IMPLEMENTATION OF SWITCH DEBOUNCE CIRCUIT USING GDI BASED SCHMITT TRIGGER

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

MASTER OF TECHNOLOGY
in
VLSI AND EMBEDDED SYSTEMS
by

Prakhar Saini

(2K23/VLS/04)

Under the Supervision of

Dr. Deva Nand

(Associate Professor)

DTU, Delhi

Department of Electronics and Communication Engineering

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Bawana Road, Delhi-110042

June, 2025

Department of Electronics and Communication Engineering
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, Prakhar Saini, Roll No. 2K23/VLS/04 hereby certify that the work which is being presented in the thesis entitled ***'Implementation of Switch Debounce circuit using GDI based Schmitt Trigger'*** in partial fulfilment of the requirements for the award of the Degree of Master of Technology (VLSI and Embedded Systems), submitted in the Department of Electronics and Communication Engineering, Delhi Technological University, is an authentic record of my own work carried out during the period from January 2025 to May 2025 under the supervision of **Dr. Deva Nand**.

I have not submitted the matter presented in the thesis for the award of any other degree of this or any other Institute.

Candidate's Signature

Prakhar Saini
(2k23/VLS/04)

Department of Electronics and Communication Engineering
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road, Delhi-110042

CERTIFICATE BY THE SUPERVISOR

This is to certify that the work reported in this thesis entitled *"Implementation of Switch Debounce Circuit using GDI Based Schmitt Trigger"* has been carried out by Mr. Prakhar Saini (2K23/VLS/04) under my supervision.

Supervisor's Signature
Dr. Deva Nand
Assosiate Professor
ECE, DTU, Delhi

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road, Delhi-110042

PLAGIARISM VERIFICATION

Title of the Thesis: **Implementation of Switch Debounce circuit using GDI based Schmitt Trigger**

Total Pages: _____

Name of the Scholar: **Prakhar Saini**

Supervisor: **Dr. Deva Nand**

Department of Electronics and Communication Engineering

This is to report that the above thesis was scanned for similarity detection. The process and outcome is given below:

Software used: **Turnitin** Similarity Index: ____ % Total Word Count: _____

Date: _____

Candidate's Signature **Signature of Supervisor(s)**

ACKNOWLEDGEMENT

The successful completion of any task is incomplete and meaningless without giving any due credit to the people who made it possible without which the project would not have been successful and would have existed in theory. First and foremost, I am grateful to Prof. O. P. Verma, H.O.D., Department of Elec- tronics and Communication Engineering, Delhi Technological University, and all other faculty members of our department for their constant guidance and support, constant motivation and sincere support and gratitude for this project work. I would owe a lot of thanks to our supervisor, Dr. Deva Nand, Associate Professor, Department of Elec- tronics and Communication Engineering, Delhi Technological University for igniting and constantly motivating us and guiding us in the idea of a creatively and amazingly per- formed Major Project 1 in undertaking this endeavour and challenge and also for being there whenever I needed his guidance or assistance.

I would also like to take this moment to show our thanks and gratitude to one and all, who indirectly or directly have given us their hand in this challenging task. I feel happy and joyful and content in expressing our vote of thanks to all those who have helped and guided me in presenting this project work for my Major project. Last, but never least, I thank our well-wishers and parents for always being with us, in every sense and constantly supporting me in every possible sense whenever possible.

Date:
Place: Delhi

Prakhar Saini
2K23/VLS/04

ABSTRACT

Switch bouncing causes multiple unwanted transitions in digital circuits when mechanical switches are pressed or released. In this work, a switch debounce circuit is implemented using a Schmitt Trigger designed with GDI-based NAND gates. The Gate Diffusion Input (GDI) technique reduces power consumption, delay, and area by minimizing transistor count compared to CMOS. The Schmitt Trigger introduces hysteresis with threshold voltages of 0.7V and 0.16V, effectively filtering bounce noise. Simulations in LTspice confirm stable output and lower power usage. This design offers an efficient, low-power hardware solution for switch debouncing in portable and embedded systems.

When a mechanical switch is pressed or released, it doesn't settle instantly; instead, it tends to bounce, creating a series of rapid, unintended on-off transitions. These glitches can lead to unexpected behavior in digital circuits if not handled properly. To address this issue, a switch debounce circuit has been developed in this work, incorporating a Schmitt Trigger that is built using GDI-based NAND gates. The Gate Diffusion Input (GDI) technique stands out by offering a significant reduction in power consumption, propagation delay, and silicon area, mainly due to its reduced transistor count when compared to conventional CMOS designs.

The use of a Schmitt Trigger is particularly beneficial here, as it adds hysteresis to the system—meaning the circuit reacts differently depending on whether the input is rising or falling. This characteristic, with defined threshold voltages of 0.7V for the high transition and 0.16V for the low, helps to cleanly filter out the noise caused by switch bouncing, ensuring that only legitimate transitions are passed to the output. Simulations conducted in LTspice further validate the design, showing that the output remains stable and consistent, even under noisy conditions, while consuming less power.

Overall, this design presents a practical and efficient hardware solution for handling switch bounce, making it highly suitable for power-sensitive applications such as portable devices and embedded systems where reliability and performance are critical.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In digital electronic systems, mechanical switches are widely used as input devices due to their simplicity and low cost. However, a common issue associated with these switches is contact bouncing, which leads to multiple unwanted transitions when the switch is toggled. These false transitions can cause erroneous behavior in digital circuits if not properly managed. To mitigate this, switch debouncing techniques are essential. This thesis presents the implementation of an efficient switch debounce circuit using a Schmitt Trigger based on Gate Diffusion Input (GDI) logic. The proposed design aims to reduce power consumption, area, and delay while maintaining signal integrity, making it suitable for compact and low-power applications in embedded systems.

1.1 Motivation and Background

In the rapidly evolving field of digital electronics, low power consumption, compact circuit design, and high performance have become essential requirements, especially with the growing proliferation of portable and battery operated devices. As technology scales down, the demand for energy-efficient circuits becomes critical to ensure reliability and prolonged battery life in embedded systems, Internet of Things (IoT) devices, and edge computing applications.

One persistent challenge in digital systems is the switch-bouncing phenomenon. Mechanical switches, though simple in construction, exhibit contact bounce when toggled, generating a series of spurious transitions before settling to a stable logic level. These unwanted transitions can cause erroneous signals in digital circuits, leading to faulty data processing, incorrect triggering of logic functions, and degraded system reliability. Traditionally, switch debouncing has been addressed using various hardware and software techniques, but these solutions often come with trade-offs in terms of circuit complexity, power consumption, and timing reliability.

An efficient approach to mitigate switch bounce is to use a Schmitt Trigger, a fundamental circuit that introduces hysteresis—a property that ensures a distinct threshold for signal rising and falling edges. By creating this hysteresis, a Schmitt Trigger effectively suppresses noise and small fluctuations at the input, thus providing a clean, stable digital output. It is an ideal solution for switch debouncing applications, where the input signal is inherently prone to noise and multiple transitions.

However, implementing Schmitt Triggers using traditional CMOS logic can lead to increased power consumption, area overhead, and complex design requirements, [1] especially when aiming for low-power, high-density systems. Therefore, it becomes crucial to explore alternative logic design techniques that can offer lower power dissipation, smaller area footprint, and reduced propagation delays, without compromising the core functionality of the Schmitt Trigger.

1.2 Research Focus: GDI-Based Schmitt Trigger Design

This research proposes a novel approach for designing a Schmitt Trigger based switch debounce circuit by leveraging the Gate Diffusion Input (GDI) technique. GDI is a well-established method for implementing digital logic gates with a minimal number of transistors, resulting in lower power consumption, reduced propagation delays, and smaller silicon area compared to conventional CMOS designs.

In this work, the Schmitt Trigger is constructed using NAND gates, which are in turn implemented using the GDI technique [6]. This combination provides a power-efficient, compact, and reliable solution for hardware debouncing applications. By replacing conventional CMOS NAND gates with GDI-based NAND gates, the Schmitt Trigger circuit inherits the advantages of reduced power dissipation and optimized performance, making it suitable for integration into modern digital systems where energy efficiency is paramount.

The proposed GDI-based Schmitt Trigger effectively eliminates unwanted glitches and noise from the switch input, stabilizing the output signal without the need for external clocking mechanisms or complex sequential logic. This makes the design not only simpler but also asynchronous, allowing for seamless integration into a wide range of digital systems, including low-power embedded devices, user-interface circuits, and control systems.

1.3 Objectives and Contributions

The primary objective of this thesis is to design, simulate, and evaluate a low-power Schmitt Trigger circuit using GDI-based NAND gates for switch debouncing applications. The key goals of this research include:

• Designing a Schmitt Trigger circuit that effectively mitigates the effects of switch bouncing by introducing hysteresis characteristics.

• Implementing GDI-based NAND gates to optimize the power and area performance of the Schmitt Trigger circuit.

• Simulating the proposed design using industry-standard tools (such as LTspice) to evaluate its behaviour in terms of power consumption, propagation delay, and stability.

• Demonstrating the application of the GDI-based Schmitt Trigger in a switch debounce circuit, validating its effectiveness through simulation results and waveform analysis.

• Comparing the performance of the GDI-based Schmitt Trigger solution against conventional logic implementations, highlighting its advantages in power and performance metrics.

1.4 Structure of the Thesis

The thesis is structured into eight chapters:

- **Chapter 1:** Introduces the motivation.
- **Chapter 2:** Literature Review
- **Chapter 3:** Design Methodology
- **Chapter 4:** Future Scopes
- **Chapter 5:** Conclusions

Chapter 2

Literature Review

2.1 Fundamentals of Switch Debouncing Phenomenon

Switch Bouncing Phenomenon

It has been observed how a push button or switch operates: it just presses to change its state. Switches are used in electronics for a variety of purposes. Mostly, they are used to drive or represent distinct voltage levels, which makes the system binary, that is, it can be either ON or OFF or have high or low voltage levels. In order to express the binary logic of 0s (zeroes) and 1s (ones), these voltage levels, for example, 5 Volts = High = ON = closed circuit and 0 Volts = Low = OFF = open circuit, are useful.

