www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 11 November, 2013 Page No. 3106-3110

Implementation of CMOS circuits In Logic optimization using logical effort technique

Akansha Rajput,Satyendra Sharma
Department of Electronics and Communication Engineering
Noida Institute of Engineering & Technology
Greater Noida, 201310
Email: akansha493@gmail.com,satyendracommn@gmail.com

ABSTRACT: - *The method of logical effort is an easy way to estimate delay and dynamic power in a CMOS circuit. We can select the fastest candidate by comparing delay estimates of different logic structures. The method also specifies the proper number of logic gates on a path and the best transistor sizes for the logic gates. Because the method is easy to use, it is ideal for evaluating alternatives in the early stages of a design and provides a good starting point for more intricate optimizations. In proposed work, the implementation of logical effort technique in static CMOS circuits like conventional adder, array multiplier, decoder and multiplexer occurred. So if its transistors sizing changed or adjusted such that its delay and PDP reduce then as a result of this bigger circuits also get the benefit of this changes*

Keywords –*Transistor sizing, Logical effort, static CMOS circuit.*

I. introduction

This is important especially with the projected slower improvement in the battery-technology compared to the progress pace of the semiconductor industry [1]. Thus, power estimation, analysis and optimization are essential for CMOS IC design. Using circuit simulators such as Spice to predict the power dissipation in large circuits is an unfeasible solution due to large computing-time. Hence, developing accurate power models is necessary for designing and optimizing very large scale integrated (VLSI) CMOS circuits.

In CMOS circuits there are two sources for power dissipation: static and dynamic. Static power dissipation is mainly due to standby leakage current [2, 3] and it is not a function to the switching frequency of a CMOS gate. This source is out of the scope of this work. Dynamic power, in contrast, is the power consumed by a CMOS gate when its output toggles between high and low logic levels [4, 5]. Short-circuit power and switching power are the main components of the dynamic power dissipation. The first component is produced by the direct DC path between the supply voltage and ground when both the NMOS and PMOS transistors are ON during the input transition. Switching power, on the contrary, contributes the major portion of the power consumption in CMOS circuits, and is the result of the charging and discharging of the output capacitance.

Reducing the power dissipation in IC designs was always a key concern and the force behind moving from one technology to another. Under specific delay constraints, power may be reduced at different levels of the design abstractions. At the circuit level, which is the target of this paper, power optimization is achieved by transistor sizing, supply voltage and/or threshold voltage scaling.

The works in [6–9] attempt to optimize switching power through transistor sizing. Turgis et al. [6] consider a chain of inverters where a tapering ratio of 4.25 is found to minimize the power dissipation. In [7] it has been proven that the sum of the input capacitances of an inverter chain is minimized when inverters bear the same fan out. For a path with general gates the minimal energy solution was obtained in [8] by numerically solving a set of equations, which was resulted from LaGrange method. BiCMOS circuits were considered in [9]. This method uses an iterative process to size and optimize the design's gates where the high drive capability buffered gates (i.e. ,BiCMOS) with sufficiently low fan-out are identified and replaced with a lower power unbuffered (i.e., CMOS) version. This work seeks the minimization of network delay subject to network power dissipation.

Optimizing the supply voltage to reduce the power dissipation was the target of many researchers. Considering micro-processors, Ca ie tal. [10] Propose a dual supply voltage technique to reduce both the static and the dynamic power dissipation of CMOS circuits. Low supply voltage and low threshold-voltage devices are used for high activity circuits while higher supply voltage and high threshold voltage devices are assigned to the low activity circuitry. In [11] the power optimization has been achieved in two steps. First, maximum delay is assigned to all gates then in the next step each individual gate is optimized iteratively for minimum power by finding the proper combination of the transistor widths; as well as threshold and supply voltages.

This paper survey of logical effort technique in section II, after that in section III shows modified work – logical

implementation and simulation result and in section IV concludes this paper

Logical Effort Theory

Logical effort theory is based on delay caused by the capacitive loads that the logic gate drives and by the topology of the logic gate. Clearly, as the load increases, the delay increases, but delay also depends on the logic function of the gate. Logical effort theory uses inverters as basic building block as it is simplest logic gate and compares the driving capabilities of other gates with it.

