IMPLEMENTATION OF SUBTHRESHOLD ADIABATIC LOGIC FOR ULTRA LOW-POWER APPLICATION

[1]M. Priyanka,[2]P. Chandrasekar

[1]PG Scholar,[2]Professor

[1,2]Department of ECE

[1,2]RVS College of Engineering and Technology,

[1]Email-priya15riya@gmail.com,[2]Email- chasek_2002@yahoo.co.in

Abstract- Behavior of adiabatic logic circuits in weak inversion or sub threshold regime is analyzed in depth for the first time in the literature to make great improvement in ultralow power circuit design. This novel approach is efficacious in low-speed operations where power consumption and longevity are the pivotal concerns instead of performance. The schematic and layout of a 4-bit carry look ahead adder (CLA) has been implemented to show the workability of the proposed logic. The effect of temperature and process parameter variations on sub threshold adiabatic logic-based 4-bit CLA has also been addressed separately. Post layout simulations show that sub threshold adiabatic units can save significant energy compared with a logically equivalent static CMOS implementation.

Introduction

The demand for implementing ultralow-power digital systems in many modern applications, such as mobile systems, sensor networks, and implanted biomedical systems, has increased the importance of designing logic circuits in sub threshold regime. These emerging applications have low energy as the primary concern instead of performance, with the eventual goal of harvesting energy from the environment. In sub threshold logic circuits operate with a supply voltage VDD lower than the transistor threshold voltage VT and utilize the sub threshold leakage current as the operating current. Conventional CMOS logic circuits utilizing sub threshold transistors can typically operate with a very low power consumption, which is mainly due to the dynamic (switching) power consumption and is quadratically dependent upon the supply voltage as CL f V2 DD (where CL, f, and VDD are the load capacitance, operating frequency, and the supply voltage, respectively).Recently, adiabatic logic (or energy recovery logic) style has emerged as a promising approach in strong inversion regime, to reduce dynamic power consumption significantly without sacrificing noise immunity and driving ability. These circuits achieve ultralow energy consumption by steering currents across devices with low voltage differences and by gradually recycling the energy stored in their capacitive loads, especially in low-frequency regime. Since the performance requirements are quite relaxed in many of these energy efficient subs threshold applications we believe that the adiabatic style can be used efficaciously in a sub threshold regime to make the circuit more energy efficient. To the best of our knowledge, no paper emphasizes the application of adiabatic logic in weak inversion regime for advanced technology node such as 22 nm. Therefore, the attempt to realize the sub threshold adiabatic logic (SAL) concept is a new endeavor. In general, the design of SAL requires a deep knowledge of the main features of the adopted logic style, such as power dissipation, leakage current, and impact of temperature variation, operating frequency, and noise immunity. In this paper, the behaviors of adiabatic logic in sub threshold regime are discussed in depth. To demonstrate the workability of the adiabatic logic circuits in sub threshold regime, a 4-bit carry look ahead adder (CLA) unit is adopted as a reference circuit. Analysis of energy dissipation along with the sensitivity of energy dissipation on supply voltage and temperature variations is also discussed in depth. Moreover, the analytical expression of optimum frequency and supply voltage under minimum energy condition has been verified through simulation in 22-nm technology. Extensive experiments are carried out using CADENCE SPICE Spectra to ensure the high energy efficiency and design feasibility of the proposed logic in weak inversion regime compared with other conventional CMOS logic.

2. Existing System:

A CMOS transistor (or device) has four terminals: gate, source, drain, and a fourth terminal that we shall ignore until the next section. A CMOS transistor is a switch. The switch must be conducting or on to allow current to flow between the source and drain terminals (using open and closed for switches is confusing-for the same reason we say a tap is on and not that it is closed). The transistor source and drain terminals are equivalent as far as digital signals are concerned-we do not worry about labeling an electrical switch with two terminals

2.1 Existing Technique:

Normal CMOS Logic

2.2 Technique explanation:

- If Vin is down (0 volts), NFET is OFF and PFET is ON pulling V_{out} to V_{dd} (high = 1)
- If Vin is up (at V_{dd}), NFET is ON hard and PFET is OFF pulling V_{out} low to Gnd (“0”)

3. Proposed System:

In adiabatic logic circuits, ramp type supply voltage is used to slow down the charge transport mechanism. Hence, the supply clock plays the pivotal role. A ramp type supply voltage φ(t) is considered in Fig. 1(a), which gradually swings in between logic 0 (Gnd potential) and logic 1 (VDD) in time duration 2T, where f (=1/2T) is the supply clock's frequency. The power supply waveform φ(t) can be divided into charging phase, when φ(t) ramps up from 0 to VDD in 0 to T unit time and discharging phase when φ(t) ramps down from VDD to 0 in T to 2T unit time.