Figure 2.1: Basic Switch However, a lot goes on behind the straightforward push buttons on our keyboards and other gadgets [3]. In essence, a basic button consists

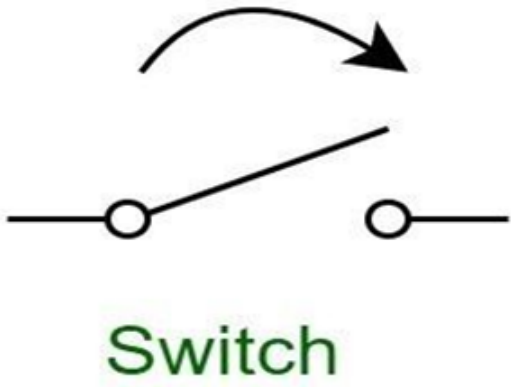

Figure 2.1: Basic Switch

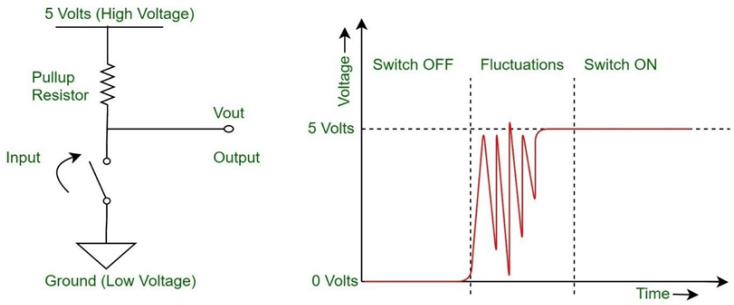

Figure 2.2: Switch Bounce Voltage Graph

of two metal contacts that come into contact when the user presses the button. The underlying circuit is then completed by these metal contacts, which also alert the sensing element—typically a microcontroller—that the button has been pressed.

2.2 Traditional Debounce Methods (SR Flip-Flop, RC filters, D Flip-Flop)

In this category, there are various implementations of circuits which can be used for eliminating the effect of switch debouncing right at the hardware level. The different types of circuits used are:

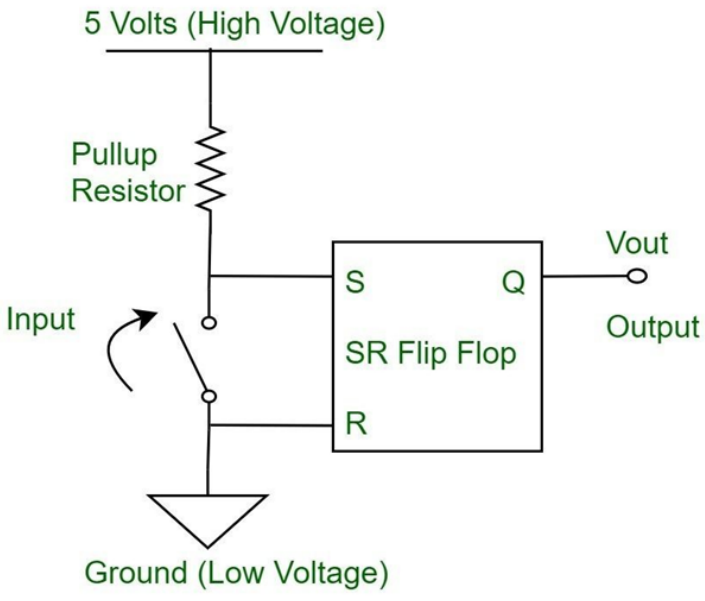

Figure 2.3: Switch Debounce using SR Flip Flop

It has been observed that the usage of D flip flop consumes less power than SR Flip Flop Figure 2.3. So, considering the same case in the switch debounce circuit

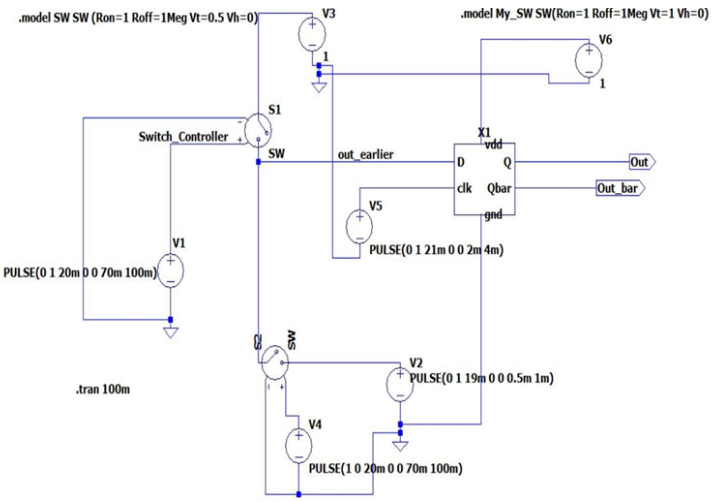

Figure 2.4: D flip flop for switch debounce

the SR Flip Flop is replaced by D Flip Flop as shown in the figure Figure 2.4.

The D flip flop used a power consumption technique that is GDI (Gate Diffusion Input) and both NAND gates using CMOS and GDI have been compared. It has been observed that the power consumption in CMOS-based NAND D flip flop is 9uW, and power consumption in GDI-based NAND D flip flop is 4.8uW. So there is almost 100 percent decrement in power consumption in case of GDI NAND D flip flop than CMOS NAND D flip flop.

2.3 Schmitt Trigger: Theory and Applications

Working Principle

A Schmitt Trigger is a fundamental circuit element used in digital and analog systems to convert a noisy or analog input signal into a clean, stable digital output. It is a comparator with positive feedback that introduces hysteresis, enabling it to reject noise and small fluctuations in the input signal [4]. Unlike a standard comparator or logic gate that changes state at a single threshold voltage, the Schmitt Trigger has two distinct threshold voltages:

- The upper threshold voltage (VUT), at which the output transitions from low to high as the input rises.
- The lower threshold voltage (V_{LT}), at which the output transitions from high to low as the input falls.
- This dual-threshold behaviour introduces a dead zone between VLT and VUT, where small variations in the input do not affect the output [2]. As a result, the Schmitt Trigger effectively cleans up signals with slow edges or noise, providing a sharp, noise-free digital output transition.

- In practical terms, when the input signal exceeds V_{UT}, the output is set to a logic high level. When the input drops below V_{LT}, the output resets to logic low.
- For input values between these two thresholds, the output retains its previous state, creating a memory-like effect that enhances noise immunity. This behavior makes Schmitt Triggers ideal for applications such as:
- Signal conditioning: Cleaning up noisy sensor inputs. Debouncing mechanical switches: Eliminating spurious transitions caused by switch bouncing.
- Oscillator circuits: Generating square waves from slow or analog input signals.
- Wave shaping circuits: Converting analog or slowly varying signals into sharp digital edges.

2.4 Hysteresis Characteristics

The defining feature of a Schmitt Trigger is its hysteresis curve—a graphical representation of its input-output relationship [9]. This hysteresis behaviour is characterized by the two distinct threshold voltages (V_{UT} and V_{LT}) and the resulting hysteresis width (ΔV), defined as:

$$\Delta V = V_{UT} - V_{LT} \tag{2.1}$$

The presence of hysteresis ensures that the Schmitt Trigger output does not rapidly toggle in response to minor, rapid fluctuations near the switching threshold. Instead, the circuit remains in its current state until the input crosses the respective threshold. The typical transfer characteristic of a Schmitt Trigger shows:

• A sharp transition from low to high output when the input crosses V_{UT}. • A corresponding sharp transition from high to low output when the input falls below V_{LT}. This behaviour provides:

• Noise immunity: Preventing false triggering due to small voltage spikes.

• Stable switching: Ensuring the output changes state only for significant input changes.

• Signal regeneration: Producing clean digital transitions from slowly changing or noisy inputs.

The hysteresis property is especially critical in debouncing applications, where mechanical switches introduce rapid, unintended transitions that need to be suppressed.

2.5 Advantages in Digital Systems

The Schmitt Trigger offers several key advantages that make it an essential building block in digital systems:

- **Noise Rejection:** By introducing hysteresis, the Schmitt Trigger ensures that small, rapid fluctuations in the input signal (such as those from noisy sensors or mechanical switch bounce) do not cause unwanted changes in the output. This makes it highly reliable in real-world conditions where signal integrity is often compromised.
- **Improved Signal Integrity:** Schmitt Triggers produce sharp, clean digital transitions from inputs with slow edges or gradual transitions. This is particularly useful in circuits where precise timing and well-defined logic levels are critical.
- **Asynchronous Operation:** Unlike flip-flop-based debounce circuits, the Schmitt Trigger does not rely on a clock signal. This makes it ideal for systems where simplicity, low latency, and asynchronous behavior are desired.
- **Low Component Count:** Schmitt Triggers can be implemented using simple gate configurations (such as NAND or NOR gates with feedback), making them compact and resource-efficient.
- **Versatility:** Schmitt Triggers are widely used in a variety of digital applications, including:
  - Switch debouncing.
  - Waveform shaping.
  - Square wave generation (oscillators).
  - Pulse width modulation. Threshold detection in analog-to-digital conversions.

By combining the GDI technique with the Schmitt Trigger design, this thesis aims to further enhance the power and area efficiency of such circuits, enabling their integration into low-power embedded systems, IoT devices, and portable electronics where energy efficiency is paramount.

2.6 Logic Design using GDI

With the growing demand for faster, more compact, and energy-efficient digital applications—especially in portable devices—researchers have been exploring new solutions to overcome the limitations of traditional CMOS technology. Over the

years, various logic design techniques have emerged, each aiming to boost circuit performance.

However, most Pass Transistor Logic (PTL) designs face two key challenges [5]. The first is the threshold voltage drop across single-channel pass transistors, which leads to lower current drive and slower operation.

Several PTL-based approaches have been proposed to tackle these issues:

1. **Transmission Gate CMOS (TG):** TG logic uses both nMOS and pMOS transistors in parallel to achieve full logic-level swings, making it possible to implement complex logic functions with fewer transistors.

2. **Complementary Pass-Transistor Logic (CPL):** CPL uses nMOS pass transistors for logic functions, complemented by CMOS inverters at the outputs. It offers the benefits of lower power consumption and a small stack height, but suffers from static power loss due to the low voltage swings at the inverter gates. To address this, variants like LCPL (with pMOS restoration transistors) and SRPL (with cross-coupled inverters) have been introduced to improve the swing and reduce power loss.

