

FPGA Based Fixed Width 4×4, 6×6, 8×8 and 12×12-Bit Multipliers using Spartan-3AN

Muhammad H. Rais and Mohamed H. Al Mijalli

King Saud University, College of Applied Medical Sciences, Biomedical Technology Department, Riyadh 11433, Saudi Arabia

Summary

In this study we investigate the Field Programmable Gate Array (FPGA) implementation of fixed width 4×4, 6×6, 8×8, and 12×12 standard and truncated multipliers using Very High speed integrated circuit Hardware Description Language. Multiplier is a core operation for digital signal processing (DSP) applications such as finite impulse response (FIR) and discrete cosine transform (DCT). The implementation of DSP algorithm requires Application Specific Integrated Circuits (ASICs). The image processing applications requires real time conditions and the algorithms should be verified and optimized before implementation which cannot be done with ASICs because they are not reconfigurable and cost is very high. The FPGA is a viable technology that could be implemented and reconfigured at the same time, since FPGA have the benefit of hardware speed and the flexibility of software. In this study we achieved remarkable reduction in FPGA resources, power and delay when the full precision of standard multiplier is not required and the truncated multiplier can be implemented with fewer resources, power and delay. The comparisons of FPGA layout show that the standard multipliers utilize lot of space as compared to truncated multipliers which could be utilized for other embedded resources.

Key words:

Digital Signal Processing (DSP), Field Programmable Gate Array (FPGA), Spartan-3AN, Truncated Multiplier VHDL.

1. Introduction

The scientific computations require intensive multiplication for signal processing (DSP) applications. Therefore, multipliers play a vital and core role in such algorithm used in computations. In digital signal processing, general purpose signal processing (GPSP) and application specific architecture for DSP the computational complexity of algorithms has increased to such extent that they require fast and efficient parallel multipliers

In particular, if the processing has to be performed under real time conditions, such algorithms have to deal with high throughput rates. In many cases implementation of DSP algorithm demands using Application Specific Integrated Circuits (ASICs). This is especially required for image processing applications such as JPEG and MPEG etc. Since development costs for ASICs are high,

algorithms should be verified and optimized before implementation.

However, with recent advancements in very large scale integration (VLSI) technology, hardware implementation has become a desirable alternative. Significant speedup in computation time can be achieved by assigning computation intensive tasks to hardware and by exploiting the parallelism in algorithms. To date, field programmable gate arrays (FPGAs) have emerged as a platform of choice for efficient hardware implementation of computation intensive algorithms [1]. FPGAs enable a high degree of parallelism and can achieve orders of magnitude speedup over general purpose processors (GPPs). This is a result of increasing embedded resources available on FPGA. FPGA have the benefit of hardware speed and the flexibility of software. The three main factors that play an important role in FPGA based design are the targeted FPGA architecture, electronic design automation (EDA) tools and design techniques employed at the algorithmic level using hardware description languages. In FPGAs, the choice of the optimum multiplier involves three key factors: area, propagation delay and reconfiguration time. Therefore, FPGA has become viable technology and an attractive alternative to ASICs [1-2].

Multiplication and squaring functions are used extensively in applications such as DSP, image processing and multimedia [3]. A full width digital $n \times n$ multiplier computes the $2n$ output as a weighted sum of partial products [4]. If the product is truncated to n -bits, the least-significant columns of the product matrix contribute little to the final result. To take advantage of this, truncated multipliers and squarers do not form all of the least-significant columns in the partial-product matrix [5]. As more columns are eliminated, the area and power consumption of the arithmetic unit are significantly reduced, and in many cases the delay also decreases. The trade-off is that truncating the multiplier matrix introduces additional error into the computation.

Other applications, which require not only a significant number of multiplication and squaring functions but also large integers, are found in the cryptography domain [6]. Achieving efficient realisation of the multiplication may have a significant impact on the specific applications in terms of speed, power dissipation and area. Many research

efforts have been presented in literature to achieve hardware efficient implementation of a truncated multiplier. The basic idea of these techniques is to discard some of the less significant partial products and to introduce a compensation circuit that partly compensates for the dropped terms, thereby reducing approximation error [7-11]. Garofalo et al [12] presented a truncated multiplier with minimum square error for every inputs' bit width. Truncated multiplication provides an efficient method for reducing the power dissipation and area of rounded parallel multiplier. High speed multiplication is desired in DSP which is normally achieved by parallel processing and pipelining, but by truncation that can be multi fold. Rais [13] also presented a study on 6×6-bit standard and truncated multipliers using FPGA device.

