Periodicals of Engineering and Natural Sciences ISSN 2303-4521
Vol. 10, No. 1, January 2022, pp.188-194 *Original Research*

A new truncation algorithm of low hardware cost multiplier

Qahtan Khalaf Omran[1], Khalid Awaad Humood [2], Tahreer Mahmood[*3]
[1,2,3]Department of Electronic Engineering, College of Engineering, University of Diyala, Iraq

ABSTRACT

Multiplier is one of the most inevitable arithmetic circuit in digital signal design. Multipliers dissipate high power and occupy significant amount of the die area. In this paper, a low-error architecture design of the pre-truncated parallel multiplier is presented. The coefficients word length has been truncated to reduce the multiplier size. This truncation scaled down the gate count and shortened the critical paths of partial product array. The statistical errors of the designed multiplier are calculated for different pre-truncate values and compared. The multiplier is implemented using Stratix III, FPGA device. The post fitting report is presented in this paper, which shows a saving of 36.9 % in resources usage, and a reduction of 17 % in propagation time delay.

Keywords: Multiplier; Truncation error; DDFS; FPGA; carry save adder (CSA)

Corresponding Author:

Qahtan Khalaf Omran
Department of Electronic Engineering, College of Engineering
University of Diyala
Diyala, Iraq
Email: qahtan@uodiyala.edu.iq

1. Introduction

The researchers in [1][2] introduces a new ROM reduction technique that allows accessing the memory cells twice at one clock cycle using time sharing. As shown in Fig 1, (a) the MUX and its coefficients represents the only main source of segments initial coefficients Ci. The key feature of the presented method is to use theses coefficients to drive the slope coefficients. Two succeeding coefficients at a time has been manipulated in such a way to achieve the targeted slope coefficients. The approach introduces a good solution to eliminate the bulky ROM-based LUT. In other hand, the computational cost, which is paid in terms of utilization extra logic gates, has been raised. It seemed to be unavoidable due to incorporating the costly multiplier. So, in this work we develop a new pre-truncation of coefficients word-length, the aim is to minimize the existing multiplier size without sacrificing the design performance. The concept here is to modify the process of multiplication rather than developing an algorithm of multiplier itself. The main heart of multiplier is the multi operand adder. In designing the target multiplier, the parallel structure is adopted for inherently unique feature. In doing so, trimming down the input word-length can help to reduce the carry save adder (CSA) array significantly. It is worth to note that the proposed technique presented in this paper has been designed to fit successfully to the mentioned work in [1] but, with a little bit modification, the same design procedure can be applied for any different number of segments with similar results. The aim of our proposal design is to understand the statistical errors of the designed multiplier. Also, to simulate this design we using Stratix III, FPGA device simulation program for different scenarios with discussion the results of these statistical errors of the designed multiplier to calculate different pre-truncate values. In this paper, the generated results with high data rate can provide meaningful foundations in both electronic and telecommunication systems such as 4G LTE, 5G Massive MIMO, [3-8]. Additionally, it can also support to determine the properties and factors affecting most types of channels and antennas [9,10]. The outline of this research starts from the current introduction in which a general overview to the main objectives is produced. An Introduction to the theory of electronic truncation algorithm of hardware multiplier is offered in section one. In section two and three the methodology or theoretical simulation of proposal design form and how does it really work. In section four, the result of simulated proposal design is discussed with theory calculations of it. Finally, section five introduction Conclusion of the development work as well as the suggestions of the work.

2. Proposed approach and mathematical model

For an $n\times m$ multiplier, the $n\times m$ partial products can be easy generated in parallel by using $n\times m$ AND gates. The hard part in designing fast multiplier is to minimize the logic utilization and time required to add these partial products. As mentioned in section 1, the main concept is to adopt the parallel structure in designing the proposed multiplier, so it can easy, for the carry save adder (CSA) array, to be reduced by truncating of the operand word length. As shown in Fig 1, (b) by using this technique we can benefit from the fact that the slope coefficients M_i has been calculated and stored in register 3 during the interval Δx. therefore it can be truncate its word length such that to reduce the multiplier size. The truncation of the multiplier's input achieves smaller partial product array, but it produces an arithmetic error. By choosing accurately the value of the truncated part we can significantly elevate the impact of this error. Further error correction can be done by rounding the most significant bit of the truncated part. The slope coefficient M_i can be calculated from the sine function as in [1], and the approximated segment lines can be written as

$$\boldsymbol{Y_i(x)} = \frac{\mathbf{(sin\ (\mathit{i}\ \Delta \mathit{x}) - sin\ (\mathit{i}-1)\Delta \mathit{x}).\mathit{x}}}{\boldsymbol{\Delta x}} + \boldsymbol{Ci} \qquad \textbf{(1)}$$

