

A BDD-based Design of An Area-Power Efficient Asynchronous Adder

***Prof. Bhagyalakshmi V. and **Prof. D.B. Mantri**

**Asst.Professor(EXTC), V.V.P.I.E.T, Solapur.*

E-mail: lakshmianand04@gmail.com

***H.O.D(EXTC), V.v.P.I.E.T, Solapur.*

E-mail: dbmantri@yahoo.co.in

Abstract

Asynchronous system design in recent years has reemerged as an important vehicle in the field of high performance, low power and secure computing. On the other hand Binary Decision Diagrams (BDDs) have found significant applications for many years in the design, synthesis, verification, and testing of VLSI circuits. In this paper we have presented the design of a hybrid Domino Pass Transistor Logic CMOS (PTL-CMOS) based 2-bit asynchronous adder, the PTL part of which is designed using the principles of BDD. Furthermore using this design as basic building block a 8-bit asynchronous adder has been implemented. The simulation results indicate a reduction in number of transistors over Minimal Energy Dual-bit Dynamic adder (MEDB) adder without any compromise in the delay. The circuit is simulated using Cadence tool, UMC 180nm, 1.5 technology.

Introduction

Asynchronous system design represents an important design methodology in recent deep sub-micron technologies. Many recent works have emphasized the significance of these self-timed systems. Though synchronous circuits have been ubiquitously used in VLSI design, these systems are based on a global clock and as a consequence the clock frequency of the circuit is always set by its worst case behavior regardless of how infrequently the worst case occurs, thereby degrading the delay of the circuits. Also, the clock distribution circuitry for generating minimal skew/jitter global clock consumes a lot of power. The above problems are circumvented using asynchronous circuit design. High speed and low power solutions are possible with asynchronous design techniques using simple handshaking and completion detection logic. Two-phase protocol and 4-phase protocols are typically used for handshaking while

completion detection is done using bundled data and using dual-rail logic styles. Addition is a fundamental operation in the working of a processor and it is used to implement more complex functions such as subtraction, multiplication, division etc. Chong proposed Type-I and Type-II asynchronous adders. In Type-I each of the individual 2-bit blocks can indicate their completion early, and this requires an n-bit AND gate completion detection for an n-bit adder. So Type-I adders are used for high speed applications but they require relatively more area and power. On the other hand 2-bit blocks in Type-II adders always have the carry-out in their critical path and as a result it suffices to detect the carry completion only at the (last) nth stage with a simple 2-bit OR gate completion and consequently Type-II adders are slightly slower but require lower area and power. It may be noted that although 2-bit blocks in Type-I adder consume less power compared to those used in Type-II adders, the total power usage in Type-II adders is considerably reduced due to the use of simpler completion detection logic circuitry as compared to a large AND gate in case of Type-I adders. Since we are targeting a low power design we have used Type-II adders for our purpose of design. Some parts of the proposed adder design are BDD-based which have been implemented using Pass Transistor Logic (PTL) with top pre-charge logic, we term it as Domino PTL. The remaining parts of the design are based on Domino CMOS logic. The design is considerably optimized in terms of area and power compared to the existing designs. Our design approach represents a novel application of BDD principles for asynchronous adder design for optimizing the transistor count and power dissipation.

Common approaches to low-power and low-energy design include the following:

- Asynchronous, delay-insensitive circuits enable operation over a wide voltage range.
- Some of them employ dual-rail logic.
- Others use single-rail design.
- While still others combine single- and dual-rail for further energy minimization.

Domino Logic and BDD Preliminaries

Any Boolean function $F(x_1, x_2, \dots, x_n)$ can be represented by a BDD, which is a Directed Acyclic Graph (DAG) with one root node and two leaf nodes labeled as 0 and 1. Except the two leaf nodes, every other node represents a variable x_i , and has exactly two outgoing (decision) edges labeled 0 and 1, which represent the corresponding Shannon decomposition at that node for $x_i=0$ and 1, respectively. The number of incoming edges on each node may be zero (for the root only), one, or more. Each directed path from the root to the leaf node 1 (0) (called a complete path), represents a product term of the function F (F'), where the polarity of a variable (node) is determined by the label of its decision edge along the path. The sum of all product terms corresponding to all complete paths leading to the leaf node 1 (0) represents a sum-of-products expression of output function F (F'). The product terms corresponding to any two complete paths are mutually conflicting, i.e., they cover

disjoint sets of min-terms. Therefore, given a completely specified input combination, exactly one complete path will be selected in a BDD. An example of a BDD representing a Boolean function $F = a_1b_1 + a_2b_2 + a_3b_3$ with initial variable ordering $(a_1, a_2, a_3, b_1, b_2, b_3)$ is shown in Figure 2.1 & 2.2, where the solid (dashed) arcs represent the 1 (0) edges.

