International Journal of Engineering Research and General Science Volume 3, Issue 1, January-February, 2015
ISSN 2091-2730

# Radix-2 Non-Restoring Asynchronous divider using Shannon based 14 Transistor Full-Adder

Sonika Choubey [1], Prof. Rajesh Kumar Paul [2]

[1] sonikachoubey18@gmail.com

**Abstract**— This paper deals with design of non-restoring divider using Shannon based adder with pass transistor logic the proposed adder using only 14 transistors for full adder implementation it is verified and implemented on Microwind 3.1 & DSCH 2 CAD tool using BSIM 4 model. The analysis is done on the basis of power consumption, delay and area occupied and theses are compared with previous papers and we are good to enhance these parameters.

**Keywords**—CAS , Shannon Theorem ,Non Restoring Division,

## INTRODUCTION

Now days, many small-sized multiplier (or) divider circuits are designing for efficient response in terms of power, delay, area and operational range. The high speed and reduced area size can be implemented with different logic style in VLSI design [1]. There are many types of logic designs; each has its own limitations in terms of speed, chip size (area) and power dissipation [3]. This trend must be ceased to obviate extortionate packaging and thermal management cost at system level. Eventually as more switching is done in a particular period of time; it would cause the circuit to perform much more gradual. However this quantity can be lowered if not solved by reducing the number of transistor that is being utilized. Many transistors in a circuit might be redundant and abstracting this redundancy can avail to lower the delay and hence incrementing the speed. This non-restoring divider circuit is designed utilizing by controlled add/subtract (CAS) cells techniques. This CAS cell is designed by gathering of full adder and 2 input XOR. The Shannon theorem predicated adder circuit can be reduced the number of transistor utilizing by redundancy reduction method.

The Shannon theorem adder circuit used only the NMOS, which would increase the switching speed .The proposed circuit factors analyzed by BSIM4 analyzer. We have compared our results with other adder cell predicated divider circuits in terms of better speed, area and power dissipation.

## RADIX-2 NON-RESTORING DIVISION

Division is the most involutes and arduous operation in the computer arithmetic. But the division algorithm is divided into two fundamental approach first one is multiplication and another is subjective approach. The multiplication approach is involutes as compared to subjective approach. Our focus is on subtractive approach to get quotient and remnant we subtract the divisor from partial remnant recursively to find the quotient and remnant.

Suppose that there are two n-digit numbers, X and D, which represent the dividend and divisor respectively. By the division operation we can find a n-digit quotient and a n-digit remainder denoted as Q and R respectively. The mathematical representations of X, D, Q, and R are as following (Koren, 1993),

$$R^{(j+1)} = r \times R^{(j)} - q_{j+1} \times D \qquad (1)$$

Where j = 0, 1, 2, …, n-1 is the iteration number.

$R_j$ is the partial remainder at iteration j.

r is the radix number.

$q_{j+1}$ is the j+1th digit of the quotient.

The final quotient is represented as

Q=$q_1, q_2, q_3 - - - - q_n$

Due to the complexity and the hardware cost, we use radix-2, i.e., r=2, for our design. Therefore, equation (1) can be rewritten and represented in equation (2)

$$R^{(j+1)} = 2 \times R^{(j)} - q_{j+1} \times D \qquad (2)$$

In the hardware design we have to check the subtraction at each step to decide the quotient in that digit. There are two ways to find the quotient of the current digit. One is the recuperating method, and the other is the non-renovating method. Without loss of generality let us discuss the two methods in the radix-2 number system. In the renovating approach, when the current partial remnant,$R^{(j+1)}$ is positive, the current quotient bit is equal to 1.

