# Summer Fellowship Report

On

**Integrated Circuit Design using Subcircuit feature of eSim**
**& Mixed Signal Simulation in eSim**
**& Device Modelling in eSim**

Submitted by

**Ganderla Chaithanya**
B.Tech(Electronics and Communication)
Rajiv Gandhi University of Knowledge Technologies, Nuzvid

Under the guidance of

**Mr. Sumanto Kar**
Assistant Project Manager, FOSSEE, IIT Bombay

**Prof.Kannan M. Moudgalya**
Principal Investigator -FOSSEE, IIT Bombay

July 4, 2025

# Acknowledgment

I would like to take this opportunity to express our heartfelt gratitude to the FOSSEE Team at IIT Bombay for granting us the invaluable opportunity to contribute to the design and integration of multiple sub-circuits using eSim. This experience has been instrumental in deepening our understanding of open-source EDA tools and their practical applications in the field of circuit simulation.

We are especially grateful to Prof. Kannan M. Moudgalya for his visionary guidance, continuous motivation, and support throughout the fellowship. His leadership has inspired us to explore and appreciate the significance of open-source contributions in engineering education and research.

Our sincere thanks extend to our mentors, Mr. Sumanto Kar Sir whose constant support and expert mentoring have played a vital role in the successful completion of our project. Their patience, availability, and insightful feedback helped us overcome every challenge and enriched our learning journey.

This internship with FOSSEE has not only enhanced our technical skills but also instilled in us a deeper appreciation for collaborative, open-source development. It has been a rewarding and enlightening experience—an important milestone in our journey toward a career in the semiconductor and electronics industry.

# Contents


**1 Introduction** **5**
- 1.1 eSim . . . 5
  - 1.1.1 NgSpice . . . 6
  - 1.1.2 Makerchip . . . 6
  - 1.1.3 KiCad . . . 6
  - 1.1.4 GHDL . . . 7

**2 Features of eSim** **8**

**3 Problem Statement** **9**
- 3.1 Subcircuit Design . . . 9
  - 3.1.1 Approach . . . 9
- 3.2 Mixed-Signal Simulation using NGVeri . . . 10
  - 3.2.1 Approach . . . 10
- 3.3 Device Modeling using Model Editor . . . 10
  - 3.3.1 Approach . . . 10

**4 Digital IC's** **12**
- 4.1 SN74ALS29827:10-Bit Buffers and Bus Drivers With 3-State Outputs 12
  - 4.1.1 Description . . . 12
  - 4.1.2 Pin Diagram . . . 12
  - 4.1.3 Subcircuit . . . 13
  - 4.1.4 Test Circuit . . . 14
  - 4.1.5 Output Waveforms . . . 15
- 4.2 SN54HC164 : 8-Bit Parallel-Out Serial Shift Registers . . . 16
  - 4.2.1 Description . . . 16
  - 4.2.2 Pin Diagram . . . 16
  - 4.2.3 Subcircuit . . . 16
  - 4.2.4 Test Circuit . . . 17
  - 4.2.5 Waveforms . . . 17
- 4.3 74HC194: 4 Bit Bidirectional Universal shift Register . . . 19
  - 4.3.1 Description . . . 19
  - 4.3.2 Pin Diagram . . . 19
  - 4.3.3 Subcircuit . . . 20
  - 4.3.4 Test Circuit . . . 21
  - 4.3.5 Waveforms . . . 21
- 4.4 IXDD414PI: 14 Amp Low-Side Ultrafast MOSFET Driver . . . 22
  - 4.4.1 Description . . . 22
  - 4.4.2 Pin Diagram . . . 22

4.4.3 Subcircuit . . . 22
4.4.4 Test Circuit . . . 23
4.4.5 Waveforms . . . 23
4.5 74FCT163501:Registered Transciever . . . 24
4.5.1 Description: . . . 24
4.5.2 Pin Diagram . . . 25
4.5.3 Subcircuit . . . 26
4.5.4 Test Circuit . . . 26
4.5.5 Waveforms . . . 27

**5 Analog IC's** **28**
5.1 LM13700MX:Operational Transconductance Amplifier . . . 28
5.1.1 Description . . . 28
5.1.2 Pin Diagram . . . 28
5.1.3 Subcircuit . . . 29
5.1.4 Test Circuit . . . 29
5.1.5 Waveforms . . . 30
5.2 CD4007UB : CMOS Dual Complementary Pair Plus Inverter . . . 31
5.2.1 Description . . . 31
5.2.2 Pin Diagram . . . 31
5.2.3 Subcircuit . . . 32
5.2.4 Test Circuit . . . 32
5.2.5 Waveforms . . . 33

