University of Cincinnati

Date: 4/14/2015

I, Phani Kameswara Abhishikth Varanasi , hereby submit this original work as part of the requirements for the degree of Master of Science in Computer Engineering.

It is entitled:

Study of Ultra Low Power Design and Power Reduction Techniques for VLSI Circuits at Ultra Low Voltages

Student's name: **Phani Kameswara Abhishikth Varanasi**

This work and its defense approved by:

Committee chair: Wen-Ben Jone, Ph.D.

Committee member: Ranganadha Vemuri, Ph.D.

Committee member: Philip Wilsey, Ph.D.

16163

Study of Ultra Low Power Design and Power Reduction Techniques for VLSI Circuits at Ultra Low Voltages

A thesis submitted to the

Division of Research and Advanced Studies
of the University of Cincinnati

in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in the School of Electronic and Computing Systems
of the College of Engineering and Applied Sciences

April, 2015

by

Phani Kameswara Abhishikth Varanasi
B.E (EEE), M. V. S. R. Engineering College, 2011

Thesis Advisor and Committee Chair: Dr. Wen Ben Jone

Abstract

The advancements and scaling in technology are continuously increasing in accordance with Moore's Law. This results in an increase in the performance of chips, but comes with a price due to the increased power consumption, and hence resources are spent on cooling, packaging and other methods to reduce the after effects. This additional cost has to be eliminated, and the most obvious solution is to reduce the power consumption of a design which would also protect the chips from permanent failure due to additional heat in the chips.

Various power reduction methods including supply voltage scaling, dynamic voltage and frequency scaling, multi voltage design, clock gating for dynamic power reduction, and multi-Vth technique, power gating for leakage power reduction have been proposed. The main aim of our research was to reduce the supply voltage which has a quadruple effect on reducing the power consumption, and hence operate the designs in or as close to the subthreshold region of operation as possible. This kind of ultra low power designs are especially useful in biomedical applications. *Carry skip adder* and *magnitude comparator* designs are considered for our research due to the extensive use of such designs in almost all arithmetic applications. Simulations are performed at 45nm CMOS technology and at very low voltages, (e.g., 0.4V) to check the functionality first, followed by the application of some of the most widely used power reduction techniques in the industry, including clock gating and power gating, to test their effectiveness at such low voltages. Error detection sequential circuits were also employed to check if they can further reduce the power consumption and improve the performance of the designs with ultra low supply voltages. The results obtained give interesting insights into the effectiveness of various power reduction techniques at ultra low voltages.

Acknowledgements

I would like to thank my academic advisor, Dr. Wen Ben Jone, for his extremely helpful and enthusiastic nature which paved way for the formation of our thesis. It was due to his constant encouragement and relentless attitude, coped with his words of wisdom, which helped me learn many new things. He was always supporting and never hesitated to clear my doubts or to discuss about the work, irrespective of the time, for which I am indebted to him. I would also like to thank Dr. Ranga Vemuri and Dr. Philip Wilsey for their valuable time to serve as my Masters' thesis committee members. I would also like to thank them for the courses they taught with such enthusiasm which helped me during various stages of my Masters program. I express my thanks to Rob Montjoy who helped me out whenever I had issues while working with the software.

I would like to greatly appreciate and thank my parents, who were always there for me when I needed, which motivated me to overcome tough situations. I cannot thank them enough for the sacrifices they have done for me. I would also like to immensely thank my brother, Suresh Kumar, for his constant support and advices right from my childhood till now, which helped me in all aspects of life. Special thanks to my cousin, Vishwanath Kotta for his constant support while in the US for which I am grateful to him.

I express my thanks to my roommates Nikhil, Naren, Ujwal, and friends, for all the good times we had, and for being there through thick and thin for making my Master's experience such an enjoyable one. Lastly, I would like to thank God, the almighty, for his love and for what I am right now.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Digital integrated circuits, each containing millions or billions of transistors fabricated on it, can function as quite a wide range of components which include memory, microprocessor or even complex design. The main feature of these ICs is to include more functionality on a relatively smaller area with a minimum weight by miniaturizing the electronic equipment [1]. The basic building blocks of integrated circuits are logic gates, which in turn contain transistors, and operate on binary data. The main advantage of integrated circuits is the cost associated with them being very low, which is bound to decrease further as more technological advances result in the generation of larger circuit functions on a single chip [1]. The ubiquitous Moore's Law pertaining to any VLSI circuit predicts that the number of transistors on an IC doubles approximately two years, and that has definitely been making a huge difference in the building of various VLSI designs. VLSI circuits are present in almost every application and hence with an increase in transistor density per a single chip, we have been empowered to accomplish more with the same available area.

1.1 Power Consumption Considerations

The number of transistors present on an integrated circuit is one of the most talked about factors by all VLSI engineers and the reasons are many, including but not limited to more area and power consumption, changing the density of an integrated chip and so on. The advantage of having more transistors on a single chip is to accomplish more functionality per the same available area. But, this has an offset associated with it too, in that, if the number of transistors increases, there is an increase in power density as more power is dissipated by the same functioning design. In order to deal with this increased power dissipation, cooling and packaging costs increase, which offsets the increased functionality. It can even lead to permanent failure in the chip, if the amount of heat produced becomes very high. As a result of this, there needs to be a balance between transistor density and power consumption.

Reducing the power consumption of a chip is an increasing area of importance for most of the circuits. One category of applications that focuses on reduced power consumption is one involving mobile/ portable communications and sensor systems [2]. The devices and equipment used especially in medical applications like pacemaker, military applications, security systems have lowering the power consumption as the main criterion as long as the performance is within acceptable limits. Devices like portable phones and handheld devices warrant the operational life to be increased, and this increase should not compromise on the operation or performance of such devices. Hence, designing chips for low power is one of the most important areas of interest in modern times, especially with the technological advancements and more components being embedded into the same chip is concerned. Whenever we are dealing with digital ICs, arithmetic circuits are present in almost all the designs, and hence an *adder* and a *magnitude comparator* are two such designs which will be present in any computing application, like in an ALU unit in a computer, microprocessors or any other device dealing with arithmetic operations, address translations etc. So, there is a need to reduce power consumed by these devices so that it presents a starting point to reduce overall power consumption of the designs. This has been studied in our thesis, by means of finding ways to reduce power consumption of adder and magnitude comparator circuits.

Technology scaling is continuing to progress at an alarming rate, and due to this scaling, performance of the designs keeps increasing at almost the same or even slightly lower costs. But, the problem with technology scaling is that as the feature size keeps reducing, it becomes increasingly difficult to fabricate designs, not only due to issues related to sensitivity to different types of variations, but also due to the increased leakage power consumption in such low feature sized devices.

Power in a CMOS circuit can be dynamic, which is mainly contributed due to the presence of capacitors in the design. These capacitors need to be charged and discharged based on the inputs applied, and there is a lot of switching activity associated with this which leads to power consumption. In higher feature sizes, the majority of power consumed is attributed to dynamic power. Another type of power, short circuit power, comes into picture when, momentarily, both NMOS and PMOS transistors involved in a designed can be partially turned on, due to slow rise times of input signals. During this situation, there is a small amount of current flowing directly from the power source to the ground, causing adverse affects to the chip if persistent for longer periods. The last type of power consumption is due to the current drawn by a circuit even when it is off or in idle period. This power is mainly contributed by subthreshold, gate, reverse-biased, and junction leakage currents. This power was not very significant when higher feature sized transistors were employed, but with technology scaling, this component of power has become comparable to dynamic power consumption, though short circuit power consumption is generally less compared to the other two types.

There have been a number of methods that are proposed to reduce the power consumption of circuit designs. Some of them are: dynamic voltage and frequency scaling, reducing the operating voltage of the circuit, using voltage islands and clock gating to reduce the switching/ active power consumption. The techniques to reduce static power consumption include power gating, multi-threshold transistor usage, biasing. Clock gating and power gating are some of the most widely used techniques in the VLSI industry, and are implemented in our designs to check their effectiveness at very low voltages.

1.2 Review of Clock Gating and Power Gating in Ultra Low Voltage Region

Leakage power consumption has to be reduced during standby periods, when no processing is occurring, especially in low voltage regions when more transistors cause more leakage power [3]. One of the popular methods of accomplishing power gating in the ultra low voltage region is a hardware controlled approach, in which individual functional units are made to sleep for short amounts of time. While performing power gating in this region, there is a trade off that exists between size of the footer, achievable leakage reduction and performance penalty that is incurred as a result of voltage drop across the footers as presented in [4].

Traditional CMOS circuit power gating has some disadvantages, one of which is the charge being stored on MOSFETs even during idle time, and the other is the overhead due to the switch from active to cut off mode of operation. A new style of power gating structure, Sense Amplifier based Pass Transistor Logic (SAPTL) was presented in [3] owing to its smaller footer size and boot-up capacitance requirement associated with it, and also because it can overcome the disadvantages posed by traditional CMOS gating structures. The work in [4] presents a detailed analysis by observing the behavior with and without cut off structures, and then three different cut off structures, including MTCMOS, DTCMOS were proposed to check leakage reduction that can be achieved. It also establishes that it is necessary to optimize both the footer width and also supply voltage (Vdd) to achieve the minimum leakage energy.

Finally, [5] presents a new power gating technique that can be used in ultra low voltage region to reduce leakage in sleep mode. It states that normal PG structures, such as the one using high-Vth scheme, does not work well in the ultra low power region due to long time to switch between different modes of operation causing voltage fluctuations and degrades frequency of operation. It considers different configurations such as a single low-Vth footer, and series-connected NMOS footers with low-Vth transistors. A new technique was also proposed based on the criticality of paths to show its effectiveness at the ultra low voltage region of operation.

The clock gating technique deals with shutting off the clock to some blocks of the circuit so that switching activity is reduced. This technique is especially useful when there are blocks used intermittently in a design. When the values latched through flip flops do not change values during the current clock period, there is no need to apply the clock to those cells, as the values are held until the next clock edge. [6] presents a novel clock gating cell optimized to use for low power and low voltage applications, and compares it to conventional clock gating cells. It consumes lower power compared to conventional cells and is advantageous. [7] presents a novel sequential selective clock gating method which is effective at ultra low voltages to maximize savings at such low voltages. Simulations were performed on several multiprocessor circuits and results were presented. Finally, [8] proposes different flip-flops that are configured to enable energy recovery from the clock network, resulting in significant energy savings. Most of these clock gating schemes were focused on slightly higher voltages, which presented a good opportunity for us to consider clock gating effects at lower voltages.

1.3 Subthreshold Region of Operation

The reduction of operating voltage is one such extremely useful technique as dynamic power is quadratically related to the supply voltage, thus providing a chance to save more power by reducing the voltage. The region in which a transistor operates depends on the supply voltage. When the supply voltage is reduced, the transistor operating mode shifts from strong inversion, to moderate and finally to weak inversion region. But, reducing the voltage too much has adverse effects as the delay associated with the designs increases quadratically as we reduce the voltage below threshold voltage, coupled with an increase of leakage power consumption. This is a problem with the weak inversion region, and is due to the increased sensitivity on PVT variations and also exponential dependence of delay and current on the threshold and supply voltages. Therefore, a balance has to be struck between reducing the operating voltage and performance requirements of the design. In particular, the threshold voltage is non-scalable and also the subthreshold slope presents a lot of limits, which has caused supply voltage scaling to slow down to maintain device performance without increasing leakage power too much [9]. Care has to be taken about the design aspects of circuits in this region of operation to avoid severe performance loss due to variations.

The subthreshold or weak inversion region of operation presents an interesting area of focus for low power applications, but the performance penalty is huge. Ultra low power design can be done in near-threshold region so that performance can be put in check and also operation can be done close to the minimum energy point, which occurs for CMOS logic families in the subthreshold region of operation [10]. Subthreshold digital logic design has grown in popularity ever since. This thesis aims to operate designs as close to this region as possible by focusing primarily on reducing power consumption when performance is within acceptable limits.

1.4 Thesis Organization

The main aim of our thesis was to operate the designs in very low voltage region, which would make them operate either in or as close as possible to the subthreshold region of operation. A very important and interesting question is related to the effectiveness of various power reduction techniques at such low voltages. This chapter gives a brief introduction to the work presented in this thesis, followed by the remaining chapters organized as follows.

Chapter 2 introduces various concepts and presents the background related to this research, including the basic design of circuits considered, power reduction techniques, and EDS design.

Chapter 3 extensively deals with selection of proper operating voltage and model file which would produce a balance between performance and power consumption. CSA and comparator designs are elaborated, followed by analysis of the worst case delay variation with the number of stages in a CSA. Finally, EDS design for CSA is presented toward the end. Also, various simulations performed and observations made are detailed.

Chapter 4 deals with the application of various power reduction techniques like power gating and clock gating for the designs under consideration, at very low voltages. The problems encountered with these methods, if any, are presented, and details about simulations and measurements performed, followed by observations, are given toward the end.

Chapter 5 presents conclusion to this thesis and provides any possible future work in this area.

Chapter 2

Background

This chapter discusses about background related to ultra low voltage region of operation and also covers topics related to our research work. The motivation for our work has already been presented in the previous chapter. Section 2.1 deals with aspects related to operating the designs in the ultra low voltage region. Section 2.2 presents topics related to various adder configurations with emphasis on carry skip adder which is used for our research work. Section 2.3 deals with design of a magnitude comparator which is also employed in our work, followed by power reduction techniques: power gating in Section 2.4 and clock gating in Section 2.5. This chapter is concluded by giving details about Error Detection Sequential (EDS) circuit in Section 2.6.

2.1 Ultra Low Voltage Design

This section deals with details about the ultra low voltage region of operation of CMOS circuits. Digital integrated circuits mostly use CMOS circuits as building blocks. The feature size of CMOS transistors is reducing day by day, and this coupled with increasing chip density where more circuitry is being fit into a smaller space, and higher operating frequencies, are a cause of concern as power consumption increases as a result of these factors. This may lead to even permanent failure of the chip due to increased temperature of the chip. Therefore, power consumption has to be minimized using different possible techniques, and one of such techniques is by operating the design in the subthreshold region of transistor operation. If speed or performance is not the major factor relating to a design (e.g. for biomedical applications), subthreshold operation provides a very good energy-saving approach to many energy constrained applications [11], where we reduce the supply voltage considerably without worrying too much about performance.

The minimum energy per operation point (MEP) in the case of static CMOS technologies is achieved in the subthreshold region of operation [10] [12]. A device enters into the subthreshold region of operation when its gate to source voltage (Vgs) is less than its threshold voltage (Vth). During this condition of operation, minority carriers present in the inversion channel are not very high, but they do correspond a current flow and hence this region is known as *weak inversion*. When the supply voltage (Vdd) is less than the threshold voltage (Vth), the major component of current is provided by subthreshold current as junction leakage and gate current are smaller when operating in the subthreshold region. The current flow is not due to the creation of inversion channel, but due to diffusion.

In the subthreshold region, subthreshold current is exponentially related to Vdd, Vth and gate to source voltage (Vgs). Our aim was to reduce the supply voltage and operate the circuits in or as close to the subthreshold region of operation as possible and to see if circuits really operate at such low voltages and if they do, can there be further savings that can be achieved by operating in this region.

2.2 Adder Configurations

This section gives details about different adder configurations with focus on carry skip adder. The work that was previously done is presented including some analysis about adder delay minimization. Our main aim is to operate the adder in very low voltages.

An adder is one of the most basic and widely used arithmetic component in all computational applications. Different types of adders like ripple carry adder, carry look-ahead adder, carry-select adder have been proposed. Ripple carry adder is very slow as the carry generated must ripple through each and every bit in case of 16-bit adder, thus increasing the delay, especially at low voltages. Carry look-ahead adder is quick, but the design complexity is very high, as it has a lot of gates associated with it for generating propagate and generate signals. Other types of adder consume a lot of power when operating at very low voltages due to the high number of gates associated in the design.

Several full adders were designed to work at very low voltages such as the one presented in [13], but the problem with this design was that it was not functioning well, in that the transition of output signals was not happening completely to logic high level for some of the input patterns applied when we used model files from NCSU and ASU PTM that have been provided in [14] and [15]. As the designs presented in [16] and [17] and others have pass transistor logic involved in the design, all of those circuits suffer from severe threshold loss problem while cascading. Also, the traditional 28-transistor-based CMOS adder presented in [18] was simulated. It was functioning well when the supply voltage was reduced below 0.5V, but even for a single bit adder, the number of transistors required was 28 which is a large number. If we consider a 16-bit cell, the area and power consumption would be very high and hence cannot be used where power is the main criterion. Hence, after considering all these designs, we decided to use carry skip adder as it presents a good balance between area, performance and power.

The carry skip adder we considered for our research was based on [19]. There were a few circuit modifications that we had done to make them operate for different model files and a wide range of operating voltages. Firstly, the XOR gate used in the design consumes slightly more power, and hence we performed simulations on XOR configurations provided in [20] and came up with a design that consumes lesser power than the former. Secondly, an inverter was provided in [19] at the end of each stage which might increase the delay associated with the worst case. So, we used the normal signal originating from a stage and used normal inputs instead of the inverted ones for the next stage inputs. This change was done keeping in mind that the worst case propagation delay is the most important aspect that needs to be concentrated while dealing with a carry skip adder.

The way full adders are grouped together into blocks and the number of levels involved in the design play an important role in determining the worst case propagation delay in a carry skip adder. The work presented in [21] uses dynamic programming algorithms to configure carry skip adder, which does not produce optimum results for actual values of skip and ripple time. A geometric approach was proposed in [22] with an assumption about ratio of skip time and ripple time and hence does not produce accurate results. This idea was extended by [23] for arbitrary skip and ripple time ratios but again bases its results on computer algorithms. An extensive mathematical analysis is presented in [24] to find out the optimal block size in a constant block and also variable block CSA.

The work in [25] presents an optimization strategy only for the case of constant block size, and suggests to use variable block size adder to further improve performance. The authors in [25] provided the relative values of maximum propagation time for deviations from the optimum group size for equal groups, but this was done through mathematical analysis only. Hence, overall, for all different kinds of work already done and proposed, delay minimization by all these previously mentioned papers is based on mathematical analysis or complex computer arithmetic programs but not through simple simulations of the design for different block sizes. This was the motivation for our design to be simulated under different block sizes and number of stages to determine the optimum configuration.

