ISSN 2278-7763

A SPST BASED 16x16 MULTIPLIER FOR HIGH SPEED LOW POWER APPLICATIONS USING RADIX-4 MODIFIED BOOTH ENCODER

Chadalavada Divya Sai
ECE, GNITS
Hyderabad, India
Email id: divyasai1994@gmail.com

Koka Vinila
ECE, GNITS
Hyderabad, India
Email id: vinila0208@gmail.com

***ABSTRACT*-----The major design constraints in any VLSI circuit design is its power dissipation and speed. These entities cannot be optimized simultaneously; improvement of one entity is possible at the expense of one or more others. Power dissipation is recognized as a critical parameter. The objective of a good multiplier is to reduce the power dissipation without compromising with its speed and compactness. In order to save the power consumption it is better to reduce its dynamic power, which is a major part of the total power dissipation.**

This paper presents a spurious power suppression technique (SPST) for a high speed low-power multiplier which can dramatically reduce the power dissipation of combinational VLSI designs for multimedia/DSP purposes. This paper also discusses the efficiency of SPST over array multiplier in terms of speed and power dissipation. The effectiveness of SPST in various applications such as Efficient multi-Transform Design (ETD) and Versatile Multimedia Functional Unit (VMFU) are also presented. The proposed SPST separates the target designs into two parts, i.e., the most significant part and least significant part (MSP and LSP), and turns off the MSP when it does not affect the computation results to save power and thus minimize the switching power dissipation. In this project we used Xilinx-ISE tool for logical verification and further synthesizing.

Keywords-Booth encoder; low power; spurious power suppression technique (SPST); SPST-Adder..

I. INTRODUCTION

There is a need for fast multipliers in digital signal processing as speed of multiplication operation determines the speed of the designed circuit. When multiplying two operands if any of the partial products is zero then adding of such a partial product is redundant. Such computations can be reduced by detecting all zero partial products and eliminating them. This reduces power consumption as it reduces the switching transitions.

In this paper modified booth encoder has been used which lesser number of partial products when compared to a radix-2 booth encoder. This reduces the no of partial products given to the adder circuitry in turn increasing the speed of operation. The adder circuitry of a general multiplier is replaced by a SPST equipped adder or subtractor. This SPST adder avoids the unwanted additions and thus minimizes power dissipation.

II. PARTIAL PRODUCT GENERATION

Radix-2 booth algorithm produces 16 partial products. The main bottleneck in terms of speed of this multiplier is the addition of partial products.Radix-4(Modified) booth's algorithm is presented in this paper ,which can reduce the number of partial products by a factor 2 (8partial products) when compared to radix-2 booth's algorithm. Modified booth multiplier operates much faster than an array multiplier for longer operands. Computation time of modified booth multiplier is proportional to the logarithm of the word length of operands.

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.

III. MODIFIED BOOTH ALGORITHM

Step 1: Insert 0 on the right side of LSB of multiplier. The sum of weights of 1's in a binary code gives the decimal value of code. 0 is appended at the end to indentify the presence of 1 in LSB of multiplier.
Step 2: Divide the multiplier into overlapping groups of 3-bits as shown in figure1. Every group has 2(Radix-2^2) new bits and one overlapped bit, which is

ISSN 2278-7763

used to carry the sign information from one group to another.

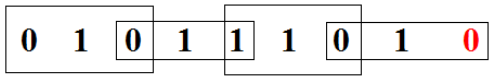

Figure1: Grouping of multiplier

Step 3: If the number of multiplier bits is odd, add an extra bit (1 if signed, else 0) on left side of MSB.

Step 4: Based on the grouping of multiplier, determine product scale factor from modified booth encoding table given in Table1. The encoded digit E_k is obtained from the equation (1).

Table 1: Booth encoding table

Block ($b_{2k+1}b_{2k}\,b_{2k-1}$)	Encoded digit(E_k)	Operation on X
000	0	0X
001	+1	+1X
010	+1	+1X
011	+2	+2X
100	-2	-2X
101	-1	-1X
110	-1	-1X
111	0	0X

$$E_k = b_{2k-1} + b_{2k} - 2b_{2k+1} \quad (1)$$

Step 5: Compute the Multiplicand Multiples

Step 6: When new partial product is generated, each partial product is added 2 bit left shifting in regular sequence.

Step 7: Sum Partial Products using SPST.

IV. SPST TECHNIQUE

Figure 2 shows a computing example of Booth, multiplying two numbers, "6D71" and "005D". The highlighted part 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 effectively reduce the power consumption caused by the transient signals.

