International Journal of Soft Computing 3 (6): 412-420, 2008
ISSN: 1816-9503

Low Power High Performance Multiplier

[1]C.N. Marimuthu, [2]Aiby Varghese and [3]P. Thangaraj
[1]Maharaja Engineering College, Avinashi, India
[2]ME-Applied Electronics, Maharaja Engineering College, Avinashi, India
[3]Department of CT, Kongu Engineering College, Perundurai, India

Abstract: There are different entities that one would like to optimize when designing a VLSI circuit. These entities can often not be optimized simultaneously, only improve one entity at the expense of one or more others. The design of an efficient integrated circuit in terms of power, area and speed simultaneously, has become a very challenging problem. Power dissipation is recognized as a critical parameter in modern VLSI design field. In Very Large Scale Integration, Low power VLSI design is necessary to meet MOORE'S law and to produce consumer electronics with more back up and less weight. Multiplication occurs frequently in finite impulse response filters, fast Fourier transforms, discrete cosine transforms, convolution and other important DSP and multimedia kernels. The objective of a good multiplier is to provide a physically compact, good speed and low power consuming chip. To save significant power consumption of a VLSI design, it is a good direction to reduce its dynamic power that is the major part of total power dissipation. In this study, we propose a high speed low-power multiplier adopting the new SPST implementing approach. This multiplier is designed by equipping the Spurious Power Suppression Technique (SPST) on a modified Booth encoder which is controlled by a detection unit using an AND gate. The modified booth encoder will reduce the number of partial products generated by a factor of 2. The SPST adder will avoid the unwanted addition and thus minimize the switching power dissipation. The proposed high speed low power multiplier can attain 30% speed improvement and 22% power reduction when compared with the conventional array multipliers.

Key words: Array multiplier, booth encoder, low power, spurious power suppression technique

INTRODUCTION

Power dissipation is recognized as a critical parameter in modern VLSI design field. To satisfy MOORE'S law and to produce consumer electronics goods with more backup and less weight, low power VLSI design is necessary. Dynamic power dissipation which is the major part of total power dissipation is due to the charging and discharging capacitance in the circuit. The golden formula for calculation of dynamic power dissipation is Pdynamic = CLV2f. Power reduction can be achieved by various manners. They are reduction of output capacitance CL, reduction of power supply voltage V, reduction of switching activity and clock frequency f.

Fast multipliers are essential parts of digital signal processing systems. The speed of multiply operation is of great importance in digital signal processing as well as in the general purpose processors today, especially since the media processing took off. In the past multiplication was generally implemented via a sequence of addition, subtraction and shift operations. Multiplication can be considered as a series of repeated additions. The number to be added is the multiplicand, the number of times that it is added is the multiplier and the result is the product. Each step of addition generates a partial product. In most computers, the operand usually contains the same number of bits. When the operands are interpreted as integers, the product is generally twice the length of operands in order to preserve the information content. This repeated addition method that is suggested by the arithmetic definition is slow that it is almost always replaced by an algorithm that makes use of positional representation. It is possible to decompose multipliers into two parts. The first part is dedicated to the generation of partial products and the second one collects and adds them.

The basic multiplication principle is two fold i.e., evaluation of partial products and accumulation of the shifted partial products. It is performed by the successive additions of the columns of the shifted partial product matrix. The 'multiplier' is successfully shifted and gates the appropriate bit of the 'multiplicand'. The delayed, gated instance of the multiplicand must all be in the same

Corresponding Author: C.N. Marimuthu, Maharaja Engineering College, Avinashi, India

column of the shifted partial product matrix. They are then added to form the product bit for the particular form. Multiplication is therefore a multi operand operation. To extend the multiplication to both signed and unsigned numbers, a convenient number system would be the representation of numbers in two's complement format.

TECHNIQUES FOR IMPLEMENTING MULTIPLIERS

Binary multiplication of positive operands can be implemented in a combinational 2 dimensional logic array (Lakshmi and Venkataramani, 2005). A 4×4 array multiplier is shown in Fig. 1. The functions of M_0, M_1, M_2 and M_4 are also shown in Fig. 1. X_3 $X_2X_1X_0$ is the 4 bit multiplicand and $Y_3Y_2Y_1Y_0$ is the 4 bit multiplier. The main component in each cell is a full adder. The AND gate in each cell determines whether a multiplicand bit, Xj is added to the incoming partial product bit, based on the value of the multiplier bit Yi. Each row adds the multiplicand (appropriately shifted) to the incoming partial product, PPi to generate the outgoing partial product PP (i +1), if Yi = 1. If Yi = 0, PPi is passed vertically downward unchanged. The worst case signal propagation delay path is from the upper right corner of the array to the high order product bit output at the bottom left corner of the array. Assuming that there are 2 gate delays from the inputs to the outputs of a full adder block. Since, the array multiplier is having low speed and consumes more power many low power multiplier design techniques are available now. The design techniques such as Glitching power minimization by selective gate freezing (Benini *et al*., 2000), Fast power efficient circuit lock switch-off schemes (Henzler *et al*., 2004), power-aware scalable pipelined Booth multiplier (Lee, 2004), Partially

