International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 6, June - 2013

FPGA Implementation Of Quaternary Adder/Subtractor&Logic-Unit.

Aditya Sharma.
M.Tech Scholar(VLSI Engg.),SGVU Jaipur.

Abstract

Complexity in circuits, limitations of number of digits& propagation delay are inevitable drawbacks that arise in diversified DSP applications involving various operations.Quaternary Signed Digit(QSD) number system involves elimination of carry propagation chains consequently reducing propagation delay to a much extent & hence results in fast arithmetic operations in comparison to binary number system. This paper presents VHDL design of quaternary number system based adder / subtractor / logical unit. The proposed VHDL design will be implemented on FPGA & detailed performance analysis will be done.

1. Introduction

Different digit devices namely signal processors and computers involve diversified arithmetic and logical operations. Quite a bunch of researchers have developed interest in arithmetic units based on QSD number system. Also new advancements in integrated circuits methodologies have paved the way for making of arithmetic circuits as per requirements in VLSI implementations. Although technical hinderance still exist in case of extent of number of bits that can be employed, complexity level of circuits and timing delay in propagation.

This paperpresents efficient quaternary signed digit based arithmetic logic unit design involving addition without carry, subtraction without borrow. The quaternary signed digit based addition or subtraction involves constant values of min-terms independent of the size of operand. The EDA tools used in this design verify the design's functioning. Xilinx Project Navigator 12.1 and Modelsim 10.2b have been employed for the same. Behavioural Description of the proposed design is coded in VHDL.

2. Quaternary Signed Digit Representation

Quaternary Signed Digit representation is based on 3-bit 2's complement system. Individual QSD numbers can be represented using the relation

$$Q = \sum_{j}^{k} a_j \, 4^j ,$$

Where a_j refers to either value from the given set(say), S = $[\bar{3}, \bar{2}, \bar{1}, 0,1,2,3]$ resulting in a suitable QSD notation.Negative quaternary signed digit number is the quaternary signed digit complement of the positive quaternary signed digit number as stated below,

$$\bar{3} = -3, \bar{2} = -2, \bar{1} = -1.$$

For instance,

$$\bar{3}2\bar{2}2_{QSD} = -182_{10},$$
$$1\bar{2}\bar{2}1_{QSD} = 25_{10}.$$

Using these 7 QSD digits as given in the above set "S" any transformation of a QSD number can be represented. In the proposed QSD-ALU 4-digit QSD numbers have been used, the magnitude of 4-digit QSD numbers is equivalent to 8-digits in binary.For instance a QSD number,$3333_{QSD} = 11111111_2 = 255_{10}$.

Unique QSD representation in 3-bit 2's complement notation is elaborated as follows:-

$$\bar{3} = 101$$
$$\bar{2} = 110$$
$$\bar{1} = 111$$
$$0 = 000$$
$$1 = 001$$
$$2 = 010$$
$$3 = 011$$

For instance,
$3333_{QSD} = 011011011011_{QSD}$(3-bit 2's complement notation),
$\bar{1}\bar{2}\bar{3}1_{QSD} = 111110101001_{QSD}$(3-bit 2's complement notation).

The Quaternary Signed Digit based adder / subtractor and logic unit will be written in VHDL and

IJERT

implemented on Spartan-3E FPGA kit using Xilinx Project Navigator. Although before implementing on FPGA kit, the simulation results of different units of the design will be checked on Modelsim 10.2b.Performance analysis of various units of the design will be done based on different parameters such as LUT utilization, average power consumption of the circuit on different FPGAs and propagation delay.

3. Quaternary Adder/Subtractor Depiction

QSD addition basically employs a carry free sort of technique for addition. Normally a carry free addition happens in two levels. The first level involves the addend & augend producing intermediate carry & sum. The second level involves the merging of the carry from the previous digit with the intermediate sum of present digit. There are two rules to stop carry from further occurrence,

I.) Magnitude of Intermediate Sum≤2.
II.)Magnitude of Carry≤1.