2.3 Magnitude Comparator Design

Magnitude comparator (i.e.,unsigned) is a very important arithmetic component which is used to compare two positive numbers and is used in almost all computational applications. The comparator circuit is a relatively easier design when compared with CSA as it is not very complex in structure and understanding.

The working of a comparator design can be explained as follows. If we consider a 4-bit comparator, which means it compares two 4-bit numbers, the comparison begins from the MSB bit pair. If one of the bits is a 1 and the other is a 0, it means the former number is greater than the latter straightaway. Same is the case when first bit is less than the other bit. In this case, former is less than the latter and both these cases have very small delay associated with them. When the MSB bit pair have the same bit associated with them, the comparison goes to the next significant bit and so on. Hence, if we select the bits in such a way that except for the LSB bit pair, all the other bit pairs are the same, and the worst case delay occurs.

From the above description, we begin by employing inverters on the input signals in a bit pair. The alternating signals between inverted and non inverted inputs of both bits are then AND-ed together, followed by NOR gates. Finally, these are sent through AND and OR gates to get the desired signals indicating whether they are equal or one number is greater/less than the other. Figures 2.1 and 2.2 show the gate diagram of an individual 4-bit block and block diagram of a 16-bit comparator using 4-bit blocks and logic gates.

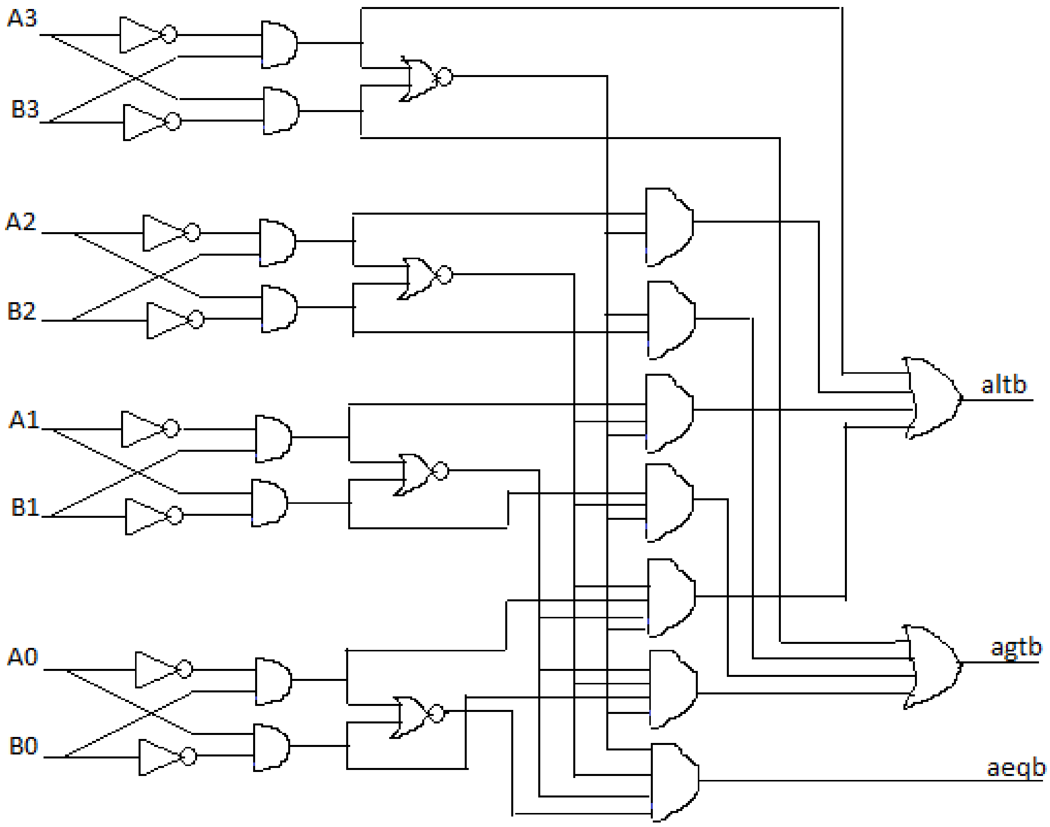


FIGURE 2.1: Gate diagram of a 4-bit comparator block

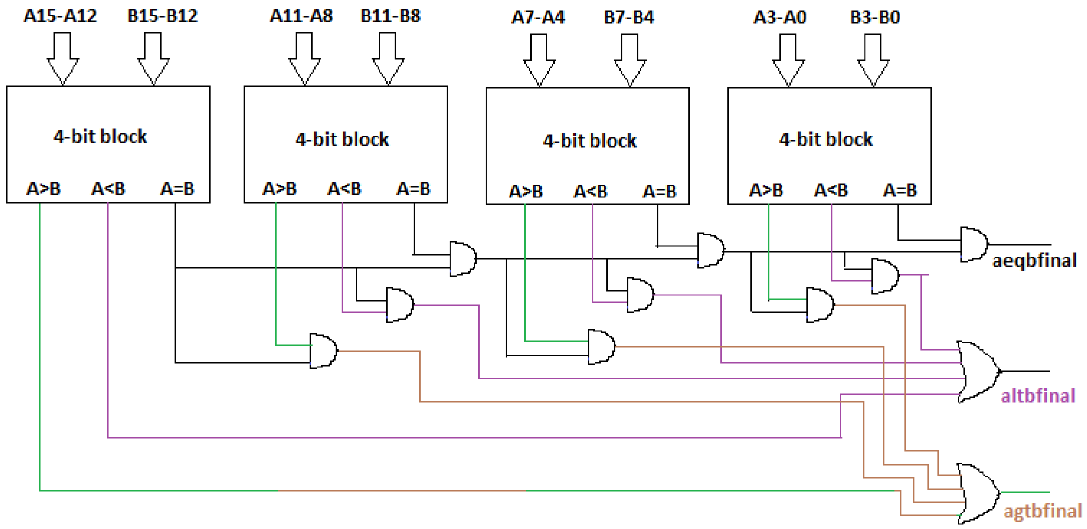


FIGURE 2.2: Block diagram of 16-bit comparator

2.4 Power Gating

The technique of power gating is discussed in this section, and is one of the most effective and widely used leakage reduction methods. Fig. 2.3 shows a general configuration used for the power gating technique. The crux of this technique lies in disconnecting the logic circuit block from the power rails in standby mode. This is accomplished by employing additional transistors operating as switches which offer a high resistance in the standby mode. This high resistance disconnects the virtual power rails from the global power rails [26]. These additional transistors can be placed either in between the pull-up network and supply voltage (Vdd) terminal, called *header* configuration, or between the pull-down network and ground (GND) terminal, called *footer* transistor. In addition to providing a high resistance in standby mode, these transistors create the stacking effect, which results in an increase in the threshold voltage of the transistors in stack. This combination of resistance and threshold voltage increase is the result of leakage current reduction with this method [26].

The operation of this method can be explained as follows. During active mode or normal operation of the circuit, the sleep transistors are turned on. The transistors in the on condition offer a low resistance and hence the voltage of virtual supply rails is almost the same as that of global supply rails. As a result of this, normal operation of the

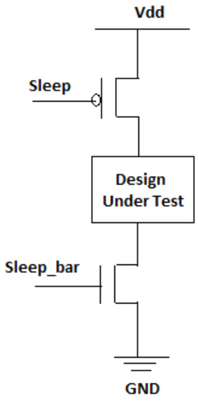


FIGURE 2.3: Power gating for a design

circuit is ensured without a significant impact on the circuit performance [27]. During the standby mode, the sleep transistors are turned off by asserting their gate signals low in case of footer transistors. This presents a large resistance between the global and virtual supply rails which ultimately cuts off the supply to the logic block, thus reducing leakage power. The virtual ground terminal voltage should not be too low under this situation which might not produce requisite savings with this method. Hence, the width of footer transistor has to be adjusted in case virtual ground terminal potential is small to make sure savings are achieved.

There are several issues that have to be taken into consideration while using this method. The size of a switch affects the circuit delay in active mode and the leakage current in sleep mode, so it should be determined carefully [28]. If the transistors are sized to be very small, the performance is affected as the high-to-low transition delay of the circuit is increased due to the voltage drop on the sleep transistor, decreasing the effective supply voltage of the logic gate [29]. If they are made very large, the result is an area overhead, small leakage power saving, and also increase in dynamic power consumption to turn the transistors on and off [29]. Hence, they have to be sufficiently big but not too big that it has adverse effects on the circuit area. There can be different configurations for power gating: one is employing both header and footer sleep transistors, the second is employing only a footer switch and the last is using only a header switch. Fig. 2.4 shows all the three possible configurations for the power gating technique. We have selected the configuration which employs only a single transistor, the footer transistor as it is sufficient and also smaller in area for the same switching current, resulting in a reduction of area and active mode voltage drop. In addition to these features, a single big switch is generally used (implemented as multiple switches in parallel) as it is one of the most widely used method in industrial applications [28].

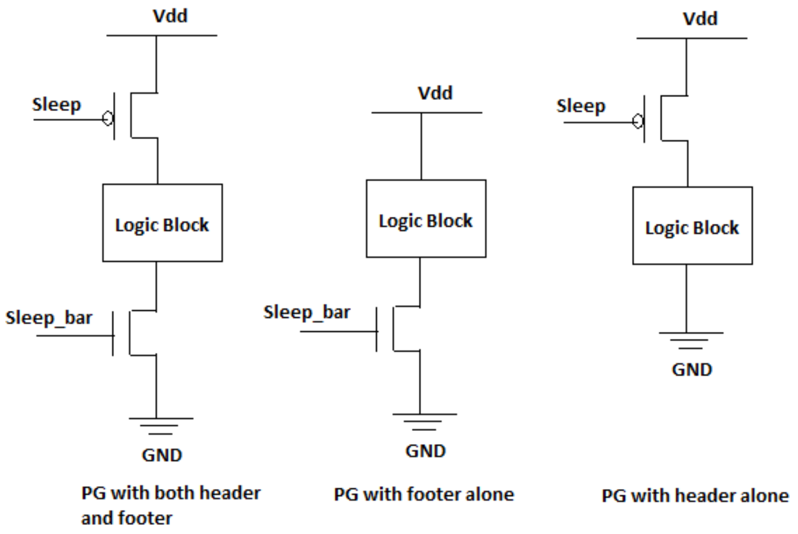


Figure 2.4: Different power gating configurations for a circuit

2.5 Clock Gating

This section presents a brief introduction to a very well known power reduction technique, clock gating, and is primarily employed when dynamic power consumption of a design has to be minimized. The clock gating technique is widely used when dealing with minimizing switching power of clock signals associated with flip-flops and their related combinational circuits. Switching power comes into picture when a signal is changing values, so energy has to be supplied or lost to charge/discharge load capacitance associated with the gate [30].

The main idea of clock gating is to reduce the switching activity of a design by minimizing the number of unused clock signals that are switching simultaneously, without losing the performance. In other words, it aims to prevent parts of the design from switching at all, by means of disconnecting them when not necessary, provided proper functionality can be achieved. The clock signal employed in sequential circuits switches every cycle and has an activity factor of one, thus consuming a lot of power, due to power contributed by combinatorial blocks, flip flops and clock distribution network, as the clock signal has to travel throughout the design passing through a lot of interconnects. Clock signal does not carry any information and is primarily used for synchronization purposes [31], and hence unnecessary toggling activity can be reduced by employing clock gating in the way as follows.

We can employ a circuitry which can control when a new signal needs to be clocked into the flip flops. This circuitry generates a gated clock signal. When the stored data or state remains unchanged, we do not need the clock signal, which may consume power unnecessarily due to its toggling activity if turned on. So we can disconnect the blocks that are dependant on clock signal during that time through the use of a clock gating circuitry and its associated gated clock output.

The clock gating circuitry consists of an Enable signal which can be controlled independently and logic gates. The enable signal is applied either through a simple combinatorial gate like AND/ NOR gate or through the use of sequential elements like flip flops or latches, based on the requirement and one such method is represented in Figure 2.5 which contains a latch and an AND gate to generate the gated clock signal.

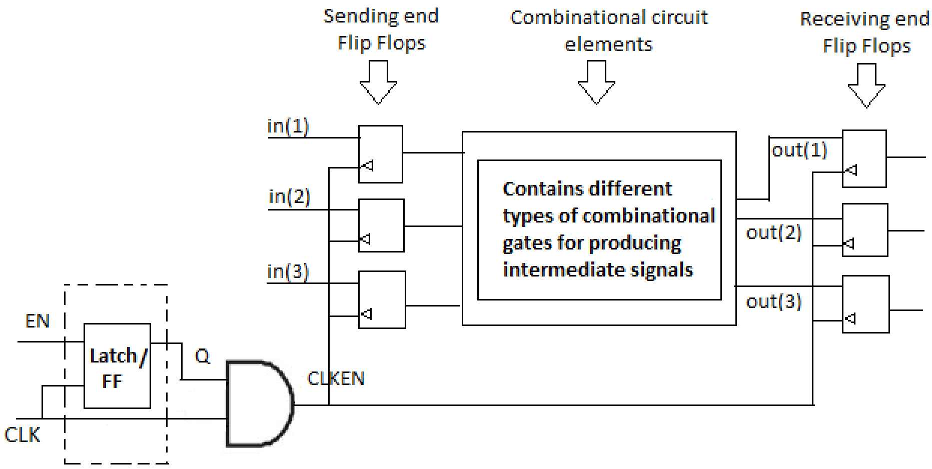


FIGURE 2.5: Clock Gating representation

The enable signal is controlled in such a way that the gated clock signal produced does not switch continuously and can be turned off to prevent it from reaching some logic modules where the current state is held and not being changed. The most common way to apply the gated clock signal is through the use of a latch and an AND gate. This method saves power well but there is a problem of testability that arises because of this, in the sense the gated clock signal depends extensively on the control input and hence is difficult to control. AND gate based, NOR gate based, Latch-based AND, Latch based NOR, MUX-based are some of the widely used techniques to generate the gated clock signal.

There are some issues that have to be carefully considered while employing clock gating. There should not be glitches occurring in the design due to the enable signal not applied properly. To avoid this, the enable signal is changed only during the low clock phase and not during the high period, as this would cause synchronization problems of the related signals in case of positive edge triggered flip flops.

2.6 EDS

This section gives details about a novel technique of using Error Detection Sequential (EDS) circuits to detect late timing transitions in sequential designs.

One of the most important factors that considerably affects the performance and energy efficiency of VLSI circuits such as microprocessors, servers and other complex designs is the variability in device and circuit parameters. These variabilities in the parameters, also called dynamic parameter variations, arise due to several reasons, either environmental or changes in the workload.

It is of paramount importance to make sure the system operates correctly even in the presence of dynamic variations. This can be achieved by employing a resilient design that contains error detection and recovery circuits. When a timing error has occurred due to a dynamic parameter variation, the resilient circuit detects and corrects the error [32]. One of the most important advantages of using the resilient circuits is that the circuit can be operated at a higher clock frequency or a lower supply voltage than the conventional design.

In a conventional design, as shown in Figure 2.6, a critical path is bounded by sending and receiving flip flops. Figure 2.7 shows timing diagrams for a conventional design under normal conditions and during worst case dynamic variations.

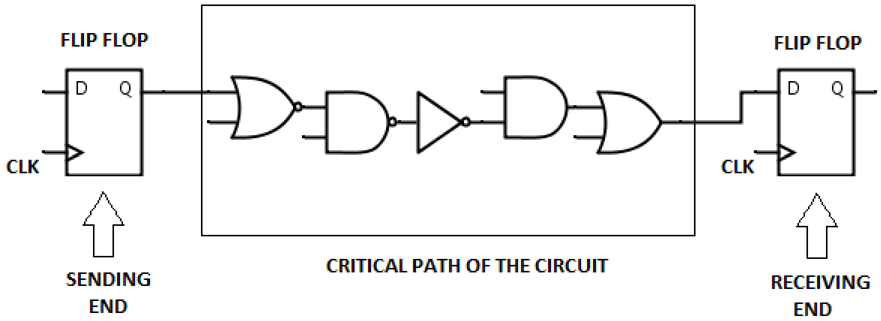


FIGURE 2.6: Conventional structure of a design

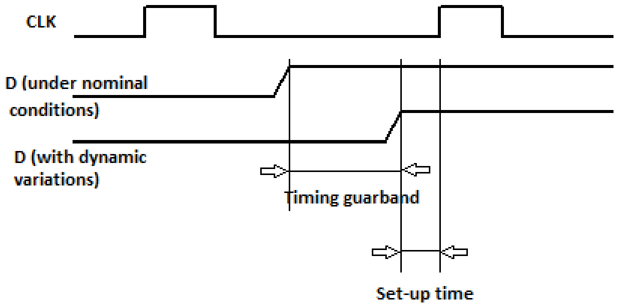


FIGURE 2.7: Timing diagram for a conventional design

Under nominal conditions of operation, the input at the receiving flip flop, arrives early to the rising edge of the clock. In the presence of dynamic variations in the design, in order to ensure proper functionality of the structure, input to the receiving flip flop should arrive at least a set-up time prior to the rising edge of the clock. If this criterion is not met, a set-up time violation is said to occur, which leads to a wrong value being latched by the flip flop. The difference between the input arrival times in the above mentioned cases is the timing guardband that has to be provided in normal designs to ensure correct behavior under dynamic variations.

The basic design of an Error Detection Sequential (EDS) circuit as proposed in the Intel 45nm Resilient Microprocessor core is given in Fig. 2.8. The resilient design has a similar structure to the normal design but the major difference between the two is that the receiving flip flop is replaced by an EDS circuit in resilient design. This EDS circuit configuration uses a positive edge triggered latch in the datapath instead of a flip flop and also a shadow flip flop which is triggered by the same input at positive edge of the clock as shown in Fig. 2.9. An XOR logic gate is also employed which compares the outputs of the latch and flip flop and produces a logic high error signal if they differ.