On the other hand, if the current partial remnant is less than 0, then the current quotient bit is set to 0, and then the partial remnant should be integrated with the divisor and instaurate back to the antecedent partial remnant, R^((j)), and it is soi-disant the "restoring" method. In the non-recuperating method, if the current partial remnant, R^((j+1)), is positive, the current quotient bit is equipollent to 1. On the other hand, if the current partial remnant is less than 0, then the current quotient bit is set to -1, and at the next step we have to integrate the divisor to the current partial remnant, R^((j+1)), to compose the next partial remnant, R^((j+2)).The quotient map is shown in Fig. 1. By this method, there is no desideratum to integrate divisor to renovate the antecedent partial remnant. However, the quotient in the non-renovating scheme is represented in the signed bit (digit) format. Therefore, after we culminate the division process, the non-recuperating method needs a supplemental step to convert the signed bit format to the binary number representation. Since we do not require to check the polarity of the partial remnant to do the renovating of the partial remnant. Therefore, the speed of the non-renovating division algorithm is more expeditious than the speed of the renovating division algorithm.

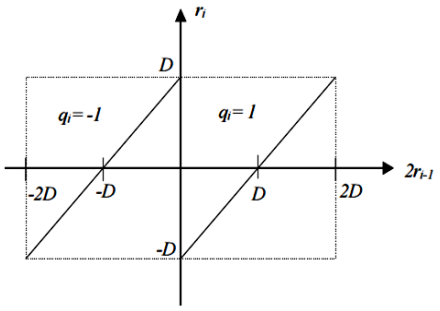


Fig 1. The Quotient map of Non restoring division

In the radix-2 division, dividend and divisor are usually represented in the normalized format. A normalized binary number can be represented in the following format.

A=$(0.1a_1a_2a_3a_4a_5 \ldots\ldots a_n)$

By the non-renovating division approach, we find the -1 of the quotient bit can be simply set to 0, and the quotient is the genuine quotient that we opted to find. We utilize a simple example to describe the normalized radix-2 non-recuperating division algorithm.

Suppose that we have two numbers, 011011111 and 01001, where the decimal points are abbreviated and the first bits of the dividend and divisor represent the denotement bits. The division is executing as the following procedures.

**Example:**

**011010000 / 01001 => *Q = 010111* &*R = 0001***

| | | |
|---|---|---|
| 011010000 | | |
| -) 01001 | (+10111) | |
| 001000000 | | $q_1 = 1$ |
| -) 01001 | (+10111) | |
| 11111000 | | $q_2 = 0$ |
| +) 01001 | | |
| 0011100 | | $q_3 = 1$ |
| -) 01001 | (+10111) | |
| 001010 | | $q_4 = 1$ |
| -) 01001 | (+10111) | |
| *R = 0001* <= 00001 | | $q_5$ |
| *=1 => Q = 010111* | | |

That signifies in the radix-2 non-recuperating division approach; we do not require to convert the signed digit quotient to the binary representation and the calculated quotient is the number that we opted in the hardware design. For the benefit of the speed and hardware cost, we utilize radix-2 non-recuperating division algorithm to build our divider. Pass transistor logic:

In electronics, pass transistor logic (PTL) describes several logic families utilized in the design of integrated circuits. It reduces the count of transistors used to make different logic gates, by eliminating redundant transistors. Transistors are utilized as switches to pass logic levels between nodes of a circuit, in lieu of as switches connected directly to supply voltages. This reduces the number of active contrivances, but has the disadvantage that the difference of the voltage between high and low logic levels decreases at each stage. Each transistor in series is less saturated at its output than at its input. If several contrivances are chained in series in a logic path, a conventionally constructed gate may be required to unsaturated the signal voltage to the full value. By contrast, conventional CMOS logic switches transistors so the output connects to one of the potency supply rails, so logic voltage levels in a sequential chain do not decrement. Since there is less isolation between input signals and outputs, designers must take care to assess the effects of unintentional paths within the circuit. For felicitous operation, design rules restrict the arrangement of circuits, so that sneak paths, charge sharing, and slow switching can be eschewed. Simulation of circuits may be required to ascertain adequate performance.

In digital circuits, an adder–subtractor is a circuit that is capable of integrating or subtracting numbers (in particular, binary). Below is a circuit fig 2 that does integrating or subtracting depending on a control signal. It is additionally possible to construct a circuit that performs both integration and subtraction concurrently.