3. **Double Pass-Transistor Logic (DPL):** DPL combines both nMOS and pMOS transistors in its design, enabling full-swing operation without needing extra restoration circuitry. It also helps reduce DC power consumption.

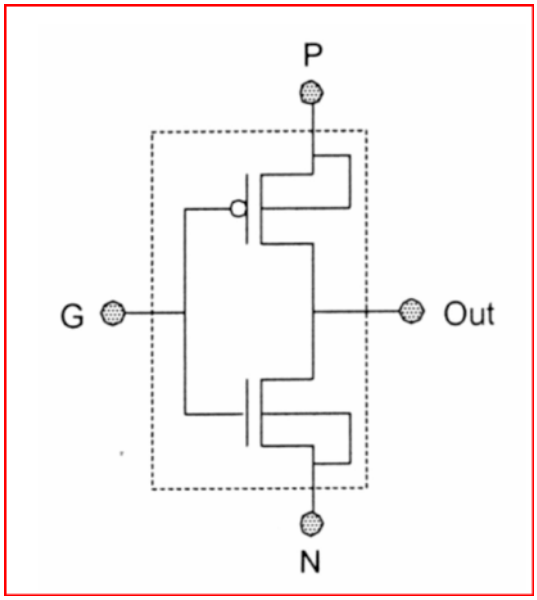

Figure 2.5: GDI Based Cell

Table 2.1: Various Logic Functions of GDI Cell

N	P	G	Out	Function
0	B	A	$\overline{A}B$	F1
B	1	A	$\overline{A}+B$	F2
1	B	A	$A+B$	OR
B	0	A	AB	AND
C	B	A	$\overline{A}B+AC$	MUX
0	1	A	$\overline{A}$	NOT

The Gate Diffusion Input (GDI) method is based on a basic cell structure that, at first glance, looks quite similar to a standard CMOS inverter [1]. However, there are key differences that set it apart.

Firstly, the GDI cell includes three inputs:

• G, which acts as the common gate for both the NMOS and PMOS transistors,

• P, which connects to the source or drain of the PMOS, and

• N, which connects to the source or drain of the NMOS.

Unlike CMOS inverters where bulk connections are fixed, in a GDI cell, the bulk terminals of the NMOS and PMOS can be tied to N or P, respectively. This flexibility allows the cell to be biased differently, depending on the logic function being implemented.

However, it's important to note that not all configurations of the GDI cell can be realized using a standard p-well CMOS process. For full functionality, more advanced technologies like twin-well CMOS or silicon-on-insulator (SOI) may be required. This aspect is discussed later in the document.

What makes GDI special is that by simply changing the way the inputs are connected, many different logic functions can be implemented using the same basic structure. These functions, which would normally require 6–12 transistors in CMOS or PTL (Pass Transistor Logic, can be built using just two transistors in a GDI cell Figure 2.5.

In this work, the focus is mainly on two GDI configurations—F1 and F2—because: They are functionally complete, meaning any two-input logic operation can be built from them. The F1 function is compatible with standard p-well CMOS, as it keeps the NMOS bulk consistently biased.

However, in some configurations, especially when P and N are set to opposite logic levels, unwanted current paths can form, causing static power loss. This is particularly an issue for logic gates like AND, OR, and MUX when implemented using standard CMOS. These effects can be minimized by using floating-bulk SOI technology, where such problems are less significant. Overall, the GDI structure offers a clear advantage over traditional logic styles like PTL, especially in terms of reducing transistor count, power consumption, and circuit complexity. To fully grasp how GDI works in different situations, a deeper look into the operation of the basic cell under various configurations is needed.

Table I shows that a single GDI cell structure can replace multiple conventional logic gates—like AND, OR, NOT, MUX, and custom logic functions (F1, F2)—just by changing input connections. This makes GDI a power-efficient and compact alternative to CMOS, especially for low-power VLSI design.

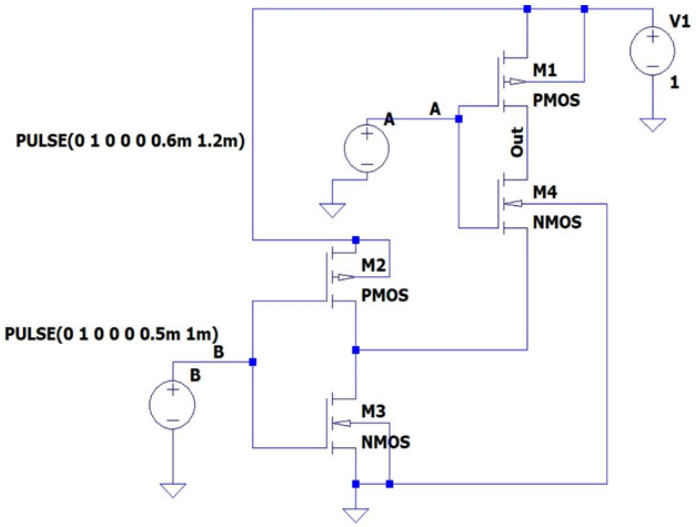

Figure 2.6: NAND gate using GDI in LTspice

2.7 Comparison between GDI, CMOS, TG and N-PG

Table 2.2: Logic gate comparisons (GDI, CMOS, TG, and N-PG) using the circuit topologies

Gate type in series	Logic expression	GDI			CMOS			TG			N-PG		
		Power (µW)	Delay (nsec)	# tr.	Power (µW)	Delay (nsec)	# tr.	Power (µW)	Delay (nsec)	# tr.	Power (µW)	Delay (nsec)	# tr.
MUX	$\overline{A}B + AC$	35.7	1.1	8	49.7	2.1	24	44.9	1.0	16	47.5	3.1	16
OR	$A + B$	26.3	1.2	8	32.9	1.7	12	36.2	1.3	16	32.6	2.7	16
AND	AB	25.7	0.9	8	34.1	1.4	12	30.8	0.8	16	30.1	2.8	16
F1	$\overline{AB}$	31.2	0.8	8	45.2	1.5	12	31.8	1.1	16	31.8	2.5	16
F2	$\overline{A} + B$	32.0	1.3	8	43.1	1.9	12	33.2	1.4	16	29.6	3.5	16

Each logic family is evaluated based on the following key performance parameters:

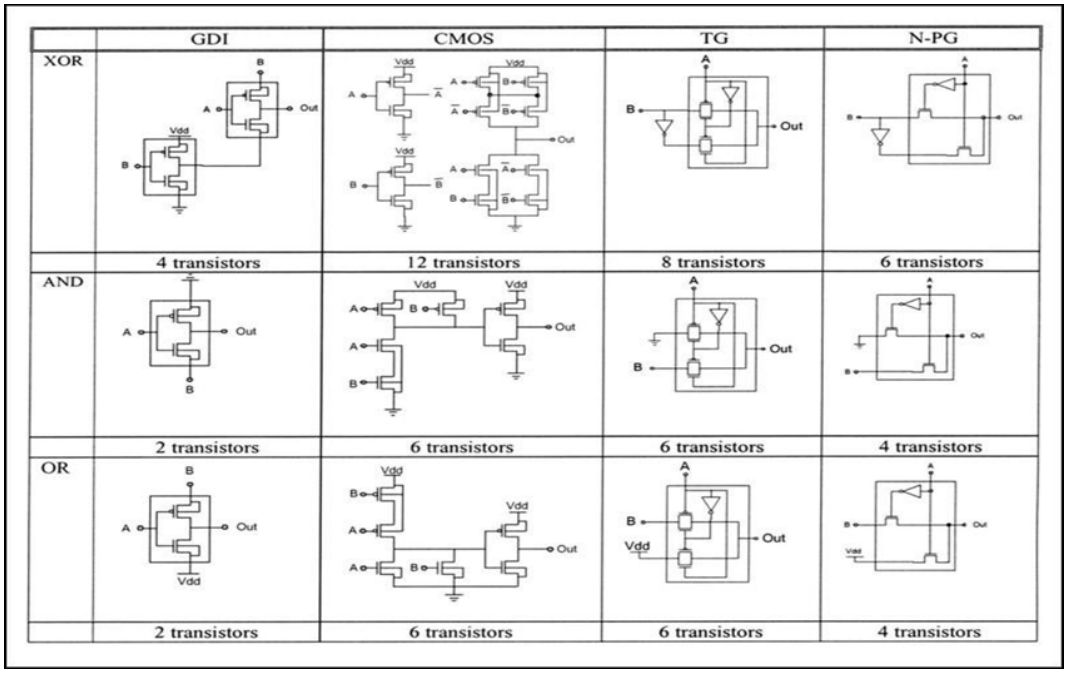

Figure 2.7: Logic gate comparisons (GDI, CMOS, Transmission gate, and NMOS pass gate) using the circuit topologies

- **Power Consumption** (in microwatts, µW)
- **Propagation Delay** (in nanoseconds, ns)
- **Number of Transistors Used** (# tr.)

Key Observations

- **GDI Logic:** GDI consistently uses fewer transistors (only 8 for all functions) and achieves lower power consumption and delay. For example, the AND gate consumes only 25.7 µW with a delay of 0.9 ns, which is significantly better than other logic styles.
- **CMOS Logic:** CMOS implementations use more transistors (ranging from 12 to 24) and show higher power consumption and delay. For instance, the MUX gate in CMOS consumes 49.7 µW and has the highest delay of 2.1 ns.
- **Transmission Gate (TG):** TG circuits offer a good balance between power and delay but require more transistors (mostly 16). The TG-based AND gate achieves a low delay of 0.8 ns.
- **nMOS Pass Gate (N-PG):** N-PG uses 16 transistors, but exhibits the highest delay across most gates (e.g., 3.5 ns for F2). While the power consumption is

reasonable, the poor delay performance makes it less favorable for high-speed applications.

Summary

From the comparison, it is evident that **GDI** offers the most efficient design in terms of power, delay, and transistor count, making it suitable for low-power and high-performance applications. While **CMOS** is widely used due to its robustness, it is relatively inefficient. **TG** offers decent performance but with higher transistor count, whereas **N-PG** is the least effective in terms of delay and should be used cautiously in timing-critical designs.

