Design and Implementation of Fast Addition Using QSD for Signed and Unsigned Numbers

S.Jakeer Hussain[1], K. Sreenivasa Rao[2]

[1]*(Department of ECE , AITS, Rajampet, Andhra Pradesh, India.)*
[2]*(Department of ECE , AITS, Rajampet, Andhra Pradesh, India.)*

Abstract: Binary Signed Digit Numbers are known to allow limited carry propagation with more complex addition process. Some of the limitations of this system are computational speed which limits formation and propagation of carry especially as the number of bits increases. Therefore it provides large complexity and low storage density. Carry free arithmetic operations can be achieved using a higher radix number system such as Quaternary Signed Digit (QSD) and it allows higher information storage density, less complexity. A high speed area effective adders and multipliers can be implemented using this technique. Carry free addition and other operations on a large number of digits such as 64, 128, or more can be implemented with constant delay and less complexity. The Design is simulated & synthesized using Modelsim6.0.

Keywords: carry free addition, QSD, VLSI.

INTRODUCTION: Now-a-days adders are mostly used in various electronic applications such as Digital signal processors and computing devices these adders are used to perform various algorithms like FIR, IIR etc. In Modern electronics, Digital systems play a prominent role in day to day life. Arithmetic operations such as addition, subtraction and multiplication still suffer from known problems including limited number of bits, propagation time delay, and circuit complexity. The speed of digital processor depends heavily on the speed of adders they have constraints like area, power and speed requirements. The delay in an adder is dominated by the carry chain.

Moderncomputers are based on binary number system(radix =2). It has two logical states ‘0’ and ‘1’. In such system, ‘1’ plus ‘1’ is ’0’ with carry ‘1’ (i.e. 1+1=10). This carry should have to add with another ‘1’, as a result further carry ‘1’ generates. This creates the delay problem in computer circuits. In adders Binary Signed Digit Numbers are known to allow limited carry propagation with more complex addition process sand very large circuit for implementation. Some of the limitations of this system are computational speed which limits formation and propagation of carry especially as the number of bits increases. Therefore it provides large complexity and low storage density.

A special higher radix-based (quaternary) representation of binary signed-digit numbers not only allows carry-free addition and borrow-free subtraction but also offers other important advantages such as simplicity in logic and higher storage density. Carry free arithmetic operations can be achieved using a higher radix number system such as Quaternary Signed Digit (QSD). In present study, QSD number system eliminates carry propagation chain which reduces the computation time substantially, thus enhancing the speed of the machine. QSD Adder or QSD Multiplier circuits are logic circuits designed to perform high-speed arithmetic operations. A higher radix based signed digit number system, such as quaternary signed digit (QSD) number system, allows higher information storage density, less complexity. A high speed area effective adders and multipliers can be implemented using this technique. The advantage of carry free addition offered by QSD numbers is exploited in designing a fast adder circuit. Additionally adder designed with QSD number system has a regular layout which is suitable for VLSI implementation which is the great advantage over the RBSD adder. An Algorithm for design of QSD adder is proposed. This algorithm is used to write the VHDL code for QSD adders. VHDL codes for QSD adder is simulated and synthesized and the timing report is generated. The timing report gives the delay time produced by the adder structure.

DESIGN OF QSD ADDER

It offers the advantage of reduced circuit complexity both in terms of transistor count and interconnections. QSD number uses 25% less space than BSD to store number. QSD numbers save 25% storage compared to BSD:

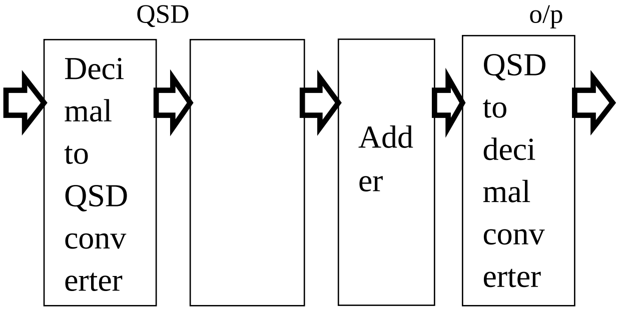
So the proposed QSD adder is better than RBSD adder in terms of number of gates, input connections and delay though both perform addition within constant time. Proposed design has the advantages of both parallelisms as well as reduced gate complexity. The computation speed and circuit complexity increases as the number of computation steps decreases. A two-step scheme appears to be a prudent choice in terms of computation speed and storage complexity.