3.1 Proposed Technique:

SAL Adiabatic Logic

3.2 Technique explanation:

- The design as in SAL basic logic gates have been implemented using either the pull-up or the pull-down transistors.

Therefore, the SAL-based CLA is also area efficient in comparison with the conventional structure.

4. Literature survey:

1) A 0.25 V 460 nW Asynchronous Neural Signal Processor With Inherent Leakage Suppression

Further power and energy reductions via technology and voltage scaling have become extremely difficult due to leakage and variability issues. In this paper, we present a robust and energy-efficient computation architecture exploiting an asynchronous timing strategy to dynamically minimize leakage and to self-adapt to process variations and different operating conditions. Based on a logic topology with built-in leakage suppression, the prototype asynchronous neural signal processor demonstrates robust sub-threshold operation down to 0.25 V, while consuming only 460 nW in 0.03 in a 65 nm CMOS technology. These results represent a 4.4 reduction in power, a 3.7 in energy and a 2.2 reduction in power density, when compared to the state-of-the-art processors.

As the supply voltage is scaled near or below the device threshold, dramatic increases in leakage and variability severely limit digital processor performances. In this paper, we present robust and energy-efficient computation architecture by employing an asynchronous self-timed design methodology. The proposed strategy allows for an adaptive adjustment to latency variations, and supports for an inherent leakage minimization under process variations and changing operating conditions, all of which are major issues in scaling regimes that favor major reduction in supply voltages. Circuit techniques specifically for leakage minimization are aggressively employed at both the logic and system levels. The prototype asynchronous neural signal processor demonstrates robust operation down to 0.25 V while consuming only 460 nW. Compared to the traditional synchronous approach, the asynchronous design shows a 2.3 reduction in power. Moreover, the self-timed operation alleviates the impact of variations on processor performance. Therefore, the asynchronous design exhibits a better statistical characteristic of power performance than the synchronous counterpart. These results demonstrate that in addition to soliciting better transistors and fabrication technology, leakage and variability issues can be tackled at the circuit and system levels with novel timing schemes and circuit innovations.

2) Design Techniques and Architectures for Low-Leakage SRAMs

In high performance Systems-on-Chip, leakage power consumption has become comparable to the dynamic component, and its relevance increases as technology scales. These trends are even more evident for memory devices, for two main reasons. First, memories have historically been designed with performance as the primary figure of merit; therefore, they are intrinsically non power-efficient structures. Second, memories are accessed in small chunks, thus leaving the vast majority of the memory cells unaccessed for a large fraction of the time. In this paper, we present an overview of the techniques proposed both in the academic and in the industrial domain for minimizing leakage power, and in particular, the sub threshold component, in SRAMs. The surveyed solutions range from cell-level techniques to architectural solutions suitable to system-level design. We can observe a couple of facts that allow a few suggestions that can have general value.

a) Exploit Orthogonality of Strategies: Although some techniques (e.g., bitline and word line design) have a moderate impact in absolute terms, they are orthogonal to techniques that are based on DPM. The same consideration applies to the customized design of the bitcell. Therefore, whenever the re-design of the internals of the SRAM architecture is allowed, designer should try to apply such techniques to decrease the leakage cost of basic memory operations (bitline/word line access and reads/writes).

b) Technology Matching: Although many techniques do not scale nicely with technology and/or scaling, designers should try to match the various techniques with the target technology. A technique may become less relevant in future technologies but might be the most suitable for the current ones. A good example is body-biasing. Although it is expected to become less efficient for nodes beyond the 32 nm, it represents a good solution for 65 nm or 90 nm nodes. Therefore, especially in the embedded domain where technology scaling is limited by the integration of other types of technology (e.g., embedded FLASH memories) and mixed technologies on the same chip are not uncommon body-bias still remains an efficient knob to control leakage.