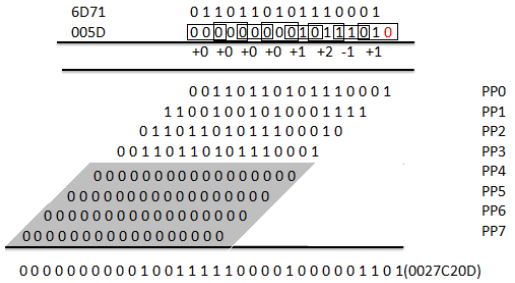

Figure 2: Multiplication using modified Booth encoding.

First four partial products are directly added. The next four partial products are given to SPST equipped adder. Considering the example given above, partial products 4 and 5 added and simultaneously partial products 6 &7 are added. All the partial products are of 17 bits. The SPST sum obtained is of 20 bits each.

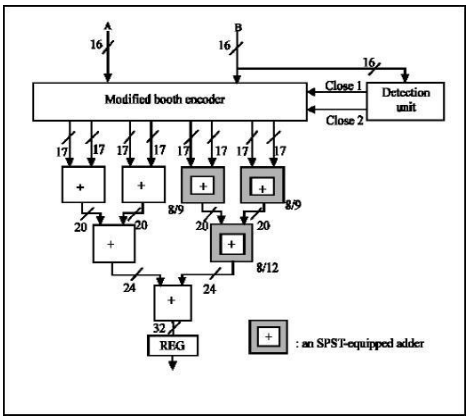

Figure 3: Proposed SPST adder

The obtained two SPST sums are again given to a SPST equipped adder and are of 24 bits as shown in figure 3.

V. ADDER/SUBTRACTOR ADOPTING THE SPST

The 16-bit SPST equipped adder is divided into two parts LSP and MSP. The MSP of the original adder/subtractor is modified to include data controlling circuits, detection logic circuits, sign extension circuits and logics for calculating carry in and carry out signals.

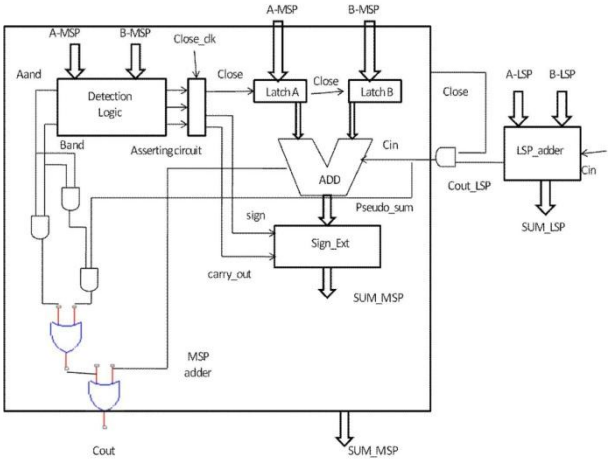

Figure 4: SPST equipped adder

ISSN 2278-7763

The MSP is always of 8 bits and LSP is flexible to have any number of bits. Latches in the Figure 4 are implemented using AND gates, where one input is either multiplier or multiplicand and the other input is close signal. The close signal is obtained from a detection logic circuitry. If close signal is 1, then the multiplier and multiplicand enter the adder circuit and gets added just like the LSP. But, if close signal is 0, then the add/sub unit doesn't receive inputs, but it can still produce the same result by concatenating the sign and carry-out bits. The sign and carry-out signal signals are also outputs of detection logic block. The close signal becomes 0 in 5 cases. There is no other case where close can become 0.

The first case illustrates a transient state in which spurious transitions of carry signals occur in the MSP, although the final result of the MSP is unchanged. Whereas, the second and third cases describe situations involving one negative operand adding another positive operand without and with carry-in from the LSP, respectively. Moreover, the fourth and fifth cases demonstrate the addition of two negative operands without and with carry-in from the LSP, respectively. In those cases, the results of MSP are predictable therefore; the computations in MSP are useless and can be neglected. To know whether the MSP affects the computation results or not, we need the 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.

$$A_{MSP} = A[15:8] \quad B_{MSP} = B[15:8] \tag{2}$$

$$A_{AND} = A[15].A[14].A[13]\ldots\ldots A[8] \tag{3}$$

$$B_{AND} = B[15].B[14].B[13]\ldots\ldots\ldots B[8] \tag{4}$$

$$A_{NOR} = \overline{A[15]+A[14]+A[13]+\ldots A[8]} \tag{5}$$

$$B_{NOR} = \overline{B[15]+B[14]+B[13]+\ldots B[8]} \tag{6}$$

$$Close = \overline{(A_{AND}+A_{NOR}).(B_{AND}+B_{NOR})} \tag{7}$$

$$Carr\_ctrl = (C_{LSP} \wedge A_{AND} \wedge B_{AND}).(A_{AND}+A_{NOR}).(B_{AND}+B_{NOR}) \tag{8}$$

$$Sign = \overline{C_{LSP}}.(A_{AND}+B_{AND}) + C_{LSP}.A_{AND}.B_{AND} \tag{9}$$

