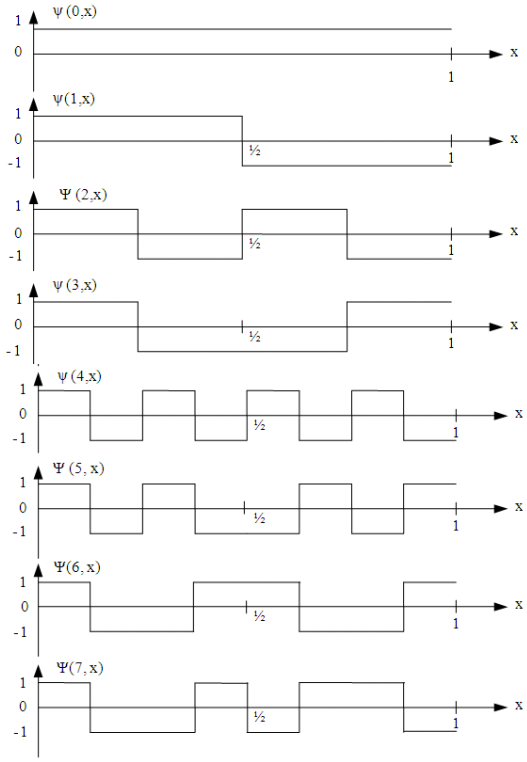

Fig. 2: The first eight Walsh functions

Generally, the set of Walsh functions can be classified into one of the five groups. The five types of orderings are: Strict Sequency ordering, dyadic Paley ordering, natural Hadamard ordering, X-ordering and reverse Gray ordering. Each order is attractive for different reasons and used for different applications (Beauchamp, 1975). The first eight Walsh functions defined in unit interval [0,1) are shown in Fig. 2. The first function ψ(0, x) has no zero crossings over the entire interval of [0,1), whereas ψ(1, x) has one zero crossing (sign changes) over the interval and so on. All the eight functions take on the values {+1,-1}. Every function starts with the value +1.

Walsh series expansion: Since Walsh functions constitute a complete set, any arbitrary function f(x) can be expressed as (Beauchamp, 1975):

$$f(x) = \sum_{n=0}^{\infty} A_n \psi(n,x) \tag{7}$$

where, A_n are the coefficients of the expansion and can be obtained by (Beauchamp, 1975):

$$A_n = \int_0^1 f(x)\psi(n,x)dx \tag{8}$$

Thus the above technique can be used to generate any periodic function/signal.

Generation of Rademacher and Walsh functions: A careful examination of the nature of Rademacher function shows that these functions may be generated using the following procedure:

- Let φ(0, x) be the function with the value "1" for the entire interval of duration T = 1, i.e., φ(0, x) = 1
- To obtain φ(1, x), divide the interval (0,1) in half and let the value of φ(1, x) in the first half interval be "+1" and in the second half of the interval "-1"
- To obtain φ(2, x), divide each of the intervals (0,1/2) and (1/2,1) in half and let the value of the function in the first half of each interval be "+1" and in the second half of the interval "-1"

The process is repeated until each interval is a single-pulse element. It has been observed that a binary counter just gives the above signals at the output of various stages. We have therefore used the VHDL code for a binary counter for the generation of Rademacher functions. We have thus eliminated the need of complicated code based upon Eq. 1. A snippet of the VHDL code is given below.

```
Rademacher: Process (Clk, Reset)
Begin
 If (Clk' event and Clk ='1') then
                If (Reset = '1') then
                  IQ <= "00000";
                Else
                  IQ<=IQ+1;
               End if;
End if;
```