2.8 Switch Debounce using D flip flop

So above there are several ways to transform many logic functions using GDI technique [1]. While the formation of NAND gate using CMOS and GDI technique it has been observed that the power dissipation in NAND gate CMOS is 1.6uW and by using GDI NAND gate the power consumption is 1uW. So there is decrement of 60 percent.

Than there is a D flip flop have been made using the NAND gate. The undefined input conditions SET = ”0” and RESET = ”0” are prohibited in the SR NAND Gate Bistable circuit. That is the SR flip flop’s disadvantage. This condition:

1. Override the action of latching feedback.

2. Make both outputs equal to one.

Give up control over the input, which initially changes to 1, leaving the other input at ”0,” which controls the latch’s final state. An inverter is required to stop this from occurring. To create a different kind of flip flop circuit known as a D flip flop, Delay flip flop, D-type Bistable, or D-type flip flop, we connect the inverter between the Set and Reset inputs.

Of all the timed flip flop varieties, the D flip flop is the most significant. It guarantees that neither of the two inputs—S nor R—ever equals 1 simultaneously. A gated SR flip-flop with an inverter linked between the inputs is used to design the delay flip-flop to enable a single input D (Data).

The ”D” designated single data input is utilized instead of the ”Set” input, and the inverter is used for the complementary ”Reset” input [2] Figure 2.8. Consequently, a level-sensitive SR flip flop is the building block for the level-sensitive D-type or D flip flop. Thus, in this case, S=D and R= D (complement of D).

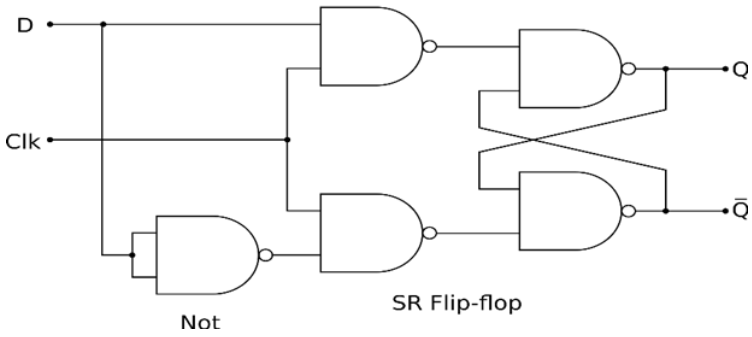

Figure 2.8: Basic D flip flop using NAND gate

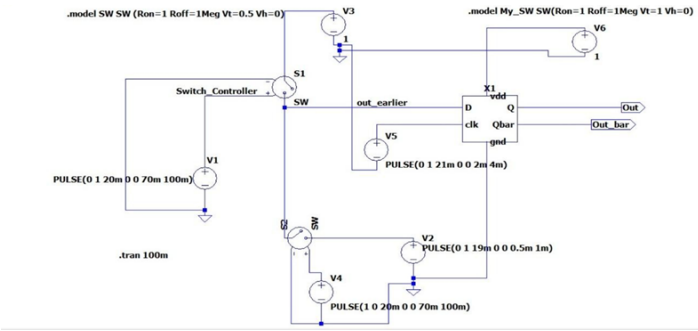

Figure 2.9: Switch Debounce using D flip flop

A D flip-flop is a basic memory device used in digital circuits. It stores one bit of data.

- The ”D” stands for Data or Delay.
- It has a data input (D), a clock input, and an output (Q).
- When the clock gets a pulse (usually rising edge), whatever value is on the D input gets ”locked in” and sent to the output Q.

The output stays the same until the next clock pulse.

2.9 GDI NAND Gate representation

NAND using CMOS

A NAND gate is implemented using CMOS technology by combining both PMOS and NMOS transistors Figure 2.10. In this design, two PMOS transistors are connected in parallel to form the pull-up network, and two NMOS transistors are connected in series to form the pull-down network.

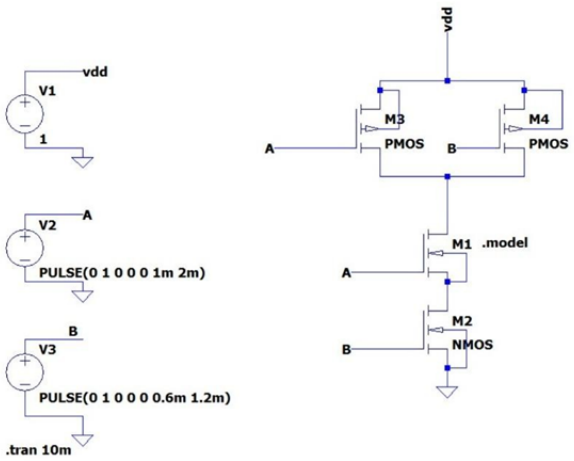

Figure 2.10: NAND using CMOS

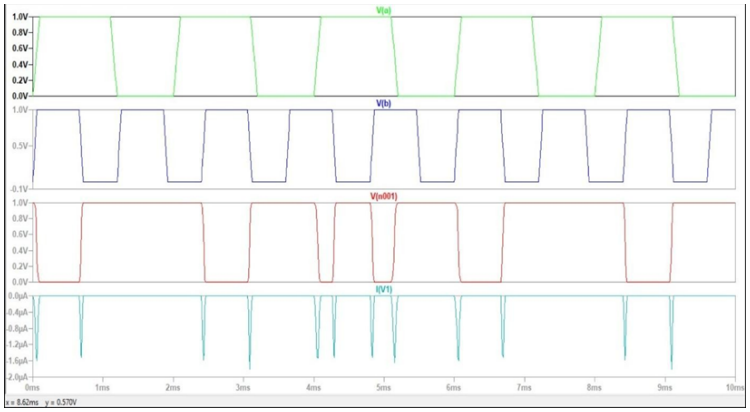

Figure 2.11: Output waveform of NAND gate using CMOS

When both inputs are given a high signal, the path through the NMOS series is completed, and the output is pulled down to logic low. In all other input combinations, at least one PMOS remains on, keeping the output at logic high Figure 2.11.

As a result, the output is kept high for all input combinations except when both inputs are high — which is the expected behavior of a NAND gate. The use of CMOS ensures that low power is consumed and noise margins are improved, as both pull-up and pull-down networks are used effectively.

NAND using GDI

One of the major advantage of using GDI technique is that they have very less leakage current as in all the stages both pmos and nmos are in stacking. In stacked MOSFET configurations, leakage power is reduced due to the ”stack effect.” When multiple transistors are turned off, the leakage current from the first transistor in the stack causes a voltage to build up at the intermediate node [1].

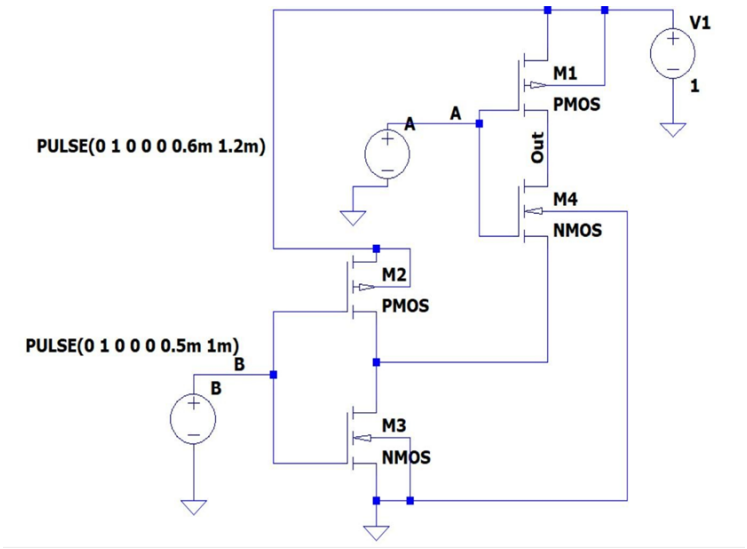

Figure 2.12: NAND using GDI

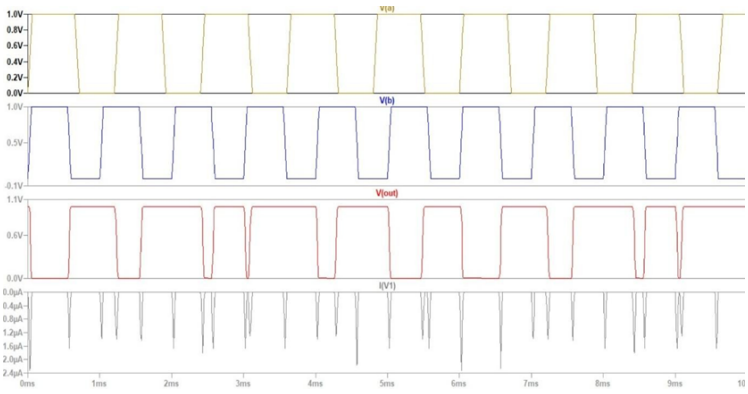

Figure 2.13: Output waveform of NAND using GDI

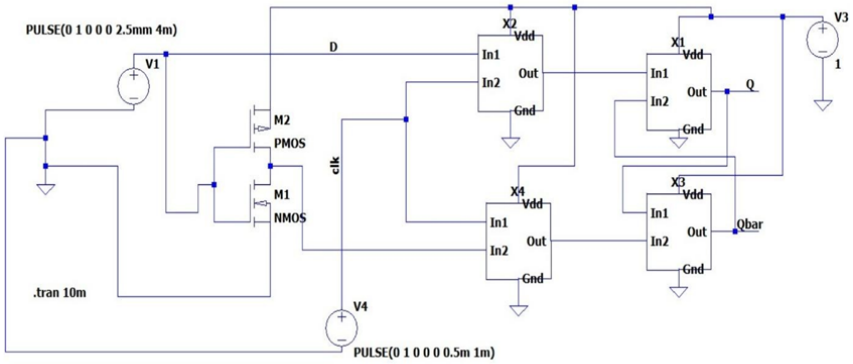

Figure 2.14: D flip flop using CMOS based NAND

This voltage, in turn, raises the source voltage of the next transistor in the stack, thereby reducing its gate-to-source voltage and further suppressing leakage. As a result, the overall leakage current is diminished in the stacked configuration, since each transistor in the chain contributes to lowering the current flow.

