UGC Approved Journal

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 6, June 2017

Low Power 4*4 Canonical Signed Digit Multiplier using 90nm Technology

Saloni[1], Dr. Neelam Rup Prakash[2]

M. Tech Student, Dept of Electronics and Communication Engg, PEC University of Technology, Chandigarh, India [1]

Professor, Dept of Electronics and Communication Engg, PEC University of Technology, Chandigarh, India [2]

Abstract: Designing of low power arithmetic circuit is done considering four levels; algorithm, architecture, circuit and system levels. The most commonly used component of many digital circuit designs is digital multipliers. In order to achieve high data throughput, digital signal processing systems rely on hardware multiplication. There are a number of multipliers available for different applications. This paper focuses on an algorithm, called Canonical Signed Digit Multiplication. This paper presents the performance comparison of the proposed multiplier ie. CSD Multiplier with Array multiplier in terms of power. A Design of 4X4 Multiplier using 90nm Technology is successfully synthesized. Cadence Virtuoso 90nm Technology is used for simulation of the Design. The simulated transient output of 4X4 Multiplier is shown. Multiplier circuit works with 1.8 V power supply. Simulation results obtained show 99.56% reduction in power.

Keywords: Multiplier, Partial Product, Array Multiplier, CSD, Power.

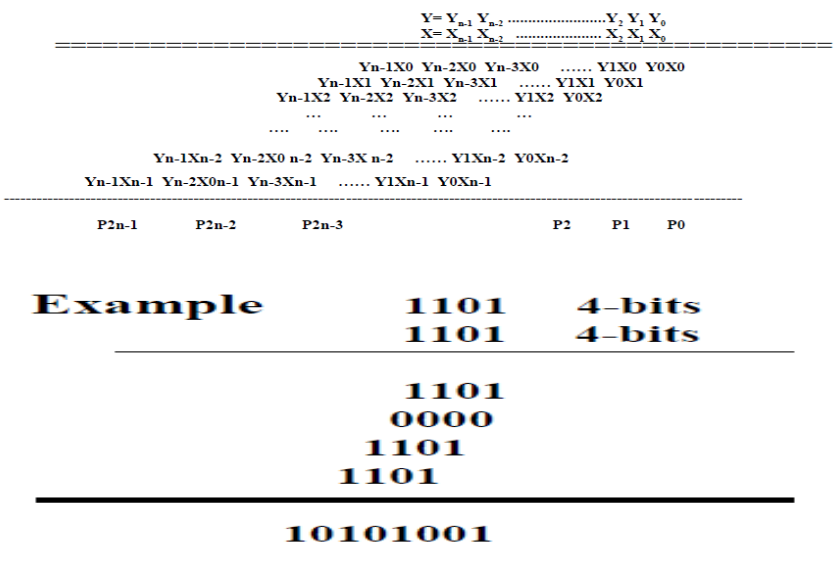
I. INTRODUCTION

1.1 INTRODUCTION TO MULTIPLIER

Multiplication is a mathematical operation which is simply an abbreviated process of adding an integer to itself a specified number of times. The operation of multiplication is rather simple in digital electronics. This algorithm uses addition and shift left operations to calculate the product of two numbers. Based upon the above procedure, an algorithm can be deduced for any kind of multiplication. The multiplication algorithm for an N bit multiplicand by N bit multiplier is shown below:

$Y = Y_{n-1}\ Y_{n-2}\ \ldots\ldots\ldots\ldots\ldots\ldots Y_2\ Y_1\ Y_0$ Multiplicand
$X = X_{n-1}\ X_{n-2}\ \ldots\ldots\ldots\ldots\ldots\ldots X_2\ X_1\ X_0$ Multiplier

Generally



Example

1101 4-bits
1101 4-bits

1101
0000
1101
1101

10101001

DOI 10.17148/IJIREEICE.2017.5635

UGC Approved Journal
IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2017

AND gates are used to generate the Partial Products, PP, if the multiplicand is N-bits and the Multiplier is M-bits then there is N* M partial product. The way that the partial products are generated or summed up is the difference between the different architectures of various multipliers. The entire process consists of three steps- partial product generation, partial product reduction and final addition [8].

