

Design of RBSD Adder and Multiplier Circuits for High Speed Arithmetic Operations and their Timing Analysis

Neelam Sharma, B.S. Rai and Arun Kumar

Sr. Lecturer, E&C Department, Institute of Technology and Management,
Gida, Gorakhpur, India
neelam_sr@yahoo.com

Abstract. RBSD Adder / RBSD Multiplier circuits are logic circuits designed to perform high-speed arithmetic operations. These high-speed arithmetic machines add and multiply numbers using Redundant Binary Signed Digit number system. In RBSD number system carry propagation chain are eliminated which reduce the computation time substantially, thus enhancing the speed of the machine. A novel design of RBSD Adder cell made of NOR gates only are proposed in this paper. A pipelined design of Multiplier cell using RBSD Adders is also suggested. The speed of these high-speed circuits is compared to the speed of CLA adder / Wallace Booth Multiplier (WBM) which are being used conventionally for high-speed calculation. Comparison yields favorable results although RBSD machines may not be convenient for manual computations but they are of great advantage for design of high-speed machines. The design of these circuits is carried out using EDA tools. The designs are simulated and synthesized using VHDL software, more specifically modelsim software is used for simulation, Xilinx project navigator and Leonardo Spectrum for synthesis. Timing analysis reports for these circuits and graphs drawn for delay time give favorable comparative results.

Index Terms: RBSD, CLA adder, WB Multiplier, Carry free addition, high speed arithmetic, carry chain, VHDL.

1 Introduction

Arithmetic operations play an important role in various digital systems such as computers, process controllers, signal processors computer graphics and image processing. Using non-conventional number system such as signed digit numbers for fast arithmetic units is particularly gaining much attention in recent years. Signed digit number system offers the possibility of carry free addition by taking advantage of redundancy associated with the representation. Several papers have been published in recent years exploring the use of signed digit numbers for fast arithmetic units. With recent development in technology of integrated circuits, various high speed circuits with regular structures have been proposed and some of them have been fabricated on VLSI chips since many digital systems are designed to operate fast with

high reliability, not only high speed operations and regular structures but also fault tolerant features should be implemented in arithmetic circuits.

RBSD number representation possesses sufficient redundancy to allow for the annihilation of carry or borrow chains and hence result is past, propagation – free addition and subtraction. Signed – digit number not only allows for carry – free addition and borrow – free subtraction but also offers important advantages for the practical implementation of arithmetic functions. These numbers are in practical use for representing intermediate values of in 2's complement and high speed multiplication schemes ever since multiplier recording was introduced by Booth. They have also been used in redundant quotient representation for the S-R-T division algorithm which was proposed independently by Sweeney, Robertson, and Tocher and which drives its name from their initials

VLSI Technology has given designers the freedom to integrate many complex components, which may not be possible in the earlier days. Multiplication is nothing but repetitive addition of partial products. It involves two basic operations 1). The generation of partial products 2) their accumulation to produce the final product. Hence if fast multiplication is desired either the numbers of partial products are reduced or the time needed to accumulate or add them is decreased.

Modified Booth Encoding (MBE) in which the partial products are reduced to half the original value is used.. After generation of partial products, they must be accumulated to obtain the final product. Fast multi-operand adders such as Wallace trees and Carry Save Adders (CSA) tree should be employed for high speed accumulation [4] . However, tree structures generally have very irregular interconnections that complete the implementation and more importantly result in Area inefficient layouts. RBSD Adders are more suitable for fast addition. They take a finite amount of time to generate the result. Since a three-input/two-output adder (3-2 Compressor, also called CSA) is most commonly used in Wallace trees, the partial product reduction rate is 3/2. The reduction rate of 2/1 can be accomplished by using RBSD Adders. Wallace trees, CLA adders and CSA adders are conventionally used for high speed addition. Similarly Wallace Booth Multiplier is used for fast multiplication. In this paper it is proved by virtue of simulation results that RBSD adder/multiplier are superior to the adders/multipliers used conventionally for high speed addition/multiplication. Adders/Multipliers of 4, 8, 16, 32, 64 bit are simulated and synthesized using EDA (Electronic Design Automation) softwares and the results yielded are discussed in the paper.