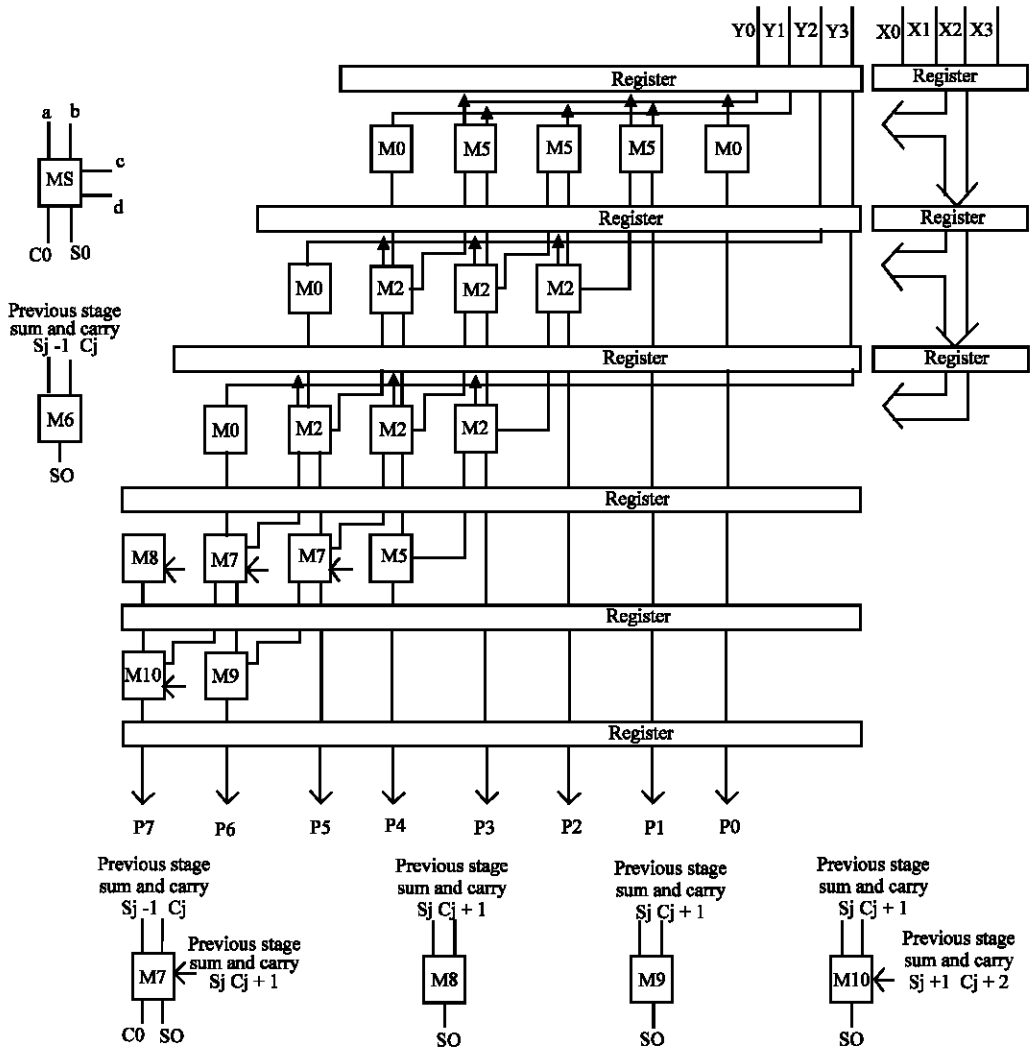

Fig. 1: Array multiplier

Guarded Computation (PGC) (Choi *et al*., 2000), High performance low power left to right array multiplier design (Huang and Ercegovac, 2005) are existing works that reduce the dynamic power consumption by minimizing the switched capacitance.

Glitching power minimization by selective gatefreezing replaces some existing gates with functionally equivalent ones that can be frozen by asserting a control signal. It can only achieve savings of 6.3% in total power dissipation since it operates in the layout environment which is tightly restricted. The Fast power efficient circuit block switch-off scheme proposes a double switch circuit block switch scheme capable of reducing power dissipation during down time by shortening the settling time after reactivation. The drawbacks of this scheme are the necessity for two independent virtual rails and the necessity for two additional transistors for switching each cell. A power-aware scalable pipelined Booth multiplier design presents a multiplier using Dynamic Range Detection (DRD) unit. is used to select the input operand with a smaller effective dynamic range to yield the Booth codes. There are three separate Wallace trees for the 4×4, 8×8 and 16×16 multiplications. This will certainly induce area and capacitance penalties. Under a 0.13 μm CMOS technology this design can obtain a 20% power reduction over the conventional multipliers. But there is a 44% n area overhead. Partially Guarded Computation (PGC) divides the arithmetic units, e.g., adders and multipliers, into two parts and turns off the unused part to minimize the power consumption. The reported results show that the PGC can reduce power consumption by 10-40% in an array multiplier but with 30-36% area overhead. In all the above techniques there is no improvement in speed. So, in this study by applying an advanced version of former spurious power suppression technique on multipliers for high speed and low power purposes is present.