Accordingly, the second level result must be less than 3 in magnitude. The representation of the second level result can be easily done using one quaternary signed digit. Hence no successive carry is produced. Level-1 includes all the feasible input combinations of the addend & augend in the range from "-6" to "$+6$" as illustrated in Table-A.

Table-A : All feasible pairs of addend(P) & augend(Q) in QSD system.

P \ Q	-3	-2	-1	0	1	2	3
-3	-6	-5	-4	-3	-2	-1	0
-2	-5	-4	-3	-2	-1	0	1
-1	-4	-3	-2	-1	0	1	2
0	-3	-2	-1	0	1	2	3
1	-2	-1	0	1	2	3	4
2	-1	0	1	2	3	4	5
3	0	1	2	3	4	5	6

The sum of any random pair of quaternary signed digits ranges from -6 to $+6$ whose notation can be done in sum & intermediate carry using QSD representation as illustrated in Table–B. More than one notations are feasible in case of few numbers but those satisfying the stated rules are selected.The selected sum & intermediate carry are illustrated in third column of Table – B.

Table-B : Sum & Intermediate Carry ranging -6 to$+6$.

Sum	*Representation of sum in QSD*	*Sum in QSD coded format*
-6	$\bar{1}\bar{2}, \bar{2}2$	$\bar{1}\bar{2}$
-5	$\bar{1}\bar{1}, \bar{2}3$	$\bar{1}\bar{1}$
-4	$\bar{1}0$	$\bar{1}0$
-3	$0\bar{3}, \bar{1}1$	$\bar{1}1$
-2	$0\bar{2}, \bar{1}2$	$0\bar{2}$
-1	$0\bar{1}, \bar{1}3$	$0\bar{1}$
0	00	00
1	$1\bar{3}, 01$	01
2	$1\bar{2}, 02$	02
3	$1\bar{1}, 03$	$1\bar{1}$
4	10	10
5	$2\bar{3}, 11$	11
6	$2\bar{2}, 12$	12

All addend, augend & sum can be written in 3-bit 2's complement binary representation. Map involvingaddend,augend and sum is carved out. Although since immediate carry ranges from -1 to $+1$,so it can be well represented in 2-bit binary format,

IJERT

accordingly resulting in six boolean relations. The sum & immediate carry circuit is shown in Figure-A.

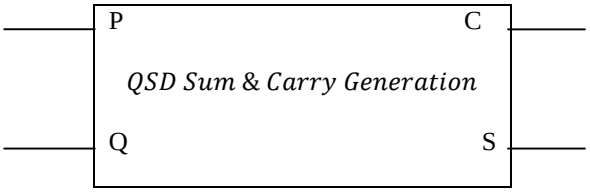

Figure-A : The QSD sum and intermediate carry-out generation.

In level 2, the intermediate carry generating from the previous digit is added up with the present digit sum evaluating to final output.The present digit always diminishes the carry in from the previous digit by combining with it. Table-C illustrates all feasible mergings of the sum and intermediate carry resulting different addition values.

Table-C : The values of all feasible summations of intermediate carry-out(P) and sum(Q).

P \ Q	−2	−1	0	1	2
−1	−3	−2	−1	0	1
0	−2	−1	0	1	2
1	−1	0	1	2	3

The output of summation in this level lasts from $-3$to $+3$. Although in this level carry is restricted, the output comes out to be in single quaternary signed digit(QSD).The 2-bit and 3-bit binary respectively represent the inputs, the intermediate carry & sum. The output is in the form of 3-bit QSD binary format. Table-D illustrates the mapping of 3-bit output & 5-bit input. Table-D as a result is used to extract three 5-variable boolean relations. Figure-B illustrates the second level adder. "n"number of QSD sum & carry generators and "$(n-1)$" number of second level adders are required for implementation of "n" digit QSD adder as illustrated in Figure-C. The output comes out to be "$(n+1)$" digit number.