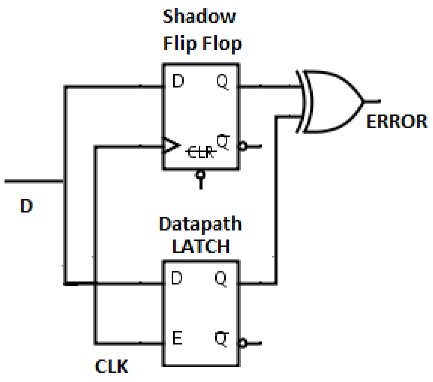


FIGURE 2.8: Gate level EDS circuit design

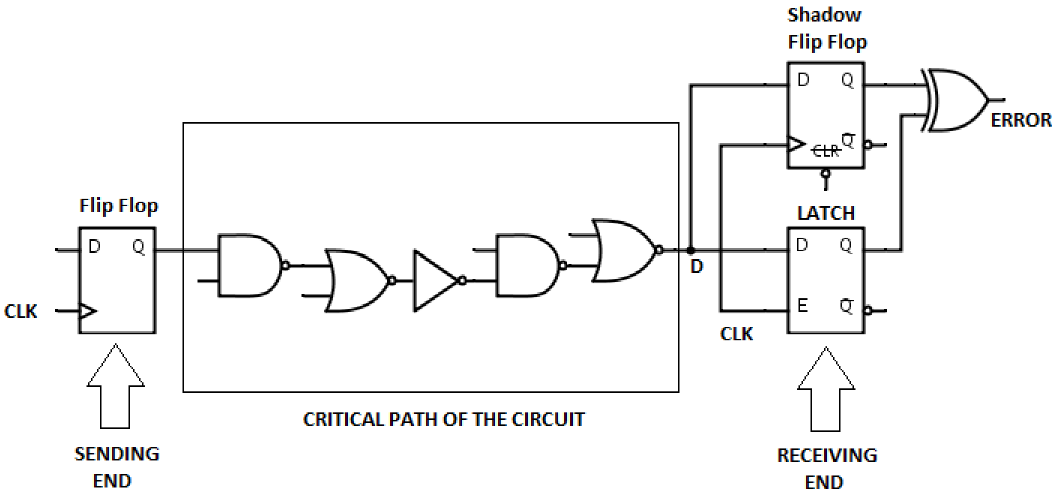


FIGURE 2.9: Structure of EDS design

Fig. 2.10 shows the timing diagram for an EDS circuit. In case the input data to the latch arrives late, the shadow flip flop output remains low but the datapath latch, being transparent during the positive clock period, latches on the late changing value. This causes the outputs to be different from the flip flop and latch, causing the ERROR signal to be asserted high as mentioned earlier, thus detecting the error due to late timing transition.

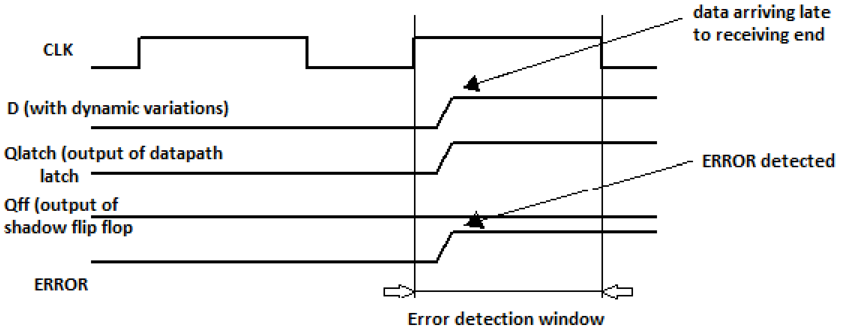

CLK
D (with dynamic variations)
Qlatch (output of datapath latch
Qff (output of shadow flip flop
ERROR
data arriving late to receiving end
ERROR detected
Error detection window

FIGURE 2.10: Timing diagram showing error detection with an EDS circuit

If the same input arrives earlier than when the error detection window begins, the latch and flip flop outputs are the same and hence there is no error. This is represented in the timing diagram Fig. 2.11.

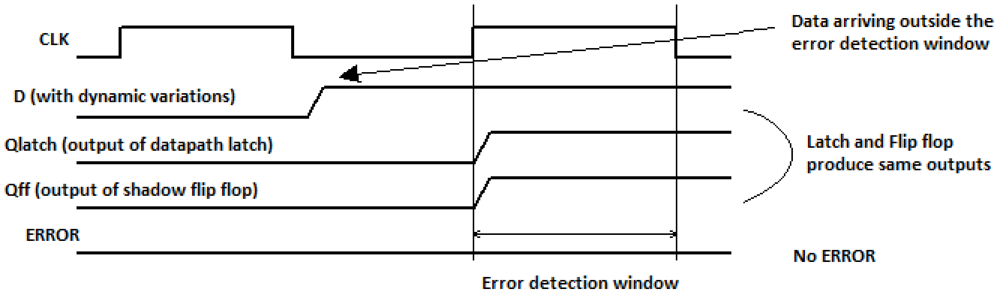

CLK
D (with dynamic variations)
Qlatch (output of datapath latch)
Qff (output of shadow flip flop)
ERROR
Data arriving outside the error detection window
Latch and Flip flop produce same outputs
No ERROR
Error detection window

FIGURE 2.11: Timing diagram showing no error with an EDS circuit

The key idea in this technique is that the error due to late timing transitions is detected only during the high clock phase, which is also known as error detection window (T_w). There are a set of timing constraints that have to be satisfied by the paths employing EDS circuits as the receiving sequential circuit. The constraint for the maximum delay path in the presence of worst case dynamic variations for EDS is given as

$$T_{max} \leq T_{cycle} + T_w - T_{setup,clk} \tag{2.1}$$

T_{max} is the maximum path delay for EDS paths, T_{cycle} is the clock cycle time, $T_{setup,clk}$ is the set up time of CLK for the datapath latch based on the rising clock edge.

The minimum path delay timing constraint during worst case dynamic conditions is given as

$$T_{min} \geq T_w + T_{hold,clk} \tag{2.2}$$

T_{min} is minimum path delay for EDS paths, $T_{hold,clk}$ is hold time of CLK for the latch based on the falling clock edge.

The next few chapters deal with design of circuits, simulations and power reduction techniques applied on the designs considered in our thesis.

Chapter 3

Design and Simulation of Carry Skip Adder and Magnitude Comparator Circuits

This chapter deals with the design of *carry skip adder* and *comparator* circuits. The first part deals with design aspects, followed by details about measurements which also cover some implementation concepts for the circuits. Different types of simulations performed on these models are presented next followed by the observations in the end.

3.1 Design Aspects of Circuits

The *carry skip adder* circuit that we have considered for our thesis is based on the work in [19]. We introduced some modifications in the design which are mentioned here. Instead of the extra inverter that was introduced in the design at the end of each stage as presented in [19], we used the normal carry output from each stage and fed it to the next stage, thus providing a chance to reduce the delay as the inverters would be in the critical path of the design. The second modification was related to the XOR configuration that was presented in [19]. After considering the simulations, delay and power values associated with different types of XOR configurations provided in [20], we selected the configuration presented in Fig 3.1 as this configuration consumes less power which is our primary requirement, even though the delay is slightly higher and hence features throughout our design.

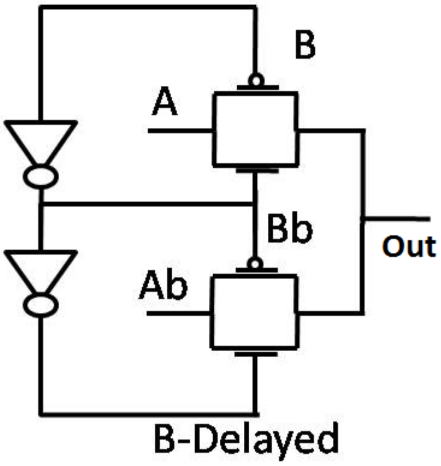


Figure 3.1: XOR configuration used

The *comparator* circuit considered in our thesis is a regular comparator that compares two positive numbers and asserts a signal high based on an operand being greater than, equal to or less than the other operand. It has gates like Inverter, AND, OR, NOR and others to generate signals which indicate if a signal is greater than, less than or equal to the other in 1 stage consisting of 4 bits and is replicated 4 times to generate a 16-bit comparator. The following sections give details about simulations, measurement of power and delay values for both the configurations.

3.2 Method of Measurement of Worst Case Delay and Power

The procedure we followed to measure the worst case delay is briefly described here. We used Synopsys HSPICE which is a powerful simulation tool and can be used for a wide range of applications and delay measurement is one such use of the tool. We considered the two signals, (i.e.transition source and transition destination signals) for which the delay has to be measured, overlapped them and selected measurement tool. In this tool, we adjust the options such that we consider the rise/fall transition for respective signals and 50% voltage levels at which the measurement was taken.

For power measurement using HSPICE, we used the built-in .measure command provided by the tool which allows us to find out the integral value of a signal over the simulation time which essentially provides the average value of a signal during that time.

3.3 Simulation and Measurement Results

After making sure that the CSA and comparator configurations are operating well at various voltages starting from 0.2V or higher based on the model file considered, we had to measure power consumption values and worst case delays at various voltages for all model files using the methods specified earlier. But, considering only a few voltages is sufficient in arriving at a reasonable combination of operating voltage and model file for further analyses which are based on a few factors explained below.

3.3.1 Power Tables

Firstly, since operating the circuit at very low power is our main criterion, we have to make sure the operating voltage is low which has a great impact on reducing power owing to the square dependence of power on operating voltage. These operating voltages have to be chosen in such a way that the least possible values (say 0.2V, 0.3V or 0.4V) for a particular model file are considered, but care has to be taken that they are indeed voltages at which circuits operate well for those models. We represented such voltages as voltages of interest given in Table 3.1, and hence considered power and delay values for these to arrive at the desired pair of voltage and model file.

Secondly, if the operating voltage was greater than 0.5V, we did not consider those even though the circuits were functioning well as they would consume more power, and our primary aim was to achieve very low power and operate in or as close as possible to the subthreshold region of operation.

TABLE 3.1: Voltages of interest for different model files

	Threshold Voltages (V)			
Model file	NMOS	PMOS	Voltages of Interest (V)	Lowest Voltages resulting in subthreshold operation (V)
ASU PTM	0.3423	-0.23122	0.2,0.3,0.4	0.2
NCSU VTL	0.322	-0.3021	0.3,0.4	0.3
NCSU VTG	0.4106	-0.3842	0.3,0.4	0.3
NCSU VTH	0.6078	-0.5044	0.6(high)	-

The other factor considered was that the performance of designs had to be within reasonable limits as designs which are extremely slow do not present useful opportunities. Hence, we had to consider the worst case delay values and select a value which is reasonably good and at reasonably low voltage.

Tables 3.2 and 3.3 show the power consumption values for the adder and comparator designs computed at voltages of interest for various model files.

TABLE 3.2: Power consumption for CSA at voltages of interest

	Threshold Voltages (V)			
Model file	NMOS	PMOS	Voltages of Interest (V)	Power consumed (uW)
ASU PTM	0.3423	-0.23122	0.2 0.3 0.4	0.4617 0.9839 1.954
NCSU VTL	0.322	-0.3021	0.3 0.4	0.8898 1.687
NCSU VTG	0.4106	-0.3842	0.3 0.4	0.4218 0.7586
NCSU VTH	0.6078	-0.5044	0.6(high)	1.529(not considered)

3.3.2 Explanation of Worst Case Delays

This section provides explanations for the worst case delays of *carry skip adder* and *comparator* configurations along with some example input patterns for showing various cases possible in an Adder. A *carry skip adder* has blocks of full adders forming a stage,

Table 3.3: Power consumption for comparator at voltages of interest

	Threshold Voltages (V)			
Model file	NMOS	PMOS	Voltages of Interest (V)	Power consumed (uW)
ASU PTM	0.3423	-0.23122	0.2 0.3 0.4	0.3421 0.7201 1.401
NCSU VTL	0.322	-0.3021	0.3 0.4	0.5519 1.081
NCSU VTG	0.4106	-0.3842	0.3 0.4	0.15 0.263
NCSU VTH	0.6078	-0.5044	0.6(high)	0.4405(not considered)

which is linked to the other stage through a multiplexer. This is shown in Figures 3.2 and 3.3 where a 16-bit adder is organized in 4 stages, each stage containing 4 full adder cells.

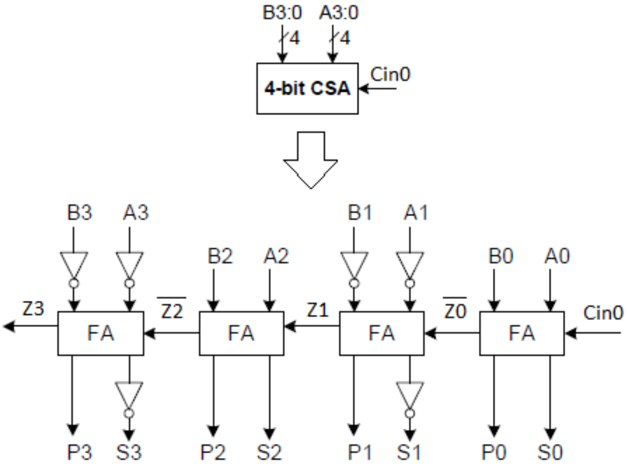


Figure 3.2: The structure of a 4-bit CSA representing a stage

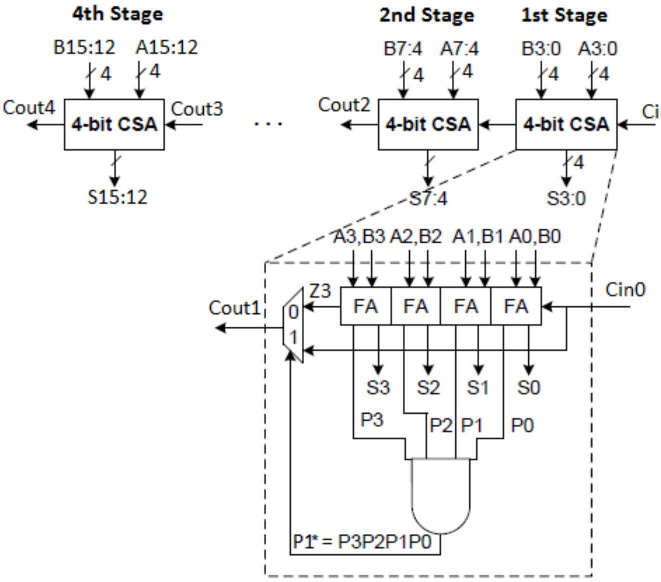


Figure 3.3: The 16-bit CSA formed through stages connected using MUXes

Each stage can either propagate the carry coming from the previous stage or skip it based on a group propagate signal which is calculated as soon as the input bits are available using NAND and NOR gates, as shown in Figure 3.4.

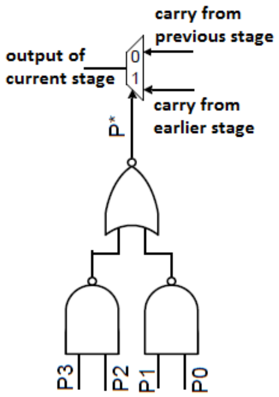


Figure 3.4: Generation of group propogate signal through NAND-NOR gates

This group propagate signal is used as the select line for the multiplexer provided at the end of each stage. The multiplexer selects from the carry of the previous stage or from earlier stages based on the select line being 0 or 1 respectively.

The worst case delay occurs in the CSA when a carry is generated in the least significant bit (LSB) and is propagated through the intermediate stages all the way to the most significant bit (MSB). This means that the intermediate stages have to propagate the carry that is generated by the least significant bit. The condition to be satisfied is shown in Figure 3.5.

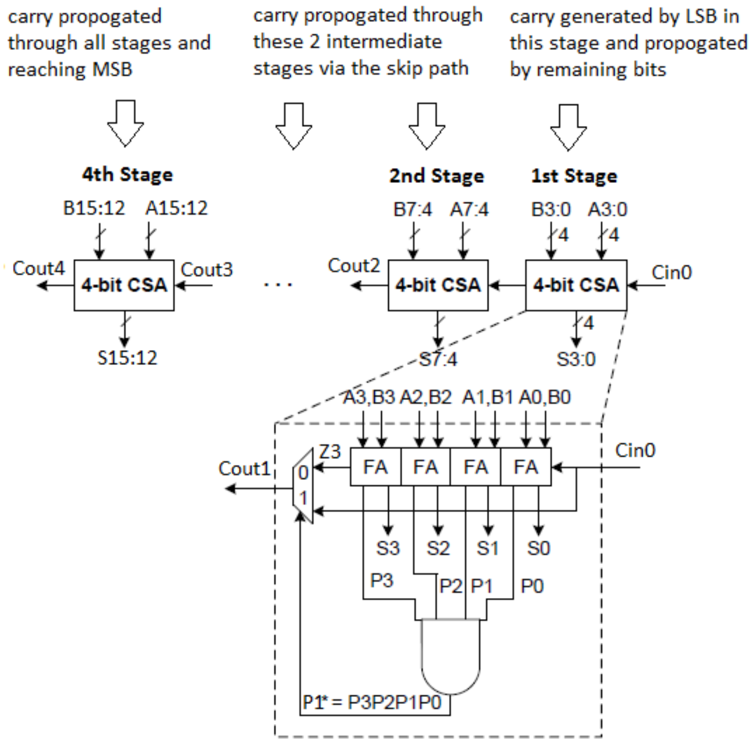
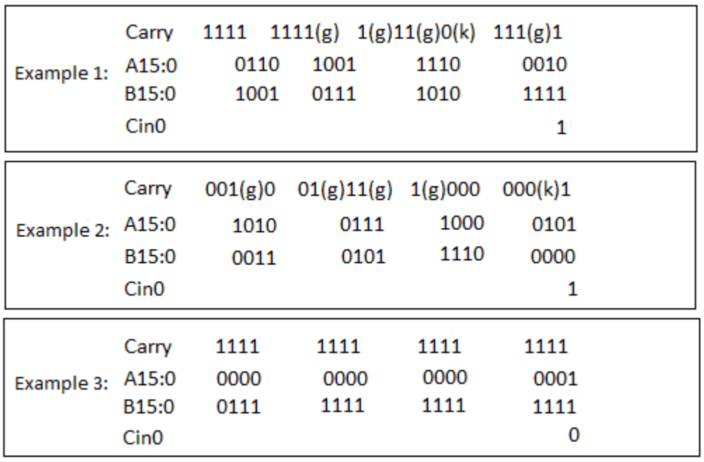


FIGURE 3.5: Condition under which worst case delay occurs for a *carry skip adder*

If an intermediate bit pair in a stage generates its own carry, that means it is not propagating the carry from earlier stages, or the carry generated by the LSB is stopped at this location instead of propagating it further. This does not correspond to the worst case delay as there is a new carry generated which will then proceed toward the MSB, instead of the old one. This is shown in Figure 3.6, and happens when we have bits 11 associated with the inputs.