We generated Walsh functions from Eq. 4 and 5. The Walsh functions of various orders are computed as products of appropriate Rademacher functions, based on the gray code conversion of the Walsh function index sequence. If we convert the ±1 amplitudes of the Walsh functions to a binary logic {0,1} representation with the conversions +1 → "0" and -1 → "1", multiplication of Rademacher functions is reduced to Exclusive-OR or modulo-2 addition operation. This greatly simplifies the procedure. A snippet of the VHDL code for the generation of Walsh functions is given below:

```
E(1) <= not IQ(4);
E(2) <= not IQ(3);
E(3) <= not (IQ(4) xor IQ(3));
E(4) <= not IQ(2);
E(5) <= not (IQ(4) xor IQ(2));
E(6) <= not (IQ(3) xor IQ(2));
E(7) <= not (IQ(4) xor IQ(3) xor IQ(2));
E(8) <= not IQ(1);
E(9) <= not (IQ(4) xor IQ(1));
E(10) <=not (IQ(3) xor IQ(1));
E(11) <=not (IQ(4) xor IQ(3) xor IQ(1));
E(12) <=not (IQ(2) xor IQ(1));
E(13) <=not (IQ(4) xor IQ(2) xor IQ(1));
E(14) <=not (IQ(3) xor IQ(2) xor IQ(1));
E(15) <=not (IQ(4) xor IQ(3) xor IQ(2) xor IQ(1));
E(16) <=not IQ(0);
E(17) <=not (IQ(4) xor IQ(0));
E(18) <=not (IQ(3) xor IQ(0));
E(19) <=not (IQ(4) xor IQ(3) xor IQ(0));
E(20) <=not (IQ(2) xor IQ(0));
E(21) <=not (IQ(4) xor IQ(2) xor IQ(0));
E(22) <=not (IQ(3) xor IQ(2) xor IQ(0));
E(23) <=not (IQ(4) xor IQ(3) xor IQ(2) xor IQ(0));
E(24) <=not (IQ(1) xor IQ(0));
E(25) <=not (IQ(4) xor IQ(1) xor IQ(0));
E(26) <=not (IQ(3) xor IQ(1) xor IQ(0));
E(27) <=not (IQ(4) xor IQ(3) xor IQ(1) xor IQ(0));
E(28) <=not (IQ(2) xor IQ(1) xor IQ(0));
E(29) <=not (IQ(4) xor IQ(2) xor IQ(1) xor IQ(0));
E(30) <=not (IQ(3) xor IQ(2) xor IQ(1) xor IQ(0));
E(31) <=not (IQ(4) xor IQ(3) xor IQ(2) xor IQ(1) xor IQ(0));
```

One of the major issues involved is the evaluation of expansion coefficients required for an arbitrary signal generation. This has been successfully achieved.

RESULTS

Arbitrary signals in digital form were generated using Rademacher and Walsh functions with a five stage process as illustrated in Fig. 3.

Simulation: A large number of arbitrary signals were generated. We present the simulation of four of them for demonstration purposes. The signals may be sampled at any number of points in the time period.

This number is used for Walsh series approximation. Larger the number of terms in the Walsh series approximation, higher is the accuracy at the cost of increased computational complexity and hardware resource utilization. A detailed theoretical study has been reported earlier (Qasim and Abbasi, 2006). It has been observed that 32-term Walsh series approximation gives very good accuracy, in many cases, zero error is obtained for the generated digital waves.