SPURIOUS POWER SUPPRESSION TECHNIQUE (SPST)

The former SPST has been discussed in (Chen *et al*., 2005, 2006). Figure 2 shows the five cases of a 16-bit addition in which the spurious switching activities occur. The 1st case illustrates a transient state in which the spurious transitions of carry signals occur in the MSP

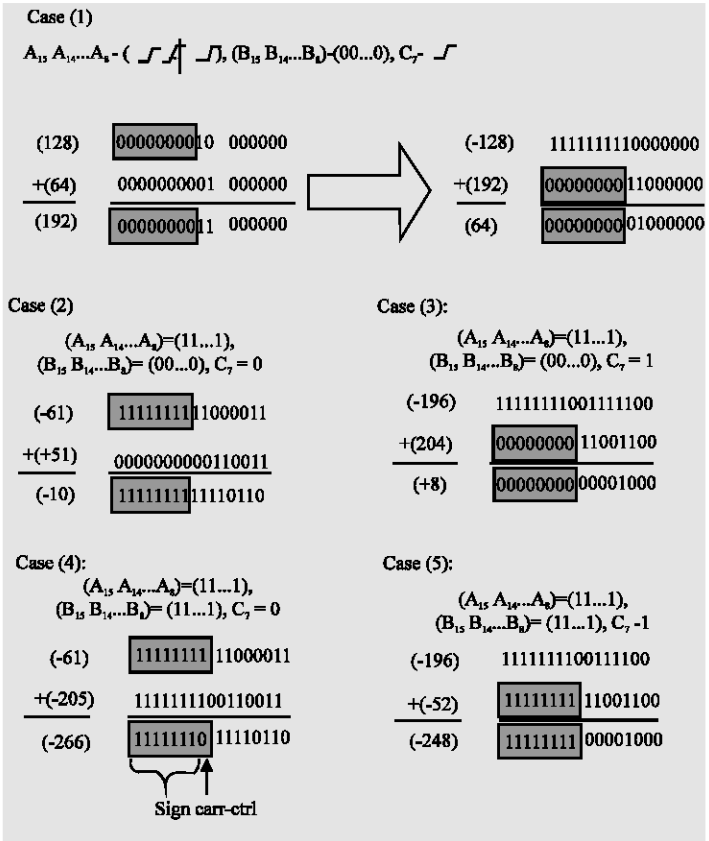

Fig. 2: Spurious transition cases in multimedia/DSP processing

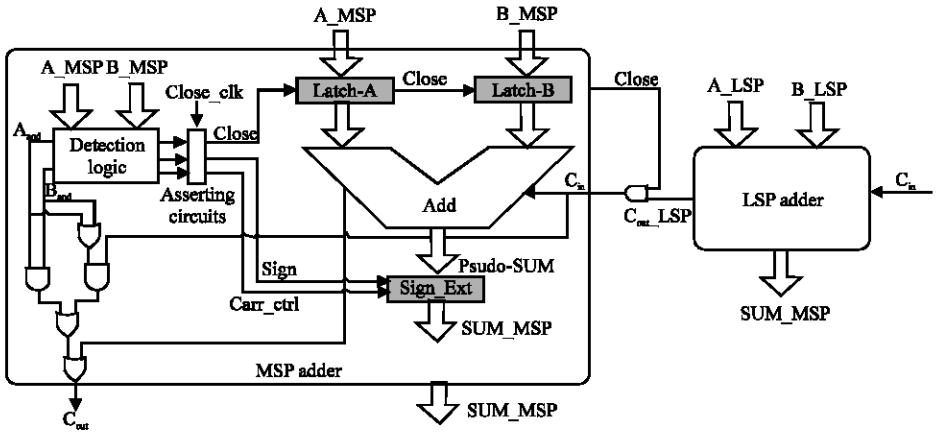

Fig. 3: Low power adder/subtractor adopting the SPST

though the final result of the MSP are unchanged. The 2nd and the 3rd cases describe the situations of one negative operand adding another positive operand without and with carry from LSP, respectively. Moreover, the 4th and the 5th cases, respectively demonstrate the addition of 2 negative operands without and with carry-in from LSP. In those cases, the results of the MSP are predictable. Therefore, the computations in the MSP are useless and can be neglected. The data are separated into the Most Significant Part (MSP) and the Least Significant Part (LSP). To know whether the MSP affects the computation results or not, we need a detection logic unit to detect the effective ranges of the inputs. The Boolean logical equations shown below express the behavioral principles of the detection logic unit in the MSP circuits of the SPST-based adder/subtractor:

$$A_{MSP} = A[15:8];\ B_{MSP} = B[15:8] \quad (1)$$

$$A\ and = A[15].A[14]\ldots\ldots A[8] \quad (2)$$

$$B\ and = B[15].B[14]\ldots\ldots B[8] \quad (3)$$

$$\overline{Anor} = \overline{A[15] + A[14] + \ldots + A[8]} \quad (4)$$

$$\overline{Bnor} = \overline{A[15] + A[14] + \ldots + A[8]} \quad (5)$$

$$Close = (Aand+Anor).\ (Band+Bnor) \quad (6)$$

where, A[m] and B[n], respectively denote the mth bit of the operands A and the nth bit of the operand B and A_{MSP} and B_{MSP}, respectively denote the MSP parts, i.e., the 9th bit to the 16th bit, of the operands A and B. When the bits in A_{MSP} and/or those in B_{MSP} are all ones, the value of A and and/or that of Band, respectively become one, while the bits in AMSP and/or those in B_{MSP} are all zeros, the value of Anor and/or that of Bnor, respectively turn into one. Being one of the three outputs of the detection logic unit, close denotes whether the MSP circuits can be neglected or not. When the two input operand can be classified into one of the 5 classes as shown in Fig. 1, the value of close becomes zero which indicates that the MSP circuits can be closed. Figure 1 also shows that it is necessary to compensate the sign bit of computing results. Accordingly, we derive the Karnaugh maps which lead to the Boolean Eq. (7) and (8) for the Carr_ ctrl and the sign signals, respectively. In Eq. (7) and (8), C_{LSP} denotes the carry propagated from the LSP circuits.

$$Carr\_ctrl = CLSP.\ Aand.\ Anor.\ Band.\ Bnor + CLSP$$

$$\begin{aligned} &Aand.\ \overline{Anor}.\ \overline{Band}.\ Bnor + CLSP.\ \overline{Aand}.\ Anor.\ \overline{Band}.Bnor + CLSP.\ Aand.\ \overline{Anor}.Band\ Bnor \\ &= C_{\overline{LSP}}\ (\overline{Aand}.\ Band + Aand.\ \overline{Band}).\ (Aand.Band + Aand.Bnor + Anor.Band + Anor\ Bnor.) + CLSP.(\ Aand.Band + \overline{Aand}.\overline{Band}). \\ &(Aand.Band + Aand.Bnor + Anor.Band + Anor) \\ &= (C_{LSP} \oplus Aand \oplus Band).(Aand + Anor).\ (Band + Bnor) \end{aligned} \quad (7)$$

$$\begin{aligned} sign &= C_{LSP}.\ (Aand + Band) \\ &+ CLSP.Aand.\ Band \end{aligned} \quad (8)$$

Figure 3 shows a 16-bit adder/subtractor design example based on the proposed SPST. In this example, the

16-bit adder/subtractor is divided into MSP and LSP at the place between the 8th bit and the 9th bit. Latches implemented by simple AND gates are used to control the input data of the MSP. When the MSP is necessary, the input data of MSP remain the same as usual, while the MSP is negligible, the input data of the MSP become zeros to avoid switching power consumption. From the derived Boolean Eq. (1-8), the detection logic unit of the SPST is designed as shown in Fig. 3 (a), which can determine whether the input data of MSP should be latched or not. Moreover, we add three 1-bit to control the assertion of the close, sign and Carr-ctrl signals in order to further decrease the glitch signals occurred in the cascaded circuits which are usually adopted in VLSI architectures designed for video coding.

PROPOSED SPURIOUS POWER SUPPRESSION TECHNIQUE (SPST)

The SPST uses a detection logic circuit to detect the effective data range of arithmetic units, e.g., adders or multipliers. When a portion of data does not affect the final computing results, the data controlling circuits of the SPST latch this portion to avoid useless data transitions occurring inside the arithmetic units. Besides, there is a data asserting control realized by using registers to further filter out the useless spurious signals of arithmetic unit every time when the latched portion is being turned on. This asserting control brings evident power reduction. In Fig. 3 SPST is illustrated through a low power adder/subtractor design example. The adder subtractor is divided into two parts, the Most Significant Part (MSP) and the Least Significant Part (LSP). The MSP of the original adder/subtractor is modified to include detection logic circuits, data controlling circuits, sign extension circuits, logics for calculating carry in and carry out signals. The most important part of this study is the design of the control signal asserting circuits, denoted as asserting circuits in Fig. 3. Although, this asserting circuit brings evident power reduction, it may induce additional delay. There are 2 implementing approaches for the control signal assertion circuits.