II. Logical Effort for individual gates

Logical effort technique [12] expresses the delay of CMOS gates D as a normalized value of τ,

$$D = \tau d, \quad (1)$$

Where, τ is technology independent parameter. The delay in logic can be expressed as sum of two components, a fixed part called the parasitic delay, p, and a part proportional to the load on the gate's output, called the effort delay, f. The total delay, measured in units of τ, is the sum of the effort and parasitic delays. Thus

$$d = f + p \quad (2)$$

The parasitic delay of a logic gate is fixed, independent of the size of the logic gate and of the load capacitance. Effort delay is dependent on load and properties of logic gate driving that load. The effort delay of the logic gate is the product of these two factors.

$$f = g.h \quad (3)$$

Where *g is* Logical Effort and h is Electrical Effort. The logical effort represents how much worse the gate is producing output current as compared to inverter. It is independent of the size of the transistors in the circuit. The electrical effort describes how the electrical environment of the logic gate affects performance and size of transistor on load driving capability.

$$h = C_{out}/C_{in} \quad (4)$$

Where Cout is the output load capacitance and Cin is the capacitance presented by presented by the input terminal of the Logic gate. Combining equations 2 and 3, we obtain the basic equation that models the delay through a single logic gate, in units of τ:

$$d = gh + p \quad (5)$$

This formulation separates τ, *g*, *h*, and *p*, the four contributions to delay. The backbone of logical theory is the calculation of logical effort *g*. it is defined as the ratio input capacitance of the gate to that of inverter that delivers same output current.

$$g = C_i/C_{inv} \quad (6)$$

Where C_i is the input capacitance of the logic gate and C_{inv} is capacitance of the reference inverter. Logical Effort is the ratio of capacitances, the units use to measure capacitance may be arbitrary. The capacitance of the transistor's gate is proportional to w, and its ability to produce output current, or conductance, is also proportional to w. in most CMOS processes , pull up transistors must be wider than pull down transistors to have the same conductance since mobility of electrons is more as compared to holes. $\mu=\mu_n/\mu_p$ is the ratio of PMOS to NMOS width in an inverter for equal conductance. Under this assumption, an inverter will have a pull down transistor of width w and a pull up transistor of width 2w so the total capacitance can be said to be 3w.Hence by definition:-

$$g = 3/3 = 1 \quad (7)$$

III. Multi-stage circuit

The logical effort along a path compounds by multiplying the logical efforts of all the logic gates along the path. We use the uppercase symbol G to denote the *path logical effort*, so that it is distinguished from g, the logical effort of a single gate in the path. So,

$$G = \Pi g_i, \quad \textbf{(7)}$$

The electrical effort along the path through a network is simply the ratio of the capacitance that loads the logic gate in the path to the input capacitance of the first gate in the path.

$$H = \frac{C_{out}}{C_{in}} \quad (8)$$

Where C_{out} and C_{in} are the output and input capacitances of the path, respectively. For a multi-stage logic circuit, another parameter should be defined in order to cover fan-out of each stages. When a logic gate has other gates connected to its output which are not in the target path, a fraction of the output current is directed along the path while the rest is directed off that path. Here we define the branching effort (b) as

$$b = \frac{C_{on} + C_{off}}{C_{off}} \quad (9)$$

Where C_{on} is the load capacitance along the path and C_{off} is the capacitance of the connections that lead of the path. For no branch b=1.

The branching effort along the entire path is nothing but multiplication of the branching effort of each stage along the path.

$$B = \pi bi \quad (10)$$

Finally the path effort (F) can be defined as

$$F = GHB \quad (11)$$

Now we can find the optimal value of the stage effort in order to have minimum delay along a given path. The path delay (D) is the sum of the delays of each stage and it can be divided into two parts consisting of the path effort delay (D_f) and the path parasitic delay (P) as

$$D = \sum d_i = D_f + P \quad (12)$$

$$D_f = \sum g_i + h_i \quad (13)$$

And the path parasitic delay is

$$P = \sum p_i \quad (14)$$

This minimum delay is achieves when the stage effort is,

$$f = g_i h_i = F^{1/N} \quad (15)$$

So, $$D = NF^{1/N} + P \quad (16)$$

we can determine the transistor sizes of gates along a path.

$$C_{in} = \frac{g_i C_{out}}{f} \quad (17)$$

Table 1: Logical effort of static CMOS gates. (γ = 2, where γ is the ratio between PMOS and NMOS transistor size.)