This phenomenon enables better control over static power consumption, making stacking an effective technique for leakage reduction in modern CMOS designs. Here also, the very short circuit power as been seen as there is no such time when there is both NMOS and PMOS are ON and there is contact between power to ground.

In CMOS logic circuits, short-circuit power is minimized during switching transitions due to the inherent design of complementary pMOS and nMOS transistors [1]. During normal switching operation, at any given time, either the pMOS or the nMOS is conducting, but not both simultaneously for long periods.

This complementary behavior ensures that the direct current path from the power supply (VDD) to ground is momentarily established only during very brief transitions when the input signal changes. However, due to the relatively short overlap time and the symmetrical, sharp switching characteristics of CMOS devices, the short circuit current flow is limited. As a result, short-circuit power dissipation is significantly reduced compared to other logic families (such as pseudo-NMOS or static logic families), making CMOS technology highly efficient for low-power applications.

2.10 D flip flop using CMOS based NAND

As there is less leakage power and leakage current observed, the behavior can be understood with respect to the characteristics of the MOSFET. For the condition where $V_{DS} > V_{GS} - V_{TH}$, the device operates in the saturation region. In this region,

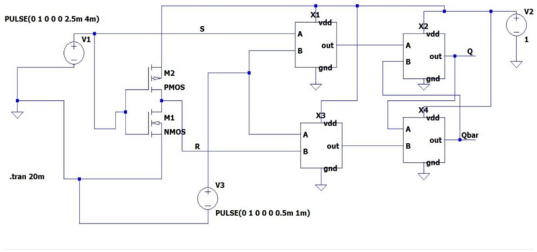

Figure 2.15: D flip flop using GDI based NAND

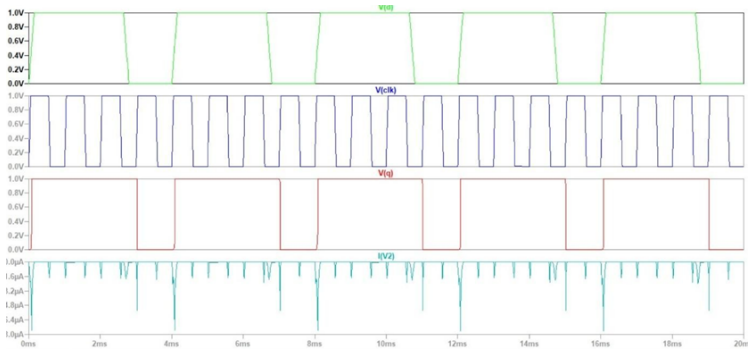

Figure 2.16: Output waveform of D flip flop using GDI based NAND

the drain current is given by:

$$I_D = \mu C_{ox} \frac{W}{2L} (V_{GS} - V_{TH})^2 \tag{2.2}$$

As the leakage current decreases, the available drain current I_D increases, which allows the output load to be charged or discharged more easily, resulting in reduced delay [4]. To further improve performance, the width of the transistors in the critical path can be increased to provide a higher current drive, enabling faster switching and reduced propagation delay.

2.11 D flip flop using GDI based NAND

This complementary behavior ensures that the direct current path from the power supply (VDD) to ground is momentarily established only during very brief transitions when the input signal changes. However, due to the relatively short overlap time and the symmetrical, sharp switching characteristics of CMOS devices, the short-circuit current flow is limited.

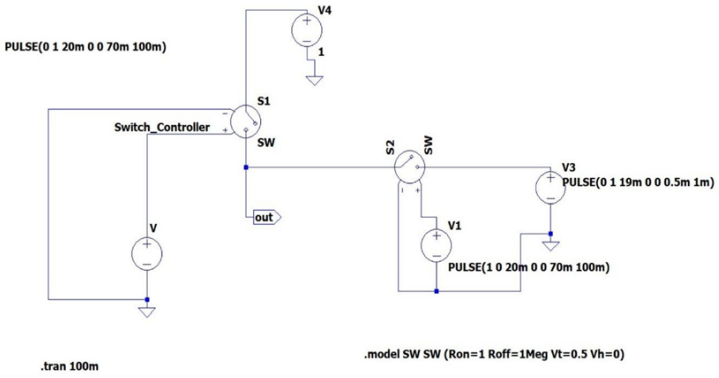

Figure 2.17: Implementation of switch bounce circuit in LTspice

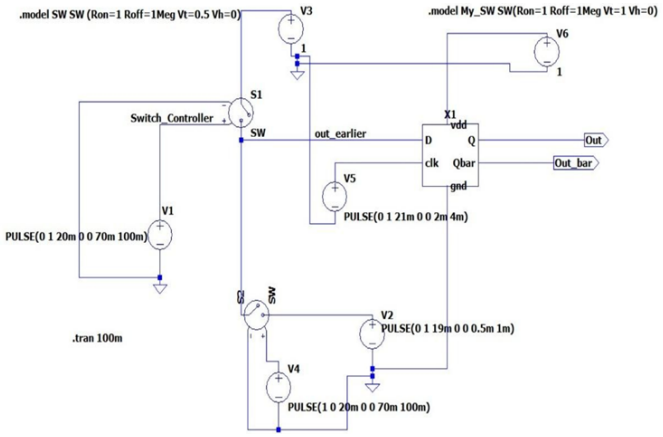

Figure 2.18: Output of Switch bounce circuit

2.12 Implementation of switch bounce circuit in LTspice

Here we can see that the noisy signal is getting converted to a neat and clean square waveform. This is because the DFF is storing it's value at the first clock than replicating the same.

2.13 Switch Debouncing Circuit using D flip flop

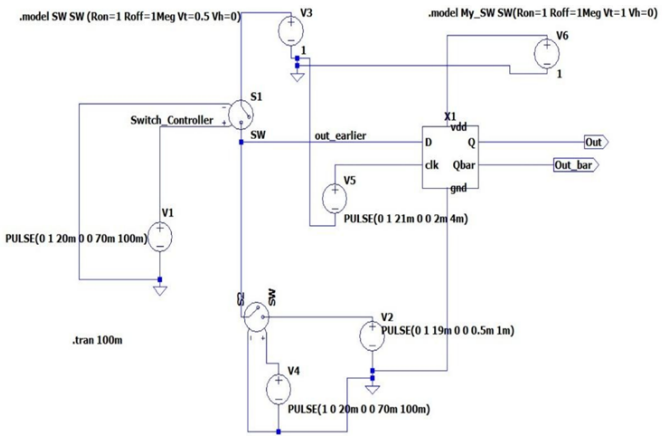

Figure 2.19: Switch Debouncing Circuit using D flip flop

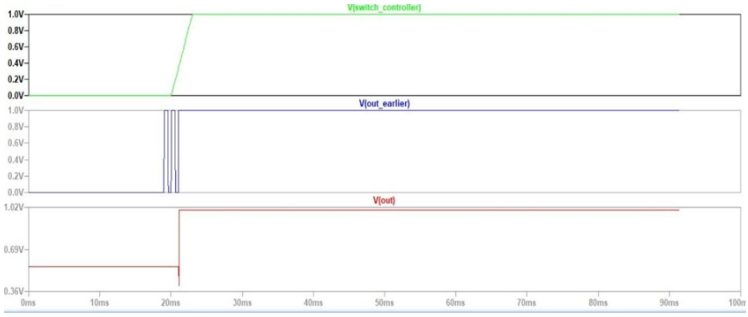

Figure 2.20: Output of Switch Debounce circuit using D flip flop

Chapter 3

Design Methodology

3.1 Overview of the proposed approach

There is a special methodology that has been followed which uses a Schmitt trigger to switch and debounce a bouncing circuit. Figure 3.1:- Switch Debouncing

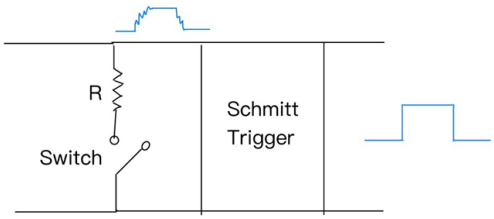

Figure 3.1: Switch Debouncing Methodology using Schmitt Trigger

Methodology using Schmitt Trigger A Schmitt trigger helps solve the problem of switch debounce by introducing hysteresis into the circuit. This means the input signal must pass a certain threshold before it triggers a change in the output, and it won't switch back until it crosses a second, separate threshold.

This hysteresis prevents small fluctuations or noise—like those caused by a bouncing mechanical switch—from creating multiple unwanted transitions in the output. So, the Schmitt trigger essentially smooths out the noisy signal from the switch and ensures a clean, stable output transition.

3.2 Implementation of Schmitt trigger using NAND gate

A NAND gate can be thought of as a decision-maker in a circuit. It is given two inputs, and it is designed to check whether both inputs are ON (1). If both inputs are ON, the output is switched OFF (0). In all other cases – whether one input is OFF

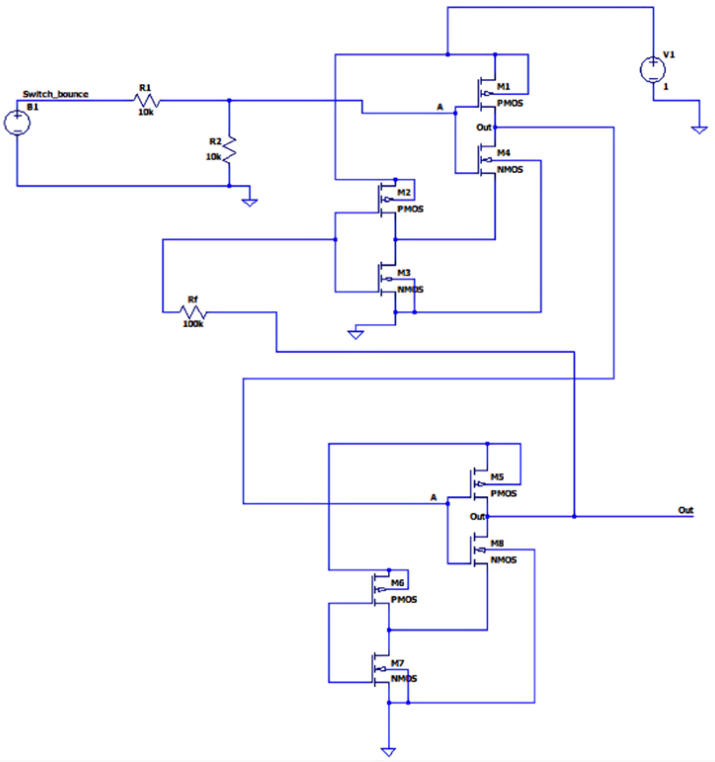

Figure 3.2: Implementation of switch debounce using NAND gate based Schmitt trigger

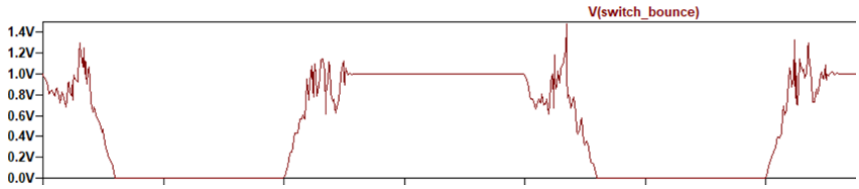

Figure 3.3: Input switch bounced signal

or both are OFF – the output is kept ON (1).