The first implementing approach of control signal assertion circuit is using registers. This is illustrated in Fig. 4. The three output signals of the detection logic are close, Carr-ctrl, sign. The three output signals the detection logic unit are given a certain amount of delay before they assert. This is demonstrated in the timing diagram shown in Fig. 5. The delay Φ, used to assert the three output signals, must be set in a range of $\Psi < \Phi < \Delta$, where, Ψ denotes the data transient period and Δ denotes

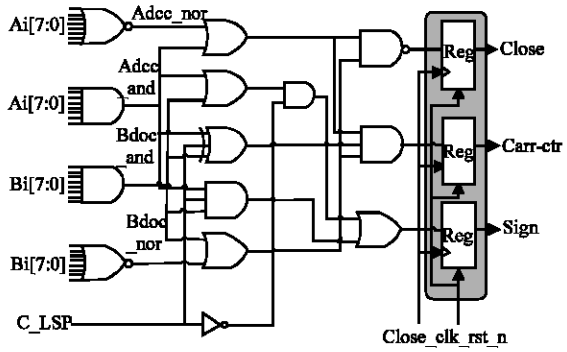

Fig. 4: Detection logic circuits using registers

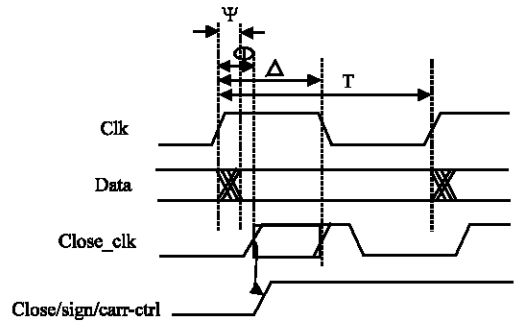

Fig. 5: Timing diagram of the control signals of detection logic circuits after assertion

the earliest required time of all the inputs. This will filter out the glitch signals as well as to keep the computation results correct. The restriction that Φ must be greater than Ψ to guarantee the registers from latching the wrong values of control usually decreases the overall speed of the applied designs. This issue should be noticed in high-end applications which demands both high speed and low power requirements. To solve this problem we adopt the other implementing approach of control signal assertion circuit using AND gate.

Detection logic circuit using an AND gate to assert control signal is sown in Fig. 6. The timing control of delay Φ in this implementation is slightly different from the one in the first implementation. That is, the range of Φ can be set as $0 < \Phi < \Delta$ to filter out the glitch signals and to keep the computation results correct. This feature allows upper level system to assert the close signal with an arbitrarily short delay closing to the positive edge of the clock signal. This will provides a more flexible controlling space for the delay Φ. When speed is seriously concerned, this implementing approach enables an extremely high flexibility on adjusting the data asserting time of the SPST equipped

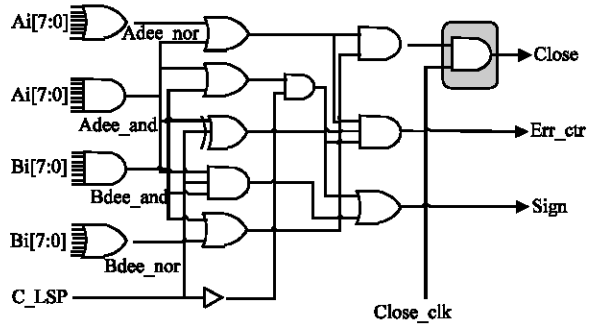

Fig. 6: Detection logic circuits using an AND gate

multipliers. Therefore, the proposed advanced SPST can benefit multipliers on both high speed and low power features.

MODIFIED BOOTH ENCODER

In order to achieve high-speed multiplication, multiplication algorithms using parallel counters, such as the modified Booth algorithm has been proposed and some multipliers based on the algorithms have been implemented for practical use. This type of multiplier operates much faster than an array multiplier for longer operands because its computation time is proportional to the logarithm of the word length of operands. Booth multiplication is a technique that allows for smaller, faster multiplication circuits, by recoding the numbers that are multiplied. It is possible to reduce the number of partial products by half, by using the technique of radix 4 Booth recoding. The basic idea is that, instead of shifting and adding for every column of the multiplier term and multiplying by 1 or 0, we only take every second column and multiply by ±1, ±2, or 0, to obtain the same results. The advantage of this method is the halving of the number of partial products. To Booth recode the multiplier term, we consider the bits in blocks of three, such that each block overlaps the previous block by one bit. Grouping starts from the LSB and the first block only uses two bits of the multiplier. Figure 7 shows the grouping of bits from the multiplier term for use in modified booth recoding.

Each block is decoded to generate the correct partial product. The encoding of the multiplier Y, using the modified booth algorithm, generates the following five signed digits, -2, -1, 0, +1, +2. Each encoded digit in the multiplier performs a certain operation on the multiplicand, X, as illustrated in Table 1.

