International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:4, No:1, 2010

Design of High-speed Modified Booth Multipliers Operating at GHz Ranges

Soojin Kim and Kyeongsoon Cho

***Abstract*—**This paper describes the pipeline architecture of high-speed modified Booth multipliers. The proposed multiplier circuits are based on the modified Booth algorithm and the pipeline technique which are the most widely used to accelerate the multiplication speed. In order to implement the optimally pipelined multipliers, many kinds of experiments have been conducted. The speed of the multipliers is greatly improved by properly deciding the number of pipeline stages and the positions for the pipeline registers to be inserted. We described the proposed modified Booth multiplier circuits in Verilog HDL and synthesized the gate-level circuits using 0.13um standard cell library. The resultant multiplier circuits show better performance than others. Since the proposed multipliers operate at GHz ranges, they can be used in the systems requiring very high performance.

***Keywords*—**multiplier, pipeline, high-speed, modified Booth algorithm.

I. Introduction

THE digital signal processing (DSP) is one of the core technologies in multimedia and communication systems. Many application systems based on DSP, especially the recent next-generation optical communication systems, require extremely fast processing of a huge amount of digital data. Most of DSP applications such as fast Fourier transform (FFT) require additions and multiplications. Since the multipliers have a significant impact on the performance of the entire system, many high-performance algorithms and architectures have been proposed to accelerate multiplication[1-10].

Various multiplication algorithms such as Booth[11], modified Booth, Braun, Baugh-Wooley have been proposed. The modified Booth algorithm reduces the number of partial products to be generated and is known as the fastest multiplication algorithm. Many researches on the multiplier architectures including array, parallel and pipelined multipliers have been pursued and the pipelining is the most widely used technique to reduce the propagation delays of digital circuits.

This paper proposes the high-speed multipliers with very deep pipelines based on the modified Booth algorithm. We determined the number of pipeline stages and the positions for the pipeline registers to be inserted by exhaustive experiments. The resultant multipliers show a good performance enough to be used in the very high-speed applications.

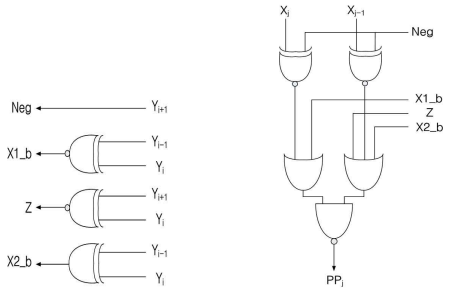
Soojin Kim is with the Department of Electronics and Information Engineering, Hankuk University of Foreign Studies, 89 Wangsan-ri, Mohyun-myun, Cheoin-gu, Yongin-si, 449-791, Republic of Korea(corresponding author to provide phone: +82-31-330-4502; fax: +82-31-330-4120; e-mail: ksjsky9888@ hufs.ac.kr).

Kyeongsoon Cho is with the Department of Electronics and Information Engineering, Hankuk University of Foreign Studies (e-mail: kscho@hufs.ac.kr).

II. Conventional Modified Booth Multiplier

A. Algorithm of the Modified Booth Multiplier

Multiplication consists of three steps: 1) the first step to generate the partial products; 2) the second step to add the generated partial products until the last two rows are remained; 3) the third step to compute the final multiplication results by adding the last two rows. The modified Booth algorithm reduces the number of partial products by half in the first step. We used the modified Booth encoding (MBE) scheme proposed in [1]. It is known as the most efficient Booth encoding and decoding scheme. To multiply X by Y using the modified Booth algorithm starts from grouping Y by three bits and encoding into one of {-2, -1, 0, 1, 2}. Table I shows the rules to generate the encoded signals by MBE scheme and Fig. 1 (a) shows the corresponding logic diagram. The Booth decoder generates the partial products using the encoded signals as shown in Fig. 1 (b).