Quaternary is the base 4 redundant number system. The degree of redundancy usually increases with the increase of the radix. The signed digit number system allows us to implement parallel arithmetic by using redundancy. QSD numbers are the SD numbers with the digit set as:

$$D = \sum_{i=0}^{n} x_i \, 4^i$$

Where x_i can be any value from the set $\{\bar{3}, \bar{2}, \bar{1}, 0, 1, 2, 3\}$ for producing an appropriate decimal representation. For digital implementation, QSD numbers are represented using 3-bit 2’s complement notation. A QSD negative number is the QSD complement of the QSD positive number. For example, using primes to denotecomplementation, we have $\bar{3}' = 3$, 3’ = $\bar{3}$, $\bar{2}' = 2$, 2’ =$\bar{2}$,$\bar{1}' = 1$, $1' = \bar{1}$.

In QSD number system carry propagation chain are eliminated which reduce the computation time substantially, thus enhancing the speed of the machine. As range of QSD number is from -3 to 3, the addition result of two QSD numbers varies from -6 to +6. The decimal numbers in the range of -3 to +3 are represented by one digit QSD number. In the two digits QSD result the LSB digit represents the sum bit and the MSB digit represents the carry bit. To prevent this carry bit to propagate from lower digit position to higher digit position QSD number representation is used. QSD numbers allow redundancy in the number representations. The same decimal number can be represented in more than one QSD representations. So we choose such QSD represented number which prevents further rippling of carry. To perform carry free addition, the addition of two QSD numbers can be done in two steps



Block diagram of QSD conversion

Step 1: First step generates an intermediate carry andintermediate sum from the input QSD digits i.e., addend and augend.

Step 2: Second step combines intermediate sum of current digit with the intermediate carry of the lower significant digit.

So the addition of two QSD numbers is done in two stages. First stage of adder generates intermediate carry and intermediate sum from the input digits. Second stage of adder adds the intermediate sum of current digit with the intermediate carry of lower significant digit. To remove the further rippling of carry there are two rules to perform QSD addition in two steps:

Rule 1: First rule states that the magnitude of theintermediate sum must be less than or equal to 2 i.e., it should be in the range of -2 to +2.

Rule 2: Second rule states that the magnitude of theintermediate carry must be less than or equal to 1 i.e., it should be in the range of -1 to +1.

According to these two rules the intermediate sum and intermediate carry from the first step QSD adder can have the range of -6 to +6. But by exploiting the redundancy feature of QSD numbers we choose such QSD represented number

TABLE I
THE INTERMEDIATE CARRY AND SUM BETWEEN -6 TO +6

Sum	QSD representation number	QSD coded number
-6	$\bar{2}2$, $\bar{1}\bar{2}$	$\bar{1}\bar{2}$
-5	$\bar{2}3$,$\bar{1}\bar{1}$	$\bar{1}\bar{1}$
-4	$\bar{1}0$	$\bar{1}0$
-3	$\bar{1}1$,$0\bar{3}$	$\bar{1}1$
-2	$\bar{1}2$,$0\bar{2}$	$0\bar{2}$
-1	$\bar{1}3$,$0\bar{1}$	$0\bar{1}$
0	00	00
1	01,1 $\bar{3}$	01
2	02,1$\bar{2}$	02
3	03,1$\bar{1}$	1$\bar{1}$
4	10	10
5	11,2 $\bar{3}$	11
6	12,2$\bar{2}$	12

Which satisfies the above mentioned two rules. When the second step QSD adder adds the intermediate sum of current digit, which is in the range of -2 to +2, with the intermediate carry of lower significant digit, which is in the range of -1 to +1, the addition result cannot be greater than 3 i.e., it will be in the range of -3 to +3. The addition result in this range can be represented by a single digit QSD number; hence no further carry is required.