Table 1: Modified both encoder

Block	Re-coded digit	Operation on X
000	0	0X
001	+1	+1X
010	+1	+1X
011	+2	+2X
100	-2	-2 X
101	-1	-1 X
110	-1	-1 X
111	0	0 X

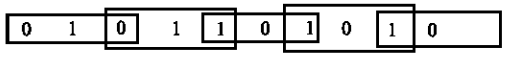

Fig. 7: Grouping of bits from the multiplier term

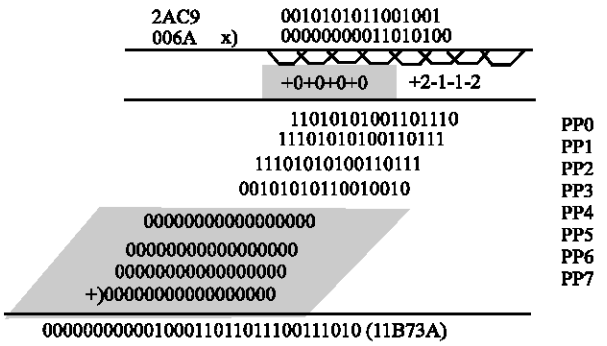

Fig. 8: Illustration of multiplication using modified Booth encoding

PROPOSED LOW POWER HIGH PERFORMANCE MULTIPLIER

The proposed high speed low power multiplier is designed by equipping the SPST on a tree multiplier. There are two distinguishing design considerations in designing the proposed multiplier.

Applying the SPST on the modified booth encoder: Figure 8 shows a computing example of Booth multiplying two numbers2AC9 and 006A. The shadow denotes that the numbers in this part of Booth multiplication are all zero so that this part of the computations can be neglected. Saving those computations can significantly reduce the power consumption caused by the transient signals. According to the analysis of the multiplication shown in Fig. 8, we propose the SPST-equipped modified-Booth encoder, which is controlled by a detection unit. The detection unit has one of the two operands as its input to decide whether the Booth encoder calculates redundant computations. As shown in Fig. 9, the latches can, respectively, freeze the inputs of MUX-4 to MUX-7 or