Timing analysis of high speed RBSD adder will be compared with that of CLA adder. Similarly, timing analysis of high speed RBSD multiplier will be carried out and compared with conventional Wallace booth multiplier.

2 Salient Features of RBSD Numbers

The following additional advantages are offered by RBSD numbers other than their carry-free addition property.

- Ease of multiplication, division, and other arithmetic operations.
- Ease of zero detection.

- Suitability of use with error codes.
- Pipelined approach.

The stored-carry representation was derived in connection with the design of past multi operand adders. Even though other arithmetic operations such as multiplication and division are performed on stored-carry numbers, the required hardware is much more complex than those of RBSD. More so the regular structure of the RBSD adder cell proposed by Kal and Rajshekhar makes the design regular, pipelined and suitable for VLSI implementation.

RBSD number system's suitability for use with arithmetic error codes is a direct result of the compatibility of this representation with Binary [10]. Particularly the effective studies performed for arithmetic error codes in connection with the binary system carry over with little change to the RBSD system unlike stored-carry system. Zero has a unique RBSD representation and can thus be easily detected when needed. This is also possible in stored carry representation but requires more hardware.

2.1 Background

The credit of redundant signed digit goes to Robertson (1959) and Avizienis (1961) [3]. They suggested a set of rules for RBSD Arithmetic. Chow and Robertson suggested the idea of logic design of the RBSD adders. Takaji and Azima proposed high speed algorithm with RBSD numbers suitable for VLSI implementation.

2.2 Redundant Signed Digit Number System Preliminaries

Signed digit representation limit carry propagation to one position to the left during the operation of addition and subtraction in digital computers. Carry-propagation chains are eliminated. Redundancy in the number representation allows a method of fast addition and subtraction in which each sum digit is a function only of the digits in two adjacent digital position. In RD representation with radix r a digit can have more than r values. SD (Signed Digit) number can assume $2\alpha+1$ values.

$$\Sigma_r = \{-\alpha, \dots, -1, 0, 1, \dots, \alpha\} \quad (1)$$

α must be within the following region

$$\begin{aligned} (r-1) / 2 &\leq \alpha \leq r-1 \\ \alpha &\geq 1 \text{ \& } r \geq 2 \end{aligned} \quad (2)$$

A SD number is represented by $n+m+1$ digits

$$Z_i = (i = -n, \dots, -1, 0, 1, \dots, m) \quad (3)$$

has the algebraic value m

And equivalent SD number

$$Y = (Y_{n-1}, \dots, Y_1, Y_0)_r$$

For every difference digit

$$d_i = x_i - r b_{i+1} \quad (4)$$

where the borrow digit

$$b_{i+1} = \begin{cases} 0 & \text{if } x_i < \alpha \\ 1 & \text{if } x_i \geq \alpha \end{cases} \quad (5)$$

Let the number 12 be changed to RBSD representation and back to binary. Note that X_i stands for RBSD number n and Y_i for binary form of the number. $X = (1100)_2$ with $r=2$, $n=4$ and $\alpha=1$ for a given SD set $(-1, 0, 1)$. The SD number Y is $(10 \ 100)$ as obtained in the Table 1.

Table 1 : Binary to RBSD conversion

Digit Position	4	3	2	1	0
X_i		1	1	0	0
d_i		$\bar{1}$	$\bar{1}$	0	0
b_i	1	1	0	0	
Y_i	1	0	$\bar{1}$	0	0

12 is represented as 10 100 i.e. 16-4 as bar stands for -1

Here d_i and b_i represent the sum and borrow digits and are same as in equations (4) and (5) respectively.

Conversion back to conventional radix

$$Y_i = Y^+ + Y^-$$

$$Y = (1 \ 0 \ \bar{1} \ 0 \ 0)_2$$

$$Y^+ = (1 \ 0 \ 0 \ 0 \ 0)_2 \rightarrow \text{positive ones are replaced by 1}$$

$$Y^- = (0 \ 0 \ 1 \ 0 \ 0)_2 \rightarrow \text{negative ones are replaced by 1}$$

$$X_i = (0 \ 1 \ 1 \ 0 \ 0)_2 \rightarrow \text{Subtract to obtain } X$$

$$X_i = \text{back to conventional binary}$$

3 RBSD Adder

RBSD adder structure with a digit set $\{-1, 0, 1\}$ per addition operation by “Two Transfer Addition Technique” which is computed in three stages

$$x_i + y_i = 2 * t'_{i+1} + w'_i \quad (6)$$

$$w'_i + t'_i = 2 * t''_{i+1} + w''_i \quad (7)$$

$$s_i = w''_i + t''_i \quad (8)$$

where x_i and $y_i \in \{1, 0, 1\}$

t''_i and t'_i are intermediate carriers called transfer digits of i^{th} stage. w'_i and w''_i are intermediate sums in the i^{th} stage.