Example 1*:* To perform QSD addition of two numbers A = 103and B = 223 (Two number is positive).

First convert the decimal number to their equivalent QSD representation:

$(103)_{10}$ = $(1\ 2\ 1\ 3)_4$ $(223)_{10}$ = $(3\ 1\ 3\ 1)_4$

A = 103 1 2 1 3

B = 223 3 1 3 1

Sum 4 3 4 6

IC 1 1 1 1

IS 0 $\bar{1}$ 0 2

Output 1 1 0 1 2

The sum output is $(11012)_{QSD}$ which is equivalent to $(326)_{10}$.

Example 2: To perform QSD subtraction of two numbers A = 103and B = -223 (One number is positive and one number is negative).

$(103)_{10}$= $(1\ 2\ 1\ 3)_4$ $(-223)_{10}$=$(\bar{3}\bar{1}\bar{3}\bar{1})_4$

A = 103 1 2 1 3

B =-223 $\bar{3}\bar{1}\bar{3}\bar{1}$

Output $\bar{2}$ 1 $\bar{2}$ 0

The sum output is$(\bar{2}\ 1\ \bar{2}\ 0)_{QSD}$ which is equivalent to $(-120)_{10}$.

LOGIC DESIGN AND IMPLEMENTATION USING OF SINGLEDIGIT QSD ADDER UNIT

There are two steps involved in the carry-free addition. The first step generates an intermediate carry and sum from the addend and augend. The second step combines the intermediate sum of the current digit with the carry of the lower significant digit. At the input side, the addend A_i is represented by 3 variable input as A_2, A_1, A_0 and the augend B_i is represented by 3 variable input as B_2, B_1, B_0. At the output side, the intermediate carry IC is represented by IC_2,

IC_1, IC_0 and the intermediate sum IS is represented by IS_2, IS_1, $IS_{0.}$ The six variable expressions for intermediate carry and intermediate sum in terms of inputs (A_2, A_1, A_0, B_2, B_1 and B_0) can be derived from. So we get the six output expressions for IC_2, IC_1, IC_0, IS_2, IS_1 and IS_0. As the intermediate carry can be represented by only 2 bits, the third appended bit IC_2 is equal to IC_1 so the expression for both outputs will be the same. The VHDL code for intermediate carry and sum generator in step 1 adder, by taking the six inputs (A_2, A_1, A_0, B $_2$, B $_1$ and B_0) and six outputs (IC_2, IC_1, IC_0, IS_2, IS_1 and IS_0). The VHDL code is compiled and simulated using ModelSim software.

SIMULATION RESULT

The NAND Implementation of single digit QSD adder is proposed where its high speed and low power QSD adder unit using NAND gate with Modelsim 6.0 is observed.

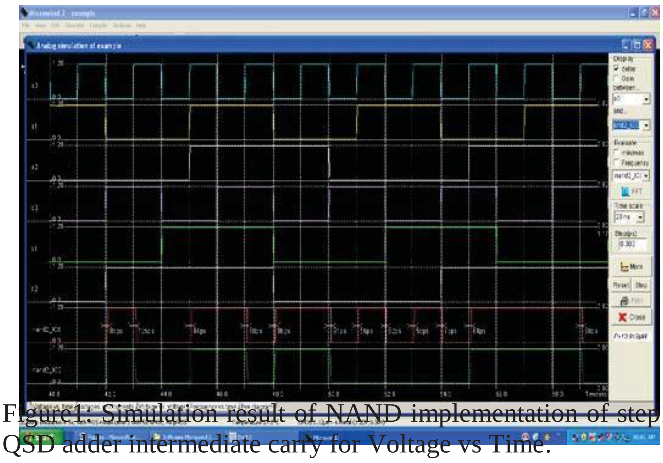


Figure1: Simulation result of NAND implementation of step1 QSD adder intermediate carry for Voltage vs Time.