The rest of this paper is structured as follows. In section 2, describes the mathematical basis of truncated multiplication. Section 3 addresses the architectural platform used in this study. Section 4 presents the FPGA design and implementation results. Finally, section 5 presents the conclusion.

2. Mathematical Basis of Truncated Multipliers

Considering the multiplication of two n-bit inputs X and Y, a standard multiplier performs the following operations to obtain the 2n bit product P

$$P = XY = \sum_{i=0}^{2n-1} P_i 2^i = \left(\sum_{i=0}^{n-1} x_i 2^i \right) \left(\sum_{i=0}^{n-1} y_i 2^i \right) \quad (1)$$

where x_i , y_i and P_i represent the i th bit of X, Y and P, respectively.

Fig. 1 shows the standard architecture of 6×6-bit parallel multiplier, where HA and FA are the half and full adders respectively. Equation (1) can be expressed by the sum of two segments: the most-significant part MP and the least-significant part LP

$$P = MP + LP = \sum_{i=0}^{2n-1} P_i 2^i + \sum_{i=0}^{n-1} P_i 2^i \quad (2)$$

The standard 6×6-bit parallel multiplier can also be divided into three subsets: the most-significant part MP, input correction IC and the least-significant part LP. Equation (2) can be rewritten as follows:

$$P = MP + IC + LP \quad (3)$$

The fixed width multiplier can be obtained directly by removing the LP region and introducing the IC region to

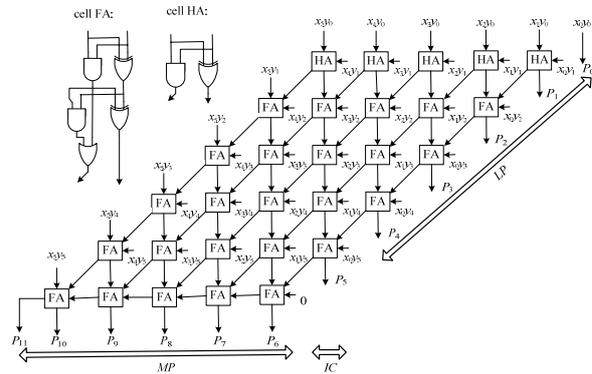


Fig. 1: The architecture of a standard 6×6-bit parallel multiplier

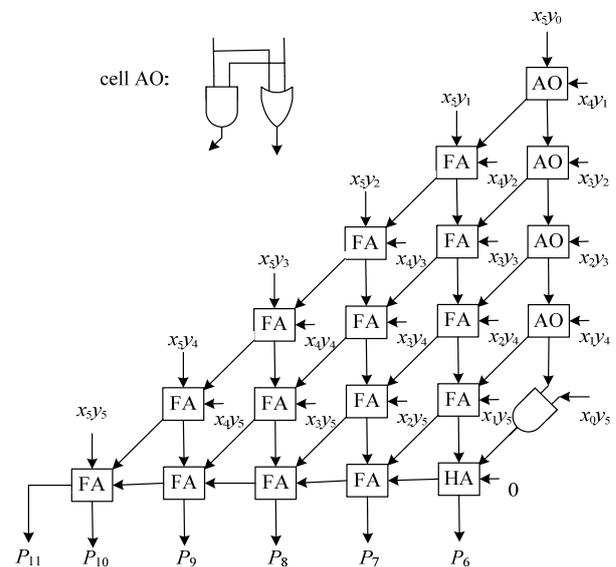


Fig. 2: The architecture of a truncated 6×6-bit parallel multiplier

obtain MP' region, which is truncated multiplier as shown in Fig. 2 and given by equation (4).

$$P = MP' + IC \quad (4)$$

3. Architecture platform

Due to the parallel nature, high frequency, and high density of modern FPGAs, they make an ideal platform for the implementation of computationally intensive and massively parallel architecture. In this section a brief introduction about state-of-the-art FPGAs from Xilinx is presented.