S_i is the final sum [1,2]. The sum of operands is realized using the following steps:

Step-1: The transfer digit $|t'_{i+1}|=1$ wherever $|x_i+y_i| \geq 1$

Step-2: The transfer digit $|t''_{i+1}|=1$ only if $|t'+w'| \geq 2$ is completed and w'' is computed using formula (2)

Step-3: $s_i = w'' + t''$

Under the above conditions w'' and t'' cannot be both +1 and -1 thus giving carry free addition. To implement the circuit the chosen set (1, 0), (0,0) and (0,1) to represent (-1), (0) and (1) respectively. RBSD adder cell for addition of two bit RBSD numbers was generated regular structures of RBSD adder cell is an added advantage over irregular structure of CLA adder cell in VLSI implementation of the structure. CLA adders circuits implemented for fast addition suffer from fan-in and fan-out limitations. These limitations depend on logic families employed, TTL circuits for example are limited to $n=12$. If m_i , b_i and d_i are binary variables with digit set $\{0,1\}$ and x_i^* , y_i^* , s_i^* are variables of digit set $\{-1,0,1\}$ represented by two bit binary (1,0),(0,0),(0,1) respectively. For this chosen representation corresponding to x_i^* , y_i^* and s_i^* are $x_i^+ x_i^-$, $y_i^+ y_i^-$ and $s_i^+ s_i^-$ respectively. For this chosen representation, the Boolean expressions for d_i , m_{i+1} , b_{i+1} , s_i^+ and s_i^- are as follows :

$$d_i = m_i \oplus \bar{x}_i^+ \bar{x}_i^- \oplus \bar{y}_i^+ \bar{y}_i^- \quad (9)$$

$$m_{i+1} = \bar{x}_i^+ \bar{y}_i^+ \quad (10)$$

$$b_{i+1} = \bar{m}_i \bar{x}_i^+ \bar{x}_i^- + x_i^+ y_i^+ + y_i^+ \bar{y}_i^- \bar{m}_i + \bar{x}_i^+ \bar{x}_i^- \bar{y}_i^+ \bar{y}_i^- \quad (11)$$

$$s_i^+ = d_i \bar{b}_i \quad (12)$$

$$s_i^- = d_i b_i \quad (13)$$

The RBSD Adder cell used to add two bit RBSD numbers ($x_i^+ x_i^-$) and ($y_i^+ y_i^-$) suggested by Kal Rajashekhar are shown in figure (1) [2]. The adder structure shown in the figure is used to perform parallel addition of two numbers four bit RBSD numbers.

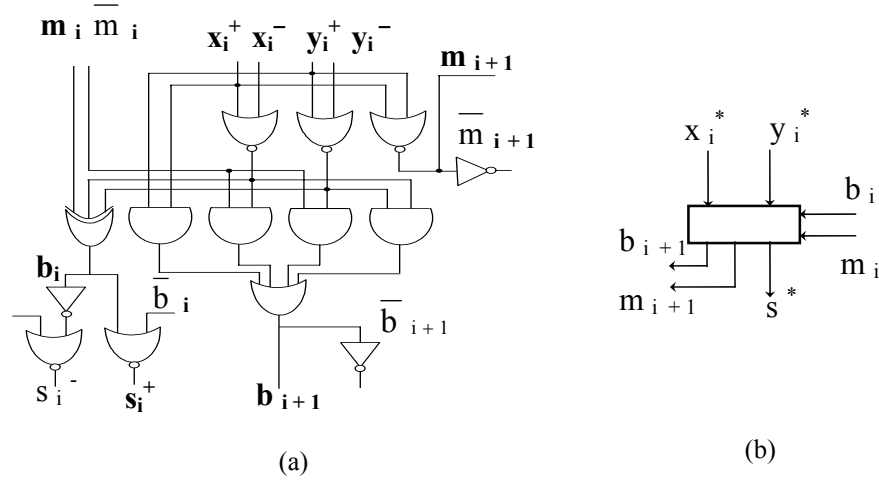


Figure 1. (a) Logic diagram of RBSD adder cell,
(b) Block diagram of RBSD adder cell