**6 Mixed-Signal Simulation** **34**
6.1 82s126 :1k bit prom . . . 34
6.1.1 Description . . . 34
6.1.2 Verilog Code . . . 35
6.1.3 Pin Diagram . . . 38
6.1.4 Test Circuit . . . 38
6.1.5 Waveforms . . . 39
6.2 AM2901 : 4-bit Microprocessor Slice . . . 40
6.2.1 Description . . . 40
6.2.2 Verilog Code . . . 40
6.2.3 Pin Diagram . . . 43
6.2.4 Test Circuit . . . 43
6.2.5 Waveforms . . . 44
6.3 TDC1008 :8×8 Multiplier with Accumulator . . . 45
6.3.1 Description . . . 45
6.3.2 Verilog Code . . . 45
6.3.3 Pin Diagram . . . 47
6.3.4 Test Circuit . . . 48
6.3.5 Waveforms . . . 48
6.4 CRC16 Generator . . . 50
6.4.1 Description . . . 50
6.4.2 Verilog Code . . . 50
6.4.3 Pin Diagram . . . 51
6.4.4 Test Circuit . . . 52
6.4.5 Waveforms . . . 52

**7 Device Modeling** **54**
7.1 SST406 : N-Channel JFET Monolithic Dual . . . 54
7.1.1 Description . . . 54
7.1.2 SPICE Model . . . 54
7.1.3 Amplifier Configuration . . . 55
7.1.4 Frequency Response Analysis . . . 56
7.1.5 Characteristics Graphs . . . 58
7.2 2N4124 : NPN General Purpose Transistor . . . 59
7.2.1 Description . . . 59
7.2.2 SPICE Model . . . 59
7.2.3 Amplifier Configuration . . . 59
7.2.4 Frequency Response Analysis . . . 60
7.2.5 Common-Base (CB) Configuration . . . 62
7.2.6 Common-Emitter (CE) Configuration . . . 63
7.3 MUR460 : Switch-Mode Power Rectifier . . . 64
7.3.1 Description . . . 64
7.3.2 SPICE Model . . . 64
7.3.3 Forward Bias Characteristics . . . 64
7.3.4 Reverse Bias Characteristics . . . 65
7.3.5 Half-Wave Rectifier using MUR460 (Ultrafast Recovery Diode) 66
7.4 2S2088U3 : Medium power transistor . . . 67
7.4.1 Description . . . 67
7.4.2 SPICE Model . . . 67
7.4.3 Amplifier Configuration . . . 67
7.4.4 Common-Base Configuration . . . 68
7.4.5 Common-Emitter Configuration . . . 70
7.4.6 Frequency Response Analysis . . . 72
7.5 1N5819 : Schottky Barrier plastic rectifier . . . 73
7.5.1 Description . . . 73
7.5.2 SPICE Model . . . 73
7.5.3 Forward Bias Characteristics . . . 73
7.5.4 Reverse Bias Characteristics . . . 74
7.5.5 Half-Wave Rectifier using 1N5819 (Schottky Diode) . . . 75

**Bibliography** **76**

# Chapter 1

# Introduction

The FOSSEE (Free/Libre and Open Source Software for Education) project, an initiative by IIT Bombay, is part of the National Mission on Education through ICT (NMEICT), Ministry of Education, Government of India. The primary goal of FOSSEE is to promote the use of Free and Open Source Software (FOSS) tools in academia and research, offering an affordable and transparent alternative to expensive proprietary software.

FOSSEE supports a wide range of open-source software including eSim, Scilab, OpenFOAM, Python, R, OpenModelica, DWSIM, and others. Through its various projects, internships, and fellowships, FOSSEE provides students, educators, and researchers an opportunity to contribute to the development and enhancement of these tools.

The project not only encourages practical learning and skill development but also aims to foster a collaborative ecosystem where individuals can contribute meaningfully to the open-source community. By integrating FOSS into academic curricula and research, FOSSEE empowers institutions across India to achieve greater self-reliance and innovation in STEM education.

## 1.1 eSim

eSim (Electronic Simulation) is a free and open-source Electronic Design Automation (EDA) tool developed as part of the FOSSEE (Free/Libre and Open Source Software for Education) project at IIT Bombay. It is created with the aim of providing a viable open-source alternative to widely used proprietary software in the field of circuit design and simulation, such as OrCAD, PSpice, and LTspice.[1]

The tool brings together the power of multiple open-source platforms—most notably **KiCad** for schematic capture and PCB design, and **Ngspice** for circuit simulation—into a single integrated environment. eSim is designed to support the entire workflow of electronics development, from circuit design to simulation, offering users a comprehensive platform for learning and innovation.

eSim is particularly well-suited for academic institutions, students, and researchers, as it eliminates the dependency on expensive licensed software. Its development is aligned with the larger objective of the FOSSEE project: to promote self-reliance, accessibility, and transparency in technical education through the use of open-source tools.

By adopting eSim, users gain exposure to industry-relevant design processes

using freely available resources. It enables hands-on experience with modern EDA methodologies and encourages active learning and experimentation. Moreover, it supports the open-source philosophy by allowing users to contribute back to its development, ensuring continuous improvement and community-driven innovation.

In an educational context, eSim plays a crucial role in fostering practical electronics skills. It enables learners to explore circuit behavior, verify designs through simulation, and even move towards prototyping through PCB layout generation—all within a unified, license-free platform. This makes eSim a valuable asset for engineering curricula and student projects.