If an intermediate bit in a stage kills the carry, it means the carry propagation path has ended prematurely as this bit cannot propagate the carry from LSB any further. This condition is also represented in Figure 3.6, and happens when we have bits 00 associated with the inputs.



Example 1:	Carry	1111	1111(g)	1(g)11(g)0(k)	111(g)1
	A15:0	0110	1001	1110	0010
	B15:0	1001	0111	1010	1111
	Cin0				1
Example 2:	Carry	001(g)0	01(g)11(g)	1(g)000	000(k)1
	A15:0	1010	0111	1000	0101
	B15:0	0011	0101	1110	0000
	Cin0				1
Example 3:	Carry	1111	1111	1111	1111
	A15:0	0000	0000	0000	0001
	B15:0	0111	1111	1111	1111
	Cin0				0

FIGURE 3.6: Different input patterns showing carry generation(g), propogation(p) and kill(k) by bit pairs

The explanation for some input patterns goes as follows: In Example 1 of Figure 3.6, the LSB generates a carry which has to be propagated to the MSB. But, bit 2 input pair generates a carry of its own, thus beginning a new path for carry, which now starts from 2nd bit instead of the first. This is obviously less than the maximum delay possible. Also, at bit 5, the carry from earlier stages is killed as this bit pair does not propagate the carry. Hence, a new path starts again at bit 6 and ends at bit 8, which starts a new path, again interrupted by bit 9, the carry generated by which continues to the MSB. This delay is way less than the maximum delay due to discontinuity in the carry path from LSB to MSB.

In Example 2 of Figure 3.6, the LSB generates a carry and is killed by the 2nd bit input pair. A new carry is generated by 8th bit pair and is ended at 9th bit owing to a newly generated carry by this input bit pair. But, this ends in bit 11 due to carry generated here which is killed again at bit 12. Bit 14 pair generates an input carry which is not propagated to the MSB at all. In this case too, the delay is not even close to being maximum delay.

This explanation suggests that in order for the intermediate stages to propagate the carry generated by the LSB, the bit-pairs in these stages have to make sure that they neither generate their own carry nor kill the carry. This condition is met when we have 10 associated with the input bits. This condition is also shown in Figure 3.6.

In Example 3 of Figure 3.6, the LSB generates a carry which is not stopped at any other bit location as none of the bit pairs generates or kills the carry coming from lesser order bits.

When 1 and 0 are associated with the input bits, propagate (P) signal which is A(XOR)B is 1, and hence all P's are 1s for these inputs. This results in group propagate signal being asserted high. As a result of this, the intermediate stages skip the carry from the previous stage and propagate the carry associated with the earlier stage instead.

This is the main distinction between a *ripple carry adder* and the CSA in that, since the group propagate signal is readily calculated upon availability of input bits, some blocks can be skipped thus reducing the delay compared to other kind of adders. This way, the worst case delay happens when the LSB generates a carry which ripples through the 1st stage through all the 4 full adder cells, is skipped by the intermediate 2 stages which ensures carry propagation and then ripples through the final 4 full adder cells in the last stage. This path is depicted in Figure 3.7.

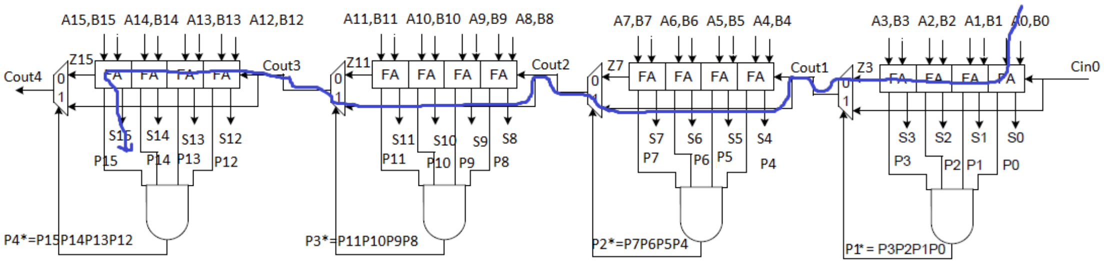


Figure 3.7: Worst case delay represented for *carry skip adder*

3.3.3 Worst Case Delay Tables

The input pattern that achieves this condition is when A[15:0] and B[15:0] change from 0000 0000 0000 0000 to 0000 0000 0000 0001 and 0000 0000 0000 0000 to 0111 1111 1111 1111 respectively. When these inputs are applied, the worst case delay is measured from the A0 input to sum15bar output. The worst case delays measured for the CSA circuit under the voltages of interest without any sequential elements such as flip flops are given in Table 3.4. These values would be different if there are other elements included in the circuit.

Table 3.4: Worst case delays for CSA at voltages of interest

	Threshold Voltages (V)			
Model file	NMOS	PMOS	Voltages of Interest (V)	Worst case delay (ns)
ASU PTM	0.3423	-0.23122	0.2 0.3 0.4	6.2469 1.3264 0.5351
NCSU VTL	0.322	-0.3021	0.3 0.4	3.4295 1.04
NCSU VTG	0.4106	-0.3842	0.3 0.4	21.646 3.8269
NCSU VTH	0.6078	-0.5044	0.6(high)	-(not considered due to high voltage)

In the case of a *comparator*, the worst case delay occurs when the least significant bit (LSB) is the one which determines the result of a comparison. If one of the most significant bits (MSB) of the 2 operands is different, it means that it is either less than or greater than the other operand. In this case, there is not much delay in generating the output on application of the inputs. Same is the case for intermediate bits too. The delay will be higher than the previous case but is not the worst case delay.

When we have all the higher stages with the same bits and just the LSB in the last stage with different bits, the computation has to wait until the last set of bits (LSB) to determine the result of the comparison. This situation is shown in Figure 3.8 and corresponds to the worst case delay measured from A15 input to altbfinal output.

The worst case delays measured for comparator for the voltages of interest without any flip flops are given in Table 3.5.

3.3.4 Selection of Operating Voltage and Model File Pair

Based on the Tables 3.4 and 3.5, we selected the VTG model file and 0.4V as the pair which would be consistent with our requirements of power-delay balance. This pair is

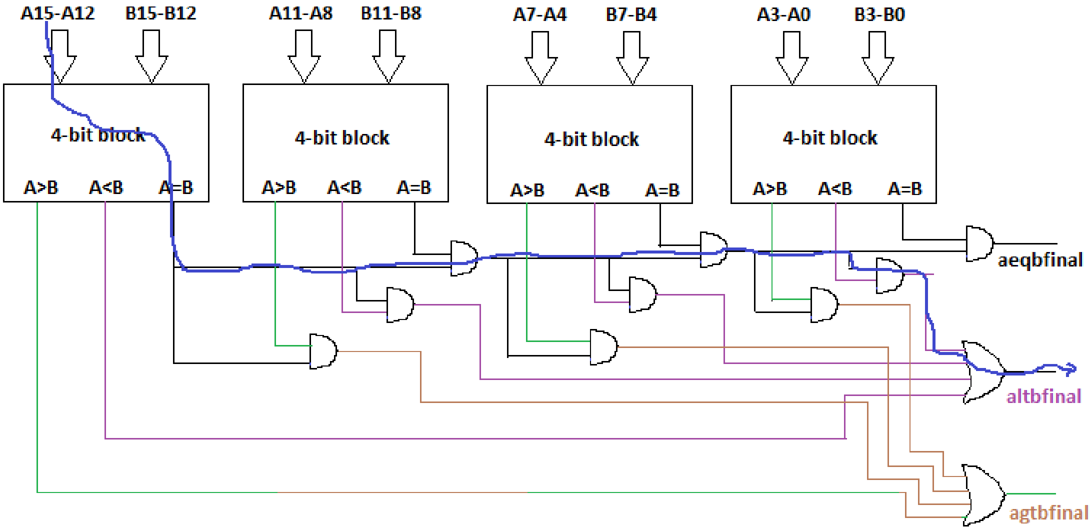


FIGURE 3.8: Worst case delay represented for the magnitude comprator

TABLE 3.5: Worst case delays for comparator at voltages of interest

	Threshold Voltages (V)			
Model file	NMOS	PMOS	Voltages of Interest (V)	Worst case delay (ns)
ASU PTM	0.3423	-0.23122	0.2 0.3 0.4	4.5623 0.7702 0.2525
NCSU VTL	0.322	-0.3021	0.3 0.4	1.4864 0.4486
NCSU VTG	0.4106	-0.3842	0.3 0.4	9.0791 1.5945
NCSU VTH	0.6078	-0.5044	0.6(high)	-(not considered since voltage is high)

used as the standard for all other operations performed henceforth on the circuits.

3.4 Observations

This section presents the observations that we could come up with based on the simulations and analysis performed. Firstly, from the minimum operating voltages at which the circuits operate, we could observe that the circuits operate at very low voltages such as 0.2V as well for ASU PTM model file. But, the delay associated is very high and hence not profitable to use at this voltage. These voltages for VTL, VTG and VTH model files from NCSU were 0.3, 0.3 and 0.6V respectively, but delays or power values were higher in the first 2 cases and the voltage is too high in the last case and hence can consume more power compared to other voltages.

We observed the patterns that were found for different model files as the voltage is varied in comparison to the threshold voltage. Based on Table 3.5, we explain some of the observations in the following discussions.

If the threshold voltage increases at a particular voltage of operation, the delay would increase. When the operating voltage is less than the threshold voltage, the device operates in the subthreshold region of operation. This is the case where we consider operating voltage of 0.2V for ASU_PTM model file, and 0.3V for NCSU_VTG model file. For these voltages, since the circuit operates in the subthreshold or weak inversion region, the driving strength of the transistors is not as high as it would be in normal operating conditions, and hence the circuit is slow resulting in higher delay values. When we increase the voltage for the same model files, the design enters normal inversion operation which causes it to speed up, thus reducing the delay.

The worst case delays were as high as 6-7 times in the subthreshold region of operation for both ASU_PTM (0.2V) and NCSU_VTG (0.3V) model files, when compared to delays observed when the operating voltage is increased to 0.3V and 0.4V for ASU_PTM and NCSU_VTG model files respectively, so that they are very close to or higher than the threshold voltages of the transistors. This result confirms the delay difference for different operating voltages for the designs based on our knowledge.

Secondly, we observed that as the operating voltage increases, power increases and worst case delay decreases even at very low voltages.

Thirdly, we used several different input patterns to check and see if they represent any anomalies as far as the worst case delay is concerned, but none of them gave results that were larger than what we had with the worst case inputs we considered in our simulations. This suggests the procedure and measurement techniques we followed were correct and accurate.

3.5 Delay Dependency on Number of Blocks per Stage

This section introduces an interesting point of view of minimizing the overall delay of the 16-bit *carry skip adder*. For applications where performance is not a worry, we can use any configuration of the circuit, in terms of the number of blocks per stage and the number of stages that make up the 16-bit adder. But, if performance also plays a key role in determining the best design, we have to make sure that the overall delay is minimized.

The following section describes about this aspect along with results from simulations performed on our design. All the simulations are performed on our design by considering NCSU_VTG model file and operating voltage of 0.4V as explained already at the end of the earlier section. Our main aim through these simulations was to effectively vary the block size that can be employed in a stage at very low voltage (0.4V), and measure the worst case delay of the design for different cases. This information is very useful in selecting the ideal configuration based on the delay requirement.

3.5.1 Motivation and Explanation

In the design that we considered for our thesis, we used 4 stages making up the 16-bit design, and each stage consists of 4 full adder cells forming a chain-like structure for propagating the carry in case the bypass path is not taken as shown in Fig. 3.7. We already presented the explanation and input cases that would provide the worst case delay for the design in the previous sections.

An interesting question that arises is what happens when the number of stages is varied, and also how the delay varies upon changing the number of full adder cells per stage. To answer this question, we started with the design that we initially considered and found the worst case delay. We then changed the number of FA cells/stage. This results in the number of stages to change as well, since a total of 16 FA cells have to be incorporated in the design.

For example, if the number of FA cells/stage is changed from 4, which was in the original design, to 2, the number of stages changes from 4 to 8, thus maintaining the same number of bits for the design. As a result of increase of the number of stages, there is an associated increase in the number of MUX circuits that are employed, the primary function of which is to select between the carry passing through all the FA cells of the present stage or the carry from the earlier stage which was skipped by the present stage FA cells, as explained in the previous section. As a result of this change, for the same input pattern that generates worst case delay, the critical path varies.

In the original design, the worst case happens when carry generated by LSB passes through all FA cells in stage 1, then skips stages 2 and 3 and finally passes through the FA cells in the last stage to arrive at the Sum15_bar output as shown in Fig.3.9.

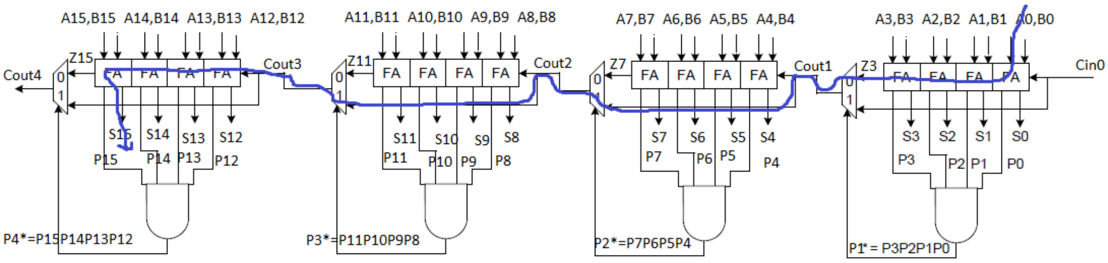


FIGURE 3.9: Worst case delay represented for the CSA with 4 blocks per stage

If the number of stages is increased as mentioned above, the critical path would now happen when the carry generated by LSB passes through FA cells in first stage, then skips the intermediate 6 stages and finally passes through FA cells in the last stage. This condition is shown in Fig. 3.10.

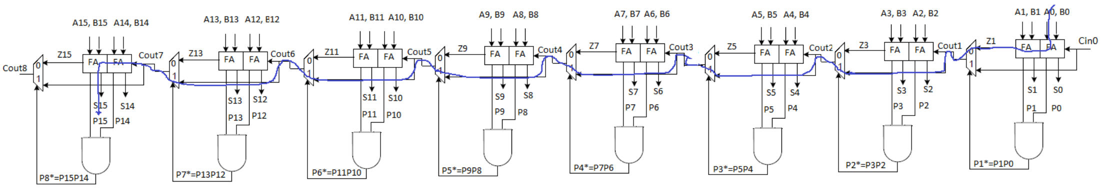


FIGURE 3.10: Worst case delay path for CSA with 2 blocks per stage

Therefore, the number of skip stages has increased now, or the computations performed by MUX circuit in selecting the appropriate signal to propagate plays an important role in determining the overall delay of the design. Similarly, the simulations are performed, now with the number of FA cells/stage changed to 6 and then to 8 and the worst case delays were observed.

3.5.2 Simulations and Results

Tables 3.6 and 3.7 show the results of simulations performed on the design by varying the number of blocks and stages. Also, these tables provide for separate cases where different sizing of transmission gates is considered. Table 3.6 gives the results of simulations performed when the transmission gates were sized similar to all other transistors in the design, wherein the PMOS transistor was sized to 900nm and NMOS to 450nm for all the transmission gates; whereas in the results mentioned in Table 3.7, a slightly conservative sizing is performed, wherein the width of PMOS transistor is made 450nm and width of NMOS transistor is made 225nm for all the transmission gates involved in the design.

Table 3.6: Worst case delays for CSA for different number of blocks and stages with T-gate sizes same as those for other transistors

Number of FA cells per stage	Worst case delay (ns)
2	8.4992
4	3.9067
6	3.7418
8	4.8076

Table 3.7: Worst case delays for CSA for different number of blocks and stages with T-gates conservatively sized

Number of FA cells per stage	Worst case delay (ns)
2	9.9113
4	3.7961
6	3.5078
8	4.4331

The curves that represent the results tabulated are given in Fig.3.11.

3.5.3 Observations

There are some important observations that can be made from these curves. Firstly, we observe that this data follows a trend in that: as the number of FA cells per stage increases, the worst case delay decreases initially until it reaches a breaking point where

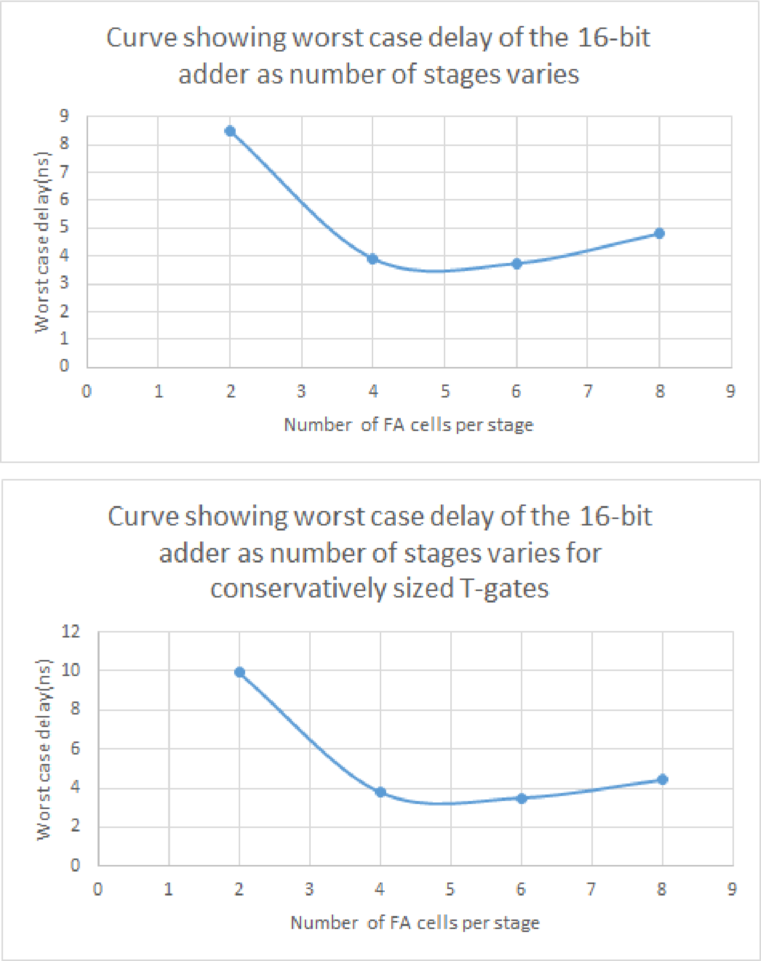


FIGURE 3.11: Worst case delay variation with number of blocks

the delay is the minimum, and then it begins to increase again as the number of blocks per stage increases. This is clearly visible in both the cases that we considered showing regularity in the results.