3) Robust Sub threshold Logic for Ultra-Low Power Operation

Digital sub threshold logic circuits can be used for applications in the ultra-low power end of the design spectrum, where performance is of secondary importance. In this paper, we propose two different sub threshold logic families:

1) Variable threshold voltage sub threshold CMOS (VT-Sub-CMOS) and

2) sub threshold dynamic threshold voltage MOS (Sub-DTMOS) logic. Both logic families have comparable power consumption as regular sub threshold CMOS logic (which is up to six orders of magnitude lower than that of normal strong inversion circuit) with superior robustness and tolerance to process and temperature variations than that of regular sub threshold CMOS logic.

In this paper, we discussed two novel sub threshold logic families. A new control circuit for the stabilization of sub threshold circuit is also discussed in detail. Both VT-sub-CMOS and sub-DTMOS logic families show superior robustness and tolerance to temperature and process variations than that of regular sub threshold CMOS logic. VT-sub-CMOS logic can be readily implemented in twin-well process technology, but it requires additional circuitry for stabilization. The additional increase in area and process complexities for sub-DTMOS logic is compensated by its higher operating frequency while maintaining comparable energy/switching as regular sub threshold CMOS logic. DTMOS has been successfully implemented in both SOI and bulk Silicon. VT-sub-CMOS logic, however, has better control on substrate bias.

4) Analysis and Design of an Efficient Irreversible Energy Recovery Logic in 0.18-um CMOS

This paper presents the design and experimental evaluation of a new type of irreversible energy recovery logic (ERL) families called complementary energy path adiabatic logic (CEPAL). It inherits the advantages of quasi-static ERL (QSERL) family, but is with improved driving ability and circuit robustness. The proposed logic style features no hold phase compared to its QSERL counterpart under the same operation conditions; thereupon no feedback keeper is required so that considerable improvements in area and power overheads can be achieved. Moreover, its throughput becomes twice as high as that of QSERL when their frequencies of power clocks (PCs) are identical. Results on the impact of variation on CEPAL are provided. Comparison between CEPAL and other known low-power logic style achieving iso-performance, namely, sub threshold logic is also given. In order to demonstrate workability of the newly developed circuit, an 8-bit shift register, designed in the proposed techniques, has been fabricated in a TSMC 0.18- m CMOS process. Both simulation and measurement results verify the functionality of such logic, making it suitable for implementing energy-aware and performance- efficient very-large scale integration (VLSI) circuitry.

Adiabatic techniques have been effective means to power minimization in deep submicron VLSI systems. In this paper, we discussed a newly developed ERL family termed CEPAL for low-power design. The proposed logic style outperforms those currently demonstrated in irreversible energy recovery literature in terms of several aspects. In addition to the summary of prior works, we analyzed CEPAL in detail, and elaborated relative strength and weaknesses of it vis-a-vis one of the known sub threshold logic styles. Specifically, we presented the efficiency of the DFFs made up of QSERL and the proposed logic style. Since the impact of leakage on CEPAL is of trifling importance, low- devices can be introduced so as to minimize the no adiabatic loss, enabling higher circuit performance. Such low-devices have been available in the processes 0.25 m and beyond.

5) Energy-Efficient GHz-Class Charge-Recovery Logic

In this paper, we present Boost Logic, a charge recovery circuit family that can operate efficiently at clock frequencies in excess of 1 GHz. To achieve high energy efficiency, Boost Logic relies on a combination of aggressive voltage scaling, gate overdrive, and charge-recovery techniques. In post-layout simulations of 16-bit multipliers with a 0.13- m CMOS process at 1 GHz, a Boost Logic implementation achieves 5 times higher energy efficiency than its minimum-energy pipelined, voltage-scaled, static CMOS counterpart at the expense of 3 times longer latency. In a fully integrated test chip implemented using a 0.13- m bulk silicon process and on-chip inductors, chains of Boost Logic gates operate at clock frequencies up to 1.3 GHz with a 1.5-V supply. When resonating at 850 MHz with a 1.2-V supply, the Boost Logic test chip achieves 60% charge-recovery.

In this paper, we have presented Boost Logic, a charge recovery logic family which is capable of efficient operation at GHz-class clock frequencies. This efficient operation is achieved through the combined use of aggressive voltage-scaling, gate-overdrive, and charge-recovery techniques. In post-layout simulations of 16-bit carry-save multipliers at 1 GHz, the Boost Logic implementation achieves energy savings in excess of 80% compared to its minimum-energy voltage scaled static CMOS counterpart at the expense of a threefold increase in computational latency. Considerable performance benefits can be achieved from the use of low devices in the evaluation tree of Boost Logic gates. The use of low threshold devices enables the logic depth of a gate to be approximately doubled, resulting in designs with lower latency and energy dissipation than regular threshold devices as confirmed through the implementation of the low 16-bit Boost multiplier. We have demonstrated the correct operation of a Boost Logic prototype chip with on-chip inductors in a 0.13- m bulk silicon process. Our measurements show that approximately 60% charge-recovery was achieved at the resonant frequency of 850 MHz.

