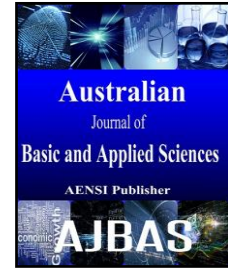




ISSN:1991-8178

## Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com



### Design And Implementation of Low Power Adder Using Recursive Approach

<sup>1</sup>Dr.M.Kathirvelu and <sup>2</sup>Ms.S.Suganya

<sup>1</sup>Professor, Department of ECE, KPR Institute of Engg & Technology, Coimbatore, India.

<sup>2</sup>PG Student, Department of ECE, KPR Institute of Engg & Technology, Coimbatore, India.

#### ARTICLE INFO

##### Article history:

Received 3 June 2015

Received in revised form 17 June 2015

Accepted 1 August 2015

Available online 15 October 2015

##### Keywords:

Full adders, parallel self-timed adder, recursive method based adder

#### ABSTRACT

In today's world there is a great need for low power design and area efficient high performance. The power consumption has been increasingly high in high performance processing units which are capable of processing an increased amount of binary data in bytes. The Full adders are the basic building blocks of the computational units such as ALU & MAC at lower levels of abstraction. Low power adders are mainly aimed at minimizing the number of transistors thereby minimizing the power consumption. In this paper the proposed method presents a parallel single-rail self-timed adder. It uses recursive method for performing multi bit binary addition. A practical implementation is provided along with a completion detection unit. The implementation is regular and does not have any practical limitations of high fan outs. The recursive method based adder consumes least power among other Self-timed adders.

© 2015 AENSI Publisher All rights reserved.

**To Cite This Article:** Dr.M.Kathirvelu and Ms.S.Suganya., Design And Implementation of Low Power Adder Using Recursive Approach. *Aust. J. Basic & Appl. Sci.*, 9(27): 655-658, 2015

#### INTRODUCTION

Low power has emerged as a principal theme in today's electronics industry. The need for low power has caused a major paradigm shift where power dissipation has become as important a consideration as performance and area. Binary addition is the single most important operation that a processor performs. In addition to explicit arithmetic (such as addition, subtraction, multiplication, and division) performed in a program, additions are performed to increment program counters and calculate effective address. The performance of processors is significantly influenced by the speed of their adders. The adders can be sequential or combinational. As the sequential adders are bound to perform slowly due to its incremental nature of operation it is not considered for parallel and fast adders. The Half-Adders (HA) are simplest single bit adders. The full adders are single bit adders with the provision of carry input and output. The full-adders are typically composed of two HAs and hence are expensive than half-adders in terms of area, time and interconnection complexity. The most common approach for

designing multi-bit adders is to form a chain of FA blocks by connecting the carry out bit of a FA to the carry in bit of the next FA block. Self-timed or asynchronous design solves the problems by removing a global clock. Most of the adders have been designed for synchronous circuits even though there is a strong interest in clockless circuits. Asynchronous circuits do not assume any quantization of time. Therefore, they hold great potential for logic design as they are free from several problems of clocked (synchronous) circuits. The Full adders are the building blocks of the ALU and MAC units of a processor.

#### 2. Various forms of full adder realization:

The Full adder realizations at transistor level is based on implementation of one of the pair of Equations shown in table 1 using Low power logic styles. Some of the common Logic styles used are DPL (Mariano Aguirre *et al* 2011), CMOS (Wallace 1994), CPL (Yano *et al* 1990), SR-CPL (Zimmerman R. and Fichtner W 1997), and Hybrid styles (Chang *et al* 2005).

**Table 1:** Various forms of Sum and Carry equations for Full adder logic

S. No	Equations
1	Sum = $A \oplus B \oplus C$ (1) Carry = $A.B + B.C + C.A$ (2)
2	Sum = $A.\bar{B}.\bar{C} + \bar{A}.B.\bar{C} + \bar{A}.\bar{B}.C + A.B.C$ (3) Carry = $A.B.\bar{C} + A.\bar{B}.C + \bar{A}.B.C + A.B.C$ (4)
3	Sum = $A.B.C + \bar{C}.(A+B+C)$ (5) Cy = $A.B + C.(A+B)$ (6)
4	Sum = $\bar{C}.(A \oplus B) + \bar{C}.(A \odot B)$ (7) Carry = $A.(A \odot B) + C.(A \oplus B)$ (8)
5	Sum = $\bar{C}.(A \oplus B) + \bar{C}.(A \odot B)$ (9) Carry = $C.(A.B) + C.(A+B)$ (10)
6	Sum = $C.(A \oplus B) + C.(A \odot B)$ (11) Carry = $A.(A \odot B) + C.(A \oplus B)$ (12)
7	Sum = $A \oplus B \oplus C$ (13) Carry = $A.B + C.(A \oplus B)$ (14)
8	Sum = $A \oplus B \oplus C$ (15) Carry = $\overline{(A.B).(B.C).(A.C)}$ (16)
9	Sum = $A \oplus B \oplus C$ (17) Carry = $\overline{A.B.(C.(A+B))}$ (18)
10	Sum = $\overline{(A \odot B) \odot C}$ (19) Carry = $A.(A \oplus B) + C.(A \odot B)$ (20)