(a) Booth encoder (b) Booth decoder

Fig. 1 Encoder and decoder for MBE scheme

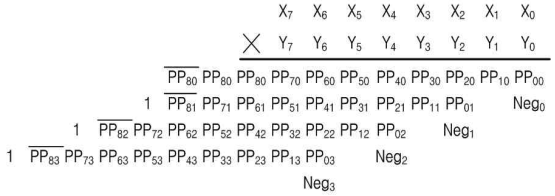


Fig. 2 Generated partial products and sign extension scheme

TABLE I
TRUTH TABLE OF MBE SCHEME

Y_{i+1}	Y_i	Y_{i-1}	Value	X1_b	X2_b	Neg	Z
0	0	0	0	1	0	0	1
0	0	1	1	0	1	0	1
0	1	0	1	0	1	0	0
0	1	1	2	1	0	0	0
1	0	0	-2	1	0	1	0
1	0	1	-1	0	1	1	0
1	1	0	-1	0	1	1	1
1	1	1	0	1	0	1	1

Fig. 2 shows the generated partial products and sign extension scheme[12] of the 8-bit modified Booth multiplier. The partial products generated by the modified Booth algorithm are added in parallel using the Wallace tree[13] until the last two rows are remained. The final multiplication results are generated by adding the last two rows. The carry propagation adder is usually used in this step.

B. Architecture of the Modified Booth Multiplier

Fig. 3 shows the architecture of the commonly used modified Booth multiplier. The inputs of the multiplier are multiplicand X and multiplier Y. The Booth encoder encodes input Y and derives the encoded signals as shown in Fig. 1 (a). The Booth decoder generates the partial products according to the logic diagram in Fig. 1 (b) using the encoded signals and the other input X. The Wallace tree computes the last two rows by adding the generated partial products. The last two rows are added to generate the final multiplication results using the carry look-ahead adder (CLA).

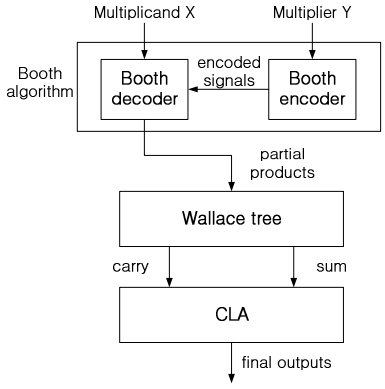


Fig. 3 Architecture of the modified Booth multiplier

III. PROPOSED PIPELINE ARCHITECTURE

The pipeline technique is widely used to improve the performance of digital circuits. As the number of pipeline stages is increased, the path delays of each stage are decreased and the overall performance of the circuit is improved. We investigated various pipeline schemes to find the optimum number of pipeline stages and the positions for the pipeline registers to be inserted in order to obtain the high-speed modified Booth multiplier.

A. Basic 3-stage Pipelined Multiplier

At first, we partitioned the modified Booth multiplier into three pipeline stages according to the functionality of the circuit as shown in Fig. 4. The delay of the critical path of the 3-stage pipelined multiplier is reduced approximately by half compared to the nonpipelined one. The critical path of the 3-stage pipelined multiplier is in the Wallace tree because it requires the most intensive computation. It means that we can reduce the delays further by adding more pipeline registers within the Wallace tree.

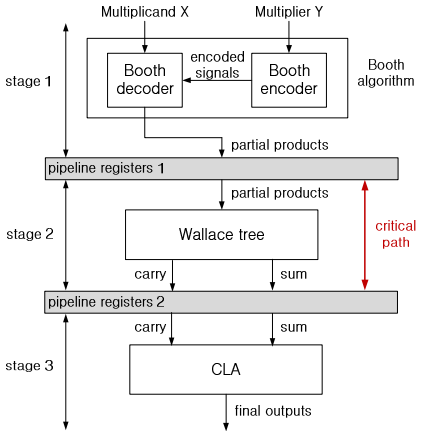


Fig. 4 Modified Booth multiplier with 3-stage pipelines

B. Pipelines in Wallace Tree

The number of partial products of the N-bit modified Booth multiplier is N/2. In case of the 8-bit modified Booth multiplier, the number of partial products is four. Instead of using an additional circuit for the 2's complement operations, we used the signals 'Neg_0' ~ 'Neg_3' as shown in Fig. 2. Therefore, the number of rows is five because of four partial products and 'Neg_3'. The Wallace tree adds three rows and generates two rows. One row represents the carries and the other row represents the sums.