5. Carry Look - Ahead Adder (CLA):

We know that Ci+1 is dependent on previous carry C_i as follow relation:

$C_i+1 = A_i B_i + A_i C_i + B_i C_i$

which can be written as $C_i+1 = G_i + P_i C_i$

G_i→ carry generate function P_i→ carry propagate function we have $A_i = 1$ or $B_i = 1$. Using these G_i and P_i we can get following equations:

C2=G1+P1*C1

C3= G2+P2*C2
=G2+P2*(G1+P1*C1) =G2+G1*P2 + P1*P2*C1

C4 = G3+P3*C3 = G3 + P3*(G2+G1*P2 + P1*P2*C1) = G3+G2*P3 + G1*P2*P3 + C1*P1*P2*P3

C5 = G4+P4*C4 = G4 + P4*(G3+G2*P3 + G1*P2*P3 + C1*P1*P2*P3)

= G4 + G3*P4 + G2*P3*P4 + G1*P2*P3*P4 + C1*P1*P2*P3*P4

These equations suggest that C2, C3, C4, C5 can be calculated from C1 directly. Hence it is called carry look ahead adder. This is a 4 stage circuit.

We have AND gates at level 1 and OR gate at level 2 in the circuit. Also fan-in of the OR gate in level 2 & that of AND gate is 5 and we have a maximum fan-in of about 8 So we can't extend this circuit to higher stage carry look ahead but can use this 4-stage circuit in cascaded form.

In the following diagram we have cascaded two 4-stage circuits to make it for 8 bit adder.

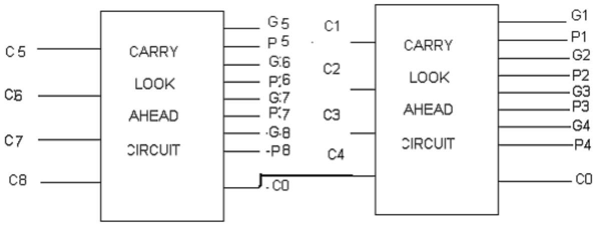


Figure 1: Block Diagram Of Carry Look Ahead Adder

For a 16-bit adder we need total delay= Δ + 2Δ + 2Δ + 2Δ + 2Δ + 3Δ = 12Δ which is also illustrated below:

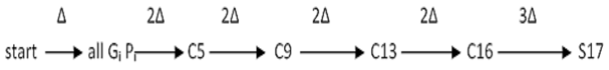


Figure 2: 16 Bit Adder

So we see that we have been able to reduce the delay for a 16 bit adder from 33Δ to 12Δ which is lesser by a factor of about 3 times.

6. Half Adder:

The half adder takes two single bit binary numbers and produces a sum and a carry–out, called "carry". We denote the sum A + B. A combinational circuit that performs the addition of two bits is called half adder while the circuit which adds 3 bits is called Full adder. The following table shows the result of different combinations of inputs:

a	b	S(sum)	C(carry)
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

Table 1: Combinations of inputs of Half Adder.

Now we can see from the table that carry is one only when both inputs are 1 while sum is 1 when only one of the two is 1 like a XOR gate.

S= a'b + ab'

C= ab

Or we can get the equations from the K-map also which are discussed on the next page K-map for the variable Sum is