Digital multiplication consists of three basic steps, these are:-
1. Generation of Partial Product Array
2. Reduction of Partial Product Array
3. Final addition

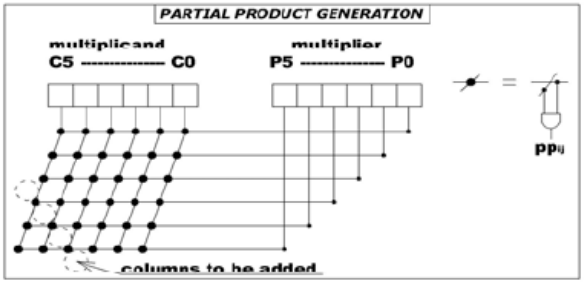


Figure1. Partial Product Generation

1.2 ARRAY MULTIPLIER
Array multiplier has a regular structure; it has a two dimensional structure that fits nicely on the VLSI planar process. There are several possible array topologies including simple, double and higher order arrays.

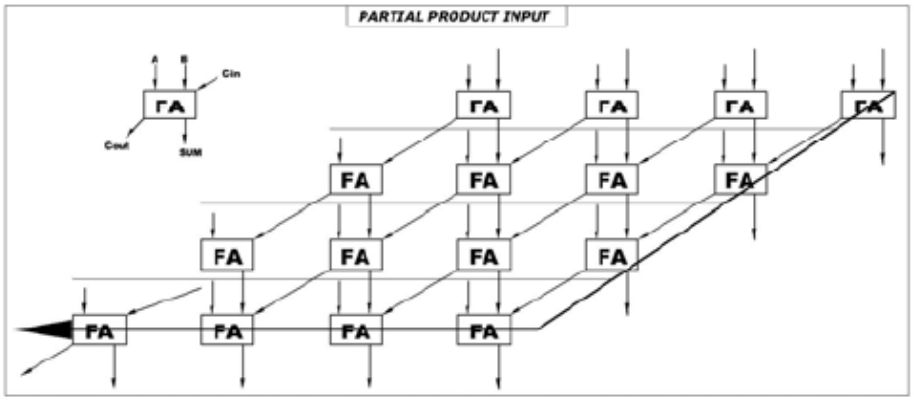


Figure2. Array Multiplication Algorithm

The multiplication parallelogram gave rise to array multiplier. As shown in Figure 2, at each level, the parallel adders receive inputs i.e. the partial products. The carry which is generated in each row is propagated into the next row. The critical path of the multiplier is shown using bold line in the above figure. In array multiplier, all of the partial products are generated at the same time. The critical path consists of two parts: vertical and horizontal. Each of them has the same delay in terms of full adder delays and gate delays. Advantage of the array multiplier comes from its regular structure. Since it is regular, it is easy to layout and has a small size.
The rest of the paper is organized as follows: Proposed algorithm is explained in section II. Simulation results are presented in section III. Discussion and concluding remarks are given in section IV.

II. PROPOSED ALGORITHM

2.1 Canonical Signed Digit:
The canonical signed digit (CSD) representation is a signed digit (SD) representation with unique features which make it useful in the applications which require low-power, efficient-area and high-speed arithmetic [1]. The CSD code is a ternary number system whose digit set is $\{1^{-}\ 0\ 1\}$, where 1^{-} stands for -1. The CSD representation is a unique representation and has two main properties:

UGC Approved Journal
IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2017

(1) the number of nonzero digits is minimal, and
(2) each non zero digit should be surrounded by zeros

According to the first property, a minimal Hamming weight is assigned to a number. This leads to a reduction in the number of additions in arithmetic operations. The second property implies that the code is unique. However, if this property is relaxed, this representation is called the minimal signed digit (MSD) representation, which has as many non-zeros as the CSD representation, but provides multiple representations for a constant [2,3]. Efficient multipliers can be implemented using CSD code and this has been adopted by a number of researchers [4–6]. It reduces the number of partial products so as to make multiplication fast and reduces power consumption and area structure of a multiplier for signal processing applications [7].
This multiplier is based on the technique of using a bypass signal for binary to CSD conversion employing lesser logic elements than the techniques using carry propagation. Now if the current slice output is non zero, the evaluation of the inputs will be bypassed in the next digit slice using the bypass signal. The generation of bypass signal is done using a single NOR gate in each digit slice.

2.2. Logic element of Binary-to-CSD converter:
In this technique, two's complement representation of a number to be encoded is divided into groups each of three bits which overlap each other. Each digit slice of the converter recodes three binary bits, b_{i+1}, b_i and b_{i-1} into a CSD digit, x_i which is binary encoded into a sign bit, $x_{i,s}$, and a magnitude bit, $x_{i,m}$. Using the sign magnitude encoding (equivalent to two's complement), bit 0 is expressed as 00, 1 as 01 and −1 as 11.

Table 1 is the truth table for the Logic Element (LE) of i-th digit slice, where p_i is the bypass input into the LE and p_{i+1} is the bypass output from the LE, respectively. If the bypass signals are not used, a redundancy in nonzero digits will be generated when the LEs of all digit-slices execute concurrently.