The reason for this trend can be explained as follows: Initially, when there are only 2 FA cells in a stage, the delay due to carry propagating through these 2 cells is less. Also, since in this case we have 7 MUX circuits which are used to propagate the carry bypassed by the intermediate stages, the delay associated with this component increases which can be explained as follows.

Each FA stage has a carry circuit that is driven by power source, which supplies energy to all transistors involved in the circuit, and hence it operates faster as the charging and discharging of the capacitive nodes can take place quickly through the power source and ground terminals. When we consider a chain of MUXs, these are nothing but transmission gates that are connected together to form a chain-like structure. The main distinction with this chain when compared to a FA chain is that there is no power source to drive the transmission gates. They contain just a bunch of PMOS and NMOS gates connected back to back in series which assist each other in the operation of this chain. As a result of this, the chain of transmission gates is slower, and usually has more delay than the delay contributed by a FA chain.

When the number of cells increases per stage, the component of delay caused by MUX is reduced as there are only 2 intermediate MUXs skipping the carry in the case of 4-bit blocks. A further increase in the number of FA cells per stage causes a reduction of delay contributed due to MUX cells, but the delay contributed by FA cells connected as a chain now dominates the total delay.

Secondly, we see that the optimal delay case occurs around the 4-6 cells/stage region. This would make sure that a balance exists between delay contributed by MUX circuit and that propagated by FA chains. If the number of MUX cells increases, the delay increases and the same applies even if the number of FA cells increases, but the increase is less compared to the former as explained earlier. Hence, we selected the configuration having 4 stages and 4 FA cells per stage for our thesis, from both power and performance point of view.

3.6 Error Resilient Circuit Design for Carry Skip Adder

This section presents details about an interesting topic: the effectiveness and applicability of Error Detection Sequential (EDS) circuits at ultra low voltages. The basic design aspects of EDS for the carry skip adder circuit that is considered in our thesis is presented first along with implementation details. Simulations performed are presented next followed by the problems with these and finally some unanswered questions that need further analysis are posed.

3.6.1 Design and Implementation Details

This part deals with the design and implementation aspects of EDS. As it was presented earlier in Chapter 2, the EDS circuits are used to detect late timing transitions occurring in a design. EDS is introduced in our design by placing it at the end of the critical path that originates from A0 input and terminates in sum15_bar output for the CSA that we have designed and was clearly explained in the earlier sections. Refer to Fig. 3.7 for the details. The presence of EDS at the output implies that if there are any late timing transitions happening for the sum15_bar signal, they will be detected by EDS and can be corrected by increasing the supply voltage or adjusting the frequency of operation.

One important aspect about the implementation of an EDS circuit that needs to be mentioned is that the datapath latch involved in the design of EDS has a delay associated with its operation, and care has to be taken such that there is sufficient time for the latch to react to the changes happening to the output signal of the design and then act accordingly. Hence, the latch design has to take this factor into consideration and also the duty cycle of the clock has to be adjusted as required.

The main idea of using EDS in our design was to check the effectiveness of EDS at very low voltages such as 0.4V that we have considered for our thesis, and also the lowest possible voltage that we can successfully scale down to and still employ EDS without any problems, if, in case it is working fine for 0.4V or even lower.

3.6.2 Simulations for Different Supply Voltages

The first thing that was done was to find out the critical path delay for a combinational circuit alone without employing the EDS design for various operating voltages and the values are tabulated in Table 3.8, and the trend of delay vs operating voltage is shown in Fig. 3.12.

Then, we introduced the EDS circuit into our design and then considered voltage pairs, a higher starting voltage and a lower voltage which is generally selected in a way that it is nearly 10% less than the higher voltage in most of the cases. If the behavior for this

Table 3.8: Worst case delays for CSA at different voltages for EDS

Operating voltage (V)	Worst case delay (ns)
1	0.2418
0.9	0.2765
0.8	0.333
0.7	0.434
0.6	0.641
0.59	0.674
0.58	0.712
0.57	0.75
0.56	0.8
0.55	0.852
0.5	1.238
0.45	2.0304
0.4	3.88
0.38	5.26
0.37	6.2
0.36	7.3
0.35	8.7
0.32	14.98
0.3	22

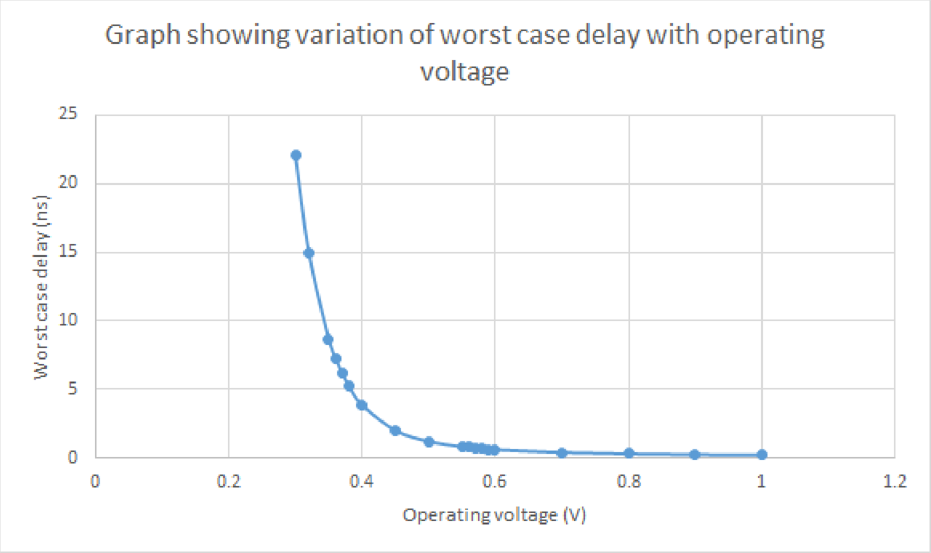


Figure 3.12: Worst case delay variation with operating voltage for CSA

voltage was not as expected, we increased the lower voltage further or adjusted clock frequency until the correct operation is ensured.

For example, let us consider a voltage pair: 0.4V and 0.35V. We already have the delays of critical paths for both voltages from the Table 3.8. The next major step is selecting a clock signal that should be applied to the EDS design. We have to select the clock in such a way that there is no error for the higher voltage (0.4V in the considered case) while there is an error which is detected for the lower voltage (0.35V in the above case) owing to the late signal transition.

As mentioned earlier, care has to be taken that the clock has to be adjusted so that there is sufficient time for the datapath latch to adjust to the transitioning Sum15_bar output signal so that the proper value is latched on. This adjustment gives us a clock period which can be used for both the voltages to get the desired functionality. As suggested earlier, if there was an issue with any of the above mentioned aspects, the lower voltage must be adjusted or clock period has to be adjusted accordingly.

3.6.3 Problems Encountered with EDS Design at Ultra Low Voltage

There were some problems that we theorized might be caused while simulating and working on this idea for ultra low voltages. This part discusses these problems in detail.

A problem that we thought would be prevalent was the transition of Sum15_bar signal which is the output for the critical path, does not happen immediately but happens in a very gradual manner, thus taking a lot of time to reach a low level when starting from a high voltage level. This large fall time associated with the data signal causes the wrong value to be latched by the datapath latch, even when the transition happens within the error detection window. In most of the cases, even when this signal starts transitioning from the start of the window, by the time it settles down to the final value, the slew is very high so the latch doesn't get a chance to reflect the changed value because of its own delay while dealing with the signal.

In order to solve this issue, the duty cycle has to be made higher by such an amount so that even after the transition of the output signal which happens inside the error detection window, there is sufficient time for the latch to know that the transition happens and then produces the correct value corresponding to the change. This ensures there is enough time for the latch to pick up the right value. But, a new problem due to the creation of short paths might come into picture if the duty cycle is too high. If there are a lot of short paths involved in the design, and a pipeline stage is employed where a single clock signal is applied to different flip flops involved in the design, because of the short paths created, the signal shoots through the entire pipeline stage hence disrupting the entire operation of EDS circuitry.

To check if this problem still persists in our design, we considered a few pairs of voltages, with the application of EDS for the CSA. When the higher voltage is employed, there is no error as data arrives early which suggests the proper operation of the design. When the lower voltage is considered, due to the increased delay involving the critical path, the flip flop and latch outputs differ, hence producing an error signal which is asserted high at the end of the error detection window for the worst case inputs as given in Figure 3.13.

Now, to check if the problem of signal shoot through exists for the lower voltages, we considered an input pattern where the 15th input bit pair (A14-B14) has a 1 1 associated with it. This means that a carry is generated by this bit pair itself and it propagates toward the MSB. This is shown in Fig. 3.13 where A[15:14] and B[15:14] are given patterns like 01 and 01 so that the 15th bit pair generates an output carry by itself. This path is a very short path as the carry has to travel only from 15th bit to the 16th bit position output and the delay associated with this path is very small. Because the delay associated is very small with this path, the Sum15_bar signal falls in the first clock high phase itself, as we thought, and since the duty cycle of the clock is high in order to detect error for lower voltages, the signal shoots through and asserts the error high even during the first clock high phase itself which confirms what we have theorized earlier about short paths is right.

The examples given in Figure 3.13 suggest that the carry skip adder circuit is a highly flexible circuit where the paths can be slow or fast depending on different input patterns applied unlike a normal combinational circuit. In a normal combinational design composed of lots of gates, we know clearly the distinction between long and short paths and hence analysis can be done easier, but in the case of CSA, a short path can also be a part of long path as they can share the same output as was presented earlier. If the same EDS is shared by both long and short paths in the design, we figured that it disrupts the functionality of EDS.

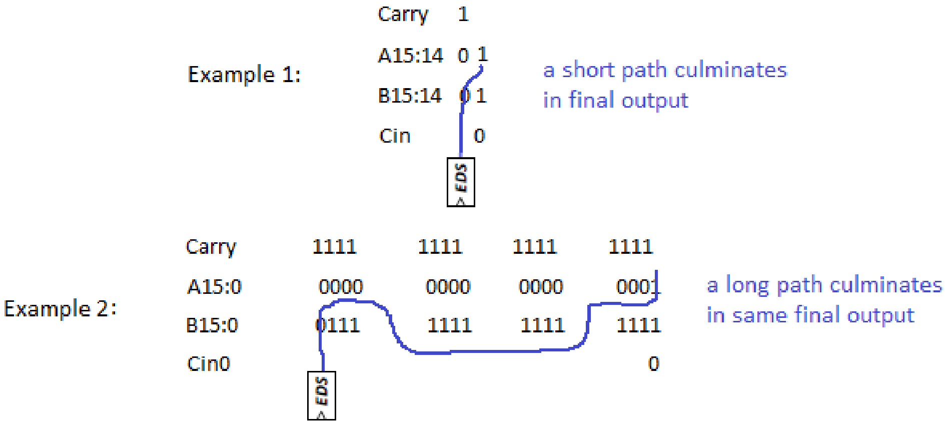

Example 1:
Carry 1
A15:14 0 1
B15:14 0 1
Cin 0
EDS
a short path culminates in final output
Example 2:
Carry 1111 1111 1111 1111
A15:0 0000 0000 0000 0001
B15:0 0111 1111 1111 1111
Cin0 0
EDS
a long path culminates in same final output

FIGURE 3.13: Example input patterns showing short and long paths culminating in same output having EDS depending on applied inputs

From these observations, we can deduce that it is very difficult to achieve further power savings by the application of EDS circuits in the ultra low voltage region. This deduction can be explained as follows: From Figure 3.12, we find that as we reduce the supply voltage below 0.4V, the delay begins to increase in an exponential manner. EDS circuitry employs datapath latch and a flip flop which contain logic gates that introduce a lot of capacitance in the design. With increased capacitance in the subthreshold region of operation, the effect on delay is increased further. As a result of this consequence, it is very difficult to adjust the clock period which helps to detect late timing errors for lower voltages, and at the same time not to produce the short path problem. This implies that ultra low voltage region is not good for EDS circuit design.

This observation also reiterates the fact that there is a need to use additional padding for the short paths, the use of which might increase the delay for some paths so that short path problem might be eliminated. But, it also increases the hardware and area required, but for further reduction in voltage, EDS might not be a very good option for reducing the power consumption as explained earlier.

3.6.4 Conclusion About EDS Design in Ultra Low Power Region

Based on the discussion which was presented earlier, we found that the design was already operating at a very low voltage of 0.4V. If we reduce the voltage further, the delay associated increases exponentially and power can be saved owing to the square dependence of power on the supply voltage. We expect that the usage of EDS circuit would facilitate further lowering of supply voltage, but with an increased capacitance and hence an increased delay at lower voltages, the ultra low power region design employing an EDS does not provide further opportunities for power reduction, unless we find a better method for the designs in this area of operation. Hence, it is very difficult to further lower the supply voltage to take advantage of EDS circuits in this region of operation with the current design employed.

The next chapter deals with power reduction techniques such as Power Gating and Clock Gating applied to designs that we have built.

Chapter 4

Power Reduction Techniques for Carry Skip Adder and Magnitude Comparator designs

The previous chapter dealt with details about the design and implementation of the circuits, followed by selecting appropriate operating voltage- model file pair for the designs for further analyses. This chapter deals with the use of power reduction techniques to reduce power consumption of the designs which is an important area of concern for modern electronic circuits. A brief introduction to Power Gating and Clock Gating is presented first, followed by application details of these techniques for our circuits. The simulation details are presented next followed by the observations in the end.

4.1 Importance of Power Reduction

When the complexity of a VLSI chip is increased, with the idea of performing more computations per the same area available, there is a problem associated with it. An increase in the complexity of a chip implies an increase in the number of components involved in the design. As the number of components involved in the design increases, the power consumed by the chip as a whole increases, resulting in excessive power leakage from a single chip. This leakage or energy loss is a situation which cannot be afforded by the manufacturers as that would mean additional costs being incurred on eliminating the problem by means of cooling, packaging and other related remedies. Consequently, the savings in resources achieved by increasing the complexity of a chip in the same area available is being lost in the form of additional unwanted aspects which is definitely not what is desired. Hence, reducing the power dissipation of VLSI circuits is increasingly becoming an important area that has to be taken care of by VLSI engineers.

4.2 Power Reduction Techniques

With more and more emphasis being laid on reducing power consumption for the designs, handheld and portable devices are becoming popular. Reducing the power dissipation of these devices is of utmost importance for surviving in the industry. The following sections discuss details about some of the widely used and relatively easy to implement power reduction techniques, their application to the circuits we selected in our thesis, simulations performed and results that we found pertaining to application of such techniques for very low voltages.

4.3 Power Gating

One of the prominent and widely used techniques for power reduction is power gating. We wanted to check if this technique can be effective at very low voltages and if it is, what are the savings that can be achieved through this technique. The first part deals with a very brief introduction to power gating followed by the implementation details in the next part. Simulations performed on our designs and observations are presented toward the end.

4.3.1 Application and Simulation Details of Power Gating for Our Designs

We used the footer transistor based power gating technique for our thesis where an NMOS footer is placed between the actual ground and a virtual ground terminal. The virtual ground node serves as the ground terminal to all NMOS transistors involved in the designs, the voltage level at which determines the operation of the design considered.

To measure leakage power consumption for a particular input pattern, we applied the pattern without any new pattern being applied during that time interval so that the output produced would remain constant for that time. We used the .measure statement which is a built-in feature provided by HSPICE as mentioned in the earlier chapter, to measure the average current flowing through the power source (iVdd) during this interval. We multiplied this value by the supply voltage (Vdd) to give the leakage power for that pattern. We repeated this procedure for different patterns to give different leakage power values.

4.3.2 Input Application Details for Power Gating

The input patterns that we applied to the designs for performing power gating on the designs were derived in a methodical manner. The following gives details about the input patterns derived for the CSA circuit.

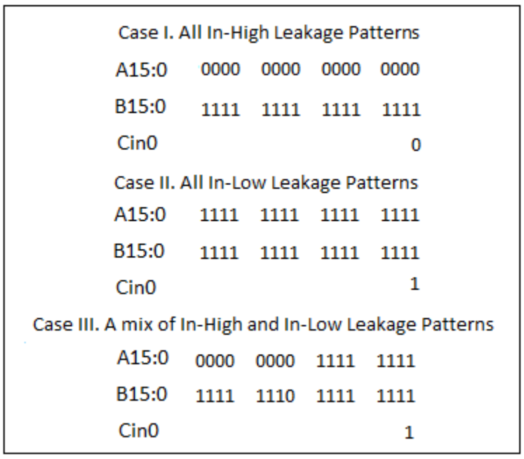
The first step that we performed was to consider a single FA cell consisting of XOR gates and also carry circuit for generating P, Sum and output carry signals respectively. Each FA cell has three inputs and hence there are eight input patterns associated with a FA cell.

We applied these eight patterns individually on the 1-bit CSA cell and measured the leakage power based on the method mentioned earlier for all the cases and are tabulated in Table 4.1. From these results for leakage power consumption of a FA cell, we found that patterns (A B Cin) 010 and 111 represent the cases which lead to highest and least leakage power for both operating voltages and are considered as In-High and In-Low leakage patterns respectively.

Table 4.1: Leakage power measurements for a single FA cell at 0.4V and 1V

Input pattern (A B Cin)	Leakage Power measured (nW)	
	0.4V	1V
000	17.064	538.5
001	15.912	500.4
010	19.192	615.5
011	17.824	601.2
100	16.044	510.2
101	14.316	464.1
110	16.572	505.9
111	12.868	413.2

Now, we extended this idea to the 16-bit CSA design where each bit was given different patterns corresponding to whether we require the bits to correspond to In-High, In-Low or a mix of both In-High and In-low leakage patterns, which are all represented in Figure 4.1.