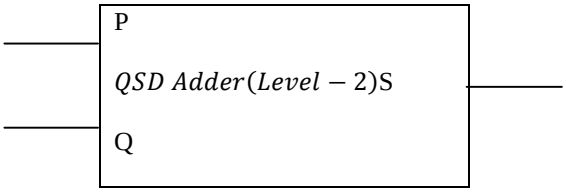

Figure-B : QSD Adder (Level-2).

Table –D : Mapping of inputs/outputs of second level QSD Adder.

INPUT				***OUTPUT***		
QSD		**Binary**		***Decimal***	***QSD***	***Binary***
P_j	Q_j	P_j	Q_j	***Sum***	S_j	S_j
1	2	01	010	3	3	011
1	1	01	001	2	2	010
0	2	00	010	2	2	010
0	1	00	001	1	1	001
1	0	01	000	1	1	001
−1	2	11	010	1	1	001
0	0	00	000	0	0	000
1	−1	01	111	0	0	000
−1	1	11	001	0	0	000
0	−1	00	111	−1	−1	111
−1	0	11	000	−1	−1	111
1	−2	01	110	−1	−1	111
−1	−1	11	111	−2	−2	110
0	−2	00	110	−2	−2	110
−1	−2	11	110	−3	−3	101

IJERT

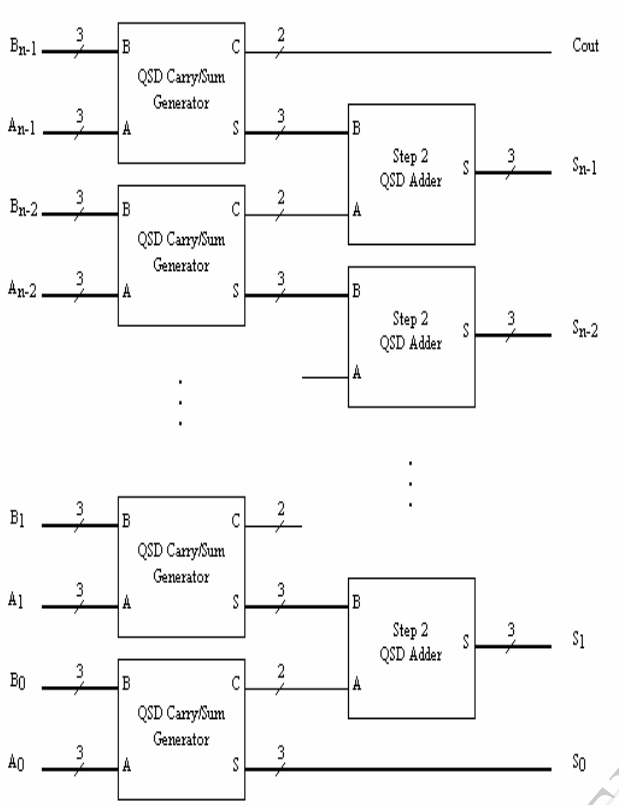

Figure-C : QSD Adder of the order " n-digits".

4. Quaternary Logic Unit Depiction

MAX function realization using OR function and MIN function realization using AND function as well as INVERTER function are all involved in quaternary logic unit specification.2-bit binary is used to express the quaternary logic levels.The truth tables for INVERTER function,MAX function & MIN function are respectively elaborated below.

Table-E : Truth Table for Quaternary Inverter Function

INPUT	*OUTPUT*
00	11
01	10
10	01
11	00

Table-F : Truth Table for Quaternary MAX Function

		P			
		0	**1**	**2**	**3**
Q	**0**	0	1	2	3
	1	1	1	2	3
	2	2	2	2	3
	3	3	3	3	3

Table-G: Truth Table for Quaternary MIN Function

		P			
		0	**1**	**2**	**3**
Q	**0**	0	0	0	0
	1	0	1	1	1
	2	0	1	2	2
	3	0	1	2	3

5. Analytical Results

This Quaternary Signed Digit based Adder/Subtractor& Logical Unit are coded in VHDL and implemented on SPARTAN-3E (xc3s250e-4pq208) using Xilinx Project Navigator 12.1. The simulation of the design is also done on Modelsim 10.2b.The magnitude of QSD number is twice the magnitude of binary.It means that a 4-digit QSD number actually will represent 8-digit binary if converted to the same. Various parameters are employed fordetailed performance analysis of this design. Propagation Delay, LUT utilization and average power consumption of various units of the proposed design are estimated.