A BDD is called ordered (OBDD) if each variable appears at most once on each complete path, and if the variables appear in the same order in all other complete paths. An ordered BDD is called reduced (ROBDD) if it is devoid of any isomorphic subgraphs or any redundant nodes. A ROBDD representation is unique for a Boolean function with a given variable ordering, and hence, it has a canonical form. Without the variable ordering the size of the BDD can become exponential while a good variable ordering reduces the size of the BDD drastically.

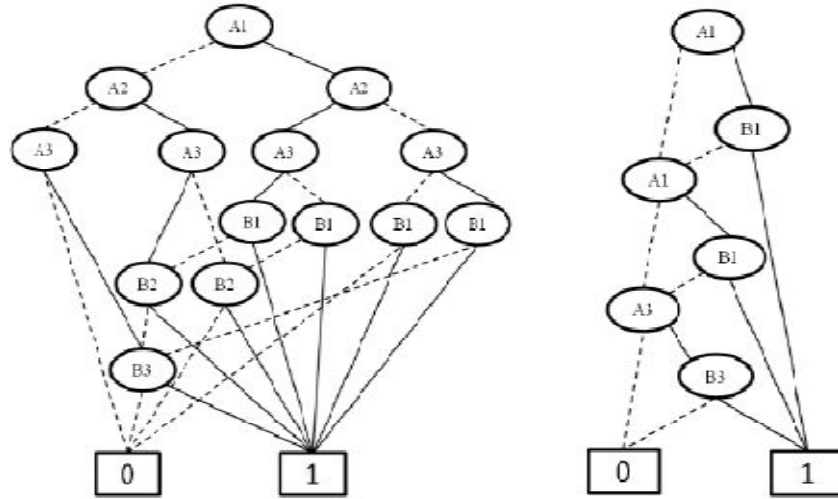


Figure 2.1: BDD & OBDD Representation.

For example, the size of the BDD in Figure 2.1 can be optimized to using the new ordering $(a_1, b_1, a_2, b_2, a_3, b_3)$ as shown in Figure 2.1. The use of BDD & OBDD makes the circuit smaller in terms of number of transistors and subsequently reduces the dissipated power during operation.

Design of a 2-Bit Asynchronous Adder

The background on the development of various asynchronous adder designs and identification of the scope of improvement in the contemporary adder designs is given in this chapter.

Analysis of Contemporary Adder Designs

Analysis of Contemporary Adder Designs Asynchronous adders take advantage of data dependent computational delays in the carry chain and indicate the end of completion using the completion detection circuitry. Asynchronous adder designs such as single bit Martin's adder, Nielsen's adder and dual-bit adders such as Chong's

adder, Dynamic Dual-Bit adder, Dual-Bit PTL adder, Minimal Energy Dual-bit Dynamic adder are the major contemporary adder designs. Martin's Adder consumes more of power as it uses PMOS transistors and dual rail encoding for both sum and carry circuits Nielsen's adder makes use of an hybrid approach, where dual rail is limited to the carry circuit (as it forms the critical path of the adder) and single rail is used for sum circuits due to which it consumes lesser power than Martin's adder. Both Martin's and Nelson's adders use 1-bit full adders as their basic building blocks. In [2] , it has been observed that 2-bit BCLA consumes lesser power, has a lower average case delay and less area compared to two separate single bit adders cascaded together in ripple carry topology. This is because of the use of complex gate realizations that lead to reduced number of nodes and buffers that need to be charged or discharged. It may be noted that in BCLA topologies the carryout bits for each block are computed using carry look-ahead technique, while carry between the blocks are rippled through. We now give a description of the existing low power (energy) Type-II dynamic dual-bit adders.