Gate type	**Number of inputs**					
	1	**2**	**3**	**4**	**5**	**n**
Inverter	**1**					
NAND		**4/3**	**5/3**	**6/3**	**7/3**	**(n+2)/3**
NOR		**5/3**	**7/3**	**9/3**	**11/3**	**(2n+1)/3**
Multiplexer		**2**	**2**	**2**	**2**	**2**
XOR		**4**	**12**	**32**		

Table 2: Parasitic delay of static CMOS gates

Gate type	**Parasitic delay**
Inverter	p_{inv}
NAND	np_{inv}
NOR	np_{inv}
Multiplexer	$2np_{inv}$
XOR	$4p_{inv}$

IV. Logic implementation with simulation work

In this work implemented logical effort technique in static CMOS circuits like conventional adder, array multiplier, decoder and multiplexer. These circuits are used very frequently in many bigger circuits. So if its transistors sizing changed or adjusted such that its delay and PDP reduce then as a result of this bigger circuits also get the benefit of this changes. Logical effort technique mainly deals with scaling of the transistors in such manner that will reduce the delay of the path in which we apply this technique. So make a module of any circuit with and without use of logical effort technique and Use this as a basic building block and compare the performance. Here first used conventional adder.

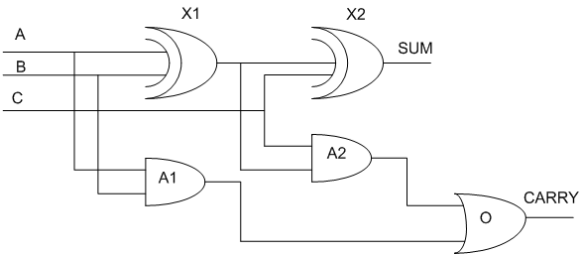

Fig.1 Conventional Full Adder

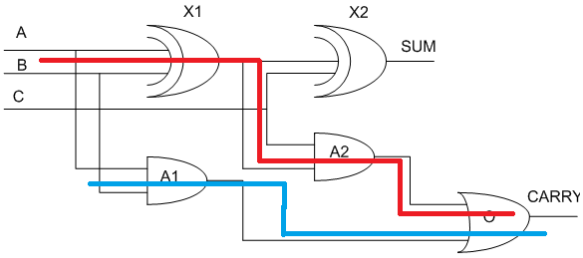

Fig.2 Full adder with highlighted paths on which logical effort applied

Now using this adder as a building block with logical effort technique, it is going to apply in multiplexer.

4:1 Multiplexer

Multiplexer is well known circuit which is used for flow of different signal through a single output path in a controlling way.