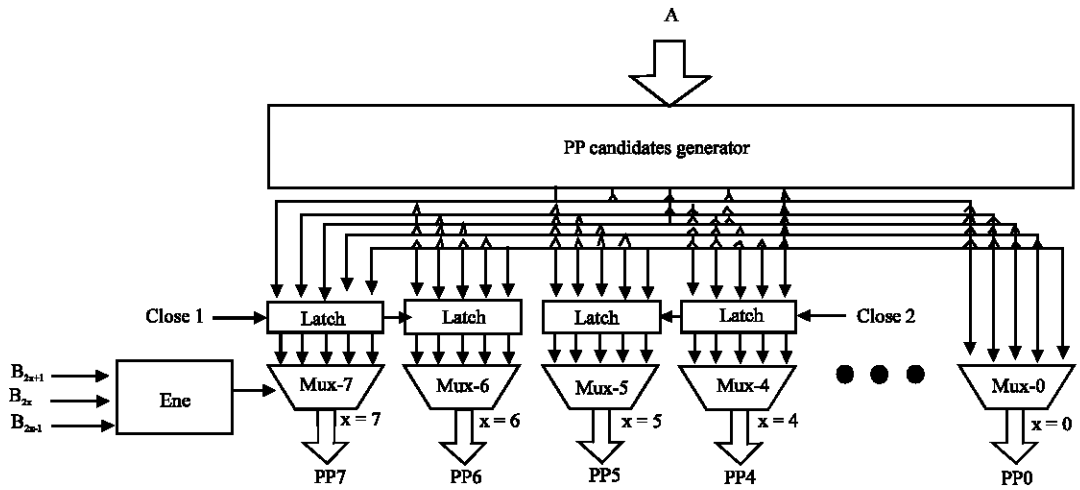

Fig. 9: SPST equipped modified booth encoder

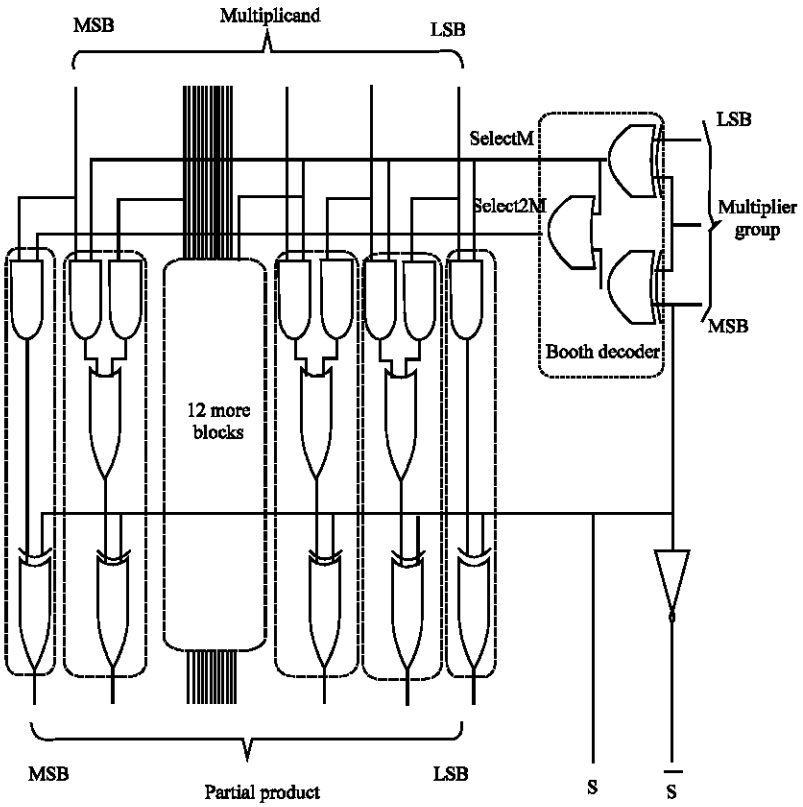

Fig. 10: Booth partial product selector logic

only those of MUX-6 to MUX-7 when the PP4 to PP7 or the PP6 to PP7 are 0, to reduce the transition power dissipation. Figure 10 shows the booth partial product generation circuit. It includes AND/OR/EX-OR logic.

Applying the SPST on the compression tree: The proposed SPST-equipped multiplier is illustrated in Fig. 11. The PP generator generates five candidates of the partial products, i.e., {-2A,-A, 0, A, 2A}. These are then selected according to the Booth encoding results of the operand B. When the operand besides the Booth encoded one has a small absolute value, there are opportunities to reduce the spurious power dissipated in the compression tree.