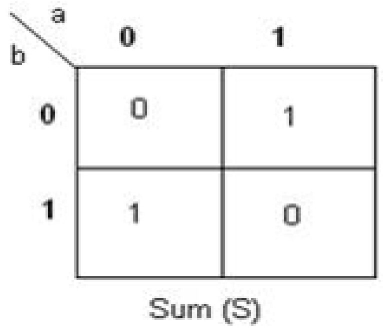


Figure 3: K-map for variable sum

The equation we get for sum is S=a´b+ab´

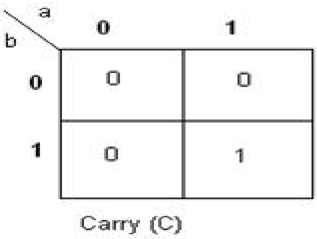


Figure 4: K-map for the variable Carry

The equation we get for carry is ab

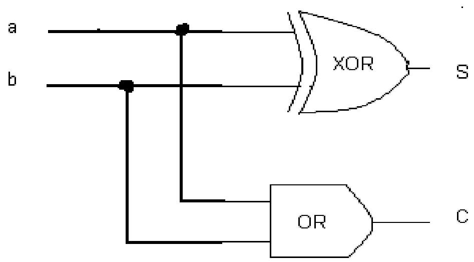


Figure 5: Half Adder

7. Full Adder:

A full adder adds binary numbers and accounts for values carried in as well as out. The full adder has 3 inputs and 2 outputs. The first 2 inputs are the 2 bits a & b to add while the 3rd input c is the carry from the previous significant bit while the outputs are the same: sum S and the carry C. The following table shows the result of different combinations of inputs:

a	b	C	S(sum)	C(carry)
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Table 2: Combinations of Input of Full Adder

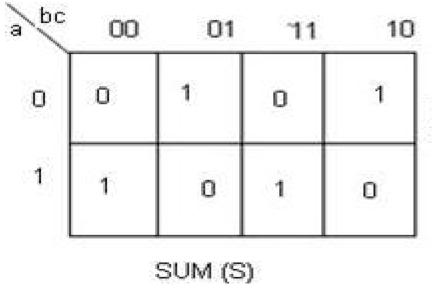


Figure 6: K-map for the output variable SUM

The equation we get for variable SUM S is S = ab´c´ + a´b´c´ + abc

This circuit is a level 3 circuit as we also need inverters at level 1, then we have 4 3-input AND gates at level 2 and 4-input OR gate at level 3.

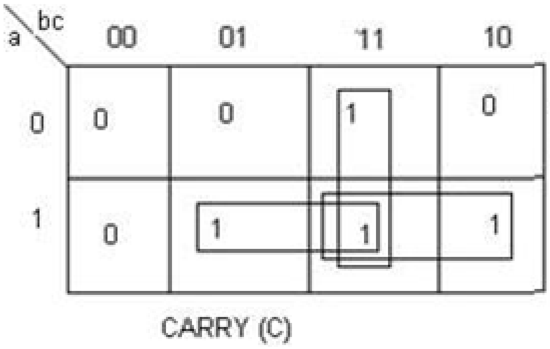


Figure 7: K-map for the variable carry

The equation we get for variable CARRY C is C = ab + ac + bc

Let’s now put the equations in different form:

S = ab’c’ + a’b’c + a’bc’ + abc = Σ (1,2,4,7)

= b’ (ac’ + a’c) + b (a’c’ + ac) = b’ (ac’ + a’c) + b (ac’ + a’c)’

=b’ (a xor c) + b (a xor c)’{We know (ac’ + a’c)’ = a’c’ + ac and a’c + ac’ = a xor c}

= b’z + z’b= b xor z

= b xor a xor c

S = a xor b xor c

and C = ab + ac + bc = ab(c + c’) + ac (b + b’) + bc (a + a’) = abc + abc’ + abc + ab’c + abc + a’bc = abc +a’bc+ab’c+abc’= Σ(3,5,6,7)

So we can draw the circuits using XOR, NOT, AND & OR gates.

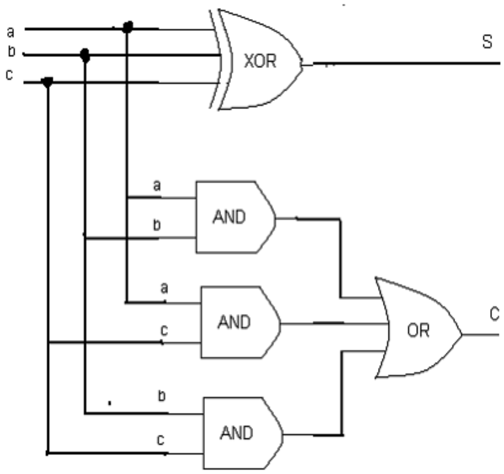


Figure 8: Full Adder

8. Full Adder using Half Adders:

We can implement the Full Adder using 2 half adders and one OR gate as follow:

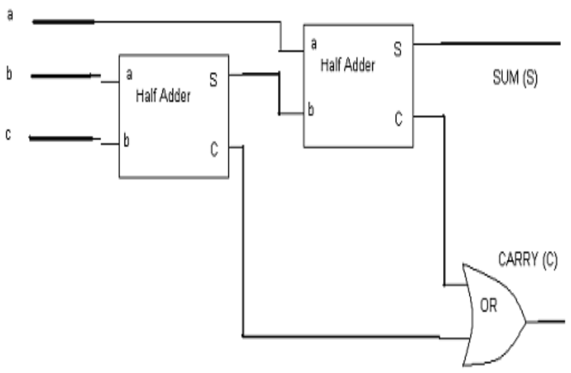


Figure 9: Circuit with half adder and OR gate.