3.1. Spartan-3 FPGAs

The Spartan-3 FPGA belongs to the fifth generation Xilinx family. It is specifically designed to meet the needs of high volume, low unit cost electronic systems. The family consists of eight member offering densities ranging from 50,000 to five million system gates [14]. The Spartan-3 FPGA consists of five fundamental programmable functional elements: CLBs, IOBs, Block RAMs, dedicated multipliers (18×18) and digital clock managers (DCMs), Spartan-3 family includes Spartan-3L, Spartan-3E, Spartan-3A, Spartan-3A DSP, Spartan-3AN and the extended Spartan-3A FPGAs. Particularly, the Spartan-3AN is used as a target technology in this paper. Spartan-3AN combines all the feature of Spartan-3A FPGA family plus leading technology in-system flash memory for configuration and nonvolatile data storage.

4. FPGA design and implementation results

The design of standard and truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers are done using VHDL and implemented in a Xilinx Spartan-3AN XC3S700AN (package: fgg484, speed grade: -5) FPGA using the Xilinx ISE 9.2i design tool.

Figs. 3, 4, 5, and 6 show the block diagram of standard 4×4, 6×6, 8×8, and 12×12-bit multipliers. Figs. 7 and 8 illustrate the internal RTL schematic of the standard 4×4 and 6×6-bit multipliers. Fig. 9 and 10 demonstrate the part of internal RTL schematic of the standard 8×8 and 12×12-bit multipliers. FPGA layouts of the standard 4×4, 6×6, 8×8, and 12×12-bit multipliers are shown in Figs. 11, 12, 13 and 14. Figs. 15, 16, 17 and 18 present the block diagram of truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers. Figs. 19, 20 and 21 show the RTL schematic of truncated 4×4, 6×6, and 8×8-bit multipliers. Fig. 22 demonstrates the part of internal RTL schematic of the standard 12×12-bit multiplier. FPGA layouts of truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers are further shown in Figs. 23, 24, 25 and 26.

Table 1 summarizes the FPGA device resources utilization for standard and truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers. Figs. 27, 28, 29 and 30 present a snapshot of simulation waveforms for standard 4×4, 6×6, 8×8, and 12×12-bit multipliers. Figs. 31, 32, 33 and 34 illustrate a snapshot of simulation waveforms for truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers respectively.

The FPGA layouts shown in Figs. 11, 12, 13, 14, 23, 24, 25 and 26 of standard and truncated 4×4, 6×6, 8×8, and 12×12-bit multipliers also show less utilization of FPGA area for truncated multiplier, which is an indication of best utilization of the FPGA resources.

The reduction in pin delay and the number of occupied slices used in truncated multiplier also show that it is one

of the viable solutions for image processing applications, where most of the redundant information can be removed. The behavioral simulation presents the utilization of MSB as the required value in truncated multiplier, for example $5 \times 8 = (290)_{10} = 000100100010 = 000100 = (4)_{10}$ is obtained as shown in Figs. 28 and 32 of the standard and truncated multipliers simulation waveforms.



Fig.3: Block diagram of standard 4×4-bit multiplier



Fig.4: Block diagram of standard 6×6-bit multiplier



Fig.5: Block diagram of standard 8×8-bit multiplier



Fig.6: Block diagram of standard 12×12-bit multiplier

5. Conclusion

In this paper we have presented hardware design and implementation of FPGA based parallel architecture for standard and truncated multipliers utilizing VHDL. The design was implemented on Xilinx Spartan-3AN XC3S700AN FPGA device using the ISE 9.2i design tool. The objective is to present a comparative study of the 4×4, 6×6, 8×8, and 12×12-bit standard and truncated multipliers. The truncated multiplier shows much more reduction in device utilization as compared to standard multiplier. The FPGA layouts of standard and truncated multipliers also show that the less area is utilized for the resources and the

left over part could be utilized for other embedded resources. Therefore, the essence is that the truncated multipliers show significant improvement as compared to standard multipliers.