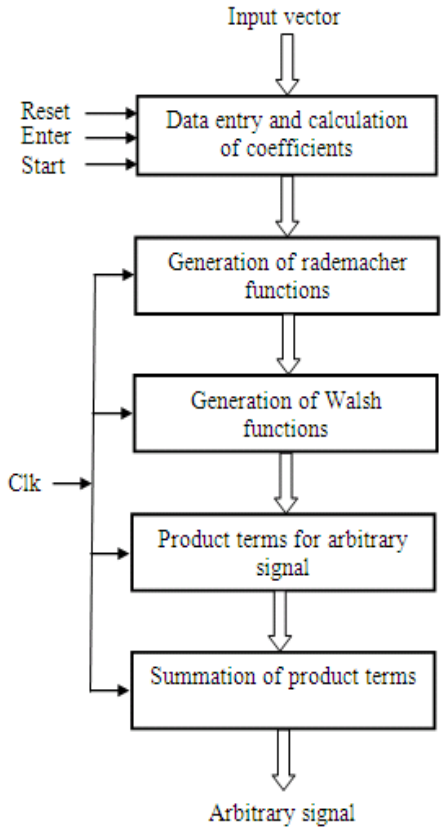

Fig. 3: Arbitrary signal generation process

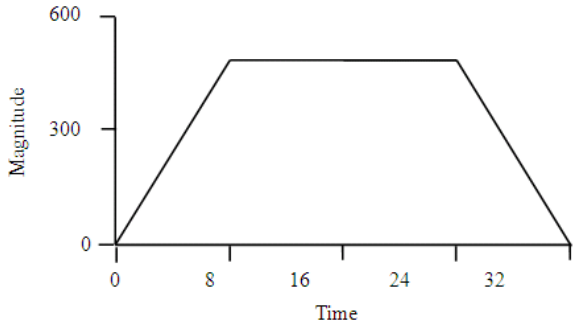

Fig. 4: Arbitrary signal no. 1

Thus 32-term Walsh series approximation with 32 samples per period has been selected for demonstration purposes. The sampled values serve as the input values to the signal generation system and are denoted by the term "desired value". The results obtained from the simulation are termed as the "obtained values". The difference between the desired and obtained values gives the error. The simulations were done using Xilinx as well as Altera EDA tools.

Simulation results of arbitrary signals: Many arbitrary signals were generated using the proposed system. The simulation results for two of them are presented. The results of signal # 1 only are presented in details.

Generation of arbitrary signal # 1: The analog form of the signal to be generated is shown in Fig. 4. It was observed that the desired and obtained values were exactly the same with zero error. The digital signal obtained from behavioral simulation is shown in Fig. 5. It should be noted that the behavioral simulation results are exactly the same with Xilinx and Altera packages since the VHDL codes are independent of technology.

Timing simulation results with Xilinx and Altera are shown in Fig. 6 and 7 respectively. A close examination of timing results from Xilinx and Altera are presented in Fig. 8 and 9 respectively in order to measure the delay from clock edge to the output signal.

Generation of arbitrary signal # 2: A signal, shown in Fig. 10, in its analog form, was generated. In this case also we found that the desired and obtained values are exactly the same at every sampling instant, thus making error zero. The simulation results, in brief are shown in Fig. 11.

Prototyping: The arbitrary signal generation system was completely designed and prototypes were developed. The design was targeted to two of the most advanced technologies-Xilinx and Altera using Xilinx ISE 9.2i and Altera Quartus II software packages respectively. In case of Xilinx, it was found that the number of gates required would fit in Spartan-3E 1600E, FPGA chip. For Altera based design, Stratix III chip was found to contain sufficient number of gates required. The simulation and implementation results are presented.

Interface design: There are many possible ways for interfacing the system. In the present study, a simple interfacing scheme has been selected, in which input samples are generated using a keyboard and passed on to the proposed arbitrary signal generation system. Each input and output was taken as a 16-bit word. This word length is quite common as it gives fairly good accuracy. However, any bit length may be selected depending upon the requirements. The proposed system has been designed in such a way that it accepts the input samples serially. This is necessary because parallel arrangement will require too many I/O pins which is not feasible.