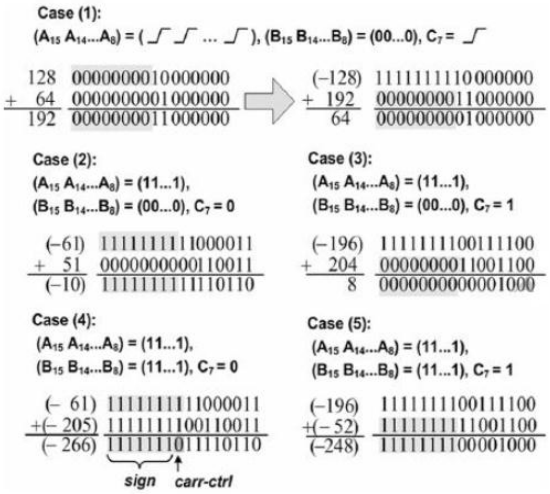

Figure 5: Cases for which close is zero

When both inputs are either all zeros or all ones, close becomes 0 which indicates that the MSP circuits can be closed to save power dissipation and the result is directly computed by the concatenation of sign and carr-ctrl bits. AND gate acts as a all ones detector and NOR gate acts as all zero detector. This design intends to avoid dynamic power consumption by feeding zero inputs into MSP add/sub unit, which freezes the switching activities in the MSP circuits.

VI. SIMULATION RESULTS

In this project we are examining the performance of the proposed high speed low power multiplier with respect of array multiplier. This multiplier can be implemented using Verilog-HDL coding. In order to get the power report and delay report we are synthesizing this multiplier using Xilinx.

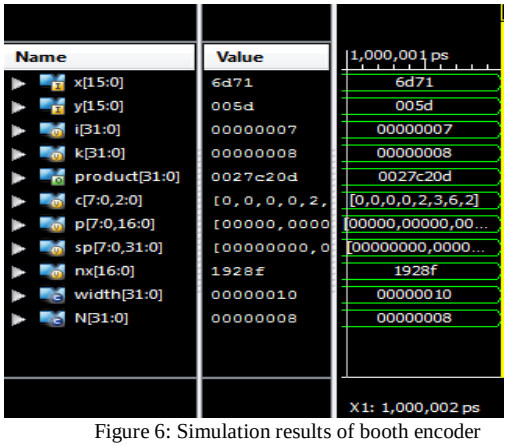

Figure 6: Simulation results of booth encoder

ISSN 2278-7763

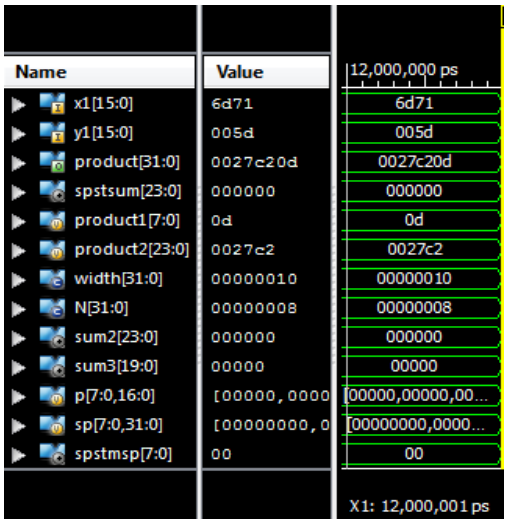

Figure 7: Simulation results of SPST equipped multiplier

A. DELAY COMPARISON

The timing report can be observed in synthesis report. The array multiplier has a delay of 23.79ns, whereas SPST multiplier has a delay of 10.391ns.This proves that SPST multiplier is 56.32% more efficient in reducing the delay when compared to array multiplier.

B. POWER COMPARISON

The power report can be observed in Xilinx power analyzer by applying a clock frequency. The dynamic power and quiescent power of array multiplier with a clock freq of 250MHz are 0.305mW and 0.003mW respectively. The dynamic power and quiescent power of SPST multiplier with the same clock freq are 0.211mW and 0.007mW respectively. The total power of array multiplier is 0.308mW and total power of SPST multiplier is 0.218mW.This proves that SPST multiplier is 41.28% more efficient than that of an array multiplier.