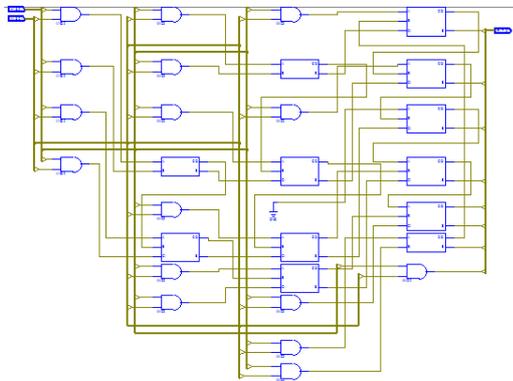


Fig.7: Complete RTL schematic of the standard 4x4-bit multiplier

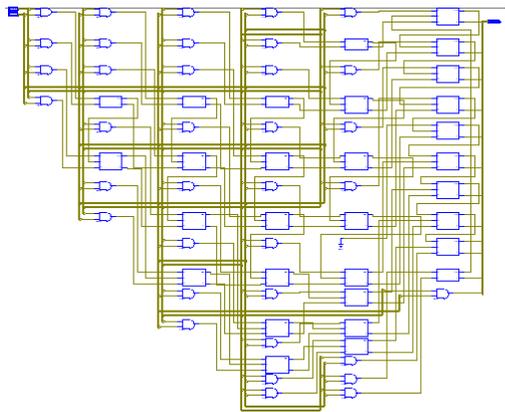


Fig.8: Complete RTL schematic of the standard 6x6-bit multiplier

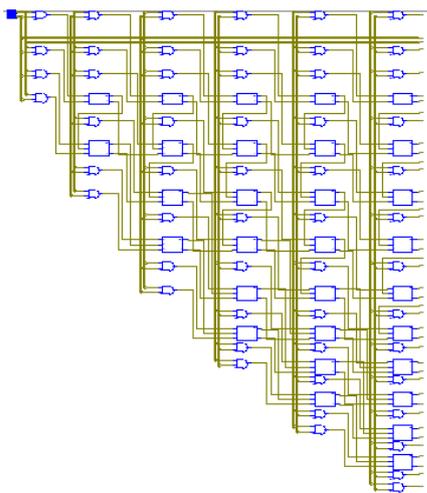


Fig.9: Part of RTL schematic of the standard 8x8-bit multiplier

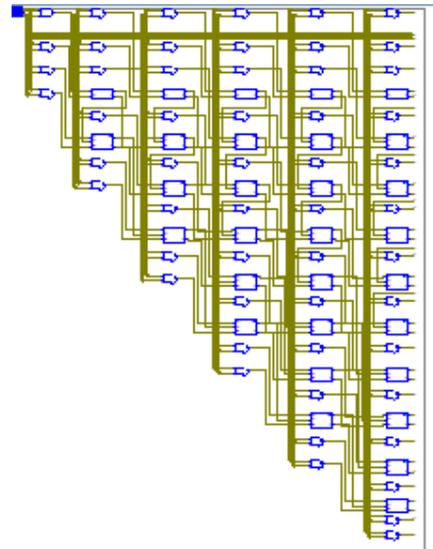


Fig.10: Part of RTL schematic of the standard 12x12-bit multiplier

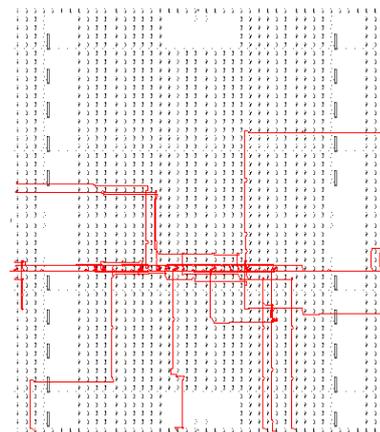


Fig.11: FPGA layout of standard 4x4-bit multiplier

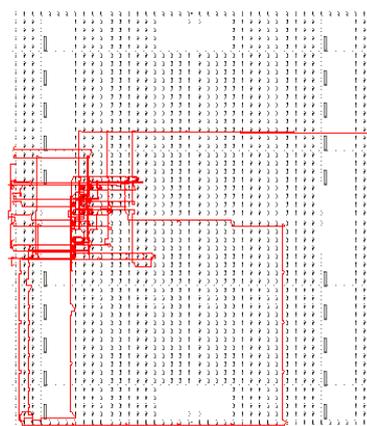


Fig.12: FPGA layout of standard 6x6-bit multiplier

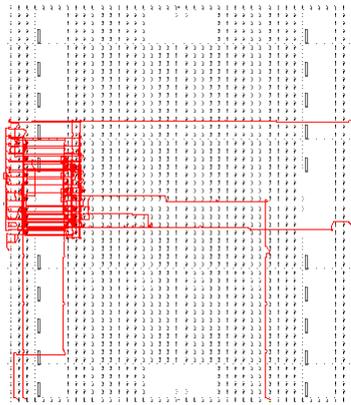


Fig.13: FPGA layout of standard 8x8-bit multiplier

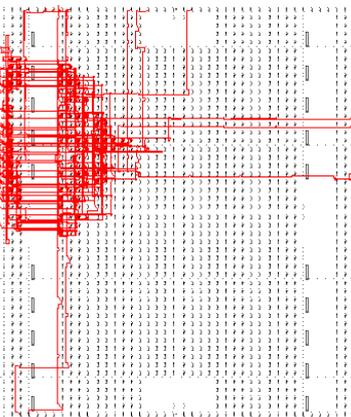


Fig.14: FPGA layout of standard 12x12-bit multiplier



Fig.15: Block diagram of truncated 4x4-bit multiplier