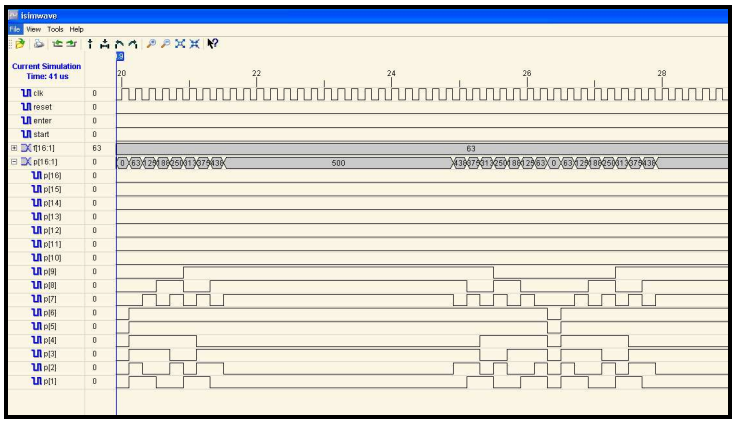

Fig. 5: Behavioral simulation results for signal no. 1

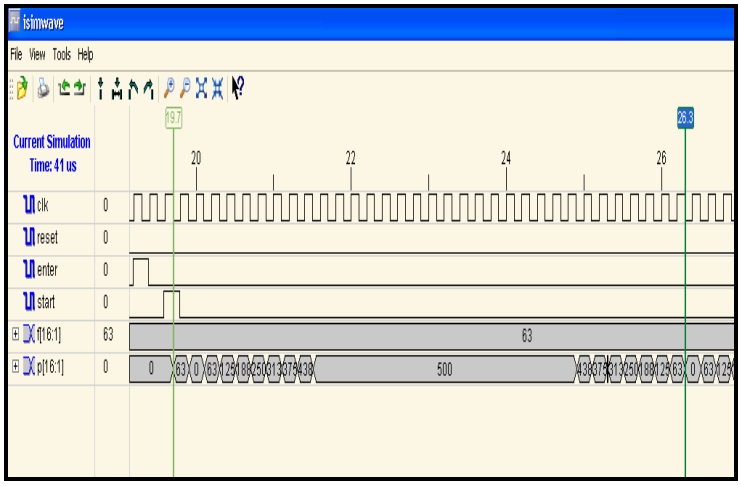

Fig. 6: Timing simulation results for signal no. 1 using Xinix

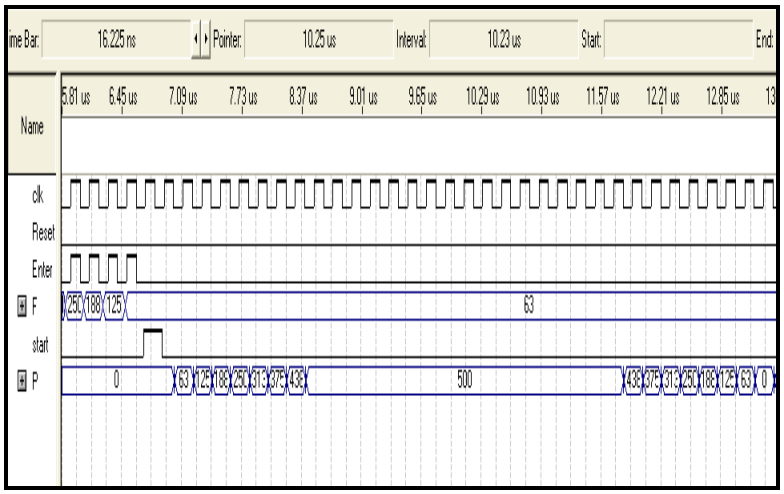

Fig. 7: Timing simulation results for signal no. 1 using Altera

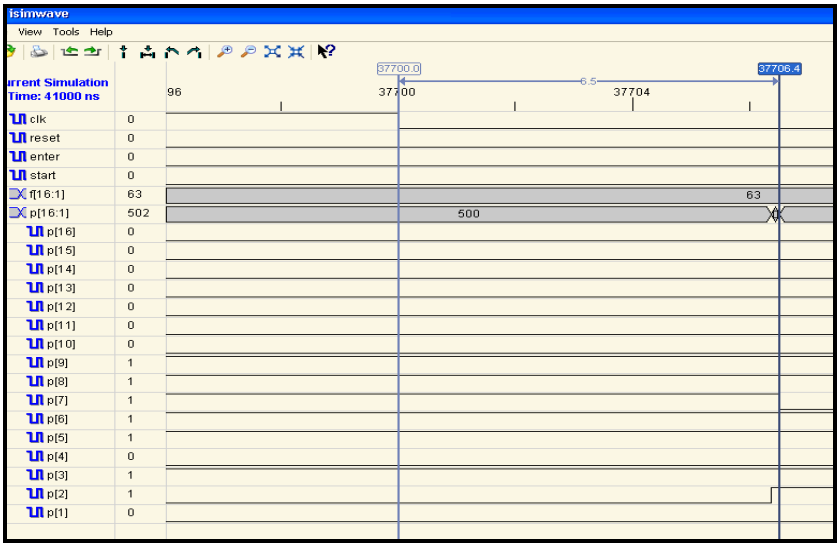

Fig. 8: Close Examination of timing results for signal no. 1 using Xilinx

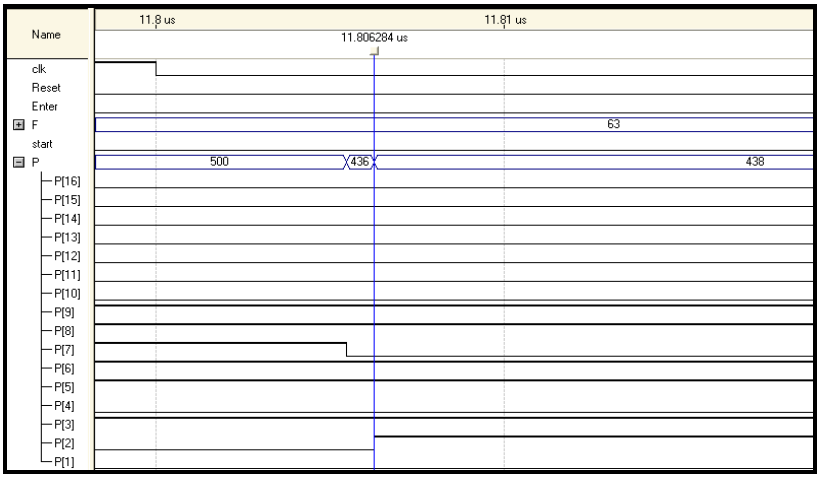

Fig. 9: Close examination of timing results for Signal #no. 1 using Altera

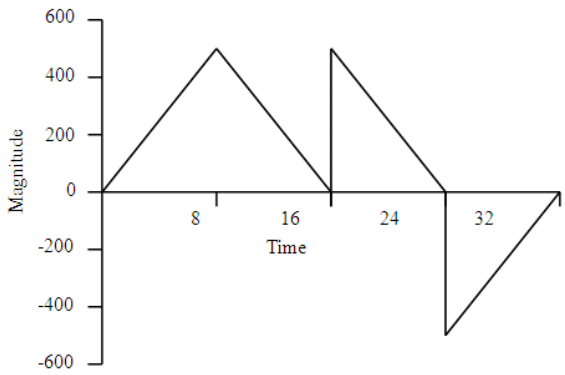

Fig. 10: Arbitrary signal no. 2

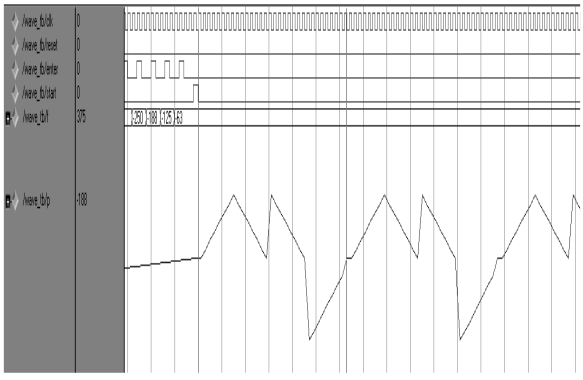

Fig. 11: Simulation results for signal no. 2

DISCUSSION

The generation of two different digital signals has been demonstrated. The signals may be sampled at any number of points in the time period. This number is used for Walsh series approximation. Larger the number of terms in Walsh series approximation, higher the accuracy at the expense of more hardware. The 32-term Walsh series approximation with 32 samples per period has been found to give fairly accurate results and has been selected for demonstration purposes. All the results showed precisely zero error and the signal generated was exactly the same as the desired one. It was observed that in the present design, the Altera devices have definite advantage over the Xilinx devices in terms of frequency of operation. However, it may be concluded that a choice between Altera and Xilinx FPGAs could only be made after a careful consideration of the requirements of the design concerned. The experience of the designer will also matter. It is also suggested that for every design both the technologies should be attempted before taking a final decision without any design prejudice.