Where C_i is the initial amplitude of ith segment, Δx = the length of segment. To avoid the division process in (1), the formula can be modified to

$$\boldsymbol{\Delta x\times\ Yi\ (x) = (sin(i\ \Delta x) - sin(i-1)\Delta x).x + Ci\ \times \Delta x} \qquad (\mathbf{2})$$

Let us define $S_D = (sin\ (i\ \Delta x) -\ sin\ (i-1)\Delta x)$ as a difference of any two consecutive sine points, thus the equation (2) becomes

$$\boldsymbol{\Delta x\times\ Yi\ (x) = S_D.\ x + Ci \times \Delta x} \qquad \textbf{(3)}$$

Δx is constant, thus the $(Ci\ \times\ \Delta x)$ product can be simply realized by costless hardwired pre-shifting, while variable multiplier is essential to perform $(S_D.x)$ product which represents the core focus of this study. The dynamic range of quantized S_{DQ} is

$$\lfloor \mathbf{2^d}\ \boldsymbol{S_{Dmin}} \rfloor \leq \boldsymbol{S_{DQ}}\ \leq\ \lfloor \mathbf{2^d}\ \boldsymbol{S_{Dmax}} \rfloor \qquad \textbf{(4)}$$

Where $\lfloor . \rfloor$ denotes the floor function and d=L-1 represents segment initial amplitude resolution. The maximum and minimum SD's word are

$$\boldsymbol{P_{min}} = \lceil \mathbf{log_2}\ \boldsymbol{S_{DQmin}} \rceil\ \ ,\ \ \boldsymbol{P_{max}} = \lceil \mathbf{log_2}\ \boldsymbol{S_{DQmax}} \rceil$$

Where $\lceil . \rceil$ denotes the ceiling function. So the multiplier must be performed, i.e the multiplication of $2^{Pmax}\times2^{B}$, since we need ($P_{max}\times B$) bit multiplier. Let $\Delta x = U\times V$, $\Delta x = 2^B$ Equation (3) becomes

$$\boldsymbol{U\times V\times\ Y_i\ (x) = S_{DQ}.\ x + C_i \times U\times V} \qquad \textbf{(5)}$$

Divide the two sides by V, we have

$$\boldsymbol{U\times\ Y_i\ (x)\ = \frac{S_{DQ}.x}{V} + C_i\ \times U} \qquad \textbf{(6)}$$

If V=2^{J}, where $1 \leq J < B$, then U=2^{B-J} and (6) becomes

$$\boldsymbol{2^{B-J} \times Y_i\ (x) = \frac{S_{DQ}.x}{2^J} + C_i\ \times 2^{B-J}} \qquad \textbf{(7)}$$

Where the range of quantized C_{qi} $0\leq C_{qi}\leq \left\lfloor 2^{L-1}\sin\frac{\pi(S-1)}{2S} \right\rfloor$.The first term of (7), $(\frac{S_{DQ}.x}{v})$ needs $(\frac{Pmax\times B}{J})$ bit multiplier. So, it is desirable to minimize $(\frac{S_{DQ}.x}{V})$ to obtain minimum multiplier size, where $(\frac{S_{DQ}.x}{v})$ somewhere in the range of $\frac{2^{Pmin}\times2^{B}}{2^J}$ to $\frac{2^{Pmax}\times2^{B}}{2^J}$ .The only possible solution for minimizing the multiplier size is by maximizing J, J can be somewhere between 1 and B-1, avoid the numbers whose values are less than one from the division process results, then J must satisfy the following condition $\left\lfloor \frac{2^{Pmin}}{2^J} \right\rfloor > 2^0$, or $2^J < 2^{Pmin}$