Device	
Family	Spartan6
Part	xc6slx4
Package	tqg144
Grade	C-Grade
Process	Typical
Speed Grade	-3

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.007	1	--	--
Logic	0.014	499	2400	21
Signals	0.015	631	--	--
IOs	0.168	65	102	64
Leakage	0.014			
Total	0.218			

Environment	
Ambient Temp (C)	25.0
Use custom TJA?	No
Custom TJA (C/W)	NA
Airflow (LFM)	0

Characterization	
ADVANCED	v1.1,2010-02-23

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	42.4	75.8	34.2

Figure 8: Power report of SPST multiplier

Device	
Family	Spartan6
Part	xc6slx4
Package	tqg144
Grade	C-Grade
Process	Typical
Speed Grade	-3

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.003	1	--	--
Logic	0.010	390	2400	16
Signals	0.017	528	--	--
IOs	0.262	65	102	64
Leakage	0.015			
Total	0.308			

Environment	
Ambient Temp (C)	25.0
Use custom TJA?	No
Custom TJA (C/W)	NA
Airflow (LFM)	0

Characterization	
ADVANCED	v1.1,2010-02-23

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	42.4	71.9	38.1

Figure 9: Power report of array multiplier

VII. APPLICATIONS

The SPST can drastically reduce the power dissipation of combinational VLSI designs for multimedia/DSP applications. There are several DSP applications, out of which one is an efficient multi-Transform Design (ETD) for MPEG-4 and the second one is a Versatile Multimedia Functional Unit (VMFU) design.

The ETD can compute forward, inverse, hadamard transforms.

On the other hand, the VMFU can compute six commonly used arithmetic operations in multimedia/DSP processing like addition, subtraction, multiplication, MAC (Multiplier and accumulator), interpolation, Sum-of-Absolute-Difference (SAD)

Encapsulating the VMFUs, designers can increase the flexibility and scalability of multimedia/DSP processors. When applying the SPST to these two designs, the realization issues in every design highly differ from each other due to the large hardware-configuration differences. However, with an elaborate design optimization, the proposed SPST can reduce power dissipation by an average of 27% and 24% for the ETD and the VMFU with 1.8V supply voltage, respectively.

VIII. CONCLUSIONS

This work presents the designing of a 16x16 multiplier with high speed low-power technique called SPST (Spurious Power Suppression Technique). A Modified Radix-4 Booth Encoder circuit is used for this multiplier architecture. Compared to array multiplier, the booth multiplier has the highest operational speed and less hardware count.

ISSN 2278-7763

The presented low-power technique called SPST and the theoretical analysis of the SPST are fully discussed. The proposed SPST can obviously decrease the switching (or dynamic) power dissipation, which decreases a significant portion of the whole power dissipation in integrated circuits. Simulation results and Power analysis reports of SPST-equipped multiplier and array multiplier are shown.

REFERENCES

[1] A. Bellaouar and M. I. Elmasry, "Low-Power Digital VLSI Design" Circuits and Systems. Norwell, MA: Kluwer, 1995.

[2] A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," Proc. IEEE, vol. 83, no. 4, pp. 498–523, Apr. 1995.

[3] J. Choi, J. Jeon, and K. Choi, "Power minimization of functional units by partially guarded computation," in Proc. IEEE Int. Symp. Low Power Electron. Des., 2000, pp. 131–136.

[4] J. S.Wang, C. N. Kuo, and T. H. Yang, "Low-power fixed-width array multipliers," in Proc. IEEE Symp. Low Power Electron. Des., Aug. 9–11, 2004, pp. 307–312.

[5] K. H. Chen, K. C. Chao, J. I. Guo, J. S. Wang, and Y. S. Chu, "Design exploration of a spurious power suppression technique (SPST) and its applications," in Proc. IEEE Asian Solid-State Circuits Conf., Hsinchu, Taiwan, Nov. 2005, pp. 341–344.

[6] K. K. Parhi, "Approaches to low-power implementations of DSP systems," IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol. 48, no. 10, pp. 1214–1224, Oct. 2001.

[7] L. Benini, G. D. Micheli, A. Macii, E. Macii, M. Poncino, and R. Scarsi, "Glitch power minimization by selective gate freezing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 8, no. 3, pp. 287–298, Jun. 2000.

[8] M. C.Wen, S. J. Wang, and Y. N. Lin, "Low-power parallel multiplier with column bypassing," Electron. Lett., vol. 41, no. 12, pp. 581–583, May 2005.

[9] O. Chen, R. Sheen, and S. Wang, "A low-power adder operating on effective dynamic data ranges," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 10, no. 4, pp. 435–453, Aug. 2002.