CONCLUSION

This study describes the modeling, simulation and prototyping of a novel arbitrary signal generation system. The technique used combined the advantages of orthogonal function based realization of digital waves, ease of implementation with high level design techniques and excellent performance of the state-of-the-art FPGAs. It was observed that Rademacher and Walsh functions can be used for hardware realization of any periodic digital signal. The designed and implemented Arbitrary signal Generation System is stand-alone and doesn't require the support of any computer hardware or software, as was needed in earlier attempts (Qasim and Abbasi, 2006). FPGA based high level design technique with VHDL has been used. The design is targeted to two state-of-the-art technologies viz. the Xilinx and Altera. The EDA tools from the respective vendors have been used for the design and implementation.

ACKNOWLEDGEMENT

The researchers thankfully acknowledge the financial support provided by the Research Center, College of Engineering, King Saud University, Riyadh, Saudi Arabia under Grant No. 431/9.

REFERENCES

Abbasi, S.A., A.R.M. Alamoud and S.M. Qasim, 2006. FPGA based digital waveform generation using Walsh functions. Proceeding of the International Conference on Computer and Communication Engineering, May 9-11, UMI, Kuala Lumpur, Malaysia, pp: 1220-1225.

Beauchamp, K.G., 1975. Walsh Functions and Their Applications. 1st Edn., Academic Press Inc., New York, ISBN: 13: 978-0120840502, pp: 249.