Type-II Dual-bit Adders

- i) *Chong's type II adder*: It includes carry in its critical path and as a result can simpler completion detection circuitry.
- ii) *Dynamic Dual Bit adder (DDBR)*: It is same as Chong's adder with some topological transistor re-ordering, it uses pre-charge cut-off transistor at the top thereby reducing the power since only output node is pre-charged during pre-charge phase while internal nodes are not. It has been reported to consume 10% less energy than the Chong's adder [7].
- iii) *Dual-bit PTL adder (DBPTL)*: It uses PTL logic for both Sum and Carry circuits. This circuit is reported to consume lesser power and delay than DDBR [7].
- iv) *Minimal Energy Dual-bit Dynamic adder (MEDB)*: The circuit consumes very less power in each cycle of operation due to the energy saving rules used in [7]. It reduces the delay as well compared to the above adders [7]. The circuit is shown in Figure 3.1. Considering the MEDB adder, it is observed that S0 and S1 circuits in the MEDB adder are realized more efficiently using PTL logic by the application of BDDs as it leads to a lesser number of transistors for its implementation and consequently provides a low power/energy solution for arithmetic functions which include XOR gates.

Hybrid PTL-CMOS Domino Dual-bit Adder

The Equations of the 2-bit CLA are shown below:

- $S_0 = A_0 \oplus B_0 \oplus C$
- $S_1 = (A_1 \oplus B_1) \oplus A_0 B_0 + (A_0 \oplus B_0) C$
- $C_1 = A_1 B_1 + (A_1 \oplus B_1) [A_0 B_0 + (A_0 \oplus B_0) C]$
- $C_1' = [A_1 B_1 + (A_1 \oplus B_1) (A_0 B_0 + (A_0 \oplus B_0) C)]'$

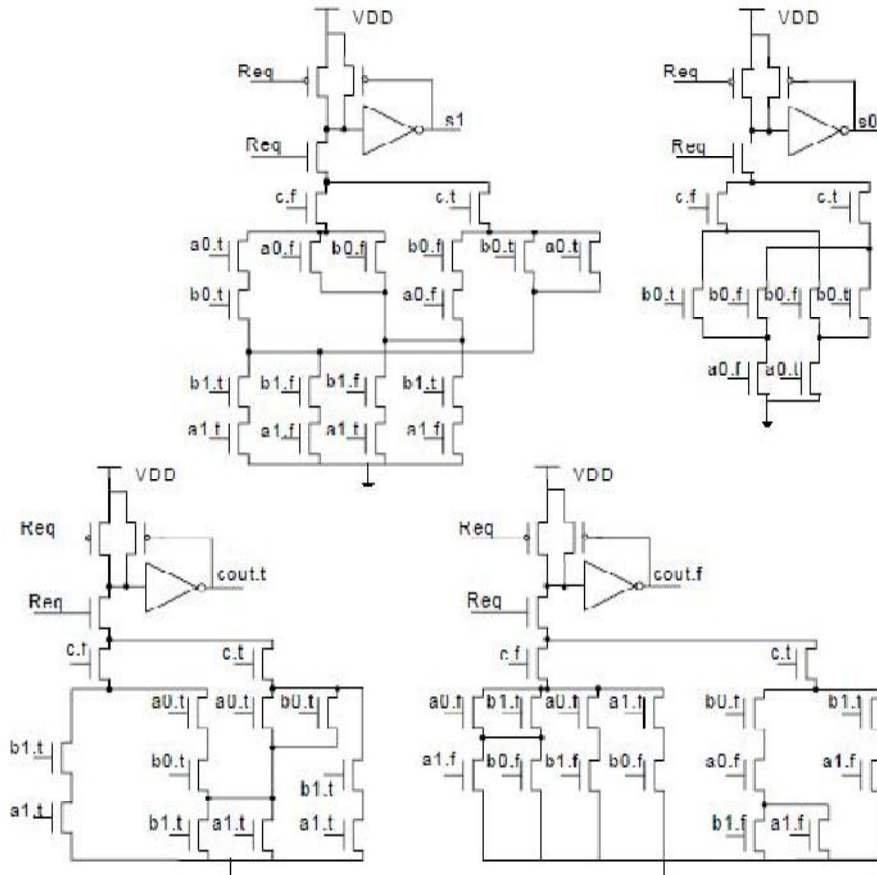


Figure 3.1: MEDB Adder.

The proposed 2-bit Domino adder incorporates all the energy saving rules proposed in and also improves the design in the following ways. Our hybrid dual-bit adder is based on the following observations derived from the above study of Type-II dual-bit adders.