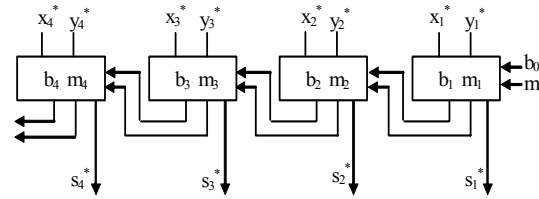


Figure 2. Block diagram of addition of two four bit RBSD numbers

3.1 Logic Design of RBSD Adder using Universal Logic

Kal and Rajashekhar designed an RBSD Adder cell made of AND, NOR, XOR, OR, and INVERTERS. However NAND and NOR operations are each functionally complete. It is highly desirable to construct digital circuits of NAND or NOR Gates because of simplicity and uniformity of circuits which have just a single primitive component [13]. And since these gates can be manufactured quite inexpensively they contribute to the major components used today by logic designers. NAND and NOR gates are also available as integrated circuit SSI packages. These packages vary in sizes and may contain from one to four gates per package. In digital design techniques, much attention has to be paid to network design in the form of repeated pattern of identical circuits. Having a similar type of gate (NAND or NOR) also makes nMOS and CMOS design easy for the designer since there is a repetition of similar design.

The proposed Adder cell in this paper is made of NOR gates. This Adder design is verified using PSPICE software and its VHDL code is also generated. VHDL code for the Adder cell proposed by Kal and Rajashekhar is also generated. Both the

designs of RBSD Adder cells are simulated and synthesized and the Timing Reports generated. The time delay for the RBSD Adder proposed by Kal and Rajashekhar is 12.11 ns and for the NOR RBSD Adder cell proposed and given in figure (2) is 10.532 ns. Thus this design proved faster. Adder cells are the building blocks of multipliers and can therefore this design can increase the speed of the multiplier multifold.

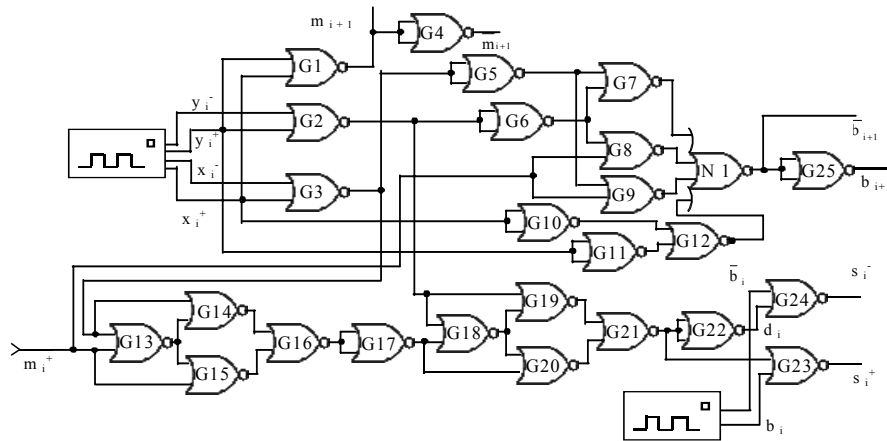


Figure 3. RBSD Adder Cell using NOR-NOR Logic

3.2 Comparison between RBSD Adder Cells

The two different types of RBSD Adder cells 1) Ordinary (the one suggested by Kal and Rajashekhar) 2) NOR-NOR – Proposed cells can be compared. The Adder structure have the same inputs and outputs but they only differ in type of gates that they are made of. The table below list the gate types and no. of gates used for proposed Adder and the Adder suggested by Kal and Rajashekhar. It also gives the time delay and hardware used on FPGA.