Fig.16: Block diagram of truncated 6x6-bit multiplier



Fig.17: Block diagram of truncated 8x8-bit multiplier



Fig.18: Block diagram of truncated 12x12-bit multiplier

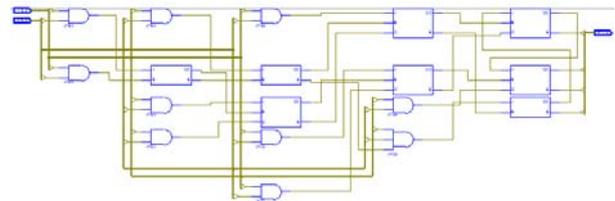


Fig.19: Complete RTL schematic of the truncated 4x4-bit multiplier

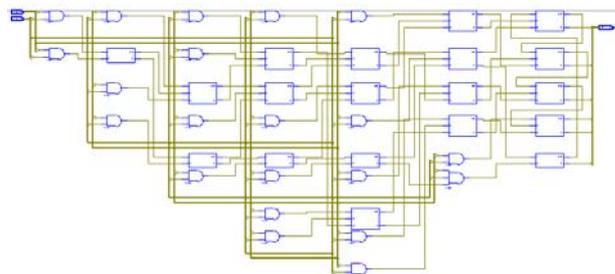


Fig.20: Complete RTL schematic of the truncated 6x6-bit multiplier

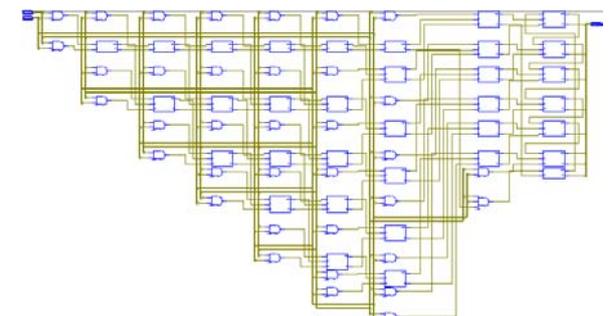


Fig.21: Complete RTL schematic of the truncated 8x8-bit multiplier

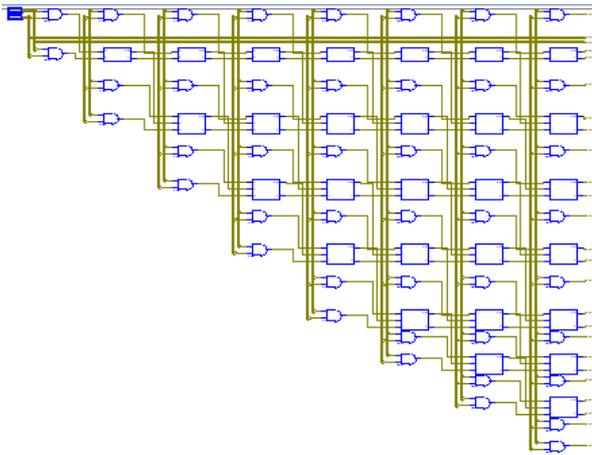


Fig.22: Part of RTL schematic of the truncated 12x12-bit multiplier

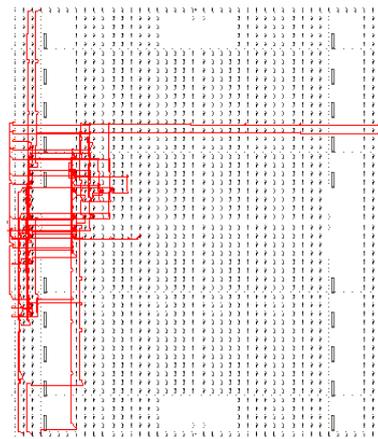


Fig.25: FPGA layout of truncated 8x8-bit multiplier

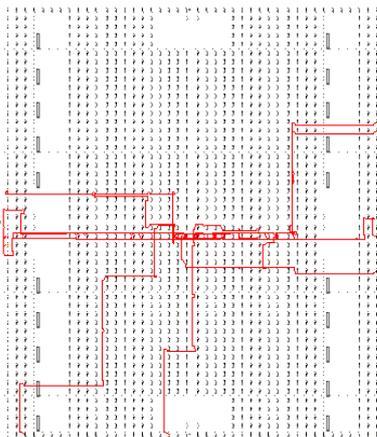


Fig.23: FPGA layout of truncated 4x4-bit multiplier

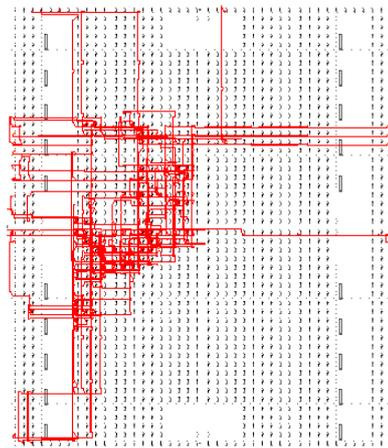


Fig.26: FPGA layout of truncated 12x12-bit multiplier