The total power consumption in a VLSI circuit is given by the equation (21).

$$V_{dd} \cdot F_{clk} \cdot \sum_i V_{swing} \cdot C_{load} \cdot P_i + V_{dd} \cdot \sum_i I_{sc} + V_{dd} \cdot I_t \quad (21)$$

The first two components of the Equation (21) are termed as the dynamic power and it is subject to the maximum amount of the total power consumed in CMOS VLSI circuits. In Layout level the load capacitance is sum of the diffusion capacitance and the cross talk capacitance. The diffusion capacitance is due to the diffusion regions and the metal layers. Hence the minimization of the diffusion regions and the metal layers can reduce the diffusion capacitance and thus lead to reduction in power consumption. The capacitance due to the cross talk between metal layers will be reduced by increasing the distance of separation and preventing the use of long metal layers as much as possible. The concept of shared diffusion can be used in the layout realization this will minimize the diffusion layer area as well as the power consumption due to diffusion capacitance.

### 3. Analysis of Various full adders:

There are many full adder realizations available in the literature survey. The most of the adders are aimed to reducing power consumption by decreasing the number of transistors. The detailed analysis of the some of the existing low power adders are given below.

#### CMOS-28T Adder:

The CMOS 28T adder is designed with equations (5), (6). In this adder, the carry is computed first using the carry equation and is used to compute the sum of the bits. In this adder, no

threshold voltage degradation and the output swings from rail to rail which makes this adder a preferable one for synthesis of tree based multipliers. The main drawback is increased delay in sum generation as the logic involved in sum generation utilizes the use of the carry produced.

#### SERF Full adder:

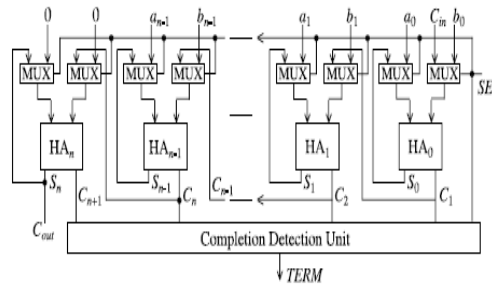
The Static Energy Recovering Full adder (SERF) is profoundly known to be a low power full adder employing reduced number of transistors based on the equation (19) & (20). In SERF, 4T Wang's XNOR is used for its sum realization and pass transistor logic for realization carry generation. The advantage is low power consumption and delay is considerably low compared to CMOS 28 T. The drawback is the output suffers from threshold voltage degradation and it is unusable for cascaded parallel structures.

#### Adder10T09A:

The adder named 10T09A discussed by (Shams M. and Bayoumi M. 1999) uses 10 transistors and the output is obtained based on the equation (19) and (20). It uses a Wang's XNOR and G-XNOR cell. The output and merits are similar to that of an SERF adder.

#### ADDER10T09B:

The adder10T09B discussed by (Yingtao Jiang *et al*, 2004) also similar to the adder 10T09A. The difference is interconnection in the second stage after the XNOR operation of the Wang's XNOR. The adder is found to exhibit some degradation problems as that of the SERF and in this case problems in pull down and pull up are worsened when compared to that of SERF.



**DPL 28T:**

The energy efficient Dual Pass transistor Logic DPL 28T is a power efficient full adder (Mariano Aguirre *et al*, 2011). The adder utilizes Dual pass transistor based multiplexer and 28 transistors for the entire adder realization. The adder is aimed at minimizing power consumption by use of multiplexers on the final stage. The logic governing the sum and carry in a DPL 28T are based on the equations (11) & (12). The adder produces an output which is free from threshold voltage degradation for majority of the input combinations with lesser delay. The demerit is, it uses the multiplexers which reduces the driving capability of the adder. Hence this adder is not capable in driving large capacitive loads. The pass transistor based XOR and XNOR gates used in the Sum realization tend to suffer from weaker Pull-up and Pull-down. The Sum output obtained suffers from weak pull up from the transition of the inputs from 001 to 010 during this period the XNOR goes to state '0' and XOR goes to state '1'.

that do not need any carry chain propagation. The design of PASTA is regular and uses half-adders (HAs) along with multiplexers requiring minimal interconnections. Thus, it is suitable for VLSI implementation. The design works in a parallel manner for independent carry chain blocks. Self-timed refers to logic circuits that depend on and/or engineer timing assumptions for the correct operation. Self-timed adders have the potential to run faster averaged for dynamic data, as early completion sensing can avoid the need for the worst case bundled delay mechanism of synchronous circuits.