- We use the fact that PTL implements XOR gates in an efficient manner, as it uses less no of transistors, so it provides a low energy solution for arithmetic functions which include XOR gates.
- As S_0 and S_1 are XOR based circuits as, we use BDD based Domino PTL implementation to realize them, (where each node of BDD is mapped to a 2x1 MUX). The BDDs for S_0 and S_1 are shown in Figure 3.4 and 3.5, respectively and the corresponding PTL MUX realization are shown in Figure 3.6 and 3.7, respectively.
- As $cout.t$ and $cout.f$ are AND-OR based functions we use Domino CMOS logic to realize them.
- The carry ($cout.t$, $cout.f$) circuit implementation is based on the reported MEDB adder.

We, however, improved the design by reducing 4 transistors in *cout.f* circuit and obtain less power dissipation and smaller IC area requirement without any compromise to the speed.

The modified carry circuit *cout.f* is shown in Figure 3.8. The circuit for the *cout.t* is shown in Figure 3.9. The schematic of the proposed 32-bit adder with OR gate as its completion detection circuitry is shown in Figure 12. In our proposed adder the critical path length is as same as the one in the MEDB adder design, which implies that the delay in our case is same as that of MEDB but with lower power consumption. This makes our design energy efficient compared to the existing asynchronous adders [9-12]. Asynchronous circuits typically require dual-rail logic for their implementation and this increases the area of the circuit.

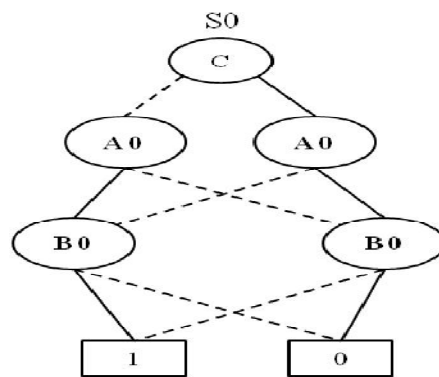


Figure 3.1: BDD for S_0 .

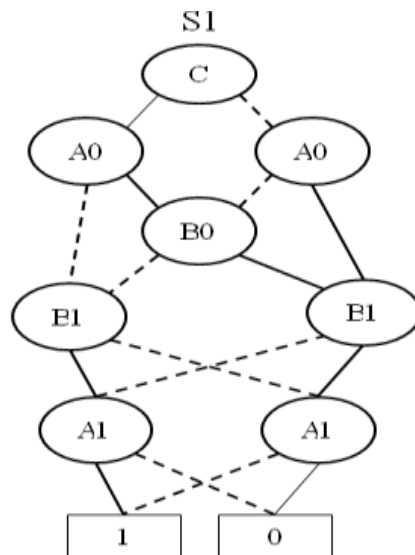


Figure 3.2: BDD for S_1 .