The Wallace tree shows a good performance by using the carry save adders instead of the ripple carry adders. It is, however, still the most critical part of the multiplier because it is responsible for the largest amount of computation. Naturally the pipeline registers should be inserted within the Wallace tree to improve the performance. Considering that the Wallace tree computes the partial products in three steps for the 8-bit modified Booth multiplier, we partitioned the Wallace tree into three pipeline stages as shown in Fig. 5. In case of the 16-bit modified Booth multiplier, the Wallace tree is partitioned into seven pipeline stages.

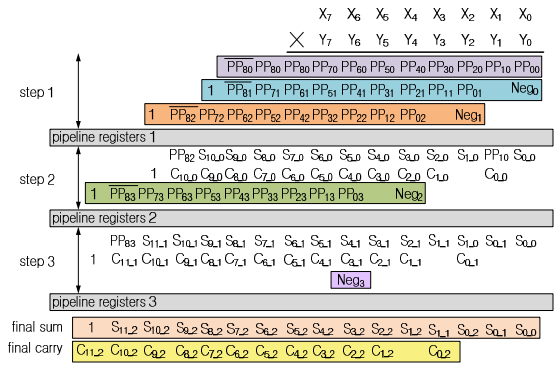


Fig. 5 Wallace tree with 3-stage pipelines

C. Pipelines in Full Adders

The delays of the basic 3-stage pipelined multiplier are reduced by half compared to the nonpipelined multiplier. By applying 3-stage pipelines in the Wallace tree, the delay of the critical path is reduced again by half. In this case, the critical path becomes the full adders in the Wallace tree. We inserted the pipeline registers in the full adders too. Fig. 6 shows the architecture of the 2-stage pipelined full adder. By using this adder in the Wallace tree, the overall performance is improved more.

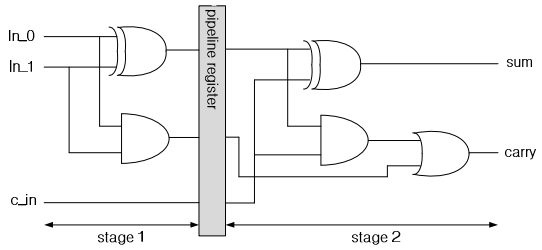


Fig. 6 Full adder with 2-stage pipelines

IV. Experimental Results

In order to improve the performance of the modified Booth multiplier, we applied the pipeline technique for the 8-bit and 16-bit multipliers. Various experiments have been conducted to find the optimum number of pipeline stages and the positions of pipeline registers. This section describes the process of obtaining the high-speed modified Booth multipliers operating at GHz ranges. We described the multiplier circuits in Verilog hardware description language (HDL) and synthesized the gate-level circuits using Design Compiler from Synopsys and 0.13um standard cell library. We verified their functions using NC-Verilog from Cadence.

A. 8-bit Pipelined Modified Booth Multiplier

Four kinds of experiments have been conducted for the 8-bit modified Booth multipliers as shown in Table II. Type A is the nonpipelined modified Booth multiplier in Fig. 3. The 3-stage pipelines are inserted into Type A to design Type B. The 3-stage pipelines are inserted into the Wallace tree of Type B to design Type C. The 2-stage pipelines are inserted into the full adders of Type C to design Type D.