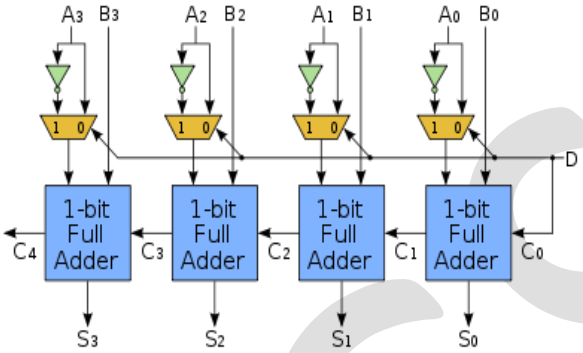


Fig 2. CAS cell using full adder and XOR gate

That has control input D and the initial carry connect is also connected to D then:

- When D=0 the modified adder performs addition
- When D=1 the modified adder performs subtraction

Having an *n*-bit adder for A and B, then S=A+B. Then, assume the numbers are in one's complement. Then to perform A-B, one's complement theory says to invert each bit with a NOT gate. This yields S=A+B'+1, which is easy to do with a slightly modified adder.

**SHANNON THEOREM**

According to this theorem any logic expression is divided into two terms. One with a particular variable set to 1and multiplying it by a variable and then set the variable to 0 and multiplying it by the inverse. The fullest reduction can be obtained by continuously repeating the Shannon theorem.

This method is useful especially to multiplier and pass transistor circuit design. The Shannon's theorem in a generalized way can be stated as a function of many variables, f (b0, b1, b2, y, bi, y, bn) can be written as the sum of two terms, say one with a particular variable ai , set to 0, and one with it set to 1.

f (b0, b1, b2, ....., bi,.....y, bn) = bi'f (b0, b1, b2, ....., 0,.....y, bn) + bi f (b0, b1, b2, ....., 1,.....y, bn)
(3)

Shannon's theorem is applied to the logical function using n-1 variables as control inputs and three data lines set to a logical '1'. These source inputs are then connected to the VDD lines (logical '0'), which are connected to the ground. The remaining nth variable is connected from the data input to the source input. The data signals flow horizontally and the

control signals flows vertically. Remove pairs of transistors when they cancel each other. The Shannon expression output depends upon the pass logic '1' or logic '0'. If it has logic'0' then the connection input is given by 0 and by '1' for the connection input '1'[5].

Using Shannon expansion theorem we get, sum=$(AxorB)\bar{C} + (\overline{AxorB})$C (4) and

carry=$(\overline{AxorB})B + (A\ xor\ B)C$ (5)

Below fig.3 shows the controlled full adder/subtractor design with Shannon based using pass transistor logic. Fig 4 shows CAS cell implementation.

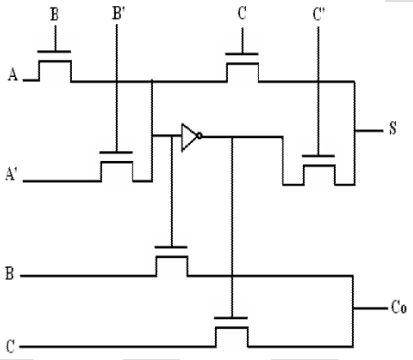


Fig 3. Full adder implementation using pass transistor logic using 14-T

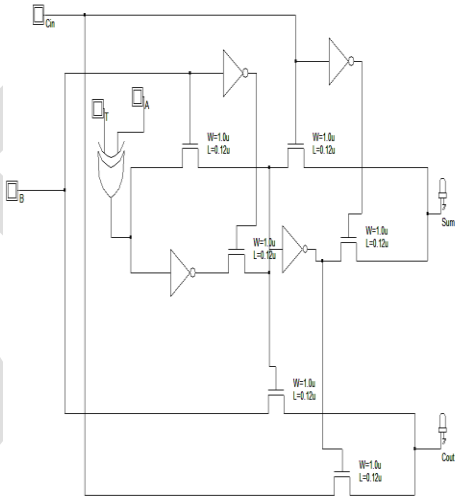


Fig 4. CAS cell using Shannon 14-T model

**NON-RESTORING DIVIDER**

The proposed divider is designed for 7bit dividend and 4bit divisor with MSB assigned to zero as a sign bit .Proposed divider designed by using controlled add/subtract (CAS) cell as a main structure. The CAS cell is designed by utilizing XOR gate and full adder and figure 3 shows the modified full adder by pass transistor logic. Figure 4 shows the CAS cell introducing xor gate with full adder. The CAS cell have A, as A as dividend, B as divisor ,Cin as carry, and T as control bit .The A, B and Cin are inputs of the full adder circuit [4]. Division is done row by row, on each row; the reminder of