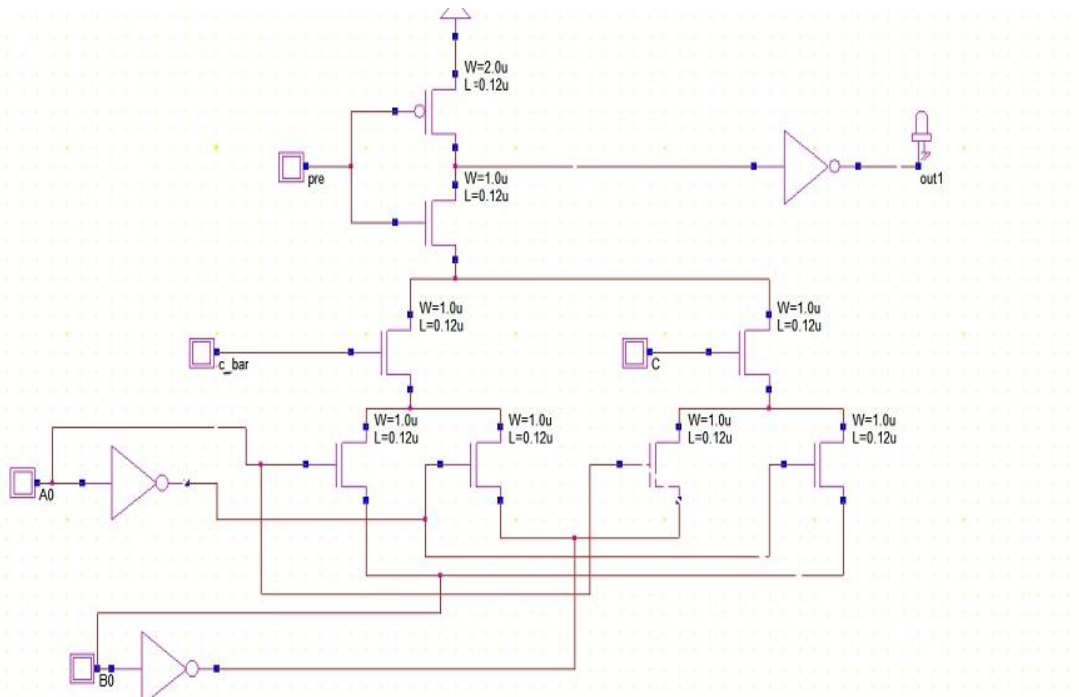


Figure 3.3: Circuit for S0.

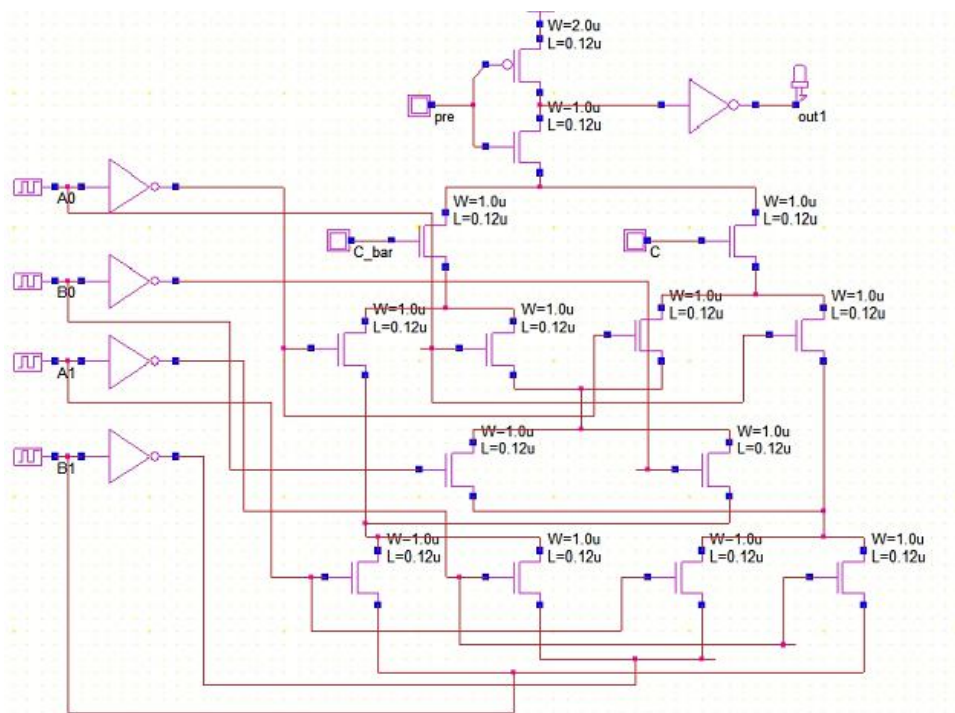


Figure 3.4: Circuit for S1.