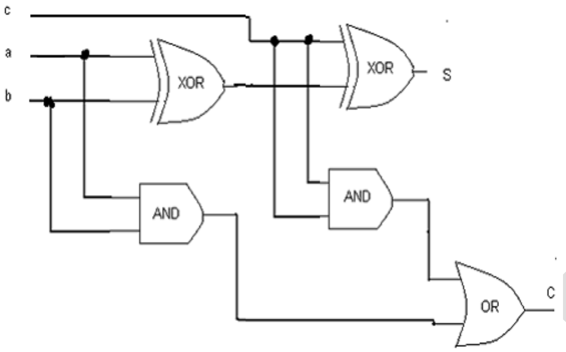


Figure 10: Full Adder Using Half Adder Circuit.

9. Sub threshold Adiabatic Logic-Based 4-Bit CLA:

Design and analysis of SAL-based 4-bit CLA are given to show the workability and the feasibility of the proposed logics. When A = B = 1 through parallel pMOS transistor, leakage currents will flow as the transistors will behave almost as a constant current source. A very small amount of charge will be stored across the load capacitor, i.e., instead of ground potential, very small voltage will be dropped across the output. Like the conventional approach, the expression of the ith sum and the (i+1)th carry output can be given as

$$S_i = A_i \oplus B_i \oplus C_i$$
$$C_{i+1} = A_i B_i + (A_i \oplus B_i) C_i$$

According to the synthesized gate level block, the SAL gate level structure of 4-bit CLA has been implemented using Virtuoso(R) Schematic Composer.

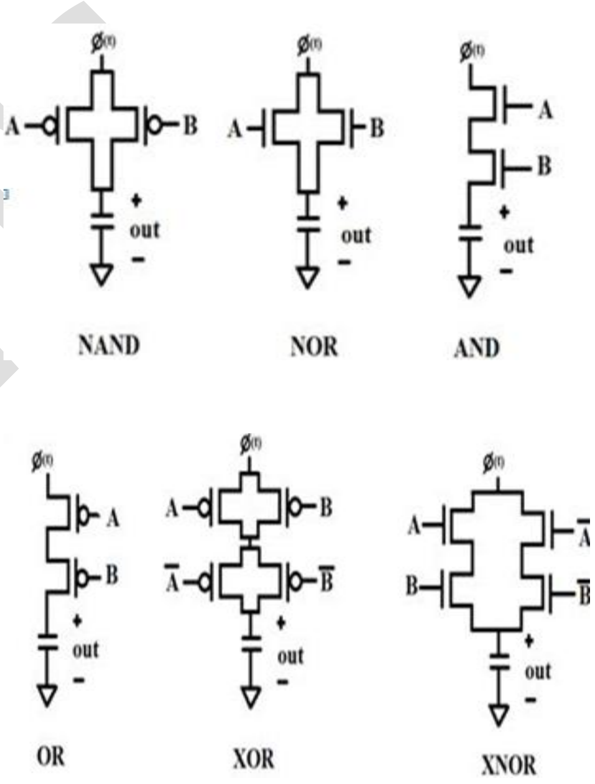


Figure 11: Logical structure of basic SAL logic gates.