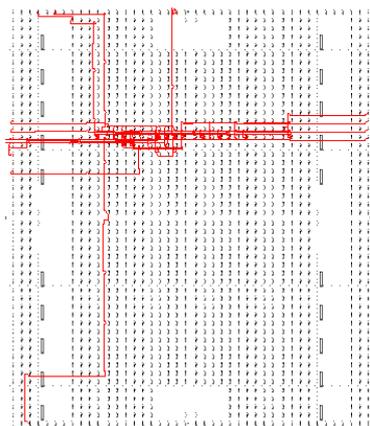


Fig.24: FPGA layout of truncated 6x6-bit multiplier

Acknowledgement

The authors acknowledge the support provided by the College of Applied Medical Sciences, King Saud University, Riyadh, Saudi Arabia.

Table 1:FPGA resource utilization for standard and truncated multiplier for Spartan-3AN XC3S700AN (package: fgg484, speed grade:-5)

Bit Width	Multipliers	Four Input LUTs (11776)	Occupied Slices (5888)	Bonded IOBs (372)	Total Equivalent Gate Count	Average Connection Delay (ns)	Maximum Pin Delay (ns)
4×4	Standard	30	16	16	180	1.421	3.598
	Truncated	18	11	12	111	1.272	2.705
6 ×6	Standard	67	36	24	402	1.238	4.873
	Truncated	43	24	18	261	1.096	2.722
8×8	Standard	121	62	32	726	1.085	3.968
	Truncated	76	40	24	456	1.072	3.641
12×12	Standard	289	148	48	1734	1.079	3.766
	Truncated	164	87	36	984	1.307	3.971