the operation will be fed into the cells in following row below. These reminders will become the inputs to full adders for bottom rows. Computation is carried out until it reaches the last rows, where the final reminder value can be calculated and the output for it is given out. For the quotient calculation, it is taken from the first till fourth columns of final cell. These cells are responsible to give out the correct quotient [5]. When quotient is 1, subtraction is done by integrating the complement of the divisor to next cell together with carry from the lower paramount-bit position. Since the initial subtraction must always be carried out, Q = 1. It is verbally expressed to be non-restoring for the reason that it has realized the CAS mentioned above. For an n x n array divider [6],[2]. Schematic design of Divider with DSCH 2 Tool is shown in figure 5 .Here the dividend(7bit) and divisor(4 bit) are not fixed excluding the sign bit; it may be in any combination of bit .

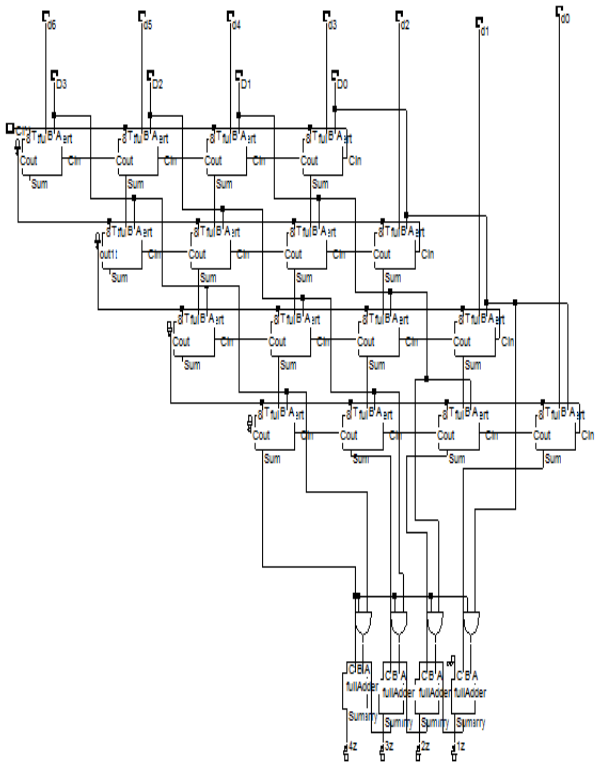

Fig 5. Radix-2 Non-Restoring Asynchronous divider using Shannon based 14 transistor full-Adder

## RESULTS AND DISCUSSION

The simulation of the proposed divider circuit is carried out using Microwind tool and DSCH tool with different bit configuration, as shown in fig. 6 to fig.9.The proposed Shannon adder circuits are designed using Microwind and DSCH VLSI CAD Tools. Here MCIT Adder, basic Shannon adder, CPL, CMOS based adder are used for comparison with our proposed modified Shannon adder circuit. The basic reason of designing this is to make an efficient adder that gives us a better performing divider when it is implemented into the cells. Our proposed Shannon adder and divider are implemented on 45nm and 65nm voltage supply is respectively 0.4V and 0.7V.