3.3 Comparative Timing Analysis of CLA and RBSD adder

VHDL implementations of CLA and RBSD adders were carried out. N bit RBSD adder structure and CLA adder were simulated with modelsim software from FPGA advantage. These adders were synthesized with Xilinx project navigator and their timing reports viewed for values $n=4, 8, 16, 32$ and 64 .

A comparative graph of CLA and RBSD adder structures is shown below. We observe that the delay produced in RBSD adder is independent of wavelength of the string to be added and has regular structure. Hence for higher values of n RBSD adders structure is ideal for VLSI implementation.

It is observed from table 2 that the delay produced by RBSD Adder is less as compared to CLA Adder for bit lengths 8, 16, 32 & 64. RBSD Adders are most suited for higher bit lengths.

Table 2. Comparative table of no. gates and gate types along with the time delay and hardware used for different RBSD Adder cells.

S. No.	Type of RBSD Adder Cell	No. of gates used	Hardware Used on FPGA	Time Delay (ns)
1	Ordinary RBSD Adder Cell	2 I/P AND – 4 4 I/P OR – 1 2 I/P NOR – 5 3 I/P XOR – 1 INVERTERS – 3	No. of Slices – 5 out 768 No. of 4 I/P LUT's – 9 out of 1536 No. of bonded IOB's – 14 out of 96	12.122 ns (7.091 ns logic 5.031 ns route)
2	NOR-NOR RBSD Adder Cell	2 I/P XOR – 25 4 I/P NOR – 1	No. of Slices – 4 out 768 No. of 4 I/P LUT's – 7 out of 1536 No. of bonded IOB's – 12 out of 96	10.538 ns (6.857 ns logic 3.681 ns route)

Table 3. Timing Analysis of CLA and RBSD Adder

S.No.	No. of Bits	Delay (ns) CLA	Delay (ns) RBSD
1	4	13.6	16.51
2	8	19.5	16.51
3	16	24.73	16.51
4	32	27.5	16.51
5	64	30.43	16.51

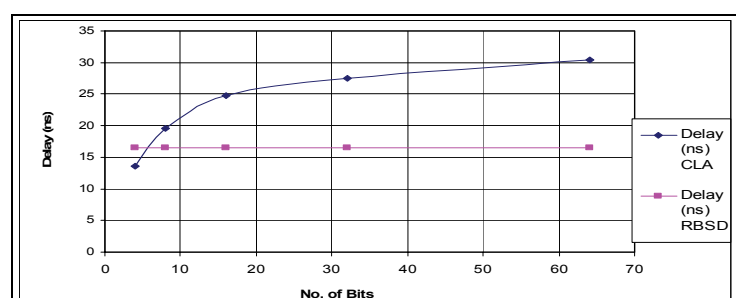


Figure 4. Timing comparison of CLA and RBSD Adder

4 RBSD Multiplier

Multiplication is the most critical operation in every computational system. In the proposed RBSD Multiplier scheme modified booth encoding is used. And n bit multiplication can be carried out with enhanced speed. The multiplier has a regular array structure thus suitable for VLSI implementation [7,8]. The combination of Modified Booth Encoding and Redundant Binary Signed-Digit (RBSD) Adders has been used to implement the multiplier using VHDL.

The various steps involved in the multiplication process are:

- Take two binary numbers one multiplicand and other multiplier
- Convert the multiplicands from two's complement form to RBSD representation (multiplier is retained in 2's complement form)
- Generate $n/2$ RBSD partial products using Booth's bit pair recording technique on the n -bit two's complement multiplier and the RBSD multiplicand.
- Add the partial products pair wise by means of a binary tree of $\log_2 (n/2)$ levels of adders.
- Convert the resulting RBSD product into two's complement form.

4.1 Pipelined Design for Multiplication using RBSD Adders

A pipelined system divides the logic into stages separated by a set of latch registers. Pipelining is a technique to provide further increase in bandwidth by allowing simultaneous execution of many tasks. By pipelining the various stages of a RBSD Multiplier structure, its efficiency will increase thus decreasing the computation rate. The major advantage of pipelining over other parallel design techniques is that improvement in performance can be obtained for less cost [11,12].