$$\text{i.e. } \boldsymbol{2^J <\ S_{DQmin}} \qquad \textbf{(8)}$$

$$\boldsymbol{S_{DQmin}} = \left\lfloor\ \boldsymbol{2^{L-1}} \times \{[\mathbf{sin}\frac{\boldsymbol{\pi(S-1)}}{\boldsymbol{2S}}] - [\mathbf{sin}\frac{\boldsymbol{\pi(S-2)}}{\boldsymbol{2S}}]\} \right\rfloor \qquad \textbf{(9)}$$

$$\textbf{So, } \boldsymbol{J} \leq \left\lfloor \mathbf{log_2}\ \{\boldsymbol{2^{L-1}} \times [(\mathbf{sin}\frac{\boldsymbol{\pi(S-1)}}{\boldsymbol{2S}}) - (\mathbf{sin}\frac{\boldsymbol{\pi(S-2)}}{\boldsymbol{2S}})]\} \right\rfloor \qquad \textbf{(10)}$$

Equation (10) represents the realizable range of J.The final multiplier becomes

$(\frac{2^{Pmax}}{2^J}) \times\ 2^B$, i.e. (P_{max} - J) × B instead of (P_{max}) ×B . Where $p_{\text{max}} = \lceil \log_2 S_{DQmax} \rceil$, and

$$S_{DQmax} = \left\lfloor\ 2^{L-1} \times [\sin\frac{\pi}{2S}]\right\rfloor \qquad \textbf{(11)}$$

By applying the division process before storing the coefficients on the Memory cells , i.e. storing $\frac{C_i}{V}$ instead of C_i, we can introduce another improvement. Where the cells word length can be reducing by amount of J bit. Consequently, there is an additional $(\frac{J}{L-J-1})$ ROM reduction ratio. So, it is useful to rewrite (7) as follows.

$$2^{B-J} \times Y_i\,(x) = \frac{S_{DQ}.x}{2^J} + \frac{C_i}{2^J} \times 2^{B-J} \times 2^J\ = \left(\frac{S_{DQ}}{2^J}\right)x + 2^B(\frac{C_i}{2^J}) \qquad \textbf{(12)}$$

In Fig 1, (b) it can be seen that a MUX and its coefficients provides the segment initial amplitudes $\frac{C_i}{2^J}$, represented with L-J-1 bits, and the $\frac{S_{DQ}}{2^J}$ coefficients with P-J bits.

An error produced by this truncation can be estimated as follows

$$\varepsilon_i = \frac{\left|C_i - \left\lfloor\frac{C_i}{2^J} + 0.5\right\rfloor \times 2^J\right|}{2^{L-1}} \qquad \textbf{(13)}$$

The worst-case errors when the value for J LSB bits of $\mathrm{C_i}$ are equal to$(2^J - 1)$) (i.e. all J LSB bits of C_i are non-zero digits (logic 1) then the error become

$$\varepsilon_{max} = 2^{J-1} \qquad \textbf{(14)}$$

The fractional part denoted by $\{\frac{C_i}{V}\}$ for real $\frac{C_i}{V}$, is defined by the formula.

$$\{\frac{C_i}{V}\} = \frac{C_i}{V} - \left\lfloor\frac{C_i}{V}\right\rfloor \text{ for all } \frac{C_i}{V},\ 0 \leq \{\frac{C_i}{V}\} < 1 \qquad \textbf{(15)}$$

To reduce the amount of error, the most significant bit, *MSB* of the fraction part $\{\frac{C_i}{V}\}$ should be rounded to the integer part $\left\lfloor\frac{C_i}{V}\right\rfloor$.

$$\textbf{round}\ (\frac{C_i}{V}) = \left\lfloor\frac{C_i}{V} + 0.5\right\rfloor \qquad \textbf{(16)}$$