Case I. All In-High Leakage Patterns

A15:0	0000	0000	0000	0000
B15:0	1111	1111	1111	1111
Cin0				0

Case II. All In-Low Leakage Patterns

A15:0	1111	1111	1111	1111
B15:0	1111	1111	1111	1111
Cin0				1

Case III. A mix of In-High and In-Low Leakage Patterns

A15:0	0000	0000	1111	1111
B15:0	1111	1110	1111	1111
Cin0				1

Figure 4.1: Input patterns applied to the CSA circuit for leakage power measurement and power gating

Once we fixed and applied the input patterns that were found out based on simulations stated earlier, we then measured leakage power consumption for the circuit after application of these patterns which would help us determine whether Low-leak patterns or High-leak patterns would save more power under power gating and these values measured are given in Table 4.2.

Table 4.2: Leakage power measurements for different cases for the 16-bit CSA for 0.4V and 1V

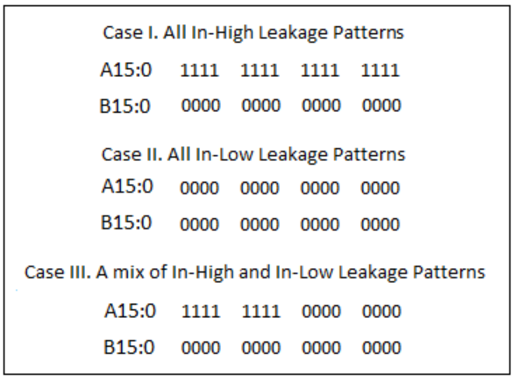
Case represented as in Figure 4.1	Input pattern (A[15:0] B[15:0] Cin)	Leakage Power measured (nW)	
		0.4V	1V
Case I	0000 0000 0000 0000 1111 1111 1111 1111 0	336.64	10990
Case II	1111 1111 1111 1111 1111 1111 1111 1111 1	235.52	7685
Case III	0000 0000 1111 1111 1111 1110 1111 1111 1	282.72	9191

The input patterns applied for the comparator design followed a similar approach which is mentioned next. First, we considered a single-bit comparator circuit consisting of INV, AND and NOR gates which has two inputs associated with it. There are a possible four input patterns for this 1-bit cell. We applied these four patterns and found out leakage power consumption using method the explained earlier and are presented in Table 4.3. From these values, we deduced that patterns (A B) 10 and 00 would produce the highest and least leakage power respectively for a 1-bit cell for both the operating voltages.

Table 4.3: Leakage power measurements for different cases for a 1-bit comparator cell for 0.4V and 1V

Input pattern (A B)	Leakage Power measured (nW)	
	0.4V	1V
00	2.1676	68.73
01	2.8736	96.67
10	3.074	110
11	2.8128	95.59

We extended this idea to the 16-bit comparator design and provided input patterns specified in Figure 4.2 based on the requirement of all In-High, all In-Low or a mix of both kinds of input leakage patterns.



Case I. All In-High Leakage Patterns

A15:0	1111	1111	1111	1111
B15:0	0000	0000	0000	0000

Case II. All In-Low Leakage Patterns

A15:0	0000	0000	0000	0000
B15:0	0000	0000	0000	0000

Case III. A mix of In-High and In-Low Leakage Patterns

A15:0	1111	1111	0000	0000
B15:0	0000	0000	0000	0000

FIGURE 4.2: Input patterns applied to the comparator circuit for leakage power measurement and power gating

We then measured the leakage power consumption values for the 16-bit comparator design for each of the patterns described above and are shown in Table 4.4. These values assist in determining which input pattern gives the greatest leakage power saving when simulations for power gating are done.

TABLE 4.4: Leakage power measurements for different cases for the 16-bit comparator for 0.4V

Case represented as in Figure 4.2	Input pattern (A[15:0] B[15:0])	Leakage Power measured (nW)	
		0.4V	1V
Case I	1111 1111 1111 1111 0000 0000 0000 0000	80.76	2882
Case II	0000 0000 0000 0000 0000 0000 0000 0000	78.76	2652
Case III	1111 1111 0000 0000 0000 0000 0000 0000	79.2	2718

From the tables showing power measurements at 0.4V and 1V for both the designs, we find that there is a pattern followed, in that, leakage power for In-high leakage patterns are the highest, followed by power for mixed in-high and in-low patterns. The least leakage power is observed when we applied in-low leakage patterns which is consistent with what we initially theorized. Also, leakage power at 1V is way higher than that at 0.4V for both the designs which is expected.

4.3.3 Procedure Followed During Simulations

While performing the simulations, the input patterns specified earlier were applied on the designs for a sufficient amount of time, around 30ns, without the use of power gating sleep transistors. Next, sleep transistor was introduced into the designs. When the sleep transistor is on, the design operates normally, now with an extra transistor in the discharge path. The outputs are produced normally as if there is no extra transistor included. When the sleep transistor is turned off, it offers a high resistance in the design which essentially cuts off the connection to the ground. The operation of the circuit under this condition is not very significant as we are interested in the savings obtained, if any, due to the disconnection with the ground terminal.

The size of the sleep transistor is an important factor to take into consideration. If the width is too high, operation of the design is guaranteed but as the resistance offered by it is less, the power saving that can be achieved by the use of sleep transistor is not very high. On the other hand, if the width is too low, the resistance offered will be very high. In this case, when the sleep transistor is turned on, there is a high voltage drop, thereby the circuit functionality cannot be guaranteed. Hence, we have to select a footer size so that it is not too big and also not too small at the same time.

We started with a width which is greater than 10% of the sum of widths of all NMOS transistors involved in the designs. At this width, when the sleep transistor is on, the design should operate normally and when it is off, it should reduce leakage. We further reduced the width and made it smaller. In doing this, we have to make sure the circuit functionality is verified and also reduction in leakage is accomplished. We kept shrinking the width until circuit malfunctions due to increased resistance. If at any stage the simulations were providing results that do not match with the expected behavior, it means the power gating technique is not working properly for very low voltages, the results of which are given in the observations section.

4.4 Observations

We have already established and explained about the input patterns and measured leakage power values for the designs when those patterns were applied in the previous section. One standard that we made for power gating in general for our thesis was the voltage level which is considered acceptable as far as normal working is considered. Generally, in a design, in order to have good noise margins and for reliability purposes, the logic low signal should not be larger than 0.2*Vdd and logic high should not be less than 0.8*Vdd. Since we are dealing with NMOS transistors predominantly for footer-based power gating, we are interested in voltage levels near GND or 0 potential. If we consider an operating voltage of 0.4V, the lower potential level which is acceptable for virtual ground terminal is smaller than 0.2*Vdd=80mV. That means, when a signal has turned low, its level should always be less than 80mV to consider it a reliable measurement. When working at 1V, this voltage level is 0.2V, which is 20% of Vdd. The virtual ground terminal should not be more than this voltage when the footer is off.

4.4.1 Simulations for CSA

We started with a single FA cell, for which we had to use a moderately low Vth footer cell instead of a high Vth one. We started with a width of 900nm which is just above 10% of total NMOS width (7650nm) in a single FA cell. It did not produce any savings when footer was turned off as the virtual ground potential was being higher than 80mV, as established earlier. We increased the width to 1800nm and it was saving power, with the virtual ground potential also within the permissible level. We then reduced the width to see when there is a change in the behavior, the details of which are given in Table 4.5.

From Table 4.5, we observe that as the width of the footer transistor decreases, % saving increases which is as expected. The width when Vss1 is reasonable and also saving is decent is around 1725nm for 0.4V case. When we further reduce the width, the virtual ground voltage would be greater than 80mV, the upper bound in the virtual ground

Table 4.5: Power gating results for 1 bit FA cell at 0.4V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (mV)	Leakage Power (nW)	
1800	14.023	12.868	76.7	9.332	27.5
1750	14.42	12.868	82.7	9.22	28.3
1725	14.634	12.868	81.1	9.164	28.8
1700	14.85	12.868	82.7	9.104	29.3

voltage, and it induces an excessive current flow from the virtual ground terminal to the actual ground, hence reliability would not be guaranteed if lower widths are used.

While the same experiment was being performed for 1V, we started with a width of 2700nm. We found that virtual ground terminal potential is around 0.33V which is very high. So, we kept increasing the width and reached a width of 9000nm when Vss1 was around 0.19V. This voltage was close to 0.2V we were considering as the limit. Some other cases were considered as well, all of which are given in Table 4.6.

For low widths, Vss1 was being very high and also leakage power when footer is off was very high. When we increased the width very much, Vss1 reduces and becomes too low. A balance is seen such that the width is not too high, and also savings and Vss1 were reasonable, and the width which gives this condition is around 8000nm. But, this width for a single bit FA cell is very high (greater than sum of all NMOS widths). If we have 16 FA cells, the width needed would be extremely high. We also observed that footer size required was smaller at 0.4V than at 1V implying power gating works well even at ultra low voltages. The % savings were higher at 1V than at 0.4V.

We performed simulations on the 16-bit CSA for 0.4V by starting with a width of 15000nm which is higher than 10% of the sum of widths of all NMOS transistors (120600nm). We observed that Vss1 for this width was very high at 0.192V. Hence we had to increase the width to 90000nm and saw that Vss1 was now 10mV, which is extremely low. Hence, we had to reduce it further and found that around 40000nm,

Table 4.6: Power gating results for 1 bit FA cell at 1V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (V)	Leakage Power (nW)	
9000	15.536	413.1	0.187	183.6	55.6
8000	17.438	413.1	0.202	172.8	58.2
7750	17.99	413.1	0.206	169.9	58.9
7500	18.579	413.1	0.21	167.1	59.5

it was around 45mV. We considered several other widths, all of which are represented in Table 4.7. From the table, we observe that as the footer width decreases, % saving increases which is expected. The width of 27500nm presents a good case as Vss1 is within acceptable limits, and also decent savings are achieved.

Table 4.7: Power gating results for 16 bit CSA at 0.4V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (mV)	Leakage Power (nW)	
40000	9.9	235.5	46	189.76	19.4
30000	13.16	235.5	69.8	169.92	27.8
27500	14.4	235.5	78	163.28	30.7
26000	15.2	235.5	84	159	32.5
25000	15.8	235.5	88	156	33.8

For 1V, the starting width itself should be very high as found from simulations for the 1-bit cell. Hence, we selected a width of 135000nm which is higher than sum of widths of all NMOS transistors (120600nm). We observed that at this width, Vss1 was around 0.19V which is acceptable, but the savings was slightly low. We then varied the width and observed that around a width of 115000nm, Vss1 was around 0.21V, which is slightly higher than 0.2V margin. After increasing the width, we observed there were savings for larger values of widths. All the cases are represented in Table 4.8. The table suggests that there are savings when a large width is employed for the footer. Also, when the footer width decreases, % saving increases which is as expected. A width of 125000nm presents a good case as savings are decent and Vss1 is acceptable. This indicates power gating is effective even at ultra low voltages for a CSA. Also, the savings obtained for 1V is higher than that at 0.4V supply voltage. The graphs showing % savings for 16-bit CSA design at 0.4V and 1V are given in Figure 4.3.

Table 4.8: Power gating results for 16 bit CSA at 1V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (V)	Leakage Power (nW)	
135000	16.23	7685	0.223	2806	58.3
130000	16.8	7685	0.1972	3145	59.1
125000	17.5	7685	0.201	3079	59.9
115000	19	7685	0.212	2946	61.7
105000	20.8	7685	0.223	2806	63.5

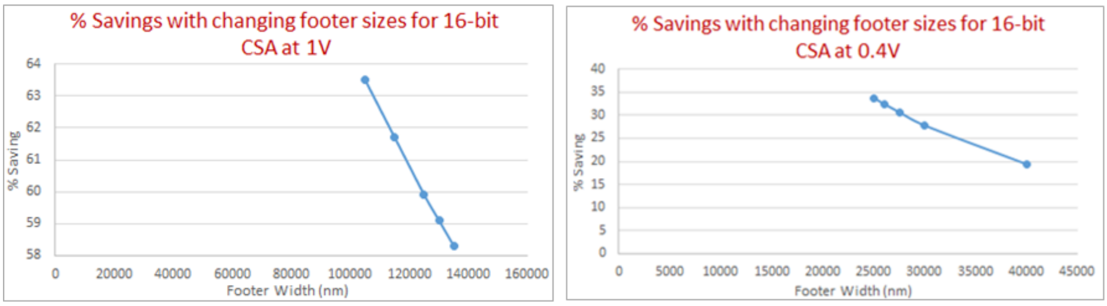


Figure 4.3: % Savings with changing footer sizes for 16-bit CSA

The simulations for a single bit comparator design were performed for 0.4V, first by starting with a footer width of 90nm, which is 10% of sum of widths of all NMOS transistors in the 1-bit comparator (900nm). For this width, Vss1 was being very high, around 0.22V. Therefore, we had to increase the width to see where it is reasonably working well. When we increased the width to 900nm, Vss1 was around 17mV which is too low. This indicates that the width has to reduced. Hence, we reduced width to 540nm and then to 270nm and observed that the latter produces a Vss1 of 85mV which is around the acceptable virtual ground terminal voltage, all of which are represented in

Table 4.9. The interesting thing is that even when Vss1 is very low, there are savings achieved. From the table, we see that as the footer width decreases, % saving increases. But, for higher widths, Vss1 is too low and only when you reach a width of 270nm will Vss1 reach around acceptable limit which can be considered a good result.

TABLE 4.9: Power gating results for 1 bit comparator at 0.4V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (mV)	Leakage Power (nW)	
1350	3	3.074	10	2.8788	6.4
900	5.8	3.074	17	2.7776	9.6
540	7.75	3.074	32	2.558	16.8
270	15.7	3.074	85	1.9588	36.3

While considering power gating at 1V, we started with a width of 1350nm and found that Vss1 was around 0.2V which is good and was saving power. We reduced width and observed that Vss1 was being too high for widths around 450nm and below, but even for these widths, there were savings achieved which is interesting, all of which are represented in Table 4.10. Even though savings were achieved, all these widths below 1250nm represent cases when Vss1 is too high, hence cannot be considered a reliable measurement. From the table, we observe that even at 1V, as the footer width decreases, % saving increases, but the problem lies with Vss1 voltage. Only 1350nm width corresponds to a good case as Vss1 voltage is around the acceptable margin.

For performing power gating on the 16-bit comparator design, for 0.4V, we considered width of 3500nm, which is just above 10% of sum of widths of all NMOS transistors involved in the design (32580nm) and observed that Vss1 was around 0.2 Volts which is slightly high. We first increased the width to 17000nm and found that Vss1 was too low. Hence, we had to decrease the width and found that around 7500nm, Vss1 was around 78mV which is acceptable. The results are shown in Table 4.11. From the table, we observe that as the footer width decreases, % saving increases. The problem with higher widths was that Vss1 was too low, and hence 7500nm represents a proper case

Table 4.10: Power gating results for 1 bit comparator at 1V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (V)	Leakage Power (nW)	
1350	18.462	110	0.2086	42.91	61
1250	21	110	0.218	41.24	62.5
1200	23	110	0.22	40.38	63.3
900	29	110	0.26	34.83	68.3
450	55	110	0.345	24.44	77.8

for savings. We should not reduce below 7500nm as Vss1 would be greater than 80mV in that case.

Table 4.11: Power gating results for 16 bit comparator at 0.4V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (mV)	Leakage Power (nW)	
17000	6.39	80.76	25	70.52	12.7
16500	6.58	80.76	26	70.2	13.1
9000	12	80.76	61	59.08	26.8
7500	14.5	80.76	78	54.36	32.7

For the simulations at 1V, we started with a width of 10000nm which is one-third of sum of widths of all NMOS transistors in the comparator. We found that for this width, Vss1 was around 0.4V which is high. Hence, we increased the width to 40000nm and found Vss1 was almost close to 0.2V. We considered a few more widths, all of which are represented in Table 4.12. From the table, we observed that as the footer width decreases, % saving increases. The width of 37500nm can be considered a good case as Vss1 is very close to 0.2V as required and also savings were decent. But, one point to be noted is that this value of width is higher than sum of widths of all NMOS transistors involved in the design which is a big number. Finally, % savings was higher for 1V than at 0.4V for the comparator design as well. The graphs showing % savings for 16-bit comparator design at 0.4V and 1V are given in Figure 4.4.

Table 4.12: Power gating results for 16 bit comparator at 1V

Width of footer transistor (nm)	Footer transistor ON		Footer transistor OFF		% Savings
	Virtual ground terminal voltage (uV)	Leakage Power (nW)	Virtual ground terminal voltage (V)	Leakage Power (nW)	
50000	13.2	2882	0.168	1341	53.5
40000	16.4	2882	0.195	1193	58.6
37500	17.45	2882	0.203	1154	59.9
30000	21.7	2882	0.23	1027	64.4

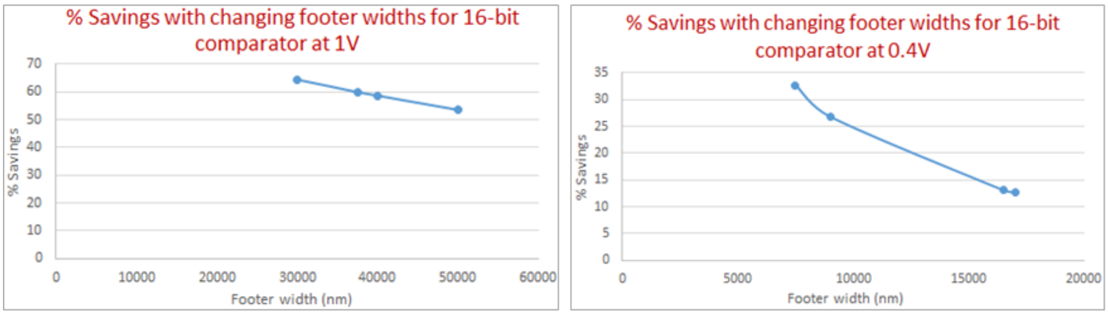


Figure 4.4: % Savings with changing footer sizes for 16-bit comparator

From all the above tables and graphs showing results for the power gating technique on both the designs, we see that power gating technique seems to be working fine and the savings at 1V are higher than those obtained at 0.4V for both the designs. The interesting thing, though is that the width of footer is lower in ultra low voltage region than at normal operating voltage of 1V. Also, as width of the footer decreases, % saving increases which is as expected for both the designs at both the supply voltages. Finally, for the CSA design, % saving was more sensitive to the footer width at 1V than at 0.4V, and for the comparator design, % saving was more sensitive to the footer width at 0.4V than at 1V supply voltage.