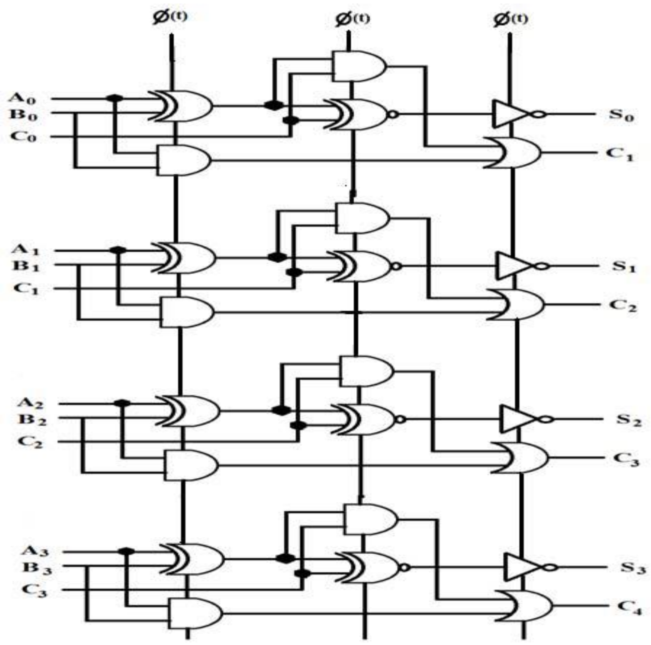


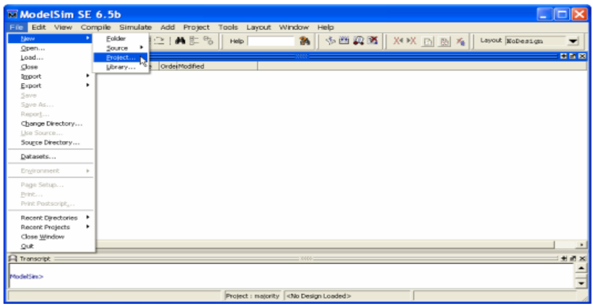
Figure 12: Logic structure of 4-bit CLA

10. Software requirement:

Modelsim

Modelsim is a powerful simulator that can be used to simulate the behavior and performance of logic circuits.

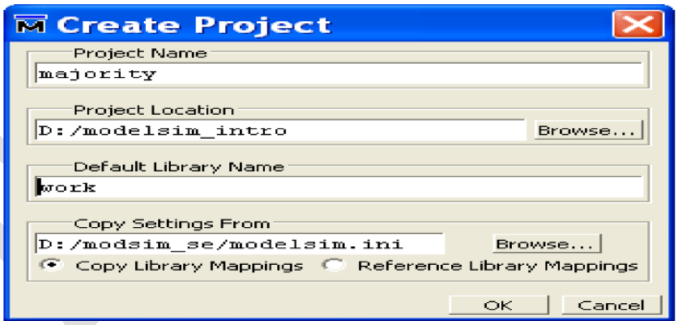
Open the Modelsim simulator. In the displayed window select File > New > Project



Project : majority <No Design Loaded>

Figure 13: Modelsim Simulator Window

A Create Project pop-up box will appear, as illustrated. Specify the name of the project; we chose the name majority. Use the Browse button in the Project Location box to specify the location of the directory that you created for the project. Modelsim uses a working library to contain the information on the design in progress; in the Default Library Name field we used the name work. Click OK.



Create Project

Project Name: majority

Project Location: D:/modelsim_intro

Default Library Name: work

Copy Settings From: D:/modsim_se/modelsim.ini

● Copy Library Mappings ○ Reference Library Mappings

Figure 14: Pop-up Box To Create Project

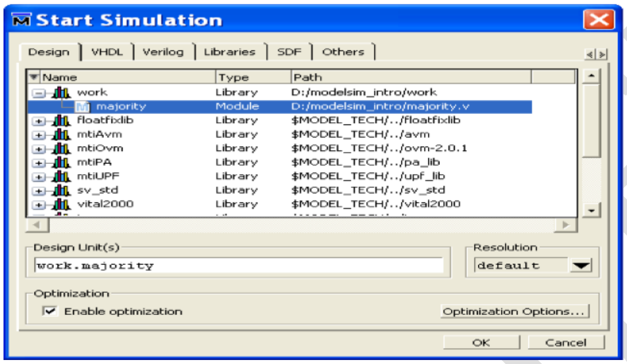
Click on Add Existing File and add the file majority. v to the Then close the windows.

At this point, the main Modelsim window will include the file as indicated. Observe that there is a question mark in the Status column. Now, select Compile È Compile All, which leads to the window. Indicating in the Transcript window (at the bottom) that the circuit in the majority. v file was successfully compiled. Note that this is also indicated by a check mark in the Status column. The circuit is now ready for simulation.

11. Creating waveforms for simulation:

To perform simulation of the designed circuit, it is necessary to enter the simulation mode by selecting simulate > start simulation. Expand the work directory and select the design called majority, as shown in the figure. Then click ok. Now, an objects window appears in the main Modelsim window.

It shows the input and output signals of the designed circuit to simulate the circuit we must first specify the values of input signals, which can be done by drawing the input waveforms using the graphical waveform editor.