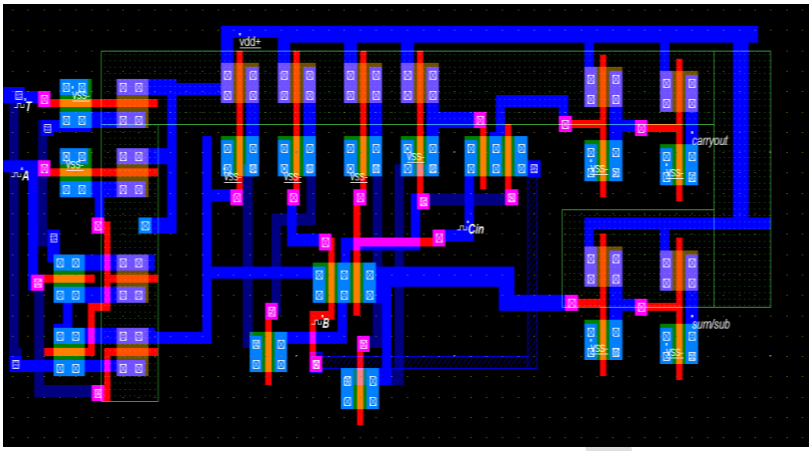


Fig.6 1-bit CAS Layout using Pass-transistor Logic in Microwind

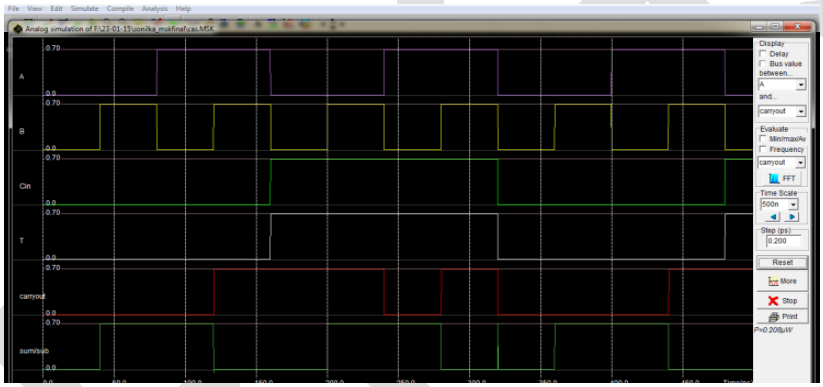


Fig.7 1-bit CAS Layout simulation using Pass-transistor Logic in Microwind

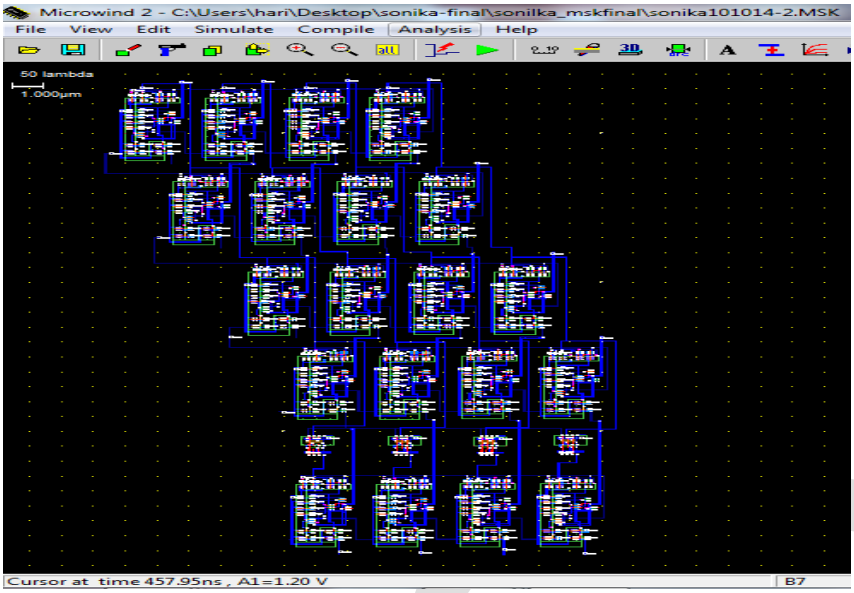


Fig 8 7÷4 bit Non-resorting divider Layout using Pass-transistor Logic in Microwind