IJERT

Table–H : Performance Analysis

OPCODE	*Design Unit*	*Delay*	*Average Power*	*LUTs (Total = 4896*
000	*Adder*	9.7*ns*	0.052 *W*	291
001	*Subtractor*	9.7*ns*	0.052 *W*	320
010	*INVERT*	5.6*ns*	0.052 *W*	16
011	*MAX*	5.6*ns*	0.052 *W*	16
100	*MIN*	5.6*ns*	0.052 *W*	16

Total average power consumption of the designed QSD ALU on Spartan-3E = 0.052 W

Total LUT utilization of the designed QSD ALU on Spartan-3E = 834(17%).

6. Conclusion

This QSD based design elaborates the better productivity over other number systems in terms of superior performance based on various parameters such as propagation delay, LUT utilization, power consumption and number of digits employed. The QSD adder designed in this paper can be used as a building block for QSD multiplier design. The future innovations can be division & square-roots using QSD representation. Furthur optimization of this design can be achieved by multivalued logic realization.

IJERT

References

[1] I. M. Thoidis, D. Soudris, J. M. Fernandez, A. Thanailakis, "The circuit design of multiple-valued logic voltage-mode adders," 2001 IEEE International Symposium on Circuits and Systems, pp 162-165, Vol. 4 , 2001.

[2] 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.

[3] A. K. Cherri, "Canonical quaternary arithmetic based on optical content-addressable memory (CAM)" Proceedings of the 1996 National Aerospace and Electronics Conference, pp. 655-661, Vol. 2, 1996.

[4] J. U. Ahmed, A. A. S. Awwal, "Multiplier design using RBSD number system", Proceedings of the 1993 National Aerospace and Electronics Conference, pp. 180-184, Vol. 1, 1993.

[5] SongpolOngwattanakul, PhaisitChewputtanagul, David J. Jackson, Kenneth G. Ricks, "Quaternary Arithmetic Logic Unit on a Programmable Logic Device", proc. IEEE conference, 2001.

[6] Reena Rani, UpasanaAgrawal, Neelam Sharma, L.K Singh, "High Speed Arithmetic Logical Unit using Quaternary Signed Digit Number System" International Journal Of Electronic Engineering Research, ISSN 0975 – 6450, Volume 2 Number 3, 2010 pp. 383–391.

[7] N. Takagi, H. Yasuura, and S. Yajima, "High Speed VLSI Multiplication Algorithm with a Redundant Binary Addition Tree, " IEEE Trans. Comp., C-34, pp. 789-795, 1985

[8] 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. 4,2000, pp. 467- 473.

[9] P. K. Dakhole, D.G. Wakde, " Multi Digit Quaternary adder on Programmable Device : Design and verification" International Conference on Electronic Design, 2008, 1-3 Dec, pp. 1-4.

[10] A. Avizienis, "Signed-Digit Number Representation for Fast Parallel Arithmetic, "IRE Transaction Electron. Comp., EC-10, pp. 389-400, 1961.

[11] M. E. Louie and M. D. Ercegovac," On Digit – Recurrence Division Implementations for Field Programmable gate Arrays" In Proc. Of the 11th symposium on Computer Arithmetic, PP. 202-209, Canada.

[12] Namba K. and Ito H., Design of defect tolerant s Wallace Multiplier ", IEEE Symposium on Dependable Computing Vol.2,pp 5-9, Dec 2005.

[13] Steve Hung Lung, "Skew Tolerant Domino Techniques for High speed Bough Wooley Multiplier circuit Design", IEEE Journal on Circuit and systems, Vol 2, pp 424-427

[14]http://www.computer.org/csdl/proceedings/dsd/2011/4494/00/4494a148-abs.html.