Consider the following example, Recoding binary number $[0101011]_2$ without bypass signals
$[1\bar{1}\bar{1}\bar{1}0\bar{1}]$

while using the recursive right to left CSD coding technique taking into account the identity:
$x_i \times x_{i-1} = 0$ yields
$[1\bar{0}1\bar{0}\bar{1}0\bar{1}]$

Since there exists two non-zero digits at bit positions i = 3, 5 (where i = 0 for the LSB), the property of non-adjacency is violated. This is corrected by cascading the LEs using the bypass signals and takes one gate delay.
For $p_i = 1$, the output bit x_i and output bypass signal p_{i+1} are zero irrespective of the inputs b_{i-1}, b_i and b_{i+1}. While for $p_i = 0$, the output bit x_i and output bypass signal p_{i+1} are generated according to the b_{i-1}, b_i and b_{i+1}. LE, $p_{i+1}=|x_i|$ and $|x_i| = 1$ if and only if $b_i = 1$ or $b_{i-1} = 1$. The sign bit, $x_{i,s} = 1$ if and only if $b_{i+1} = |x_i| = 1$. The last row shows that the inputs to the LE are bypassed or ignored by the LE, while setting p_{i+1} to zero allowing a new bypass signal to be generated at the next LE. The LE circuit realizing Table 1 is shown in Fig. 3.

In case of carry propagation, there exists a delay of atleast two cascaded 2-input gates while using bypass signal, its generation takes a delay of only a NOR gate. In negative-positive binary code, where 0 is expressed as 00, 1 as 01 and −1 as 10, the same bypass technique can be applied to the CSD digit. However, additional logic gates will be required.

TABLE 1: Truth Table for LE

p_i	bi+1	b_i	bi−1	x_i	xi,s	xi,m	pi+1
0	0	0	0	0	0	0	0
	0	0	1	1	0	1	1
	0	1	0	1	0	1	1
	0	1	1	0	0	0	0
	1	0	0	0	0	0	0
	1	0	1	−1	1	1	1
	1	1	0	−1	1	1	1
	1	1	1	0	0	0	0
1	D	D	D	0	0	0	0

UGC Approved Journal
IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2017

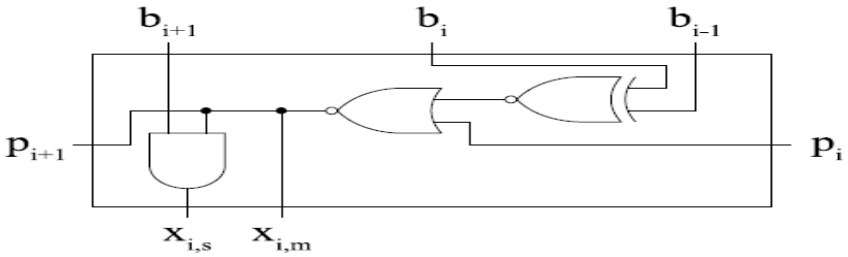


Figure 3: Logic Element Circuit

At the LSB and MSB of LEs i.e. i=0 and i=n-1, there is a requirement of inserting input paddings. The inputs b_{-1} and p_0 to the rightmost LE are set to zero. For the leftmost PE, if the input is signed and represented in two's complement form, the MSB b_{n-1} is sign extended to b_n. However, for the unsigned input, an additional LE will be needed to generate x_n to cover the largest positive number and the inputs b_n and b_{n+1} are set to 0. In both cases p_{i+1} of the leftmost LE is discarded.

2.2 CSD Multiplication:

Constant multiplication can be carried out by addition or subtracting a number of a partial product terms corresponding to the nonzero bit position in the constant multiplier. As CSD encoded multiplier contains the least number of nonzero bits, requires the minimum number of addition/subtraction operations in CSD representation. The partial product generation is done obeying the truth table as shown in table 2. Using this table the multiplier circuit for each bit is drawn as shown in figure 3.

Table2. CSD Multiplication Truth Table

x_i	y_i	p_i
0	0	0
0	1	0
1	0	0
1	1	1
0	-1	0
-1	0	0
1	-1	-1
-1	1	-1
-1	-1	1

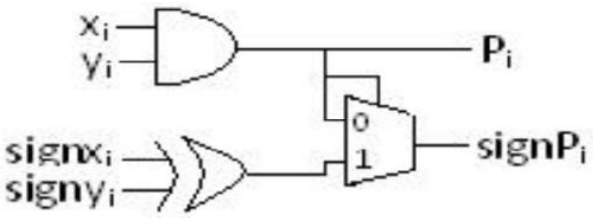


Figure4. CSD Multiplication

III. SIMULATION RESULTS

Table3. Simulation Output Results

No. of bits=4	Power Dissipation(µW)
Array Multiplier	**3808.9**
CSD Multiplier	**16.38**

Table 3 shows the comparison of the power of array multiplier and CSD multiplier for 4-bits. The CSD multiplier is found to consume less power.