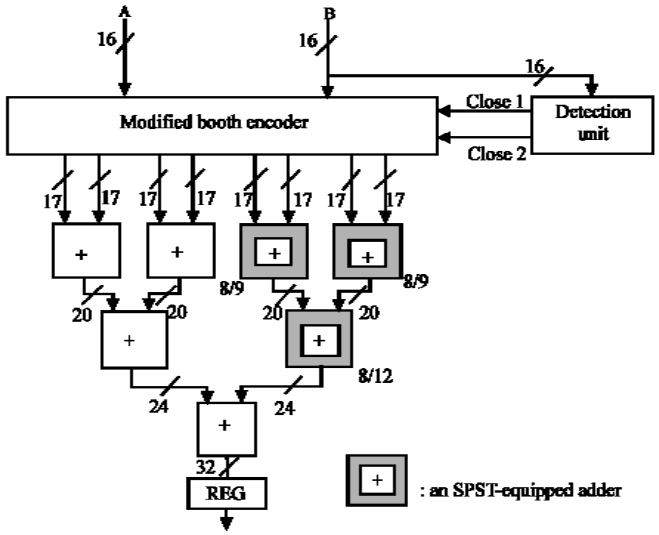

Fig. 11: Proposed high performance low power equipped multiplier

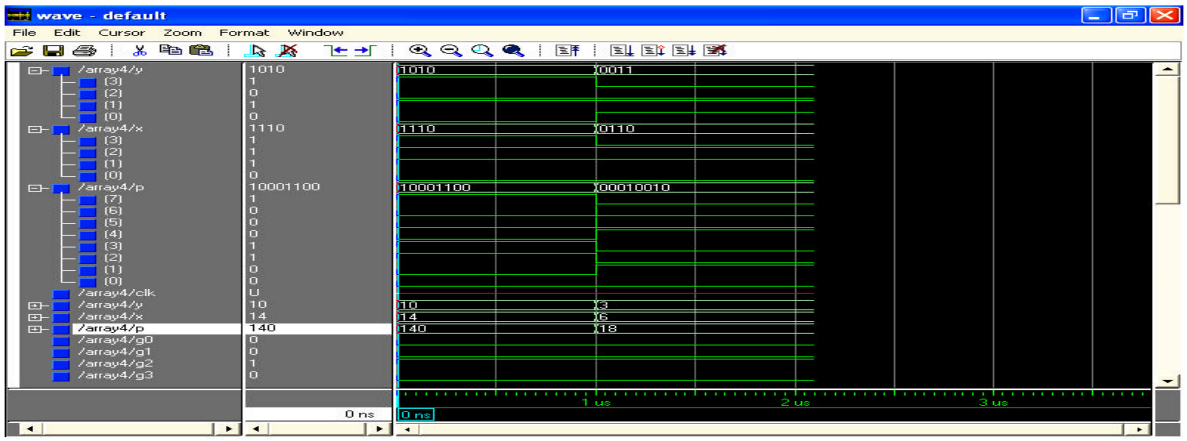

Fig. 12: Simulation results for array multiplier

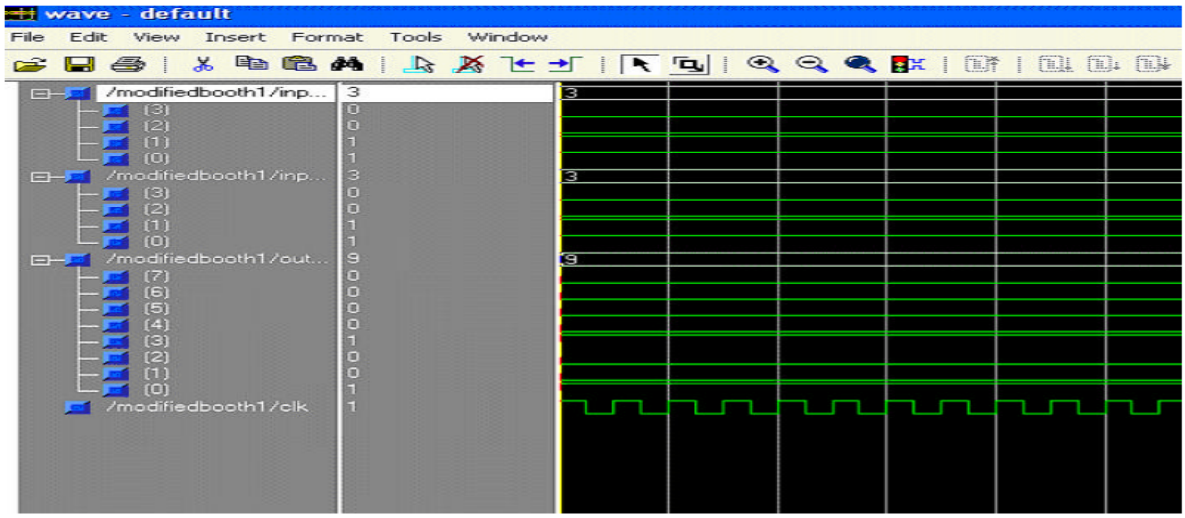

Fig. 13: Simulation results for proposed multiplier

Table 2: Synthesis report for 2 multipliers

Multiplier type	Array multiplier	Proposed multiplier
Vendor	Xilinix	Xilinix
Device and family	Spartan 2	Spartan 2
Estimate delay	21.068 ns	14.886 ns
Power dissipation	44 mW	35 mW

RESULTS AND DISCUSSION

In this study we are evaluating the performance of the proposed high speed low power multiplier by comparing this design with a conventional array multiplier. Both multipliers can be implemented using VHDL coding. In order to get the power report and delay report we are synthesizing these multipliers using Xilinx. Simulation results for the array multiplier is given in Fig. 12 and for proposed multiplier is given in Fig. 13.

The comparison of synthesis report for two multipliers is given in Table 2.

CONCLUSION

In this study, we propose a high speed low-power multiplier adopting the new SPST implementing approach. This multiplier is designed by equipping the Spurious Power Suppression Technique (SPST) on a modified Booth encoder which is controlled by a detection unit using an AND gate. The modified booth encoder will reduce the number of partial products generated by a factor of 2. The SPST adder will avoid the unwanted addition and thus minimize the switching power dissipation. The SPST multiplier implementation with AND gates have an extremely high flexibility on adjusting the data asserting time. This facilitates the robustness of SPST can attain 30% speed improvement and 22% power reduction when compared with the conventional tree multipliers. This design can be verified using Modelsim 6.5 using VHDL coding.

REFERENCES

Benini, L., G.D. Micheli, A. Macii, E. Macii, M. Poncino and R. Scarsi, 2000. Glitching power minimization by selective gate freezing. IEEE. Trans. Very Large Scale Integr. (VLSI) Syst., 8 (3): 287-297.

Choi, J., J. Jeon and K. Choi, 2000. Power minimization of functional units by partially guarded computation. In: Proc. IEEE. Int. Symp. Low Power Electron. Des., pp: 131-136.

Chen, K.H., K.C. Chao, J.I. Guo, J.S. Wang and Y.S. Chu, 2005. An efficient spurious power suppression technique (SPST) and its applications on MPEG-4 AVC/H.264 transform coding design. In: Proc. IEEE. Int. Symp. Low Power Electron. Des., pp: 155-160.

Chen, K.H., Y.M. Chen and Y.S. Chu, 2006. A versatile multimedia functional unit design using the spurious power suppression technique. In: Proc. IEEE. Asian Solid-State Circuits Conf., pp: 111-114.

Henzler, S., G. Georgakos, J. Berthold and D. Schmitt-Landsiedel, 2004. Fast power-efficient circuit-block switch-off scheme. Electron. Lett., 40 (2): 103-104.

Huang, Z. and M.D. Ercegovac, 2005. High-performance low-power left-toright array multiplier design. IEEE. Trans. Comput., 54 (3): 272-283.

Lee, H., 2004. A power-aware scalable pipelined Booth multiplier. In: Proc. IEEE. Int. Soc. Conf., pp: 123-126.

Lakshmi, G. Narayanan and B. Venkataramani, 2005. Optimization Techniques for FPGA-Based Wave Pipelined DSP Blocks. IEEE. Trans. Very Large Scale Integr. (VLSI) Syst., 13 (7): 783-792.