This rounding alleviates the amount of error by enforcing the J LSB of all $\mathrm{C_i}$ to be less than$(2^J - 1)$ and this can be done by modifying the sine points before storing in ROM to insure that the truncation of C_i coefficients will not produce a binary results with J LSB nonzero digits equal to ($2^J - 1$). This minor modification will not contribute noticeable error from the system's perspective. As the number of coefficients small, (S Coefficients) the rounding process can be applied by modifying the coefficients C_i before storing in the ROM and that means eliminating the additional rounding hardware. Fig. 2 (a), (b). shows the Array structure of (P_{max} × B) and (P_{max} - J) × B parallel multipliers. The $\left(\frac{S_{DQ}}{2^J}\right).x$ product can be expressed as follows

$$\left(\frac{S_{DQ}}{2^J}\right).x = \sum_{i=J}^{p_{max}-1} S_D.2^i.\sum_{k=0}^{B-1} x_k\,2^k\,. = \sum_{i=J}^{p_{max}-1}\sum_{k=0}^{B-1} S_D.x_k\,2^{i+k} \qquad (17)$$

It can be seen that the new multiplier has (P_{max} - J) ×B partial product term $(S_D.x_k)$ i.e. we just need (P_{max} - J) × B Full Adder (FA) instead of P_{max} × B F.A, and the time delay become (P_{max} - J + B) × $T_{PD,\,FA}$ instead of (P_{max} + B) × $T_{PD,\,FA}$ Where $T_{PD,\,FA}$ represent the propagation delay of FA. Hence the proposed multiplier offers an improvement of (J × $T_{PD,\,FA}$) time delay and has (J ×B) FA fewer component counts.

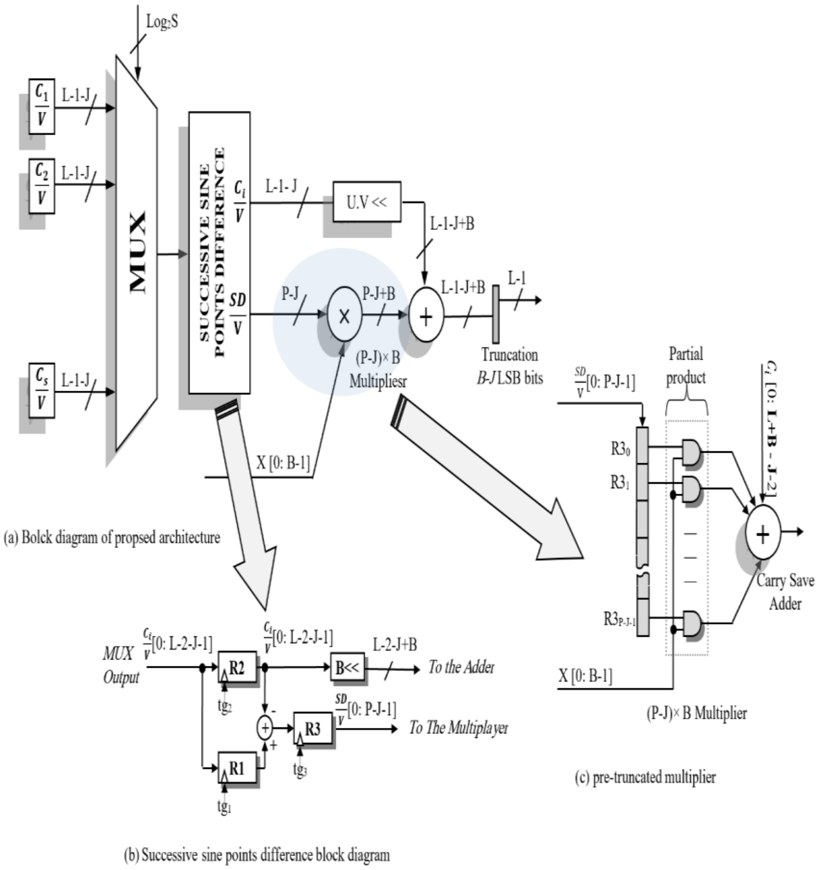
3. Calculation of multiplier size

Based on design requirement derived in previous section, we start the design of multiplier by computing the SD_{qmax} as follows $S_{DQmax} = \left\lfloor 2^{14} \times [\sin\frac{\pi}{64}] \right\rfloor = 804$ Using this value to obtain the maximum S_D's word length; $P_{max}= \lceil \log_2 804 \rceil = 10$ That is mean we need 10×8 bits multiplier to perform S_D. x product, Obviously the multiplier's size is so large, hence in the following step, we will attempt to reduce the multiplier size so as to reduce the gate count with an acceptable error. To find the realizable range of J we have to calculate S_{DQmin} sing (9).$S_{DQmin} = \left\lfloor 2^{14} \times \{[\sin\frac{\pi(31)}{64}] - [\sin\frac{\pi(30)}{64}]\} \right\rfloor = 60$. Then by using (10) we have $J \le \lfloor \log_2 60 \rfloor = 5,\ \ 1 \le J \le 5$ And using formula (13) we can calculate the average absolute error for J=1, 2, 3, 4, 5, the calculated results are reported in Table 1.