The NAND gate is known as a NOT AND gate because it performs the opposite of an AND gate. It is considered a fundamental building block in digital logic design, and many circuits are built using NAND gates.

3.3 Hysteresis Calculations

A Schmitt trigger has a feature called hysteresis that helps clean up noisy signals [9]. It does this by setting two different points where the output changes. One point is for when the signal goes up, and the other is for when the signal goes down. So, the output doesn't switch back and forth wildly every time there's a small glitch or

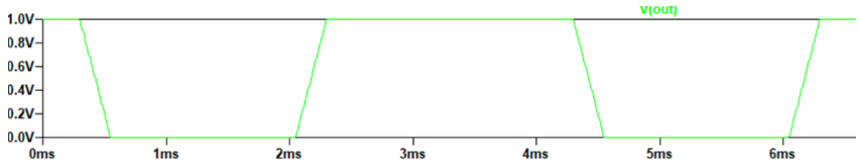

Figure 3.4: Output of the Switch Debounce

noise in the signal.

This makes a Schmitt trigger really useful for things like switch debounce, where a mechanical switch might bounce and cause messy signals. The Schmitt trigger waits for the signal to go past a certain point before changing the output, helping to keep the output smooth and stable.

Considering the Threshold Voltage of the NAND gate = 0.35V, which implies that the NAND gate will only provide it's function to the signal whose voltage value exceeds 0.35V.

$$R_1 = 10\,\text{k}\Omega \tag{1}$$
$$R_2 = 10\,\text{k}\Omega \tag{2}$$
$$R_f = 100\,\text{k}\Omega \tag{3}$$

For Upper Threshold Voltage (V_{UT}):

Using equations (1), (2), and (3):

$$V_{\text{UT}} = \left(\frac{R_1 + R_2}{R_2}\right) \times V_{\text{th}} \tag{4}$$
$$= \left(\frac{10\,\text{k}\Omega + 10\,\text{k}\Omega}{10\,\text{k}\Omega}\right) \times 0.35$$
$$= 2 \times 0.35 = 0.7\,\text{V} \tag{5}$$

For Lower Threshold Voltage (V_{LT}):

Using equations (1), (2), and (3):

Using the parallel combination of resistors:

$$V_{\text{LT}} = \left(\frac{(R_1 \parallel R_2) \parallel R_f}{R_2}\right) \times V_{\text{th}} \tag{6}$$
$$= \left(\frac{R_{\text{eq}}}{R_2}\right) \times V_{\text{th}}$$
$$= \left(\frac{4.76}{10\,\text{k}\Omega}\right) \times 0.35 = 0.16\,\text{V} \tag{7}$$

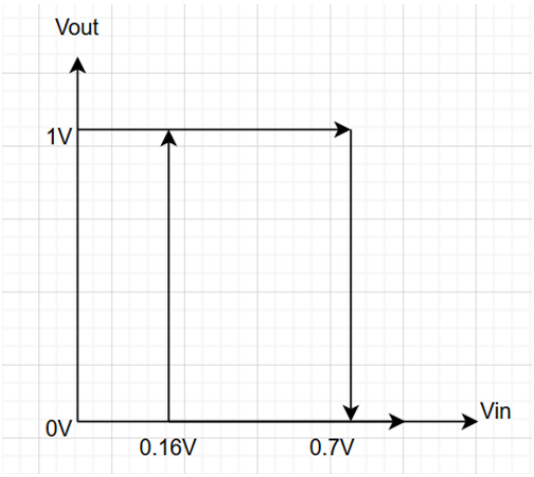

Figure 3.5: Hysteresis of the applied Schmitt Trigger

A Schmitt trigger has this cool feature where it waits for the signal to cross a certain level before it flips the output. In this case, it has two thresholds:

- If your signal goes above 0.7 volts, the output jumps up (to, say, 1 or 5 volts).

- If your signal drops below 0.16 volts, the output drops down.

Between these two points—0.16 volts and 0.7 volts—the output stays put. It doesn't react to small wobbles or noise in the signal. So, imagine a bouncy switch that keeps flipping up and down randomly when it's pressed. If you feed that noisy signal into a Schmitt trigger, the trigger won't care about those tiny ups and downs. It'll only change the output when the input clearly crosses those specific points.

That way, the Schmitt trigger gives you a nice, clean, stable output that doesn't flip-flop every time there's a tiny spike in the input. That's why it's great for fixing problems like switch bounce.

3.4 Applying Switch Debounce using Schmitt Trigger

When you press a physical switch, it doesn't always give a clean 0 to 1 voltage change [11]. Instead, because of mechanical imperfections, it bounces—the voltage might jump between 0 and 1 several times very quickly before settling. This is called switch bounce. Now, if you directly feed this bouncy signal into a digital circuit, the circuit might get confused and think the switch is being pressed and released multiple times. That's a problem! Here's where the Schmitt trigger comes to the rescue.

Let's say the switch voltage bounces up and down between 0 and 1, but gradually increases overall:

• The Schmitt trigger ignores the small, random fluctuations and only switches the output when the signal crosses the upper threshold—in your case, 0.7 volts.

• Once it switches high, even if the signal dips back down slightly (but stays above 0.16 volts), the output remains high. • The output only switches low if the signal drops below 0.16 volts.

So, the Schmitt trigger effectively filters out the bouncing noise from the switch. It doesn't react to every little change—it waits for the signal to cross a clear boundary, either high or low.

Chapter 4

Future Scopes

4.1 Future research methodology

This is our proposed figure of the switch debounce circuit. The input from the switch is passed to the input in the figure above. For the input, we will be producing the noisy waveform using the LTspice [13]. The noisy input produced by the LTspice circuit is given to the XOR gate, as well as to a DFF, which has an output port from which a clean signal can be obtained. Here output of the RC circuit is passed to the output port after a delay of five clock. cycles. The switch debounce circuit, after sampling the first transition (initial state) on the switch, waits for a predetermined time (settling time) and checks whether the current state matches the sampled state.

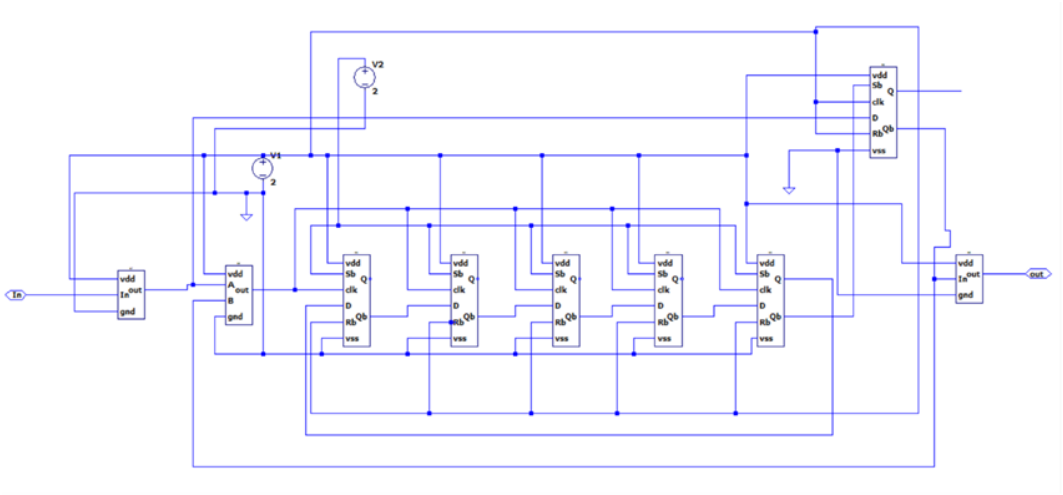

Figure 4.1: Cascading DFF to avoid the problem of switch bouncing

If there is a match, it is a valid transition (state change of switch), otherwise, it is considered as a glitch (bounce) and ignored. The circuit further continues to sample the switch state. In this way, the circuit will respond to only one pulse generation and not several state transitions caused by contact bouncing.

Here one of the inputs to XOR gate is the output from the last D-FF and another is from the terminal which is connected to bouncing input. If both the inputs of the XOR gates are equal, the output of XOR gate will be LOW else it will be HIGH which serves as clock to the final stage DFF from which Qout which is the final output without any bouncers are obtained.

The output of the final stage DFF will retain the previous state till the clock goes high. The clock will go high only after pre-determined time during which the sampled state (bounce-free) has to be stable [17].

In D flip flop, reset is used because flip flop is often used for implementation of circuits that can have many possible states, where the response of circuit depends not only the inputs but also on the states that the circuit is in at that time. Hence reset is used to obtain the initial state of the DFF.

Electrostatic discharge (ESD) [7], which has become one of the most important reliability issues in IC products, must be taken into consideration during the design phase of all ICs. Due to electrical over stress (EOS) which is caused due to high voltages exceeding the data sheet values, gate oxide dielectric breakdown which might damage the ICs. Hence input and output pads are used for the debounce circuit to avoid the damage.

In the above figure it has been done that the there are DFF made into cascades to basically make the samples of the input switch bouncing signals. There are samples made and then the system will decide the constant samples to keep so judge the following signal [5].