NAND-NAND implementation of the QSD single digit adder is simulated.

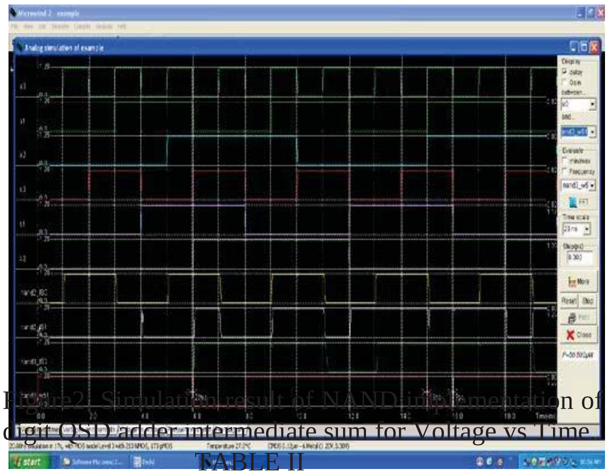


Figure2: Simulation result of NAND implementation of single digit QSD adder intermediate sum for Voltage vs Time

TABLE II

THE COMPARISON BETWEEN QSD ADDER CIRCUITS

Circuit type	Max. Power supply	Propagation delay	Transistor count	Dynamic power dissipation
Quaternary full adder (Based on Quaternary mux) equivalent Binary add	3V	2.2V	332	181µW(250 MHz)
Quaternary full adder (Based on output generator sharing)	1.2V	113ps	252	55µW(1GHz)
Quaternary full adder(Based on Quaternary Differential logic)	1.8V	1.4ns	194	194µW(300 MHz)
Quaternary full adder(using Nand gate)	1.2V	2ns	122	36.255µW(5 GHz)

CONCLUSION

In the proposed design of Quaternary Signed Digit adder using NAND-NAND implementation for single digit addition, the dynamic power dissipation is 36.255—W at 5GHz frequency. These circuits consume less energy and less energyand power, and shows better performance. The delay of the proposed design is 2ns.The design is simulated using Modelsim 6.0.

REFERENCES

i. *O. Ishizuka, A. Ohta, K. Tannno, Z. Tang, D. Handoko, "VLSI design ofa quaternary multiplier with direct generation of partial products," Proceedings of the 27th International Symposium on Multiple-Valued Logic, pp. 169-174, 1997.*

ii. *A.A.S Awwal, Syed M. Munir, A.T.M. Shafiqul Khalid, Howard E. Michel and O. N. Garcia, "Multivalued Optical Parallel Computation Using An Optical Programmable Logic Array", Informatica, vol. 24, No. 4, pp. 467-473, 2000.*

iii. *Behrooz perhami "generalized signed digit number systems, a unifying frame work for redundant number reperesentation ".IEEE transactions on computers,vol 39,no.1,pp.89-98,January 1999.*

iv. *F. Kharbash and G. M. Chaudhry, "Reliable Binary Signed Digit Number Adder Design", IEEE Computer Society Annual Symposium on VLSI, pp 479-484, 2007.*

v. *John Moskal, Erdal Oruklu and Jafar Saniie, "Design and Synthesis of a Carry-Free Signed-Digit Decimal Adder", IEEE International symposium on Circuits and Systems, pp 1089-1092, 2007.*

vi. *Kai Hwang, "Computer Arithmetic Principles, Architecture and Design", ISBN 0-471-03496-7, John Wiley & Sons, 1979.*

vii. *A.A.S. Awwal, J. Ahmed, "Fast Carry free Adder Design Using QSD Number System". IEEE, CH3306-8/93/0000-1085, Pp. 1085-1088, 1993.*

viii. *T. Chattopadhyay, J.N. Roy, "Easy conversion technique of binary to quaternary signed digit and vice versa." Physics Express, Vol. 1, No. 3, Pp. 165-174, 2011.*