Truncation(bit)	Maximum Error	Average Error	Variance
J=5	16	8.25	24.437499
J=4	8	3.5625	4.80859
J=3	4	2.0625	1.24609
J=2	2	1.1875	0.5898437
J=1	1	0.375	0.2343749

Table 1. The truncation error (×2^{L-1})

Consecutive sine points difference $\frac{S_D}{V}$	Initial amplitude coefficient $\frac{C_i}{V}$ (Quantized with 11 bits)
[100,101,100,99,98,97,95, 94,92,89,88,85,82,79,76, 73,69,66,62,58,54,49,45, 40,36,32,27,22,17,12,8,2]	[0,100,201,301,400,498,595,690, 784,876,965,1053,1138,1220,1299, 1375,1448,1517,1583,1645,1703,1757, 1806,1851,1892,1928,1960,1987,2009, 2026,2038,2046]



(a) Bolck diagram of propsed architecture

(b) Successive sine points difference block diagram

(c) pre-truncated multiplier

Figure 1. Propsed architecture

Table 2. Design coefficients, 32 segments, J=3

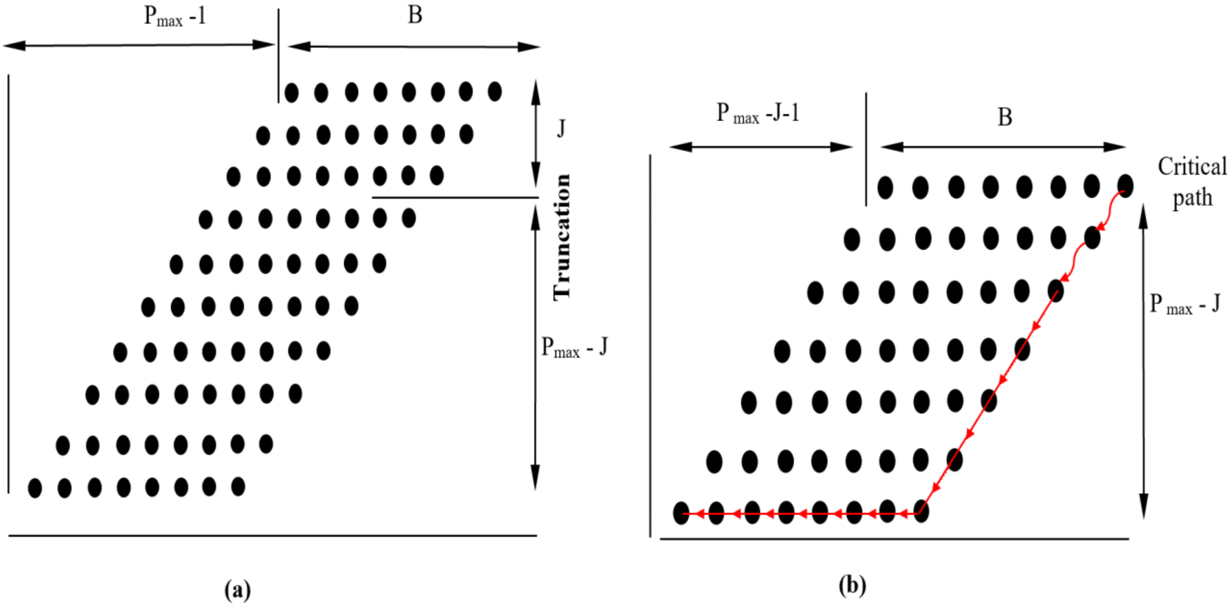


Figure 2. Array structure of parallel multiplier (a) P_{max}×B bit multiplier (b) (P_{max}-J) ×B bit

It can be seen that the average error for *J*=3 has acceptable values and thus considered as reasonable compromising between reduction of multiplier size and tolerable error. Using this value of *J*, hence calculating the $\frac{c_i}{V}$ coefficients, the coefficients are included in Table II. Finally, a truncated 7×8 multiplier has been placed in the targeted DDFS architecture instead of the 10×8 multiplier. The proposed multiplier offers an improvement of (3 × $T_{PD,\,FA}$) time delay and the component less by (3 × 8) FA.