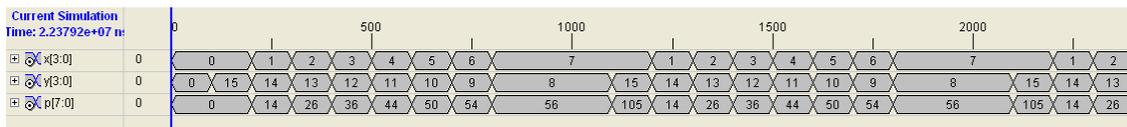


Fig.27: Simulation of standard 4×4-bit multiplier

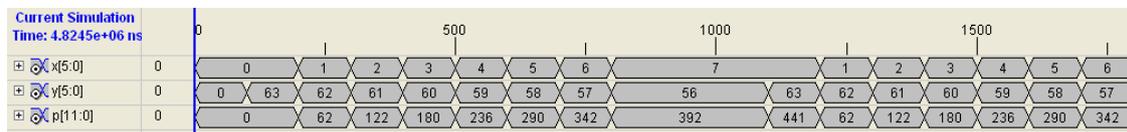


Fig.28: Simulation of standard 6×6-bit multiplier

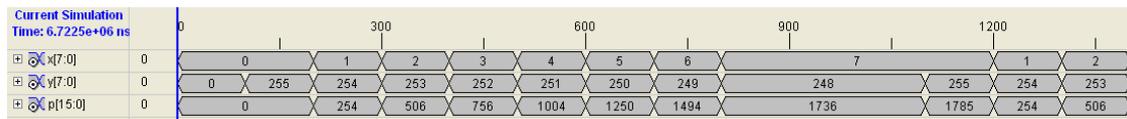


Fig.29: Simulation of standard 8×8-bit multiplier

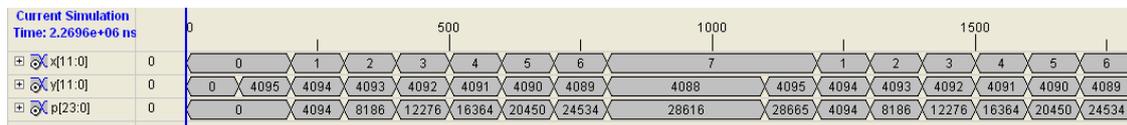


Fig.30: Simulation of standard 12×12-bit multiplier

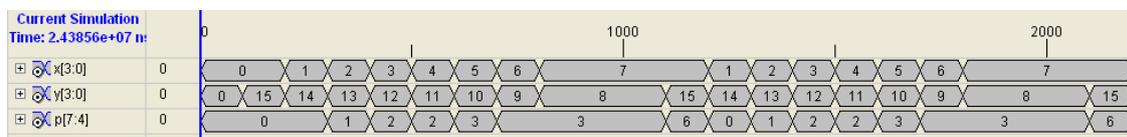


Fig.31: Simulation of truncated 4×4-bit multiplier

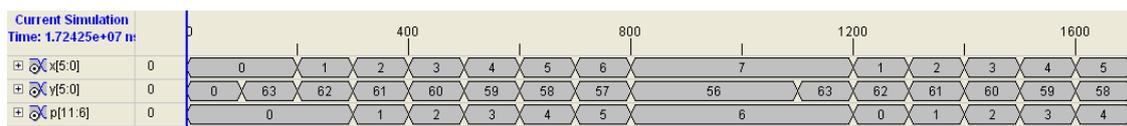


Fig.32: Simulation of truncated 6×6-bit multiplier

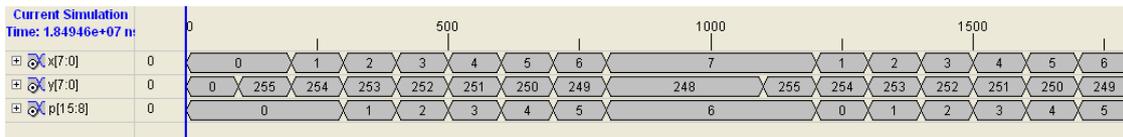


Fig.33: Simulation of truncated 8x8-bit multiplier

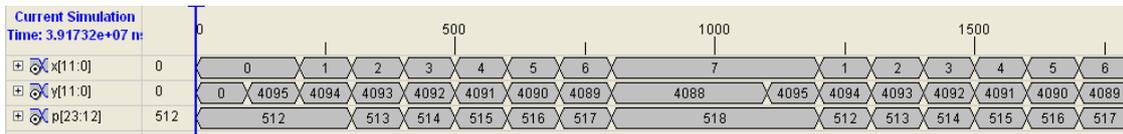


Fig.34: Simulation of truncated 12x12-bit multiplier

References

- [1] T.J. Todman, G.A. Constantinides, S.J.E. Wilton, O. Mencer, W. Luk and P.Y.K. Cheung, "Reconfigurable computing: architectures and design methods", in IEE Proc. of the Computer and Digital Techniques, 2005, Vol. 152, No. 2, pp. 193-207.
- [2] C. Maxfield, The Design Warrior's Guide to FPGAs: Devices, Tools and flows. Newnes Publishers, MA, 2004.
- [3] E.III. Walters, M.G. Arnold, and M.J. Schulte, "Using truncated multipliers in DCT and IDCT hardware accelerators", in Proc. of the XIII SPIE Advanced Signal Processing Algorithms, Architectures, and Implementations, 2003, pp. 573-584.
- [4] C.R. Baugh and B.A. Wooley, "A Two's Complement Parallel Array Multiplication Algorithm", IEEE Transactions on Computers, 1973, Vol. C-22, No. 12, pp. 1045-1047.
- [5] E.E. Swartzlander Jr., "Truncated Multiplication with Approximate Rounding", in Proc. of the 33rd Asilomar Conference on Signals, Systems, and Computers, 1999, Vol. 2, pp. 1480-1483.
- [6] W. Stallings, Cryptography and Network Security: Principles and Practices. Prentice-Hall, 4th edn., Upper Saddle River, NJ, 2006.
- [7] S.S. Kidambi, F.El-. Guibaly, and A. Antonious, "Area-Efficient Multipliers for Digital Signal Processing Applications", IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, 1996, Vol. 43, No. 2, pp. 90-95.