The 16×16 fixed-point multiplication units can be divided into six pipeline stages. Stage S_1 generates the partial product and stage S_2 converts the partial products to RBSD numbers. Stages S_3 , S_4 and S_5 stages add the RBSD numbers. Here of is important to note that since RBSD adder delay time is independent of word length with increased complexity, stages S_3 , S_4 and S_5 will have the same delay time. Stage S_6 is a CLA adder used to convert the RBSD number in two's complement form. Delay times of stages S_1 , S_2 and S_6 can be further adjusted. The pipeline is thus divided into six stages and the clock rate can be fixed as desired to give appropriate results. Figure (6) shows the pipelined unit of a RBSD multiplier.

4.2 Comparative Timing Analysis of WBM and RBSD Multiplier

VHDL implementation of WBM and RBSD multipliers are carried out and the designs of these multipliers are simulated with the help of modelsim. The designs are synthesized using Xilinx Project Navigator. Timing Analysis report is generated for 4, 4, 16, and 32 bit WBM and RBSD multipliers and a comparative graph is

generated. It is observed that 4 bit WBM is faster than 4 bit RBSD adder but as the bit length of the multiplier and multiplcand to be multiplied increases the delay of RBSD Multiplier is much less than that of WBM. The graph also shows that with the increase of bit length the RBSD Adders prove much faster. Thus even though RBSD operations are unsuitable for manual computations but they are ideal for design of fast ALU.

5 Computation of Elementary Functions

Multipliers can be used as building blocks for computation of elementary functions. Fast RBSD multipliers discussed in this paper can be used for these computations thus enhancing the speed of these circuits as well. The computation of reciprocal and square root has been considered of importance for many years since there functions appear is many applications. Recently inverse square root has also received attention because of increased significance of multimedia and graphics applications [9]. More so, because if its similar characteristics, it is considered advantages to have a single scheme to implement all three functions. Computation of all there elementary functions can be done using small tables, small multipliers, and for since functions, a final large multiplication. The strength of this method is that the same allows the computation of all the functions. Computation of logarithms and Exponentials can also be done using the same scheme. Here it is noted that the same basic computations are performed for all various functions. Computational decay of elementary functions performed by this method is quite close to other related methods, but it is significantly farther for square root and inverse square root.

Table 4. Timing Analysis of WBM and RBSD Multiplier

S.No.	No. of Bits	Delay (ns) WBM	Delay (ns) RBSD
1	4	31.63	41.2
2	8	59.39	55.07
3	16	107.19	63.72
4	32	399.25	229.4

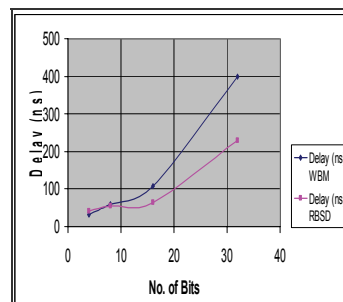


Figure 5. Timing Comparison of WBM and RBSD Multiplier

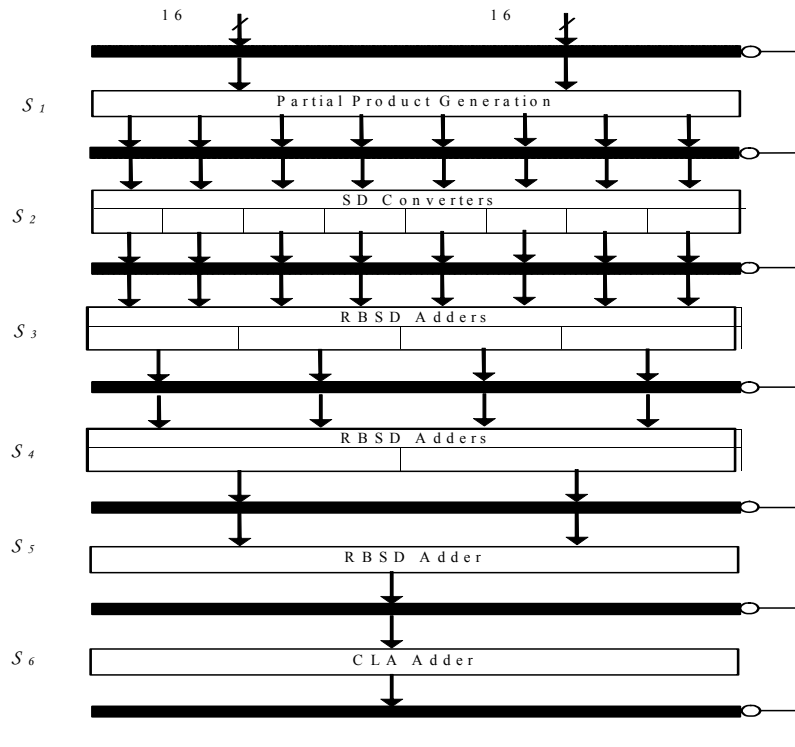


Figure 6. A pipeline unit for fixed-point multiplication of 16 x 16 bit numbers.