TABLE II
Synthesis Results of Proposed 8-bit Multipliers

Type	# of stages	Gate count	Area increase	Delays(ns)	Speed improvement
A	-	1,179	-	1.49	-
B	3	1,211	x1.03	0.82	x1.82
C	5	1,573	x1.33	0.47	x3.17
D	8	2,022	x1.71	0.34	x4.38

As shown in Table II, the overall delays of the multiplier circuits decrease as the pipeline stages increase. Note that the speed improvement ratios are much larger than the area increase ratios. Among the 8-bit modified Booth multipliers, Type D has the largest number of pipeline stages (eight) and the smallest delays (0.34ns). It operates at 2.95GHz and the speed improvement ratio is 4.38 compared to Type A, nonpipelined multiplier. The critical path of Type D multiplier is in the full adder circuit.

B. 16-bit Pipelined Modified Booth Multiplier

We have conducted 14 kinds of experiments for the 16-bit modified Booth multipliers. Type A ~ Type D multipliers have the same structure as the 8-bit modified Booth multiplier described above. There are six additional experimental results for Type C and Type D multipliers since there are seven steps in the Wallace tree instead of three steps. Table III shows the number of pipeline stages and the positions of pipeline registers in Type C and Type D multipliers. For example, 'Step 12/34/567' means that the pipeline registers are inserted between step 2 and 3, and between step 4 and 5. The synthesis results for the various 16-bit modified Booth multipliers are shown in Table IV.

As shown in Table IV, the overall delays of the 16-bit modified Booth multiplier decrease as the pipeline depth becomes deeper. Note that the speed improvement ratios are much larger than the area increase ratios and this trend is manifested as the number of pipeline stages increases. In some cases, the delays are affected by the positions of the pipeline registers even though the number of pipeline stages remains unchanged. Among the 16-bit modified Booth multipliers, Case 6 of Type D multiplier has the largest number of pipeline stages (16) and the smallest delay (0.38ns). It operates at 2.63GHz and the speed improvement ratio is 6.32 compared to Type A, nonpipelined multiplier. The critical path of Case 6 of Type D multiplier is in the Booth encoder circuit.

Table V shows the comparison results of the proposed multipliers with others. Among the 8-bit multipliers, the proposed one shows the best performance. The multiplier in [10] shows a good performance too. Note that it is designed at

the transistor level. The performance of our circuit will be improved further by optimizing at the transistor level. Our 16-bit multiplier shows the speed approaching to the 8-bit multiplier. The multiplier in [6] is slower than ours in spite that the number of bits is smaller and it uses more advanced processing technology.

TABLE III
VARIOUS CASES OF TYPE C AND TYPE D MULTIPLIERS

Case	Pipeline positions in Wallace tree	# of stages in Wallace tree	
		Type C	Type D
1	Step 12/34/567	3	10
2	Step 12/345/67	3	10
3	Step 123/45/67	3	10
4	Step 1234/567	2	9
5	Step 123/4567	2	9
6	Step 1/2/3/4/5/6/7	7	14

TABLE IV
SYNTHESIS RESULTS OF PROPOSED 16-BIT MULTIPLIERS

Type	Case	# of stages	Gate count	Area increase	Delay(ns)	Speed improvement
A	-	-	3,629	-	2.40	-
B	-	3	4,017	x1.11	1.66	x1.45
C	1	5	4,526	x1.25	0.82	x2.93
	2	5	4,528	x1.25	0.91	x2.64
	3	5	4,528	x1.25	0.91	x2.64
	4	4	4,452	x1.23	1.05	x2.29
	5	4	4,268	x1.17	1.10	x2.18
	6	9	6,069	x1.67	0.47	x5.11
D	1	12	6,834	x1.88	0.57	x4.21
	2	12	6,750	x1.86	0.61	x3.93
	3	12	6,754	x1.86	0.60	x4.00
	4	11	6,400	x1.76	0.73	x3.29
	5	11	6,419	x1.77	0.74	x3.24
	6	16	8,246	x2.27	0.38	x6.32

TABLE V
COMPARISON TO OTHER MULTIPLIERS

	Process	Input bits	Maximum frequency	Pipelined	Post-layout
[5]	0.18um	8x8	1.1GHz	-	-
[6]	0.09um	12x9	2GHz	-	-
[7]	0.35um	8x8	1GHz	○	-
[8]	0.35um	8x8	1.2GHz	○	○
[9]	0.13um	8x8	320MHz	○	○
[10]	0.18um	8x8	2.86GHz	○	○
Ours	0.13um	8x8	2.95GHz	○	-
		16x16	2.63GHz	○	-