Schematic and waveforms of multiplexer without using of logical effort:-

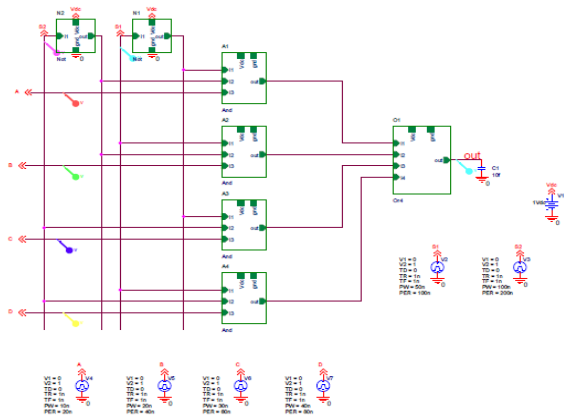

Fig.3 Schematic view of 4:1 Multiplexer

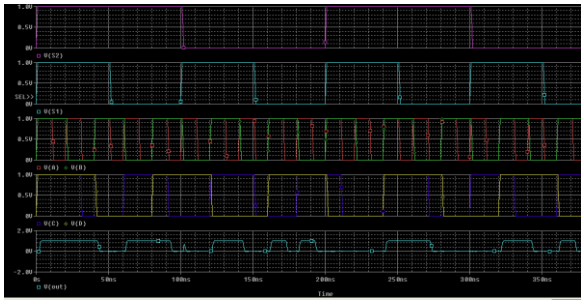

Fig.4 Waveform of 4:1 multiplexer

Schematic and waveforms of multiplexer with using of logical effort:-

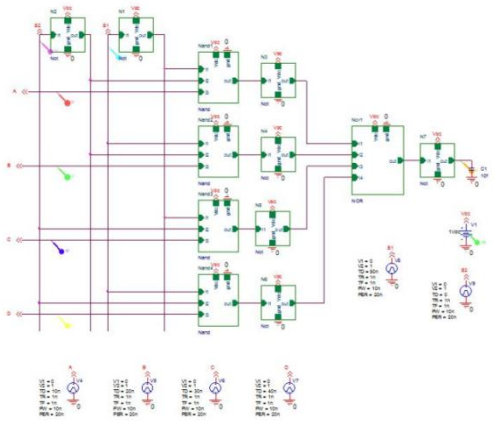

Fig.5 Schematic view of 4:1 Multiplexer with logical effort

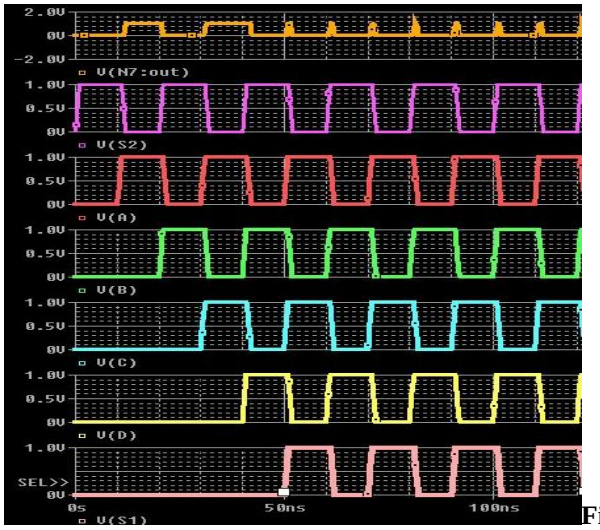

Fig.6 Waveforms of 4:1 Multiplexer with logical effort

From the comparison we can see delay decrease nearly 47% and PDP saving nearly 61.6%. So that the performance of adder has been improved by using logical effort technique.

V Conclusion

In this work found that Logical effort technique is prove to be good option to choose better circuit for particular task very easily by taking the example of multiplexer.

It is best transistor sizes for the logic gates and has to be proven an easy way to estimate power and delay.

References: -

[1] Logical effort based dynamic power estimation and optimization of static CMOS circuits Original Research Article Integration, the VLSI Journal, Volume 43, Issue 3, June 2010, Pages 279-288, A. Kabbani

[2] K. Lahiri, A. Raghunathan, S. Dey, D. Panigrahi, Battery-driven system design: a new frontier in low power design, Proceedings of the IEEE 15th International Conference on VLSI Design, 2002, pp. 261–267.

[3] I. Sutherland, B. Sproull, D. Harris, in: Logical Effort: Design Fast CMOS Circuits, Morgan Kaufmann publishers, 1999 January.

[4] P. Rezvani, A.H. Ajami, M. Pedram, H. Savoj, LEOPARD: A logical effort-based fan-out optimization for area and delay, Proceedings of IEEE International

Conference on CAD, 1999, pp. 516–518.

[5] S. Turgis, N. Azemard, D. Auvergne, Design and selection of buffers for minimum power-delay product, Proceedings of the European Design and Test Conference, March, 1996, pp. 224–228.

[6] H.Q. Dao, B.R. Zeydel, V.G. Oklobdzija, Energy minimization method for optimal energy-delay extraction, Proceedings of IEEE Solid-State Circuits Conference, 2003, pp. 177–180.

[7] Logical_effort_model_extension_with_temperature_and voltage variations Chun-Hui Wu; Shun-Hua,Lin;,Herming,Chiueh Thermal Inveatigation of ICs and Systems, 2008. THERMINIC2008. Publication Year: 2008 , Page(s): 85 – 88

[8] D. Markovic , V. Stojanovic , B. Nikolic , M.A. Horowitz, R.W. Brodersen, Methods for true energy-performance optimization, IEEE Journal of Solid-State Circuits 39 (8) (2004) 1282–1293 Aug.

[9] K.S. Lowe, P.G. Gulak, A joint gate sizing and buffer insertion method for optimizing delay and power in CMOS and BiCMOS combinational logic, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 17 (5) (1998) May.

[10] J. Cai, Y. Taur, S.F. Huang, D.J. Frank, S. Kosonocky, R.H. Dennard, Supply voltage strategies for minimizing the power of CMOS in: processors, IEEE Symposium on VLSI Technology, Digest of Technical Papers, 2002 pp. 102–103.

[11] P. Pant, V. De, A. Chatterjee, Device-circuit optimization for minimal energy and power consumption in CMOS random logic networks, Proceedings of the 34th Design Automation Conference, 1997, pp. 403–408.

[12] A. Shebaita, Y. Ismail, Multiple threshold voltage design scheme for CMOS tapered buffers, IEEE Transactions on Circuits and Systems II 55 (1) (2008) 21–25 Jan.