Name	Type	Path
work	Library	D:/modelsim_intro/work
majority	Module	D:/modelsim_intro/majority.v
floatfixlib	Library	$MODEL_TECH/../floatfixlib
mtiAvm	Library	$MODEL_TECH/../avm
mtiOvm	Library	$MODEL_TECH/../ovm-2.0.1
mtiPA	Library	$MODEL_TECH/../pa_lib
mtiUPF	Library	$MODEL_TECH/../upf_lib
sv_std	Library	$MODEL_TECH/../sv_std
vital2000	Library	$MODEL_TECH/../vital2000

Design Unit(s): work.majority

Resolution: default

Optimization

☑ Enable optimization

Figure 15: Object window

Select view ç wave which will open the wave window depicted .the wave window may appear as a part of the main Modelsim window; in this case undock it by clicking on the dock/undock icon in the top right corner of the window and resize it to a suitable size. if the wave window does not appear after undocking, then select view è wave in the main Modelsim window. For our simple circuit, we can do a complete simulation by applying all eight possible valuations of the input signals x1, x2 and x3. The output f should then display the logic values defined by the truth table for the majority function. We will first draw the waveform for the x1 input. in the objects window, right-click on x1. Then, choose create wave in the drop-down box that appears.

Conclusion:

SAL has been presented in this paper for the first time in the literature to advance the ultralow power research. A closed form expression of the energy dissipation has been derived, from which insight is gained into the dependence of energy dissipation on design and process parameters. SAL saves considerable energy compared with the static conventional logic counterpart over a wide range of frequency. In particular, the impact of temperature variation on leakage dissipation, output swing, etc., has been discussed thoroughly in this paper. Hence, the predicted values of optimum frequency and optimum supply voltage almost match the simulated ones. Post layout simulations using CADENCE SPICE Spectra and the comparison with the static counterpart explain the workability of SAL. This proposed logic scheme can be used in future energy-saving embedded circuits and mainly for energy efficient devices where ultralow power and longevity are the pivotal issues.

References:

[1] J. Kwong, Y. K. Ramadass, N. Verma, and A. P. Chandrakasan, “A 65 nm sub-Vt microcontroller

with integrated SRAM and switched capacitor DC-DC converter," IEEE J. Solid-State Circuits, vol. 44, no. 1, pp. 115–126, Jan. 2009.

[2] T.-T. Liu and J. M. Rabaey, "A 0.25 V 460 nW asynchronous neural signal processor with inherent leakage suppression," IEEE J. Solid-State Circuits, vol. 48, no. 4, pp. 897–906, Apr. 2013.

[3] A. Calimera, A. Macii, E. Macii, and M. Poncino, "Design techniques and architectures for low-leakage SRAMs," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 9, pp. 1992–2007, Sep. 2012.

[4] H. Soeleman, K. Roy, and B. C. Paul, "Robust sub threshold logic for ultra-low power operation," IEEE Trans. Very Large Scale Integer. (VLSI) Syst., vol. 9, no. 1, pp. 90–99, Feb. 2001

[5] C.-S. A. Gong, M.-T. Shiue, C.-T. Hong and K.-W. Yao, "Analysis and design of an efficient irreversible energy recovery logic in 0.18-μm CMOS," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 9, pp. 2595–2607, Oct. 2008.

[6] D. Maksimovic, V. G. Oklobdzija, B. Nikolic, and K. W. Current, "Clocked CMOS adiabatic logic with integrated single-phase power clock supply," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 8, no. 4, pp. 460–463, Aug. 2000.

[7] V. S. Sathe, J.-Y. Chueh, and M. C. Papaefthymiou, "Energy-efficient GHz-class charge-recovery logic," IEEE J. Solid-State Circuits, vol. 42, no. 1, pp. 38–47, Jan. 2007.

[8] N. S. S. Reddy, M. Satyam, and K. L. Kishore, "Cascadable adiabatic logic circuits for low-power applications," IET Circuits, Devices Syst., vol. 2, no. 6, pp. 518–526, Dec. 2008.

[9] Y. Tsividis, Operation and Modeling of the Transistor MOS, 2nd ed. Oxford, U.K.: Oxford Univ. Press, 2003.

[10] X. Xi et al., "BSIM4.3.0 MOSFET Model," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, Tech. Rep. 94720, 2003.

[11] Y. Cao, Predictive Technology Model for Robust Nanoelectronic Design. New York, NY, USA: Springer-Verlag.