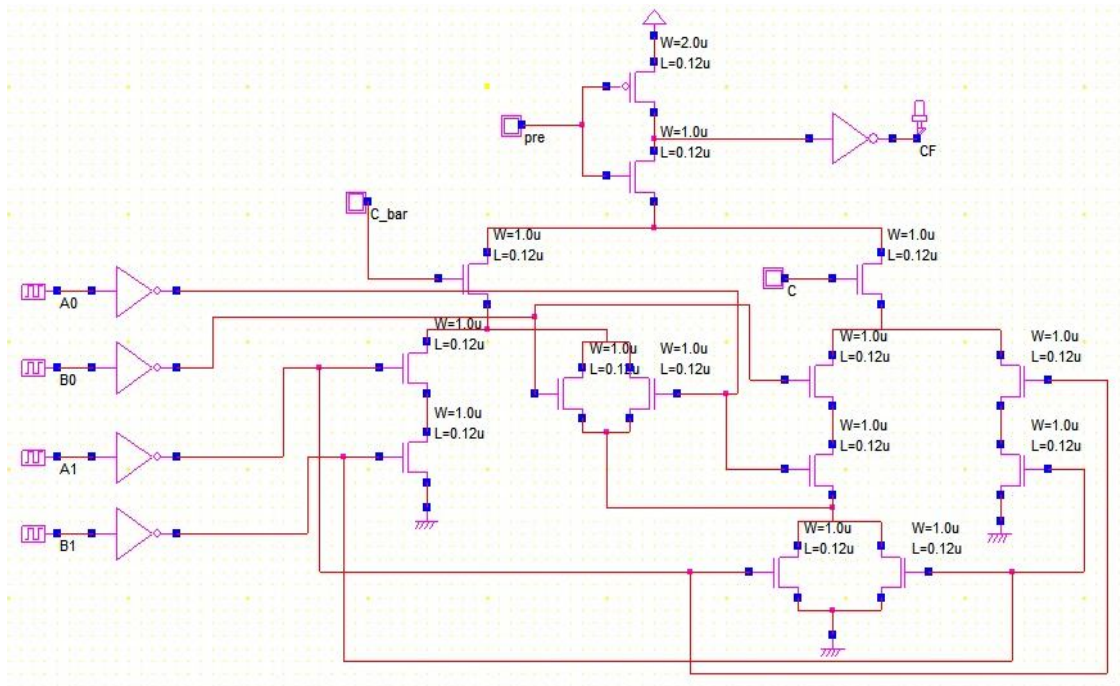


Figure 3.5: Circuit for cout. F.

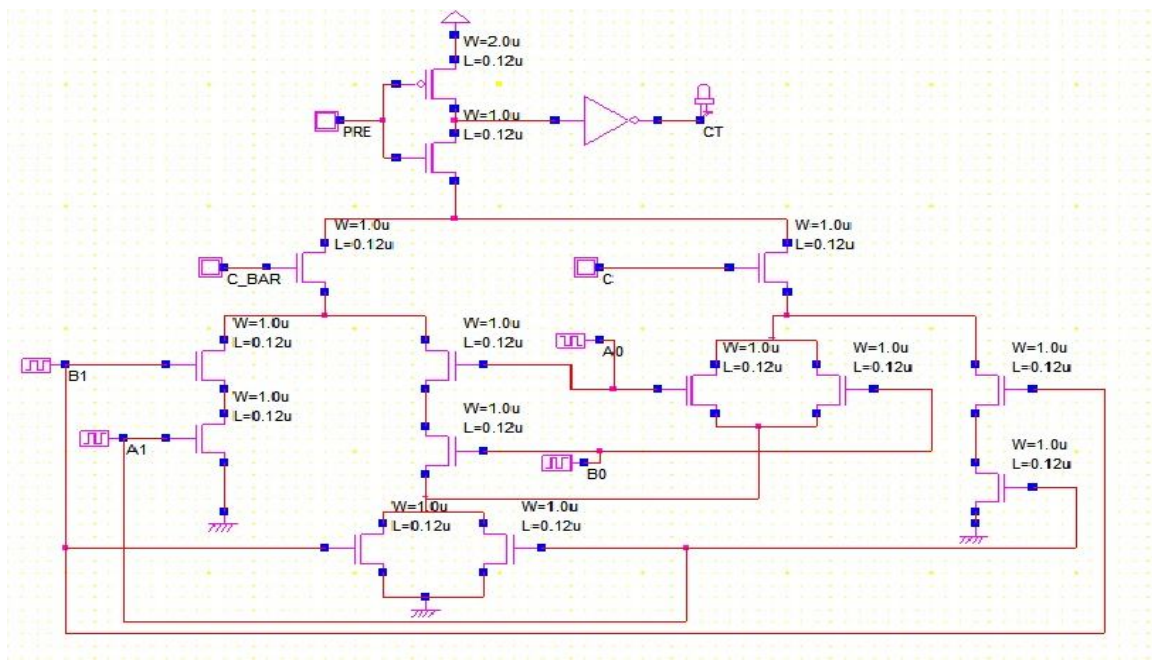


Figure 3.6: Circuit for cout. t.

The proposed design is much more area efficient compared to the existing MEDB adder due to the reduced number of transistors required for its implementation. As area is an important concern we also present comparative results on the Power-Delay-Area Product (PDAP) in the result section. Another point to note in Asynchronous BDD adder there is no static short circuit path due to the use of BDD, while such static short-circuit path can be found in DBPTL design when $b0.t=1$ and $b1.f=0$ as shown in Figure 3.10.