- [8] Y.C. Lim, "Single-Precision Multiplier with Reduced Circuit Complexity for Signal Processing Applications", IEEE Transactions on Computers, 1992, Vol. 41, No. 10, pp. 1333-1336.
- [9] J.M. Jou, S.R. Kuang and R.D. Chen, "Design of Low-Error Fixed-Width Multipliers for DSP Applications", IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, 1999, Vol. 46, No. 6, pp. 836-842.
- [10] L. Van, S. Wang, and W. Feng, "Design of the Lower Error Fixed-Width Multiplier and Its Application," IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, 2000, Vol. 47, No. 10, pp. 1112-1118.
- [11] S.R. Kuang and J.P. Wang, "Low-error configurable truncated multipliers for multiply-accumulate applications", Electronics Letters, 2006, Vol. 42, No. 16, pp. 904-905.
- [12] V.Garofalo, N.Petra, D. DeCaro, A.G.M. Strollo, and E. Napoli, "Low error truncated multipliers for DSP applications", in Proc. of the 15th IEEE International Conference on Electronics, Circuits and Systems, 2008, pp. 29-32.
- [13] M. H. Rais, "FPGA design and implementation of fixed width standard and truncated 6x6-bit multipliers: A comparative study", in Proc. of the 4th IEEE International Design and Test Workshop, IEEE Xplore Press, 2009, pp. 1-4.
- [14] Xilinx, Spartan-3 FPGA family datasheet, 2008.

Muhammad H. Rais received the Ph.D. degree in Electronics Engineering from the University of Western Australia, in 2000. He is a Researcher at Department of Biomedical Technology, King Saud University, Riyadh, Saudi Arabia. His major interest includes microelectronics, logic design, FPGA, VHDL, and characterization and modeling of semiconductor devices.

Mohamed H. Al Mijalli received the Ph.D. degree in Bioengineering, Strathclyde University, Glasgow UK. He is an Associate Professor at Department of Biomedical Technology, King Saud University, Riyadh, Saudi Arabia. His major interest includes Biomedical Instrumentation.