4.5 Clock Gating

After checking the effectiveness of Power Gating at very low voltages, we wanted to see if Clock Gating saves power at such low voltages and if it does, what is the comparison to savings at a higher voltage. The first part deals with general details about Clock Gating technique followed by the technique we used in our thesis. Simulations performed on our designs is presented next followed by observations in the end.

4.5.1 Application and Simulation Details of Clock Gating for Our Designs

The first factor that needed attention for clock gating technique to be applied was that the designs we considered are pure combinational designs having a bunch of combinational gates. In order to apply clock gating, we had to transform these circuits into designs which have sequential elements like flip flops working on the edges of a clock, or latches, which are level sensitive circuits, thereby providing an opportunity to control the activity of the circuits based on the application of proper controlling signals. Hence, we placed latches at the sending and receiving ends of the designs so that inputs are applied through the latches, and are then acted upon by the combinational circuit elements to produce the desired outputs. These outputs are then passed through the latches at the receiving end to get the final adjusted outputs. This is shown in Figure 4.5.

The latch we used is a positive level sensitive design with a clock signal which controls when to allow the signals to pass through and is shown in Figure 4.6. This is a simple design used predominantly in VLSI systems and contains only ten transistors which is easy to implement in HSPICE. Also, this circuit consumes less power when compared with other latch or flip flop designs. Hence, usage of this circuit is advantageous for our thesis. One important consideration that has to be taken into account concerns the sizing of the transistors employed in the latch design. In Figure 4.6, transistors M7 and M8 must be sized in such a way that they overpower the transistor M4 and help in bringing Q from 1 to 0 properly. This sizing is important to reduce Q to less than the threshold of M2 and M1 transistors. Proper sizing is done, considering the fact that the mobility of NMOS transistors is nearly 2-3 times that of PMOS transistors.

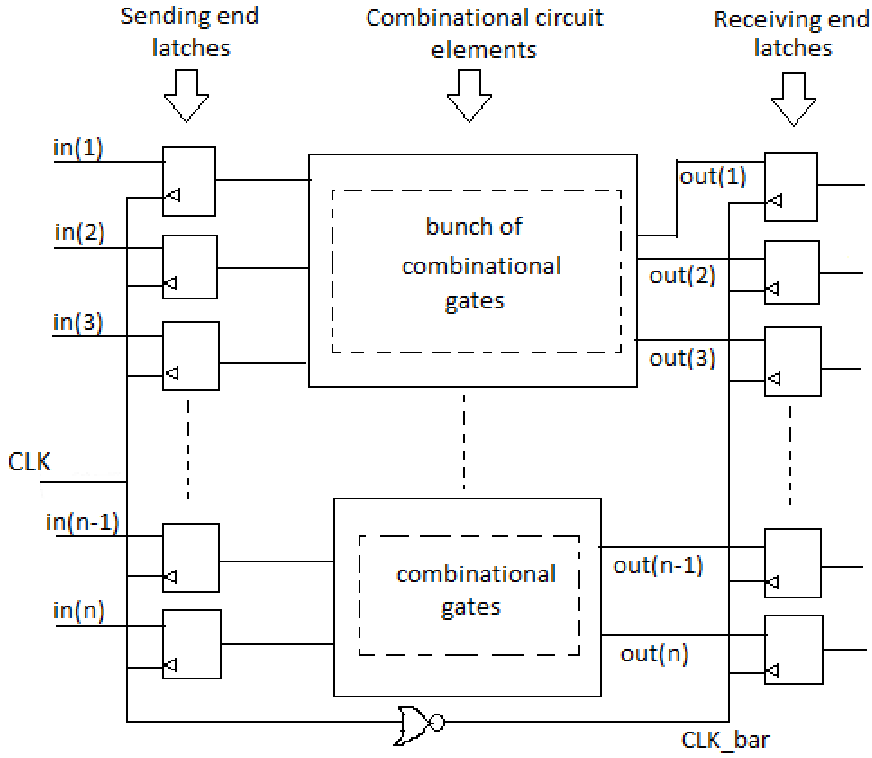


Figure 4.5: Combinational circuit transformed into a sequential circuit with help of latches

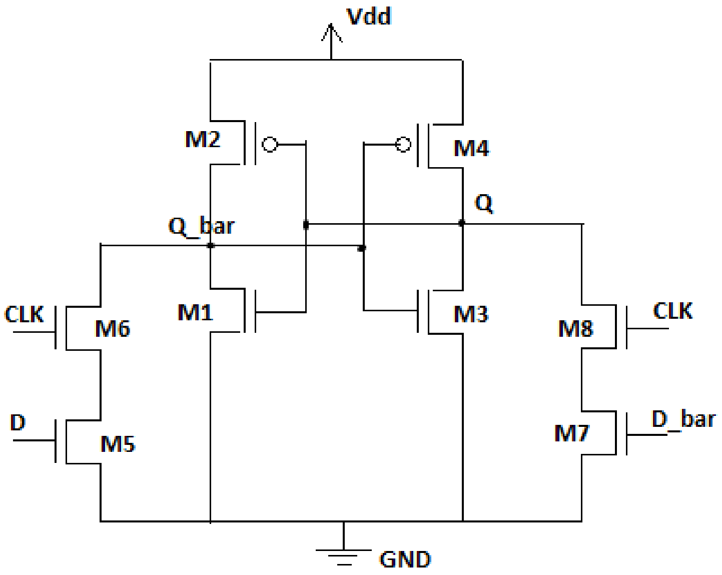


Figure 4.6: Latch used for clock gating

One important factor that needs to be taken into account with the usage of latches is to ensure that the data remains constant throughout the period when clock is on, in other words, the datum has to change only after the positive phase of the clock has ended, and not during the clock high phase. If it changes during clock high phase, there is a possibility that the value would not be latched owing to the delay associated with its operation and also delay associated with the design itself. Also, the clock signal given to the input side latches and the output side latches should not overlap each other. Hence, we used an inverter to generate an out of phase clock signal to apply for the latches at the output side as given in Figure 4.5.

This ensures that during the time when clock is high, the inputs are latched on by the latches at the input side, and when the clock turns low, the data which are latched remain constant, so that the proper values are captured by the latches at the receiving end. Finally, the inputs and also the enable signal which is employed for clock gating purpose should not change exactly at clock edges but it has to make sure that the proper values are seen during the high phase of the clock, which ensures proper operation of the designs. So, it is important to make them remain constant until the high phase of clock is done, and then change during the low phase of clock which would avoid wrong values being latched on by the designs.

The technique we used in our thesis is AND-gate based Clock Gating where a simple AND gate is employed to accomplish gated clock signal as given in Figure 4.7. Instead of the normal clock signal applied to the latches as depicted in Figure 4.5, Figure 4.7 shows a gated clock signal generated through an AND gate. This signal is used as the subsequent clock signal to all the latches involved in the designs, thus enabling the circuit to be switched off when desired, based on the enable input.

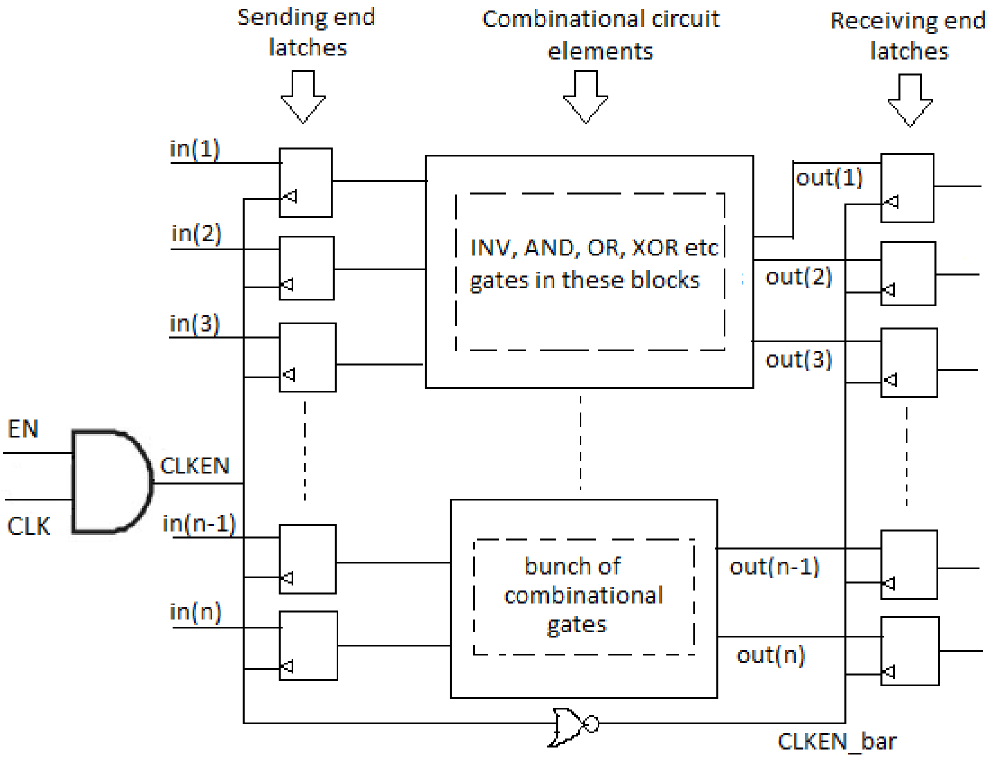


Figure 4.7: AND-gate based Clock Gating technique

This method is simple both in terms of the logic required and also the resources needed as we only need to implement a 2-input AND gate. One input to the AND gate is the clock signal and the other input is enable, which is used to control the output by means of controlling the clock to the sequential circuit. This enable signal basically enables the circuit at selected intervals of time by means of providing the opportunity to fully control the signal. There is no signal synchronization problem as a result of independently controlling the enable signal, hence preventing conditions like glitches in the system. This makes the technique even more attractive for our thesis.

Figures 4.8 and 4.9 show the clock gating technique for the magnitude comparator and carry skip adder circuits by inserting an AND gate. As the enable signal can be controlled independently, we make sure that it is changed during the low phase or after the falling edge of the clock in case of positive level-sensitive latch system instead of during the positive phase to avoid any glitches occurring in the system due to problems with signal synchronization as explained earlier.

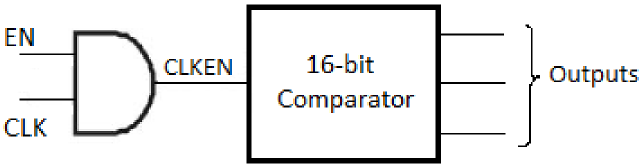


FIGURE 4.8: Clock gating technique for magnitude comparator

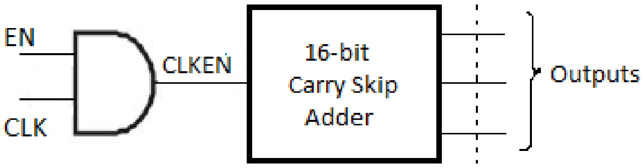
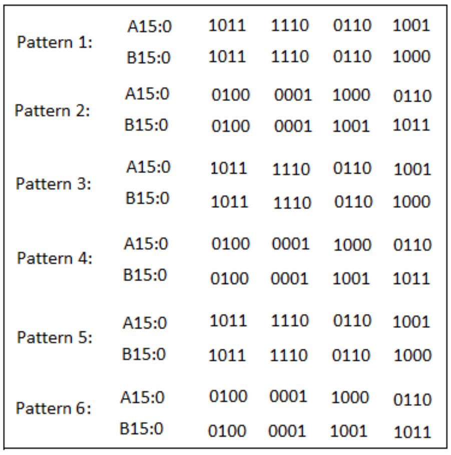


FIGURE 4.9: Clock gating technique for Carry skip adder

4.5.2 Input Application Details for Clock Gating

The amount of time the clock signal is turned off (i.e. off period of clock) can be adjusted by selecting the enable input accordingly. After properly providing the enable input, off periods of 20, 40, 60, 80 and 100% were achieved for the CSA design and 33, 66 and 100% were achieved for the comparator design. These conditions imply that clock signal is turned off for percentages of enable signal time mentioned above. For example, consider a case where the clock signal has a total on and off time intervals of ten units, each of five intervals. A 60% off time would mean that the enable signal is off for six time intervals and on for four time intervals. Similarly, the explanation holds good for other cases too.

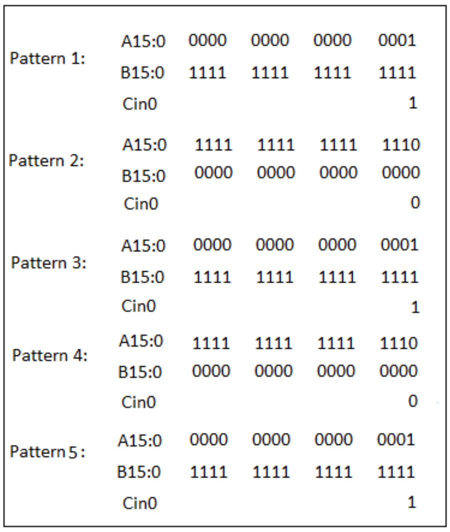
The input patterns that were applied to the comparator circuit which would cause a reasonably high switching and hence high power consumption for the designs are presented in Figure 4.10. Each of the patterns represented in the figure is applied to the comparator circuit for a certain amount of time, around 20ns, so that six different patterns are applied throughout the entire simulation period of 120ns. This is done so that the circuit switches while changing from one input pattern to the other and also switches while performing the required computations. The patterns applied make sure that the computations of the result have to wait until lower order bits change, after passing through the earlier stages asserting a equal to b (a=b) signals. There might be repetitions of the applied patterns but this ensures that there is a reasonable amount of switching happening, though this might not be the best case that gives the maximum savings. The built-in .measure statement is used to find the average power consumed during the entire simulation time.



Pattern 1:	A15:0	1011	1110	0110	1001
	B15:0	1011	1110	0110	1000
Pattern 2:	A15:0	0100	0001	1000	0110
	B15:0	0100	0001	1001	1011
Pattern 3:	A15:0	1011	1110	0110	1001
	B15:0	1011	1110	0110	1000
Pattern 4:	A15:0	0100	0001	1000	0110
	B15:0	0100	0001	1001	1011
Pattern 5:	A15:0	1011	1110	0110	1001
	B15:0	1011	1110	0110	1000
Pattern 6:	A15:0	0100	0001	1000	0110
	B15:0	0100	0001	1001	1011

FIGURE 4.10: Input patterns applied to the comparator for switching power measurement and clock gating

The input patterns that were applied to the CSA circuit are presented in Figure 4.11 which creates a high switching activity. In Figure 4.11, the odd numbered patterns applied make sure that there is high switching, in terms of asserting carry output signal of the design high by propagating the carry all the way from LSB to MSB. During this process, all the sum outputs are being asserted low. When the even numbered patterns are concerned in Figure 4.11, they are applied in such a way that switching is maximized, in terms of guaranteeing maximum switching during even to odd or vice versa transitions, by making the final carry output to be deasserted, and asserting all the intermediate sum bits. This ensures each of the outputs and also input bits have switched from their previous values. This process is continued which would ensure a high switching activity design.



Pattern 1:	A15:0	0000	0000	0000	0001
	B15:0	1111	1111	1111	1111
	Cin0				1
Pattern 2:	A15:0	1111	1111	1111	1110
	B15:0	0000	0000	0000	0000
	Cin0				0
Pattern 3:	A15:0	0000	0000	0000	0001
	B15:0	1111	1111	1111	1111
	Cin0				1
Pattern 4:	A15:0	1111	1111	1111	1110
	B15:0	0000	0000	0000	0000
	Cin0				0
Pattern 5 :	A15:0	0000	0000	0000	0001
	B15:0	1111	1111	1111	1111
	Cin0				1

FIGURE 4.11: Input patterns applied to the CSA for switching power measurement and clock gating

Also, by turning off the clock at different time intervals, i.e., by adjusting the enable input in such a way that the clock is shut off at different time intervals and during times when different input patterns are applied, different cases were achieved all of which are presented in Figures 4.12 and 4.13.