4. Design implementation and verification

The proposed design is written in VHDL code using the Quartus II 11.0 sp1 software. Stratix III, EP3SE50F484C2 FPGA device is used for implementing both the truncated and conventional full width multiplier. Table III show the post fitting report of the implementations. The project is then verified for desired output using the Modalism-Altera 6.6d simulation software. Figure 3 shows RTL simulated result for sample data SD/V (49,45,40,36) multiply by all the possible combination of phase sample inputs. Figures shows the truncated and rounding process, hard-wired shifting of the truncated value in decimal and binary Radix, it's worth to note that the truncated multiplier result present after one clock cycle which is highlighted in Figures due to output register.

Table 3. post fitting report of the multiplier

	Truncated Multiplier (7×8)	Full-width Multiplier (10×8)
Combinational ALUTs	123	195
Dedicated logic registers	30	36
F clock	211.37 MHz	175.8 MHz

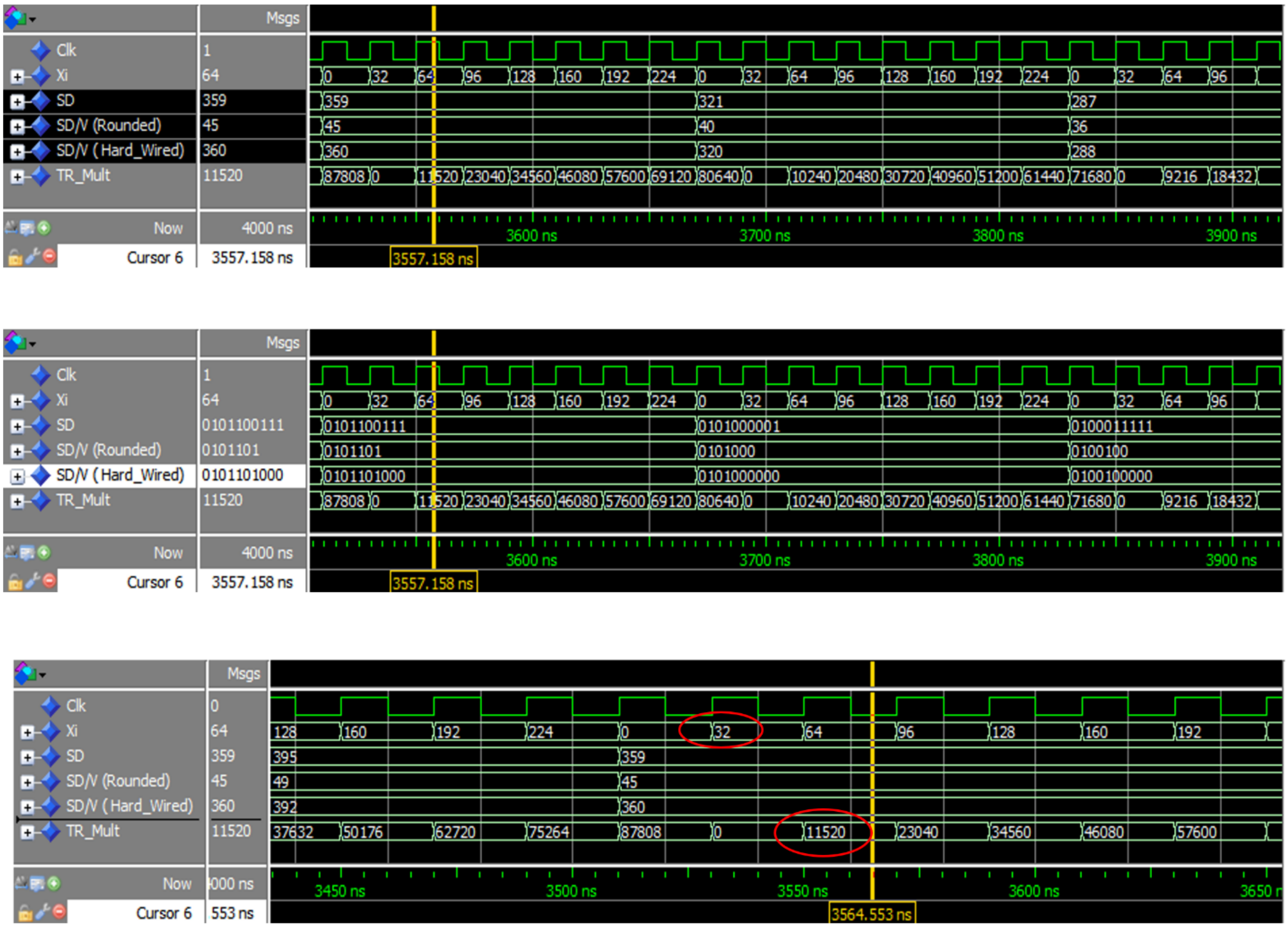


Figure 3. Simulation result of truncated 7×8 multiplier

5. Conclusion

A pre-truncation of initial amplitude value has been employed to reduce the gate count and time delay for the multiplier with an acceptable error. It was shown that an improvement of (3 × $T_{PD,\ FA}$) time delay and 24 less Full Adder count (the component fewer by 24 FA) was achieved. The developed version of multiplier has been placed in the DDFS system and tested. The proposed multiplier has shown 36.9 % less resources logic utilization, 17 % more speed than the conventional multiplier.

References

[1] Q. K. Omran, M. T. Islam, and N. Misran, “A new approach to the design of low-complexity direct digital frequency synthesizer,” *Przegląd Elektrotechniczny (Electrical Review)*, vol. 89, no. 5, pp. 157–160, 2013.

[2] Q.K Omran., M.T. Islam,“An efficient ROM compression technique for linear-interpolated direct digital frequency synthesizer,” *IEEE Conf. Semicond. Electron.*, vol. 48, pp. 2409–2418, 2014.