That if the signal is going from 0 to 1 then in between the transition there is so many voltages achieved between them then the system will take multiple samples of the signal, in the above figure the number of samples that are taken are 5 and then these samples are compared such that there values average is taken and then the side values are compared that is initial 2 samples are compared with the last 2 samples, if these samples are different then it will be assumed that the signal is changing from 1st sample value to 2nd sample value.

If the samples average value of the first 2 and the last 2 are same that means the signal is constant and this constant signal can be judged by watching that how close they are actually to the 0 or 1 [4]. By using this methodology it becomes easy to get the proper refined signal out of the switch bounced signal which is the target of this thesis and our research.

4.2 Comparison between switch debounce circuit using CMOS and GDI

So here we are reducing the power a delay on the cost of the increase in area. The reduction of circuit area is achieved by minimizing the number of transistors required for logic gate implementation [6]. In conventional CMOS logic, multiple transistors are used to realize even basic gates, which leads to an increase in the layout size. However, by applying techniques such as the Gate Diffusion Input (GDI) method, the same logic functions can be achieved with fewer transistors.

As a result, the overall area occupied by the circuit is significantly reduced. Additionally, the complexity of interconnections is minimized, which further contributes to a more compact layout. This reduction in area not only enables the design of smaller and more efficient circuits but also supports the integration of a larger number of functionalities within a limited chip space.

By optimizing the circuit in this way, lower power consumption and improved performance can also be achieved, making the design more suitable for portable and battery-powered applications. Table 4.1 presents a comparison between two switch

Table 4.1: Comparison of Switch Debounce Circuits using CMOS and GDI based D Flip-Flops

Circuit Description	**Average Power (W)**	**Area (μm^2)**	**Output Delay (ns)**
Switch Debounce using CMOS based DFF	2.46×10^{-7}	228568	3.95
Switch Debounce using GDI based DFF	1.95×10^{-7}	23487.3	2.8

debounce circuit implementations: one using a CMOS-based D Flip-Flop and the other using a GDI-based D Flip-Flop. The evaluation is done based on three key parameters: average power consumption, silicon area, and output delay.

From the data, it is evident that the GDI-based design significantly outperforms the CMOS-based design in terms of power efficiency and speed. The average power consumed by the GDI circuit is 1.95×10 W, which is lower than the 2.46×10 W consumed by the CMOS version. This reduction in power is mainly due to the reduced transistor count and simpler topology provided by the GDI technique.

In terms of area, the GDI circuit occupies only 23,487.3 μm^2, whereas the CMOS implementation uses a much larger 228,568 μm^2. This demonstrates a substantial reduction in chip area, which is a major advantage for modern VLSI systems where

layout compactness and integration density are critical.

Finally, the output delay is also improved in the GDI-based design. The GDI circuit exhibits a delay of just 2.8 ns, compared to 3.95 ns in the CMOS circuit. This indicates that the GDI-based D Flip-Flop is not only more power-efficient and compact, but also faster in switching performance.

Overall, the GDI-based switch debounce circuit shows a clear improvement over the traditional CMOS-based design in all the evaluated parameters. This makes it highly suitable for low-power, area-constrained, and high-speed digital applications, especially in embedded systems and portable devices.

4.3 Proposed Improvements

While the current GDI-based switch debounce circuit shows improvements in power, area, and delay, there are still opportunities to enhance its overall performance and robustness. One potential improvement is the inclusion of a programmable hysteresis control in the Schmitt Trigger stage, which would allow the threshold voltages to adapt dynamically based on the input noise level or supply variations.

Additionally, the circuit could be optimized further by using multi-threshold CMOS (MTCMOS) techniques to reduce leakage power during standby modes. Integration with clock gating logic may also help minimize unnecessary switching, especially when no input transitions occur. Furthermore, replacing the conventional D Flip-Flop with a pulse-triggered or sense-amplifier based flip-flop can further reduce dynamic power and improve speed.

Lastly, scalability should be considered for future versions, where the debounce logic could be designed as a reusable IP block that can support multiple switch inputs simultaneously.

4.4 Possible Applications

The proposed GDI-based switch debounce circuit can be utilized in a wide range of low-power and embedded systems where mechanical switches are commonly used for user input. This includes:

- Wearable electronics such as fitness trackers or health monitoring devices, where compact size and battery life are critical.
- Consumer electronics, including TV remotes, calculators, and game controllers.

- Industrial control systems, where reliable and fast switching responses are necessary in harsh environments.
- IoT-based devices where physical buttons are interfaced with microcontrollers that require debounce handling.
- Keypad-based security systems, where a clean and bounce-free signal is essential for accurate password entry.

Due to its low power consumption and small area, the circuit is well-suited for battery-powered and space-constrained devices.

4.5 Layout and Fabrication Scope

The designed debounce circuit, once functionally verified through simulation, can be extended to the layout level using CAD tools like Cadence Virtuoso or Electric VLSI. The GDI-based logic components would require careful transistor-level layout planning to maintain signal integrity and avoid issues such as DRC violations or latch-up.

Special attention must be paid to matching parasitics, optimizing interconnect lengths, and minimizing IR drop in power rails. The design should conform to a standard cell layout methodology, ensuring it can be integrated easily into a larger chip or system-on-chip (SoC).

Following layout design, the circuit could be prepared for fabrication using a low-power CMOS process node, such as 65nm or 45nm, to leverage the full power savings of GDI logic. Post-layout simulation (with parasitic extraction) would further validate performance metrics and ensure that the fabricated chip meets the desired specifications under real-world conditions.

Chapter 5

Conclusions

In this thesis, I set out to tackle a common yet critical problem in digital systems—switch bouncing—by designing a hardware debounce circuit that is both low-power and efficient. The approach I followed was to build a Schmitt Trigger-based debounce circuit, where the Schmitt Trigger itself was implemented using NAND gates. To make the design even more optimized, these NAND gates were created using the Gate Diffusion Input (GDI) technique, a proven method for reducing power consumption, propagation delay, and overall area in digital circuits.

Through simulation and analysis, I demonstrated that the GDI-based NAND gates are highly effective in reducing power usage compared to traditional CMOS designs [9]. The Schmitt Trigger designed using these GDI NAND gates exhibited a clear hysteresis behavior, with an upper threshold voltage of 0.7V and a lower threshold voltage of 0.16V.

This hysteresis characteristic was key in stabilizing the output signal, effectively suppressing the noise and transient glitches caused by switch bouncing. As a result, the final debounce circuit was able to deliver a stable and clean output in the presence of noisy and fluctuating inputs—solving the switch bounce problem efficiently.

Beyond just solving the problem at hand, this thesis also highlights the broader potential of using GDI logic techniques in real-world digital designs. By integrating GDI into fundamental circuits like NAND gates, [10] and then building higher-level functions like Schmitt Triggers and debounce circuits, we can achieve significant improvements in power efficiency—something that is especially important for today's battery-powered devices and embedded systems.

In conclusion, this project not only provides a practical solution for switch debouncing using a Schmitt Trigger, but also serves as a proof-of-concept for how GDI-based logic design can be used to build more power-aware and area-efficient digital systems. This approach opens doors for future exploration into applying GDI techniques in other sequential and combinational circuits—like counters, registers,

and memory elements—paving the way for low-power, high-performance digital electronics in the era of portable and energy-conscious devices.

Furthermore, the choice to implement the Schmitt Trigger using NAND gates—the universal building blocks of digital logic—ensures that the design remains modular, scalable, and easy to integrate into a variety of digital circuits. By employing the GDI technique for constructing these NAND gates, the circuit achieves remarkable reductions in power consumption, propagation delay, and silicon area when compared to traditional CMOS-based implementations [11].

This is particularly significant in the context of modern digital systems, where energy efficiency is a top priority due to the growing demand for portable, battery-operated, and resource-constrained devices such as wearables, IoT nodes, and embedded systems.

The simulations conducted as part of this work validate the effectiveness of the proposed design [13]. The GDI-based Schmitt Trigger demonstrates robust behavior in the presence of noisy, bouncing input signals, producing a clean, stable output suitable for interfacing with microcontrollers, digital counters, and other logic circuits. The power consumption analysis reveals substantial savings compared to conventional CMOS logic, confirming that GDI is a viable and superior alternative for low-power digital design. The waveform results clearly illustrate the suppression of bounce-induced glitches, thereby solving the core problem of mechanical switch instability in a hardware-efficient manner.

Beyond just solving the switch bounce problem, this thesis contributes to the broader field of low-power digital design by showcasing how fundamental circuit elements—like the Schmitt Trigger—can be reimagined using innovative design techniques like GDI. The approach taken here is modular and generalizable; the same design philosophy can be extended to other digital components, such as counters, shift registers, oscillators, and even small-scale memory cells. This opens up exciting possibilities for further research and development in the domain of ultra-low-power VLSI circuits.

Moreover, this work emphasizes the importance of rethinking standard design practices in the light of power, area, and performance constraints. While CMOS technology has been the industry standard for decades, this project demonstrates that alternative techniques like GDI can offer tangible benefits in specific use cases, particularly when power efficiency is critical. The GDI-based Schmitt Trigger design thus stands as a testament to the potential of non-conventional design methodologies in meeting the demands of modern electronics.

In conclusion, this thesis presents a practical, effective, and power-conscious

solution to the problem of switch bouncing—a solution that is not only applicable in theory but also validated through simulations and analysis. It underscores the role of GDI logic as a valuable tool in the arsenal of digital designers seeking to build circuits that are both functional and energy-efficient [19].

This work paves the way for further exploration of GDI-based circuits in more complex applications, potentially contributing to the design of next-generation low-power digital systems that are increasingly relevant in our technology-driven world.

Further we can get more better techniques of low powering in Switch Debouce circuit [17]. Also, we can make the layout of the circuit and can check the possible input and the compatibility of the outer noise with the different layers of the layout. The applications that are implemented using GDI can be used to further the implementation of circuitlike counters where the particular gate can be replaced with this low-power technique.

Similarly, some memories include so many transistors which require synchronization and very little delay, and for this application, we can use this low-power technique. Similarly, it can be used in shift registers. The application of switches has always under consideration in digital as well as analog circuits. Further we can generate a better switch debounce circuit which uses active element like BJT, MOSFET as switches [19]. However, some still remain unexplored and have the ability of providing promising benefits in terms of improvisation in area, power consumption and speed of the circuits

Bibliography

[1] N. Weste and K. Eshraghian, *Principles of CMOS Digital Design*. Reading, MA: Addison-Wesley, pp. 304–307.