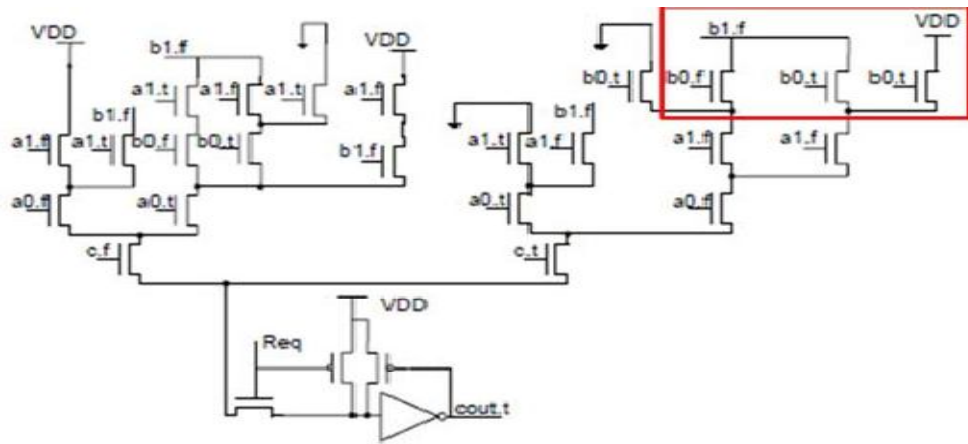


Figure 3.7: DBPTL adder.

The designed 8-bit adder by cascading 2-bit adder's with DONE output is as shown in Figure 3.8.

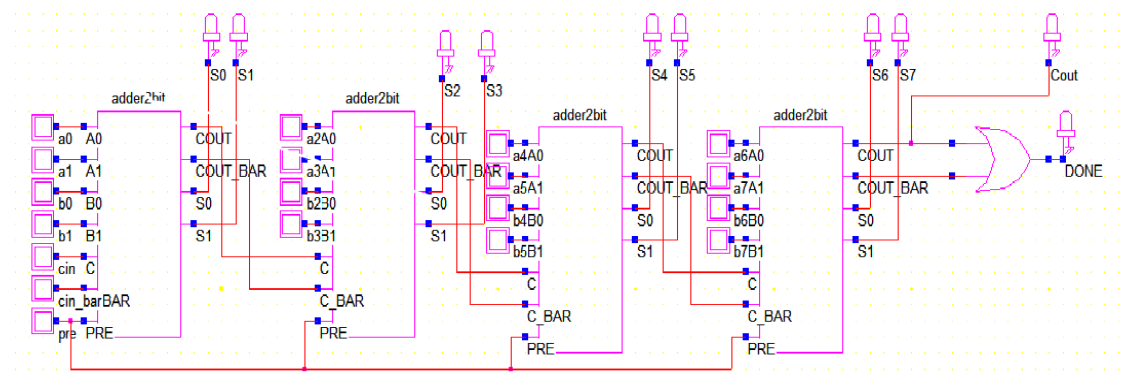


Figure 3.8: 8-bit adder.

Result

The hybrid BDD-based design for a 8-bit asynchronous adder is simulated in Dsch tool. It has a total of 268 transistors whereas this number is 376 in the case MEDB adder (considering the number of transistors in the logic block only). Hence the

reduction in the number of transistors in BDD-based design. Pre-charge signal is driven through an inverter chain so as to drive large loads. Another point to note in Asynchronous BDD adder there is no static short circuit path due to the use of BDD, while such static short-circuit path can be found in DBPTL.

For $a=\{1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\}$ $b=\{0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\}$ & $c_{in}=1$, the output S is found to be $S=\{0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\}$ & $C_{out}=1$. The output waveform is as shown in Figure 4.1.