V. CONCLUSIONS

The pipelining is the most widely used technique to improve the performance of digital circuits. We proposed the high-speed modified Booth multipliers with very deep pipelines. The proposed multipliers consist of three modules: 1) the modified Booth encoder and decoder module to generate N/2 partial products; 2) the Wallace tree module to add the partial products in parallel; 3) the carry look-ahead adder module for the final addition.

Various experiments have been conducted to find the optimum number of pipeline stages and the positions of pipeline registers for the 8-bit and 16-bit modified Booth multipliers. Using 0.13um standard cell library, we obtained the 8-bit and 16-bit multipliers with the operating frequencies of 2.95GHz and 2.63GHz. Since the speed improvement ratio is much greater than the area increase ratio, they can be used in the application systems requiring very high performance while the area penalty is tolerable.

ACKNOWLEDGMENT

This work was supported by the IT R&D program of MKE/KEIT[2009-F-010-01], and the CAD tools of IDEC.

REFERENCES

[1] Wen-Chang Yeh and Chein-Wei Jen, "High-speed Booth encoded parallel multiplier design," IEEE Trans. on Computers, vol. 49, isseu 7, pp. 692-701, July 2000.
[2] Jung-Yup Kang and Jean-Luc Gaudiot, "A simple high-speed multiplier design," IEEE Trans. on Computers, vol. 55, issue 10, Oct. pp. 1253-1258, 2006.
[3] Shiann-Rong Kuang, Jiun-Ping Wang and Cang-Yuan Guo, "Modified Booth multipliers with a regular partial product array," IEEE Trans. on Circuit and Systems, vol.56, Issue 5, pp. 404-408, May 2009.
[4] Li-rong Wang, Shyh-Jye Jou and Chung-Len Lee, "A well-structured modified Booth multiplier design," Proc. of IEEE VLSI-DAT, pp. 85-88, April 2008.
[5] A. A. Khatibzadeh, K. Raahemifar and M. Ahmadi, "A 1.8V 1.1GHz Novel Digital Multiplier," Proc. of IEEE CCECE, pp. 686-689, May 2005.
[6] S. Hus, V. Venkatraman, S. Mathew, H. Kaul, M. Anders, S. Dighe, W. Burleson and R. Krishnamurthy, "A 2GHZ 13.6mW 12x9b mutiplier for energy efficient FFT accelerators," Proc. of IEEE ESSCIRC, pp. 199-202, Sept. 2005.
[7] Hwang-Cherng Chow and I-Chyn Wey, "A 3.3V 1GHz high speed pipelined Booth multiplier," Proc. of IEEE ISCAS, vol. 1, pp. 457-460, May 2002.
[8] M. Aguirre-Hernandez and M. Linarse-Aranda, "Energy-efficient high-speed CMOS pipelined multiplier," Proc. of IEEE CCE, pp. 460-464, Nov. 2008.
[9] Yung-chin Liang, Ching-ji Huang and Wei-bin Yang, "A 320-MHz 8bit x 8bit pipelined multiplier in ultra-low supply voltage," Proc. of IEEE A-SSCC, pp. 73-76, Nov. 2008.
[10] S. B. Tatapudi and J. G. Delgado-Frias, "Designing pipelined systems with a clock period approaching pipeline register delay," Proc. of IEEE MWSCAS, vol. 1, pp. 871-874, Aug. 2005.
[11] A. D. Booth, "A signed binary multiplication technique," Quarterly J. Mechanical and Applied Math, vol. 4, pp.236-240, 1951.
[12] M. D. Ercegovac and T. Lang, *Digital Arithmetic*, Morgan Kaufmann Publishers, Los Altos, CA 94022, USA, 2003.
[13] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. on Computers, vol. BC13, pp. 14-17, Feb. 1964.