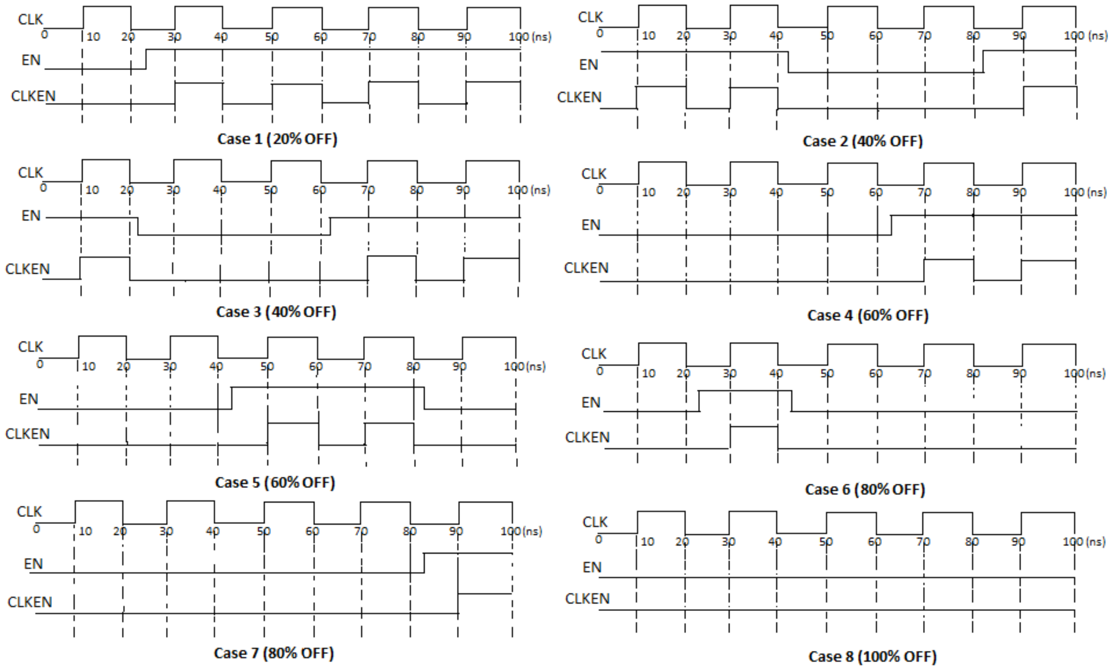
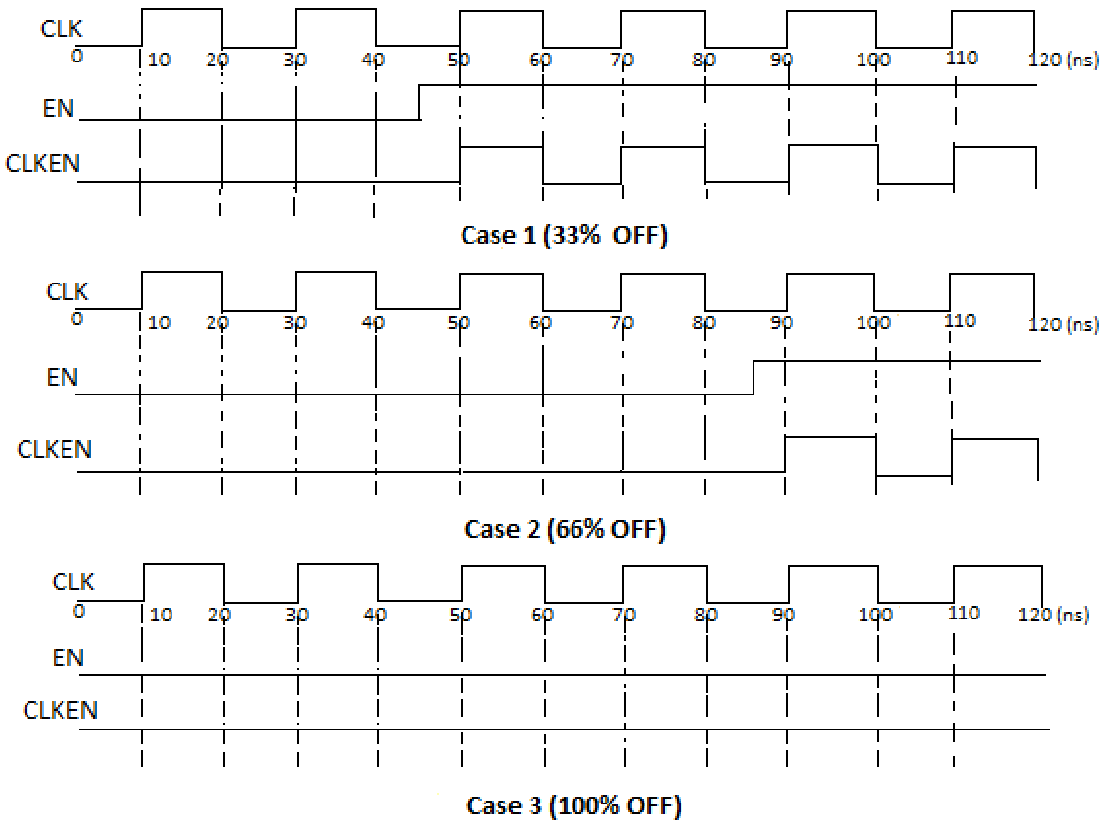


FIGURE 4.12: Different cases considered for clock gating for CSA



Case 1 (33% OFF)

Case 2 (66% OFF)

Case 3 (100% OFF)

FIGURE 4.13: Different cases considered for clock gating for comparator

4.5.3 Procedure Followed During Simulations

To perform the simulations, the designs were first analyzed for power consumption using methods discussed in the earlier chapter without employing any latches or other sequential elements. Then, the power measurements were repeated for the circuits, now with latches inserted into the designs to transform the designs into sequential circuits. As expected, power with the latches included is higher than that without the latches for the designs. This is because the latches are one of the elements that consume a major portion of the power supplied as a result of switching activities happening at these elements due to toggling of the clock signal continuously. Also, the usage of latches introduces a number of additional transistors into the design which switch continuously based on the inputs and hence consume power.

The next set of simulations were performed under all the conditions mentioned in Figures 4.12 and 4.13 to perform clock gating for the circuits and observe the savings that were achieved. The inputs for CSA were applied in this way: Pattern 1 inputs given in Figure

4.11 were applied for 23ns, then pattern 2 inputs were applied for the next 20ns and so on. The results for savings achieved are tabulated in Tables 4.13 and 4.14 for the comparator and carry skip adder circuits for 0.4V. In each of the tables from 4.13 - 4.16, the first column has the value of the average power measured when latches are included at the sending and receiving ends of the designs considered. These latches at the input and the output ends operate at the clock and the inverted clock signals to provide out of phase configuration as explained earlier. The second column presents different cases for generating the gated clock signal as given in Figures 4.12 and 4.13. The third column gives the values of the power consumed by the designs with the latches included at the sending and receiving ends, now with the gated clock and its associated inverted signals applied, instead of the normal clock signal. The last column presents the percentage savings obtained upon the application of different gated clock signals on the designs.

Table 4.13: Power savings for comparator with clock gating at 0.4V

Power consumed without Clock Gating (uW)	Clock OFF period(% time off)	Power consumed with Clock Gating (uW)	Percentage Savings(%)
1.452	33	3.229	Increase
	66	1.592	Increase
	100	0.3026	79.1

Table 4.14: Power savings for CSA with clock gating at 0.4V

Power consumed without Clock Gating (uW)	Clock OFF period(% time off)	Power consumed with Clock Gating (uW)	Percentage Savings(%)
4.481	20	6.594	Increase
	40	5.237	Increase
	40	5.235	Increase
	60	2.855	36.3
	60	3.516	21.5
	80	1.185	73.6
	80	1.64	63.4
	100	0.4075	90.9

Finally, to test the effectiveness and also to compare the results obtained at 0.4V after applying clock gating technique, simulations were also performed at 1V for all the situations given earlier and are tabulated in Tables 4.15 and 4.16.

Table 4.15: Power savings for comparator with clock gating at 1V

Power consumed without Clock Gating (uW)	Clock OFF period(% time off)	Power consumed with Clock Gating (uW)	Percentage Savings(%)
18.83	33	19.15	Increase
	66	11.96	36.5
	100	6.133	67.4

Table 4.16: Power savings for CSA with clock gating at 1V

Power consumed without Clock Gating (uW)	Clock OFF period(% time off)	Power consumed with Clock Gating (uW)	Percentage Savings(%)
37.57	20	39.7	Increase
	40	32.79	12.7
	40	32.75	12.8
	60	19.39	48.4
	60	23.2	38.2
	80	11.09	70.5
	80	12.94	65.6
	100	6.605	82.4

4.6 Observations

Based on tables for the clock gating technique applied for the CSA at both 0.4V and 1V for the applied high switching patterns, we found that as the percentage of off time increases, or as period of inactivity due to clock being turned off increases, power savings increase which is as expected. When we shut the clock off to the latches, toggling and hence switching activity is reduced considerably, thereby reducing power consumption.

But, if the shut off interval is not selected properly or is too small, there might be an overhead associated with this method. This is confirmed by our measurements for 0.4V for the low off period cases considered, during which the power is being increased. One of the reasons for this might be that even though there are one or two periods of inactivity due to the clock being turned off, the overall time for which the clock is operational is still very close to the original case when there is no gated clock available. This might suggest that the switching has reduced, but ever so slightly. The problem lies in the fact that the small amount of saving that we get at such low off periods is being offset by the power consumed by the circuitry added to generate the gated clock signals.

The main reason of power consumption being slightly higher after applying gated clock design is because of the AND gate employed. This gate was sized in such a way that the gated clock signal was produced without a large delay and it is not skewed too much, otherwise it could cause the wrong output to be asserted by the output side latches. The width of PMOS transistor for the AND gate is made 20 times the minimum feature size of 45nm, and for NMOS, the width was 10 times the minimum feature size, thus maintaining a ratio of PMOS to NMOS widths of two, which accounts for the mobility difference. Thus, the AND gate consumes power, and is higher than the savings that can be achieved due to reduced switching of the design. This pattern is observed in the case of comparator as well for 0.4V. This suggests that at very low voltages, for low periods of inactivity, there is a trade off as far as the power reduction is concerned.

Another reason for this kind of behavior in the CSA design might be because of the T-gate design of the circuit and also the patterns that are acting on the design during the time the clock is applied. The time each pattern operates on the design might be very high so that it can cause significant switching and hence, there might not be tremendous savings when low off periods are considered. This is confirmed by the fact that when the period of inactivity is increased, there is higher savings which is due to clock being off for majority of the time. Also, at very low voltages such as 0.4V, the switching due to constantly changing inputs and outputs poses a problem of constantly charging and discharging the capacitances involved in the design. As this process of charging and discharging takes longer time at very low voltages, there might be an issue with power saving at low off periods.

The maximum savings were obtained at 100% off period of the clock for 0.4V and 1V for carry skip adder circuit, upon the application of the patterns considered, and were 91% and 82% respectively which confirm clock gating method is working effectively even at very low voltages. The savings obtained at 0.4V were higher than that obtained at 1V when high off period cases are considered. The reason for this behavior can be explained as follows: based on the inputs applied, the P signals are computed, and then P* signals are ready for each block. Since the inputs are chosen in such a way that each block apart from the first block is skipped as P*=1 for each corresponding MUX, the carry output is almost readily available, which does not change as the simulation progresses due to less or negligible switching activity. Thus, there is not much effect on T-gate based circuits when there is less switching, as capacitance charging and discharging problem would not be there. This results in higher savings when higher off periods are considered for 0.4V, which is not the case when slightly lower off periods are considered as can be inferred from Tables 4.14 and 4.16.

When high switching activity patterns are considered, each time the inputs change, the outputs also change, which means a lot of activity is involved with charging and discharging various intrinsic capacitances involved in the design. The internal of each stage designed is composed of complex gates like transmission-gate based XOR and MUX circuits which are designed to operate efficiently in ultra low voltage region to save power, which is typically in voltages less than 0.5V. The patterns considered are such that the carry propagates through MUX, and XOR circuits for carry propagation from LSB to MSB, which have T-gates in them. This might cause the behavior to slightly differ from the original.

When we consider the comparator design, the maximum savings obtained at 0.4V and 1V are 79 and 67% respectively. Even in this case, savings were higher for 0.4V than at 1V, but only when 100% off time of the clock was considered, implying that clock gating is effective even at very low voltages. As the percentage of off time increases, the savings increase for 1V as expected. This design performs well under clock gating for 1V as it does not have T-gate based XOR and MUX circuits which might inhibit the savings at higher voltages. It just contains normal AND, OR, NOR and other combinational gates and hence shows improvement even in higher voltages.

As it was presented earlier, even for the comparator design, at low off periods of the clock, the small amount of savings achieved due to the gated clock signal is offset by the power consumed by the AND gate circuit and hence shows slightly higher power for these cases. Further, to confirm this result, we considered another case where we switched the clock off for only 10% time interval, and observed that power for both the designs at both the voltages was increased, proving that the AND circuitry is the reason for this additional power consumption at low off periods.

The next chapter deals with the future research that can be performed or extended based on our work and some of the unanswered questions that can be considered as a starting point, which require further analysis.

Chapter 5

Conclusions and Future Work

One of the methods which significantly help in reducing power consumption of the designs is to operate at very low voltages. The idea of power reduction is gaining popularity significantly due to the present day technologies consuming more power when dimensions are shrunk. This factor, coupled with the advent of gadgets which strive to acquire good power savings and enhanced battery life at very low voltages, has been the basis for our research.

We have seen how the designs can be operated at very low voltages, aided by circuit modifications, in terms of employing gates and components which respond well in such low voltages, and also by careful selection of proper model files, which also helps the performance aspects of the designs.

Reducing the power consumption further at such low voltages is an important area of focus for modern day integrated circuits. This was the driving force for us to employ power reduction techniques and EDS circuits to check their functionality at such low voltages.

The idea of replacing constant block sizes which contain a fixed number of cells per stage to a design containing variable block sizes and variable cells per stage can be considered a good starting point for future work related to this thesis. Also, a single level skip circuit was employed in our work, but the behavior can also be observed when the design is extended to introduce multi-level skip circuits. Simulations can be performed manually, instead of mathematical analysis which was previously established by works performed using various designs given in [25] [22] [21] [24]. Different cases for the configuration can be considered and a relation can be established for the worst case delay based on the number of cells per stage and also on number of stages through these simulations.

Various other power reduction techniques such as Multi Voltage Design and Multi-Vth optimization can be checked for effectiveness on the designs considered in our thesis to see if they present any further improvements. The testability analysis of such designs is another important aspect that can be investigated. With the present day industry focused on minimizing power consumption and with shrinking feature sizes as already mentioned, the above specified work presents several challenges which could pave way for some interesting results.

Bibliography

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Proceedings of the IEEE*, vol. 86, pp. 82–85, January 1998.

[2] R. H. Reuss and M. Fritze, "Introduction to special issue on circuit technology for ULP," *Proceedings of the IEEE*, vol. 98, pp. 139–143, February 2010.

[3] M. B. Henry and L. Nazhandali, "Design techniques for functional-unit power gating in the ultra-low-voltage region," *IEEE*, pp. 609–614, 2012.

[4] M. Seok, S. Hanson, D. Sylvester, and D. Blaauw, "Analysis and optimization of sleep modes in subthreshold circuit design," *Analog IC Signal Proc.*, vol. 8, pp. 83–114, July 1995.

[5] K. K. Kim, H. Nan, and K. Choi, "Power gating for ultra-low voltage nanometer ICs." University of Michigan, Ann Arbor.

[6] M. Saint-Laurent and A. Datta, "A low-power clock gating cell optimized for low-voltage operation in a 45-nm technology," pp. 159–163.

[7] L. Li, W. Wang, K. Choi, S. Park, and M.-K. Chung, "SeSCG: Selective sequential clock gating for ultra-low-power multimedia mobile processor design," *IEEE*, 2010.

[8] H. Mahmoodi, V. Tirumalashetty, M. Cooke, and K. Roy, "Ultra low-power clocking scheme using energy recovery and clock gating," *IEEE Transactions on VLSI systems*, vol. 17, pp. 33–44, January 2009.

[9] L. Chang, D. J. Frank, R. K. Montoye, S. J. Koester, B. L. Ji, P. W. Coteus, R. H. Dennard, and W. Haensch, "Practical strategies for power-efficient computing technologies," *Proceedings of the IEEE*, vol. 98, pp. 215–236, February 2010.

[10] M. Dejan, C. C. Wang, L. P. Alarcon, T.-T. Liu, and J. M. Rabaey, “Ultra low power design in near-threshold region,” *Proceedings of the IEEE*, vol. 98, pp. 237–252, February 2010.

[11] B. H. Calhoun, A. Wang, and A. Chandrakasan, “Modeling and sizing for minimum energy operation in subthreshold circuits,” *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 1778–1786, September 2005.

[12] A. Wang, B. Calhoun, and A. P. Chandrakasan, *Sub-Threshold Design for Ultra Low-Power Systems, 1st ed.* Springer: New York, NY, USA,, 2006.

[13] S. Singh, T. Sharma, K. Sharma, and B. Singh, “9T full adder design in subthreshold region,” vol. 2012.

[14] “HSPICE Nominal Models: Version 1.4 of the FreePDK45 kit, online,” April 2011. Available: http://www.eda.ncsu.edu/wiki/FreePDK45:Contents.

[15] “P. T. Model: Sub-45nm Bulk CMOS, online,” October 2007. Available: http://www.ptm.asu.edu/.

[16] S. Veeramachaneni and M. Srinivas, “New improved 1-bit full adder cells,” *Proceedings of the IEEE*, pp. 000735–000738, 2008.

[17] S. R. Chowdhury, A. Banerjee, A. Roy, and H. Saha, “A high speed 8 transistor full adder design using novel 3 transistor xor gates,” *International Journal of Electronics, Circuits & Systems 2;4*, pp. 217–223, 1992.

[18] H. Weste, Neil and K. Eshraghian, *Principles of CMOS VLSI Design: A Systems Perspective.* Massachusetts, USA: Addison-Wesley, 1985.

[19] A. T. Tran and B. M. Baas, “Design of an energy-efficient 32-bit adder operating at subthreshold voltages in 45-nm cmos,” *International Conference On Communications And Electronics (ICCE)*, pp. 87–91, August 2010.

[20] F. Moradi, D. T. Wisland, T. V. Cao, A. Peiravi, and H. Mahmoodi, “1-bit sub threshold full adders in 65nm CMOS technology,” *International Conference on Microelectronics*, pp. 268–271, 2008.

[21] P. Chan, M. Schlag, C. Thomborson, and V. Oklobdzija, “Delay optimization of carry-skip adders and block carry-lookahead adders using multidimensional dynamic programming,” *IEEE Trans. Comput.*, vol. 41, August 1992.

[22] V. Oklobdzija and E. Barnes, “Some optimum schemes for ALU implementation in VLSI technology,” *Proc. 7th Computer Arithmetic Symp.*, pp. 2–8, 1985.

[23] A. Guyot, B. Hochet, and J. Muller, “A way to build efficient carry-skip adders,” *IEEE Trans. Comput.*, vol. C-36, October 1987.

[24] M. Alioto and G. Palumbo, “A simple strategy for optimized design of one-level carry-skip adders,” *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 50, pp. 141–147, January 2003.

[25] M. Lehman and N. Burla, “Skip techniques for high-speed carry propagation in binary arithmetic circuits,” *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 691–698, December 1961.

[26] J. Rabaey, *Low Power Design Essentials.* 233 Spring Street, New York, NY 10013, USA: Springer Science and Business Media, LLC, 2009.

[27] M. M. Hinnwar and H. malviya, “Comparison of various leakage power reduction techniques for CMOS circuit design,” *International Journal of Engineering Research & Technology (IJERT)*, 2013.

[28] Y. Shin, J. Seomun, K.-M. Choi, and T. Sakurai, “Power Gating: Circuits, design methodologies ,and best practice for standard-cell VLSI Designs,” *ACM Transactions on Design Automation of Electronic Systems*, vol. 15, September 2010.

[29] F. Fallah and M. Pedram, “Standby and Active Leakage Current Control and Minimization in CMOS VLSI Circuits,” *IEICE Transactions on Electronics*, 2005.

[30] P. R. Panda *et al.*, *Basic low power digital design.* Springer Science and Business Media, LLC, 2010.

[31] J. Kathuria, M. Ayoubkhan, and A. Noor, “A review of clock gating techniques,” *MIT International Journal of Electronics and Communication Engineering*, vol. 1, pp. 106–114, August 2011.

[32] B. Keith *et al.*, “A 45 nm resilient microprocessor core for dynamic variation tolerance,” *IEEE Journal of Solid-State circuits*, vol. 46, January 2011.