6 Conclusion

RBSD Machines are excellent in both the computation speed and regularity in layout. RBSD number system offers faster add times because of carry free addition. Regularity of adder structure is of great advantage for its VLSI implementation. RBSD machines offers higher comparative speed if the number of bits to be added is greater. A 4, 8, 16, 32 and 64 bit conventional high speed machines compared with RBSD arithmetic machines so that higher number of bits show great advantages. Adder and multiplier circuits discussed can be used as building blocks for other RBSD arithmetic machines which is a topic of continued research. If we use ternary logic it offers reduced circuit complexity in both transistor form and interpolations. The design of RBSD Adder cell made of NOR gates only proposed in this paper proved faster than the one proposed by Kal and Rajashekhar. Adder cells are the building blocks of multipliers and can therefore this design can increase the speed of the multiplier multifold. Pipelined Design of Multiplier can also make compact circuits with reduced cost and increased speed. A further effort for implementing RBSD Adder cells with NAND gates only is also been carried out.

Even though this technique can be applied to intermediate computation results in order to speed up conventional binary computers, its main application area is in the design of special purpose arithmetic “engines” which deal with long sequences of computations and / or long operands. Since conventional binary numbers are easily converted to RBSD numbers, and since re-conversion can be accomplished by a single binary subtraction, compatibility between the two (conventional and RBSD) computational systems can be achieved.

References

1. Hwang K.(1979). Computer Arithmetic/Principles, Architecture and Design .New York :Wiley
2. Rajashekhar T.N.and Kal O. (1990). Fast Multiplier Design using Redundant Signed- Digit Numbers, International Journal of Electronics vol .69,No. 3, pp – 359-368.
3. Avizienis A. (1961).Signed digit number representation for fast parallel arithmetic, IRE Trans. Electron Computers.,vol EC-10,pp.389-400
4. Pucknell D.A. and Eshraghian K.(2002).. Basic VLSI Design PHI Third Edition
5. Yeh,Wen-Chang and Jen Chein Wie (2000).High speed Booth encoded parallel multiplier design , IEEE Transaction on Computers,vol.49,No.7 .
6. Darninger J. ,Halthaniany David J.H., Koenemann, B., Lavin M. and others .(2000). EDA is IBM : Past Present and Future, IEEE Transaction on computers Aided Integrated circuit and system, vol. 19, No. 12.
7. Besli N. and Deshmukh R.G(2002). A novel Redundant Binary Signed –Digit (RBSD) Booth’s Encoding., Proceedings IEEE Southeastcon
8. Rajashekhar T.N. and I-Shi Eric Chen. (1990). A Fast Adder Design Using Signed-Digit Numbers and Ternary Logic.,Proc.1990 IEEE Southern Tier Technical Conference, pp.187-194,Binngamton,New York.
9. M D Ercegovac, J M Muller, Tisseran ,(2000). Reciporcation, square root, Inverse square root and some Elementary functions using small multipliers, IEEE Transactions on computers, vol 49, No. 7. .
10. Parhami B(1988). ‘Carry free Addition of Recorded Binary Signed-Digit Numbers’, IEEE Transactions on computers Vol. 37 No. 11, pp. 1470-1476.
11. Hallin T. G. and Flynn M. J. ,(1972)., Pipelining of Arithmetic Functions, IEEE Transactions of computers, pp. 880-885.
12. Jump J. R.and Ahuja S. R,(1978).Effective Pipelining of Digital Systems, IEEE Transaction on computers Vol. C-27, No. 9, pp. 855-865.
13. Kohavi Zvi (2002). Switching and Finite Automata Theory, Tata McGraw-Hill, Second Edition