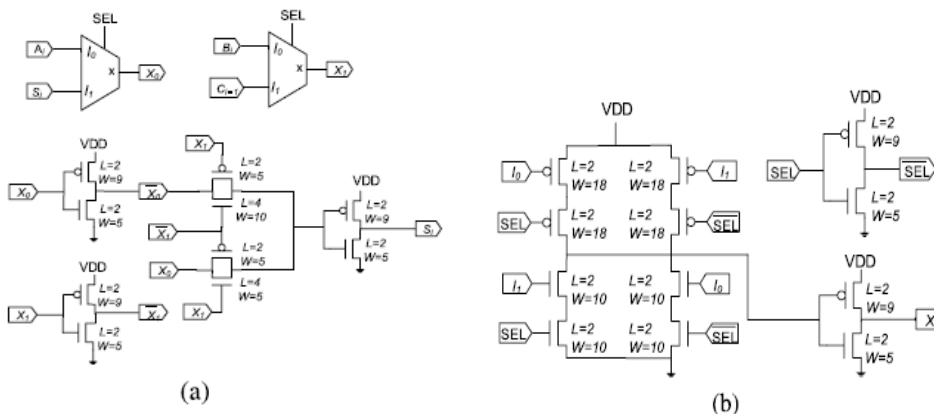
**Working:**

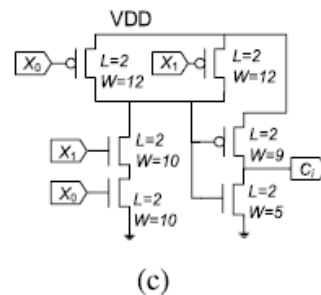
The adder first accepts two input operands to perform half additions for each bit. Subsequently, it iterates using earlier generated carry and sums to perform half-additions repeatedly until all carry bits are consumed and settled at zero level. The selection input for two-input multiplexers corresponds to the Req handshake signal and will be a single 0 to 1 transition denoted by SEL. It will initially select the actual operands during SEL = 0 and will switch to feedback/carry paths for subsequent iterations using SEL = 1. The feedback path from the HAs enables the multiple iterations to continue until the completion when all carry signals will assume zero values.

**4. Proposed Parallel Self Timed Adder:**

General block diagram of Parallel Self Timed Adder

Parallel Self Timed Adder is based on recursive formulation. The operation is parallel for those bits





CMOS implementation of PASTA.

- (a) Single-bit sum module.
- (b) 2×1 MUX for 1-bit adder.
- (c) Single-bit carry Module.

### Conclusion:

**Table 2:** Performance comparison of various full adders

Full adder Architecture	No. of transistors	Power ( $\mu$ W)	Delay (nS)
CMOS 28T	28	11.39	0.031
SERF	10	4.36	0.051
Adder10T09A	10	5.35	0.052
Adder10T09B	10	4.37	0.053
DPL28T	28	5.68	0.029
Proposed Parallel Self Timed Adder	26	4.28	0.025

This brief presents comparison of various full adder and Parallel Self Timed Adder. The analysis of Adder architectures is done using the standard library cells available with Microwind 3.5. The design achieves a very simple  $n$ -bit adder that is area and interconnection wise equivalent to the simplest adder namely the RCA. Moreover, the circuit works in a parallel manner for independent carry chains, and thus achieves logarithmic average time performance over random input values.

### REFERENCES

- Geer, D., 2005. "Is it time for clockless chips? [Asynchronous processor chips]," *IEEE Comput.*, 38(3): 18-19.
- Sparsø, J. and S. Furber, 2001. *Principles of Asynchronous Circuit Design*. Boston, MA, USA: Kluwer Academic.
- Choudhury, P., S. Sahoo and M. Chakrabarty, 2008. "Implementation of basic arithmetic operations using cellular automaton," in *Proc. ICIT*, pp: 79-80.
- Rahman, M.Z. and L. Kleeman, 2013. "A delay matched approach for the design of asynchronous sequential circuits," *Dept. Comput. Syst. Technol.*, Univ. Malaya, Kuala Lumpur, Malaysia, Tech. Rep. 05042013.
- Riedel, M.D., 2004. "Cyclic combinational circuits," Ph.D. dissertation, Dept. Comput. Sci., California Inst. Technol., Pasadena, CA, USA.
- Liu, W., C.T. Gray, D. Fan and W.J. Farlow, 1994. "A 250-MHz wave pipelined adder in 2- $\mu$ m CMOS," *IEEE J. Solid-State Circuits*, 29(9): 1117-1128.
- Cheng, F.-C., S.H. Unger and M. Theobald, 2000. "Self-timed carry-look-ahead Adders," *IEEE Trans. Comput.*, 49(7): 659-672.
- Nowick, S., 1996. "Design of a low-latency asynchronous adder using speculative completion," *IEE Proc. Comput. Digital Tech.*, 143(5): 301-307.
- Cornelius, C., S. Koppe and D. Timmermann, 2006. "Dynamic circuit techniques in deep submicron technologies: Domino logic reconsidered," in *Proc. IEEE ICICDT*, pp: 1-4.
- Anis, M., S. Member, M. Allam and M. Elmasry, 2002. "Impact of technology scaling on CMOS logic styles," *IEEE Trans. Circuits Syst., Analog Digital Signal Process.*, 49(8): 577-588.
- Weste, N. and D. Harris, 2005. *CMOS VLSI Design: A Circuits and Systems Perspective*. Reading, MA, USA: Addison-Wesley.