Bin Ateeq, A.M.A., S.A. Abbasi and A.R.M. Alamoud, 2002. Hardware realization of Walsh functions and their applications using VHDL and reconfigurable logic. Proceeding of the IEEE International Conference on Microelectronics, Dec. 11-13, IEEE Xplore Press, USA., pp: 58-61. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1161496

Cheng, S., J.R. Jensen, R.E. Wallis and G.L. Weaver, 2004. Further enhancements to the analyze of spectral purity in the application of practical direct digital synthesis. Proceeding of the IEEE International Frequency Control Symposium and Exposition, Aug. 23-27, IEEE Xplore Press, USA., pp: 462-470. DOI: 10.1109/FREQ.2004.1418502

Golubov, B., A. Efimov and V. Skvortsov, 1991. Walsh Series and Transforms: Theory and Applications. 1st Edn., Kluwer Academic Publishers, USA., ISBN: 13: 978-0792311003, pp: 384.

Kroupa, V.F., V. Cızek, J. Stursa and H. Svandov, 2000. Spurious signals in direct digital frequency synthesizers due to the phase truncation. IEEE Trans. Ultrason. Ferroelect. Freq. Control, 47: 1166-1172. DOI: 10.1109/58.869062

Qasim, S.M., S.A. Abbasi, 2006. A New Approach for Arbitrary Waveform Generation using FPGA and Orthogonal Functions. Proceedings of the IEEE 6th International Workshop on System-on-Chip for Real-Time Applications, Dec. 2006, IEEE Xplore Press, Cairo, pp: 28-32. DOI: 10.1109/IWSOC.2006.348259

Tie-Liang, L. and Q. Yu-Lin, 2001. An approach to the single-chip Arbitrary Waveform Generator (AWG). Proceeding of the IEEE 4th International Conference on ASIC, Oct. 23-25, IEEE Xplore Press, Shanghai, China, pp: 506-509. DOI: 10.1109/ICASIC.2001.982611