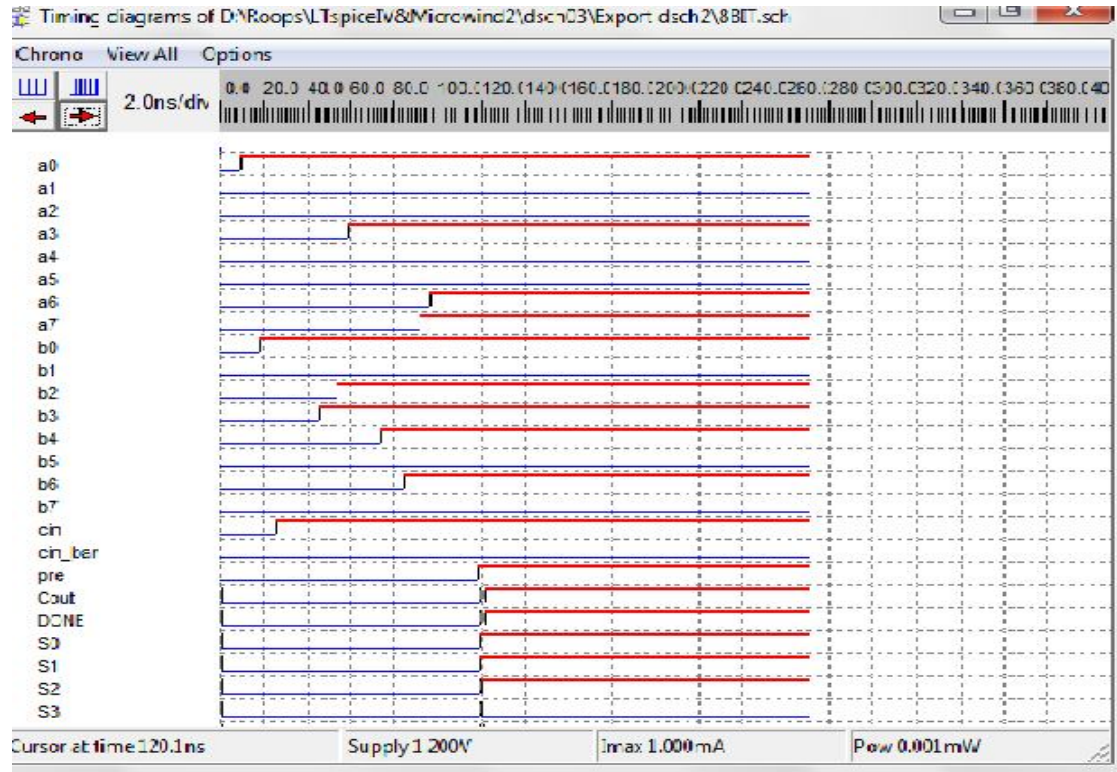


Figure 4.1: output waveform of BDD adder.

Conclusion

An area-power efficient asynchronous adder that employs a hybrid combination of Domino PTL and CMOS logic styles is simulated using Cadence tool. The adder combines merits of both the logic styles and is found to have comparatively better performance compared to the existing asynchronous adders without any sacrifice in the delay.

It may be noted that full BDD based implementation is not used in the design of the asynchronous adder while it is optimized for area and power because the carry circuit being AND-OR based is not found to be efficient when mapped to a BDD.

References

- [1] Mi Lu, "Arithmetic and Logic in Computer Systems", A John Wiley & Sons, Inc.; Publication 2004.
- [2] D. Johnson, V. Akella; "Design and Analysis of Asynchronous Adders", IEEE proc .on Com.& Dig. Tech; Vol. 145. Pp 1-8, 1998.
- [3] Chaudhry R., Liu T.-H., Aziz A.; "Area-Oriented Synthesis for Pass-Transistor Logic", Proceedings, ICCD-1998, pp 160-167.
- [4] J. M. Rabaey, A. Chandrakasan and B. Nikolic,; "Digital Integrated Circuit Design", a Design Perspective, 2nd Edition. Prentice Hall.
- [5] J. Martin; "Asynchronous Data Paths and the Design of an Asynchronous Adders", Formal Methods in System Design, 1(1), 119-137, July 1992.
- [6] L. S. Nielsen, "Low Power Asynchronous VLSI Design", Ph.D. Thesis, Dept. of Information technology, technical university of Denmark,1997.
- [7] K. S. Chong, B. H. Gwee and J. S. Chang "Low-Voltage Asynchronous adders for low power and high speed applications", Int. Symp. Circuits and Systems, 2002, page(s): I-873 - I-876 vol.1.
- [8] Obridko and R. Ginosar, "Minimal Energy Asynchronous Dynamic adders", IEEE Trans. On VLSI 2006, Volume 14, Issue 9, Pages: 1043–1047.