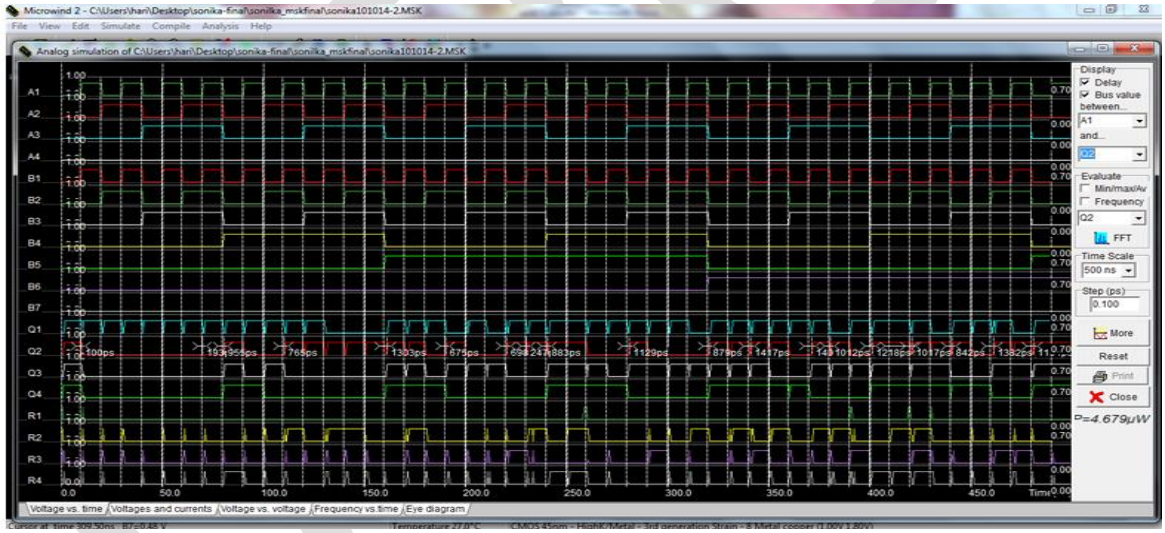


Fig 9 7bit ÷ 4 bit Non-resorting divider Layout simulation using Pass-transistor Logic in Microwind

Our proposed Shannon adder cell based divider and other adder based dividers are checked validate for dividend varied from 0111111-0000000 and divider from 0111-0000. For the analysis of various parameters such as area, propagation delay, dissipated power, are calculated and given below in table II, III. Comparison of our proposed design with previous papers is shown in table II at 45nm technology for full range of dividend and divisor. In TABLE III power dissipation at 65nm is calculated for only fixed dividend 0111111 and variable divisor from 0000-0111 because of the proper comparison.[2]

**TABLE 1 Comparison of proposed Shannon Based Full Adder**

| | | 45nm | | | 65nm | | |
|---|---|---|---|---|---|---|---|
| Adder Type on 65nm (0.7V) | No of Transistors | Power µW | Delay (ps) | Area (µm²) | Power µW | Delay (ps) | Area (µm²) |
| PROPOSED | 14 | 0.036 | 2 | 8 | 0.208 | 3 | 18 |
| MCIT ADDER [1] | 16 | 0.042 | 2 | 15*7 | 0.371 | 4 | 220 |
| SHANNON BASED [2] | 18 | 0.19 | 10 | 120 | 0.28 | 12 | 224 |
| CPL [1] | 18 | 0.817 | 14 | 43*8 | 2.247 | 23 | 190 |
| CMOS [1] | 28 | 1.49 | 654 | 25*7 | 3.76 | 854 | 530 |
| CMOS 10T [2] | 10 | 18 | 19 | 180 | 25 | 26 | 348 |

**TABLE II Comparison of Proposed Divider Circuit at 45nm (0.4V)**

| Type of Adder used | Power (mW) | Delay (ps) | Area (µm²) |
|---|---|---|---|
| PROPOSED | 0.0047 | 40 | 36*12 |
| MCIT ADDER [1] | 0.617 | 260 | 1512x76 |
| Mixed Shannon [1] | 1.231 | 295 | 1562*89 |
| CPL [1] | 1.977 | 2264 | 1818x105 |
| CMOS [1] | 2.793 | 522 | 1858x141 |

**TABLE III Comparison of Proposed Divider Circuit at 65nm (0.7V)**

| Type of Adder used | Power µW | Delay (ps) | Area (µm²) |
|---|---|---|---|
| PROPOSED | 0.98 | 44 | 51*17 |
| Shannon Based Adder [2] | 1.1 | 50 | NA |
| CMOS [2] | 5.472 | 1460 | NA |