[2] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 27, pp. 473–484, Apr. 1992.

[3] A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," *Proc. IEEE*, vol. 83, pp. 498–523, Apr. 1995.

[4] W. Al-Assadi, A. P. Jayasumana, and Y. K. Malaiya, "Pass-transistor logic design," *Int. J. Electron.*, vol. 70, pp. 739–749, 1991.

[5] I. S. Abu-Khater, A. Bellaouar, and M. I. Elmastry, "Circuit techniques for CMOS low-power high-performance multipliers," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1535–1546, Oct. 1996.

[6] K. Yano, Y. Sasaki, K. Rikino, and K. Seki, "Top-down pass-transistor logic design," *IEEE J. Solid-State Circuits*, vol. 31, pp. 792–803, June 1996.

[7] Z. Zhou, W. Nie, Z. Xi, and X. Wang, "A high-electrical-reliability MEMS inertial switch based on latching mechanism and debounce circuit," *IEEE Access*, 2021.

[8] A. Morgenshtein, A. Fish, and I. A. Wagner, "Gate-diffusion input (GDI) – a technique for low power design of digital circuits: analysis and characterization," *IEEE Trans. VLSI Systems*, 2002.

[9] I. M. Filanovsky and H. Baltes, "CMOS Schmitt trigger design," *IEEE Trans. Circuits Syst. I: Fundam. Theory Appl.*, vol. 41, no. 1, pp. 46–49, 2002.

[10] S. F. Al-Sarawi, "Low power Schmitt trigger circuit," *Electron. Lett.*, vol. 38, no. 18, pp. 1009–1010, 2002.

[11] C. Zhang, A. Srivastava, and P. K. Ajmera, "Low voltage CMOS Schmitt trigger circuits," *Electron. Lett.*, vol. 39, no. 24, pp. 1696–1698, 2003.

[12] S. Sarkar, H. Chatterjee, P. Saha, and M. Biswas, "8-bit ALU design using m-GDI technique," in *Proc. IEEE*, 2021.

[13] K. G. Dharani, "Design of low power and area efficient 8-bit USR using mGDI technology," *Int. J. Comput. Commun. Instrum. Eng.*, 2021.

[14] P. A. Babu and R. R. Vallabhuni, "8-bit carry look ahead adder using mGDI technique," in *Proc. IEEE*, 2021.

[15] K. Jafari and A. Amirany, "High-performance spintronic nonvolatile ternary flip-flop and universal shift register," in *Proc. IEEE*, 2021.

[16] G. Tejaswini, V. J. Naveen, and J. V. Suman, "Design of low power full adder using MGDI logic," *Eur. J. Mol. Clin. Med.*, vol. 7, no. 7, pp. 5127–5132, 2020.

[17] P. Tyagi, S. K. Singh, and P. Dua, "Gate diffusion input technique for power efficient circuits and its applications," *Int. J. Recent Technol. Eng.*, vol. 8, no. 4, pp. 511–515, 2019.

[18] R. K. Mistri, P. Prasad, and Anupriya, "IC layout design of 4-bit universal shift register using Electric VLSI design system," *Int. J. Eng. Res. Technol.*, vol. 6, no. 4, pp. 112–115, 2017.

[19] D. Soni and M. V. Shah, "Review on modified gate diffusion input technique," *Int. Res. J. Eng. Technol.*, vol. 4, no. 4, pp. 3450–3453, 2017.

Appendices

Appendix A: LTspice Simulation Code

This appendix contains the LTspice simulation code used to model and test the switch debounce circuit. It includes the testbench setup for both CMOS and GDI-based D Flip-Flop designs, along with the voltage sources, control signals, and probe configurations. These scripts were used to generate the timing waveforms and analyze power and delay performance.

Appendix B: Simulation Waveforms

This section presents the detailed waveform outputs obtained from LTspice simulations. The plots include input switch bouncing signals, corresponding outputs after debouncing, and comparative switching delays for CMOS and GDI circuits. Enlarged plots are provided for clarity and to support the analysis discussed in Chapter 3 and 4.

Appendix C: Transistor-Level Schematics

This appendix includes the transistor-level schematics of the GDI-based logic gates and the Schmitt Trigger used in the design. These were developed and simulated to verify correct logical behavior and to measure performance characteristics like area and power consumption.

Appendix D: Layout Snapshots and Fabrication Notes

If physical layout work was done using tools like Electric VLSI or Cadence Virtuoso, screenshots of the layout, DRC/LVS checks, and design specifications are included in this section. While actual fabrication was not performed, this appendix outlines the readiness of the design for a future tape-out.

MAJOR THESIS FINAL 30.05.2025_V2_removed.pdf

Delhi Technological University

Document Details

Submission ID **trn:oid:::27535:98654356**

Submission Date

May 31, 2025, 11:28 AM GMT+5:30

Download Date

May 31, 2025, 11:29 AM GMT+5:30

File Name

MAJOR THESIS FINAL 30.05.2025_V2_removed.pdf

File Size

1.5 MB

31 Pages

5,580 Words

29,430 Characters

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

50 Not Cited or Quoted 13%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

1 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

11% Internet sources
11% Publications
15% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

50 Not Cited or Quoted 16%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

1 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

11% Internet sources
11% Publications
15% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	dokumen.tips 3%
2	Submitted works	Grimsby College, South Humberside on 2024-06-02 1%
3	Submitted works	University of Technology, Sydney on 2022-09-09 1%
4	Internet	www.geeksforgeeks.org 1%
5	Publication	Animesh Bhattacharya, Amai K. Ghosh, Goutam Kumar Maity, Alina Roy. "CMOS b... <1%
6	Submitted works	Henley College Coventry, Coventry on 2023-06-29 <1%
7	Publication	Magesh Kannan, P., and K. Prathyusha. "Implementation of low power RAM in GD... <1%
8	Publication	Arkadiy Morgenshtein, Idan Shwartz, Alexander Fish. "Gate Diffusion Input (GDI) ... <1%
9	Internet	mantechpublications.com <1%
10	Submitted works	University of Dundee on 2021-03-08 <1%

11	Submitted works	University of North Carolina, Charlotte on 2013-02-17 <1%
12	Submitted works	AlHussein Technical University on 2024-05-19 <1%
13	Publication	Owen Bishop, Owen Bishop. "Electronics - Circuits and Systems", Routledge, 2019 <1%
14	Submitted works	University of Liverpool on 2013-05-08 <1%
15	Submitted works	University of North Texas on 2018-11-07 <1%
16	Internet	www.coursehero.com <1%
17	Internet	www.scribd.com <1%
18	Submitted works	Sheffield Hallam University on 2015-05-04 <1%
19	Submitted works	University of Huddersfield on 2021-12-06 <1%
20	Submitted works	University of the Highlands and Islands Millennium Institute on 2014-05-01 <1%
21	Internet	user.eng.umd.edu <1%
22	Submitted works	Taibah University on 2022-02-07 <1%
23	Internet	pdfcoffee.com <1%
24	Publication	Anantha P. Chandrakasan, Robert W. Brodersen. "Low Power Digital CMOS Desig... <1%

25 Submitted works
Charotar University of Science And Technology on 2016-10-18 <1%

26 Publication
G.S. Lee, D.A. Peterson. "Superconductive A/D converters", Proceedings of the IEE... <1%

27 Publication
Kishore Sanapala, Ramachandran Sakthivel. "Ultra-low-voltage GDI-based hybrid ... <1%

28 Submitted works
Visvesvaraya Technological University on 2014-12-05 <1%

29 Submitted works
Yonsei University on 2023-11-26 <1%

30 Internet
dokumen.pub <1%

31 Internet
www.ampheo.com <1%

32 Submitted works
Multimedia University on 2012-11-13 <1%

33 Submitted works
Savitribai Phule Pune University on 2018-03-04 <1%

34 Publication
Hafiz Md. Hasan Babu. "Multiple-Valued Computing in Quantum Molecular Biolog... <1%

35 Submitted works
SASTRA University on 2016-04-09 <1%

Prakhar Saini

Muzaffarnagar, UP, India
+91-9456866448
prakharsaini2002@gmail.com
www.linkedin.com/in/prakhar-saini-011857194

Proficient in debugging PI analysis signoff flows, optimization of simulation resources to boost computational efficiency for multi-million instance SoC's designs using Redhawk-SC.

Experience

Product Specialist Intern – Ansys Software Pvt. Ltd., Bengaluru (June 2024-Present)

- Supported multiple tier-1/2 customers in debugging static / dynamic IR and Grid Robustness check.
- Optimizing simulation resources via ROM technique for billion node count designs.
- Collaborating with peer teams to enhance product/feature quality & efficiency.

Projects

- Flat VS ROM design correlation check using their current heatmap. (Oct 2024-Dec 2024)
  - Conducted a comprehensive correlation between the flat run and the Reduced-Order Model (ROM) run of a complex design using both current & voltage heatmaps along with their respective waveforms. This analysis provided detailed insights into IR drop consistency across the design & helped in identifying disconnected or unpowered instances, ensuring the accuracy & reliability of the ROM approach.
- RTL-to-GDSII flow implementation with EMIR analysis. (Jan 2025-March 2025)
  - Reproduced a customer-reported issue in-house by replicating the scenario through a complete RTL-to-GDSII design flow using Synopsys ICC2. Performed detailed debugging to identify and resolve the root cause, improving design reliability and customer support efficiency.

Skills

- **Technical:** PDN Analysis, Debugging, Optimization
- **Programming:** Python, C, Verilog
- **Tools:** Redhawk-SC, Cadence Virtuoso, Xilinx Vivado, LTspice

Research Work

- **Switch Debounce Circuit Using GDI** (In-progress)
  - Developing a debounce circuit in a Schmitt Trigger using 45nm technology meter in Cadence Virtuoso.
  - Observed decrease in 30% power and 90% delay.
- **UAV-Based Applications in Sports Stadiums** (June 2023)
  - Investigated UAVs in indoor-outdoor stadiums under 5G interference.

Education

M.Tech in VLSI & Embedded Systems | Delhi Technological University, Delhi – CGPA: 8.24 (2023-Present)

B.Tech in ECE | Lovely Professional University, Punjab - CGPA: 7.52 (2019-2023)