[3] T. Mahmood, O. A. Mahmood, and K. A. Humood. “An efficient technique to PAPR reduction for LTE uplink using Lonzo’s resampling technique in both SC-LFDMA and SC-DFDMA systems,” *Applied Nanoscience*, 2021.

[4] H. K. AL-Qaysi, T. Mahmood, and K. A. Humood, “Evaluation of different quantization resolution levels on the BER performance of massive MIMO systems under different operating scenarios.” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 3, pp. 1493-1500, 2021.

[5] A. H. M. Alaidi, A. S. Abdalrada, and F. T. Abed, "Analysis the Efficient Energy Prediction for 5G Wireless Communication Technologies," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 14, no. 08, pp. 23-37, 2019.

[6] H. T. S. Al-Rikabi, Enhancement of the MIMO-OFDM Technologies. California State University, Fullerton, 2013.

[7] A. Al-Dawoodi, H. Maraha, S. Alshwani, A. GHAZI, A. M. FAKHRUDEEN, S. Aljunid, S. Z. S. IDRUS, A. A. MAJEED, and K. A. AMEEN, "Investigation of 8 x 5 Gb/s mode division multiplexing-fso system under different weather condition," *Journal of Engineering Science Technology*, vol. 14, no. 2, pp. 674-681, 2019.

[8] A. Ghazi, S. Aljunid, S. Z. S. Idrus, R. Endut, C. Rashidi, N. Ali, A. Al-dawoodi, A. M. Fakhrudeen, A. Fareed, and T. Sharma, "Hybrid WDM and Optical-CDMA over Multi-Mode Fiber Transmission System based on Optical Vortex," J*ournal of Physics: Conference Series*, vol. 1755, no. 1, p. 012001, 2021.

[9] T. Mahmood, H. AL-Qaysi, and A. Hameed, “The Effect of Antenna Height on the Performance of the Okumura/Hata Model Under Different Environments Propagation,” *International Conference on Intelligent Technologies (CONIT)*, pp. 1-4. IEEE, 2021.

[10] T. Mahmood, W. Q. Mohamed, and O. A. Imran, “Factors Influencing the Shadow Path Loss Model with Different Antenna Gains Over Large-Scale Fading Channel," *International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)*, pp. 1-5, 2021.