The following figures show the schematic of a 4-bit CSD conversion circuit and 4-bit CSD multiplier circuit.

UGC Approved Journal

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 6, June 2017

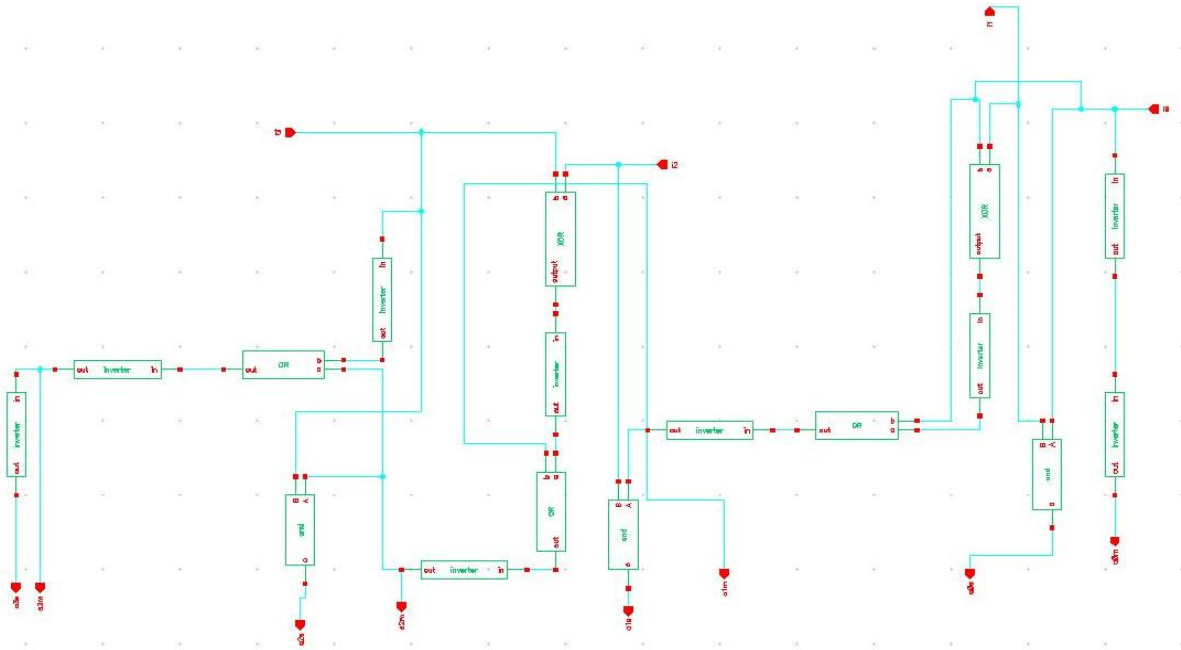


Figure6. CSD Conversion Circuit Schematic

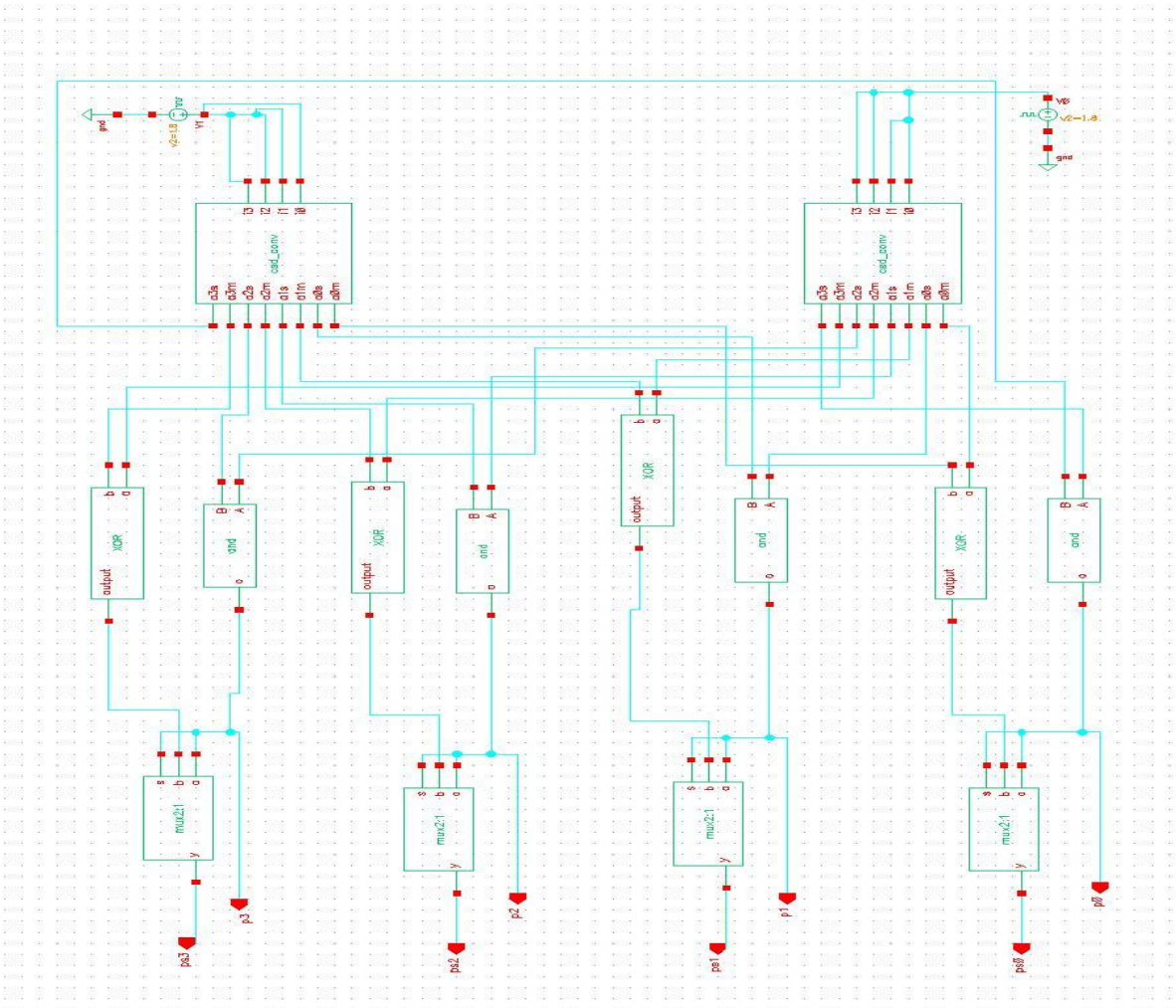


Figure6. CSD Multiplier Circuit Schematic

DOI 10.17148/IJIREEICE.2017.5635

UGC Approved Journal

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 6, June 2017

IV.RESULTS AND DISCUSSION

Multiplication is one of the basic functions used in digital signal processing. The proposed circuit- Canonical Signed Digit 4X4 Multiplier using 90nm Technology is successfully synthesized and is found to consume 99.56% less power. Cadence Virtuoso 90nm Technology is used for simulation of the Design. Multiplier circuit works with 1.8 V power supply.

REFERENCES

[1] Koren I., 2002, Computer Arithmetic Algorithms, A.K. Peters, Ltd. (Ed.).
[2] Phillips B, Burgess N., 2004 June, "Minimal weight digit set conversions", IEEE Transactions on Computers, June 2004, Volume 53, Issue 6, pp.666–677.
[3] Phatak D.S., Koren I., 1994 August, "Hybrid signed-digit number systems: a unified frame- work for redundant number representations with bounded carry propagation chains", IEEE Transactions on Computers, Volume 43, Issue 8, pp. 880–891.
[4] Ma G.K., Taylor F.J., 1990 January, "Multiplier policies for digital signal processing", IEEE ASSP Magazine, Volume 7, No. 1, pp. 6–20.
[5] Soderstrand M.A., 2003 May, "CSD multipliers for FPGA DSP applications", Proceedings of the International Symposium on Circuits and Systems (ISCAS), Volume 5, pp.469–472.
[6] Iacono D.L. and Ronchi M., 2004, "Binary canonic signed digit multiplier for high- speed digital signal processing", Proceedings of the 47th IEEE International Midwest Symposium on Circuits and Systems, Volume 2, pp. 205–208.
[7] Wang Y., DeBrunner L.S., Zhou D., DeBrunner V.E., 2007, "A multiplier structure based on a novel real-time CSD recoding", IEEE International Symposium on Circuits and Systems (ISCAS), pp.3195–3198.
[8] Maheshwari N, January 2013, "A Design of 4X4 Multiplier using 0.18 um Technology", International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 2 Issue 1, pp. 251-257.