## CONCLUSION

The proposed adder circuit is designed by utilizing modified Shannon theorem and this adder implemented into the design of proposed divider. Predicated divider circuit gives more preponderant performance than subsisting author's design if compared its delay and area efficiency. Furthermore, the divider circuit is utilizing less number of transistors than other divider circuits due to the utilization of Shannon predicated adder design approach. The customary arrangement transistors tree structure is reducing critical path in the circuit, which yields lower area, and less propagation delay, lower power dissipation. The proposed adder predicated divider circuit may utilized in Digital Signal Processing circuits, like DFT ,FFT etc ,due to the lower area, and less propagation delay ,lower power dissipation.

## REFERENCES:

[1] Owk Prasanth Kumar "Non-Restoring Divider Circuit Using a MCIT Based Adder Cell having Low Energy and High" International Journal of Scientific & Engineering Research Volume 3, Issue 4, April-2012 ISSN 2229-5518
[2] C.Senthilpari, S.Kavithaa and Jude Joseph "lower delay and area efficient non-restoring array divider by using shannon based adder technique" ICSE 2010 Proc. 2010, Melaka, Malaysia
[3] D. Markovic´ , B. Nikolic" V.G. Oklobdf ija, A general method in synthesis of pass-transistor circuits, Microelectronics. J. 31 (2000) pp. 991—998.
[4] Kiat-seng Yeo and Kaushik roy "Low-voltage, low power VLSI sub systems "Mc Graw-Hill publication. ISBN: 0-07-143786-X
[5] R. Zimmermann, W. Fichtner, Low-power logic styles: CMOS versus pass-transistor logic, IEEE J. Solid-State Circuits 32 (7) (1997) pp. 1079—1090.
[6] Qiao Tong and Niraj K. Jha "Design of C-Testable DCVS Binary Array Dividers" Volume 26, Issue 2, Feb. 1991 pp.134 — 141.
[7] Scott C. Smith "Design of a NULL Convention Self-Timer Divider" ACM Conference proceeding ESA/VLSI 2004, pp. 447 — 453.
[8] Goel, S. Kumar, A. Bayoumi, M. A. "Design of Robust, Energy-Efficient Full Adders for Deep-Sub micrometer Design Using Hybrid-CMOS Logic Style" IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Dec. 2006 Volume: 14, Issue: 12, pp. 1309-1321
[9] I-Chyn Wey; Chun-Hua Huang; Hwang-Cherng Chow "A new low- voltage CMOS 1-bit full adder for high performance applications" ASIC, 2002. Proceedings. 2002 IEEE Asia-Pacific Conference, 2002, pp. 21 — 24.
[10] M. Alioto, G. Palumbo, "Analysis and comparison on full adder block in submicron technology" IEEE Transactions on VLSI Systems, VOL. 10, NO. 6, DEC. 2002.
[11] Tung N. Pham Earl E. Swartzlander "Design of Radix 4 SRT Dividers for Single Precision DSP in Deep Submicron CMOS Technology" Signal Processing and Information Technology, 2006 IEEE International Symposium on Publication Date: Aug. 2006 pp. 236-241.
[12] A.Th. Schwarzbacher, M. Brutscheck1, O.Schwinge,J.B.Foley "Constant Divider Structures of the Form 2n11 1" Irish Signals and Systems Conference 2000, Dublin, Ireland, pp. 368-375, June 2000.
[13] Donald A. Neamen "Microelectronics: Circuit-Analysis-andDesign"third international edition, ISBN 007-125443-9, 2007, pp. 137-139.
[14] Wai-Kai Chen "the VLSI hand book 2nd edition" CRC press, Taylor & Francis Group 2007.
[15] IACSIT International Journal of Engineering and Technology, Vol.2, No.3, June 2010 ISSN: 1793-8236
[16] A Shannon Based Low Power Adder Cell for Neural Network Training K.Nehru, *Member, IACSIT,* Dr.A.Shanmugam, S.Deepa and R.Priyadarshini