### 1.1.1 NgSpice

NgSpice is an open-source circuit simulator integrated into eSim for performing analog and mixed-signal simulations. Based on the SPICE simulation engine, NgSpice supports DC, AC, transient, and parametric analyses, making it suitable for analyzing a wide range of circuits. [2]
It serves as the simulation backend in eSim, enabling users to simulate circuits designed using schematic editors like KiCad.Ngspice has become an essential resource in education and research, enabling learners and engineers to simulate and test circuits efficiently, without relying on proprietary software.

### 1.1.2 Makerchip

Makerchip is a browser-based integrated development environment (IDE) for designing and simulating digital circuits using TL-Verilog (Transaction-Level Verilog). It is a powerful platform that supports modular, scalable, and clean digital design using modern hardware description methodologies. Makerchip enables real-time simulation, visualization, and debugging of digital logic, making it highly suitable for both learning and development.[3]
As part of its ongoing effort to expand support for open-source tools, FOSSEE, IIT Bombay has integrated Makerchip with eSim. This integration allows users to work with TL-Verilog-based digital components alongside analog subcircuits within the same simulation environment. It bridges the gap between analog and digital design, enabling mixed-signal circuit development in a unified and open-source platform.
The integration enhances eSim's capability as a complete circuit design platform—supporting schematic entry, analog simulation, digital design using TL-Verilog, and combined verification workflows. It also aligns with the broader goals of the FOSSEE project to make advanced design and simulation tools freely accessible for academic and professional use.

### 1.1.3 KiCad

KiCad is an open-source electronic design automation (EDA) tool integrated into eSim, a platform developed by FOSSEE, IIT Bombay, for circuit simulation and PCB design. In eSim, KiCad serves as the primary environment for schematic capture and PCB layout, allowing users to design circuits visually, assign footprints, and generate board layouts with precision. Its integration with eSim enables a seamless workflow from schematic creation to simulation using Ngspice. This allows

users—especially students and educators—to design, simulate, and validate electronic circuits in a free and open-source ecosystem without the need for expensive proprietary software. The inclusion of KiCad in eSim promotes hands-on learning, enhances understanding of electronics, and supports the development of custom ICs and systems using open tools.[4]

### 1.1.4 GHDL

GHDL is a powerful and open-source VHDL simulator that plays a crucial role in digital system design and verification. Unlike traditional simulators that may rely on proprietary software, GHDL compiles VHDL code directly into native machine code, which makes simulations faster and more efficient. It supports behavioral, structural, and RTL-level modeling, enabling users to simulate complex digital designs, such as processors, communication systems, and finite state machines. In the context of eSim, GHDL is fully integrated to allow users to design circuits using VHDL and verify their functionality through simulation, all within a single unified environment. It also supports waveform generation (via tools like GTKWave), enabling users to analyze signal behavior over time. This integration makes eSim a comprehensive platform for both analog and digital circuit simulation, supporting mixed-mode design and fostering a strong foundation in electronics and hardware description languages for students, researchers, and hobbyists.[5]

# Chapter 2

# Features of eSim

eSim offers similar capabilities and ease of use as any equivalent proprietary software for schematic creation, simulation, and PCB design, without having to pay a huge amount of money to procure licenses. Hence it can be an affordable alternative to educational institutions and SMEs. It can serve as an alternative to commercially available/licensed software tools like OrCAD, PSpice, LTspice, Xpedition, and HSPICE.

- **Schematic Design with KiCad**:Design electronic circuits easily using KiCad's user-friendly schematic capture tool integrated into eSim.
- **Netlist Creation and Circuit Simulation with Ngspice**: Generate netlists from the schematic and simulate the circuit behavior using Ngspice.
- **PCB Design and Gerber File Generation**: Design PCB layouts using KiCad and generate Gerber files required for PCB fabrication.
- **Custom Component Modeling**:Add or modify SPICE models and subcircuits using the built-in Model Builder and Subcircuit Builder tools.
- **Mixed-Signal Circuit Simulation**: Simulate circuits containing both analog and digital components in a single environment.
- **Cross-Platform Compatibility**: Available on both Ubuntu and Windows operating systems for broad accessibility.
- **Integration with OpenModelica**: Enables co-simulation with OpenModelica for system-level modeling and cyber-physical system design.
- **Cost-Effective Alternative to Proprietary Software**: eSim offers capabilities similar to commercial tools like OrCAD, PSpice, LTspice, Xpedition, and HSPICE.
- **No Licensing Cost**:Being open-source, eSim does not require costly licenses, making it highly affordable for educational institutions and small to medium enterprises (SMEs).
- **Ideal for Learning and Research**: Enables students, educators, and researchers to perform complete circuit design workflows from schematic to simulation and PCB layout in a free and open-source environment.

# Chapter 3

# Problem Statement

The objective of this project is to design and implement digital integrated circuits utilizing the Subcircuit feature of eSim, develop accurate device models using the Model Editor tool, and conduct mixed-signal simulations through NGVeri. The digital models developed as part of this work will contribute to the enhancement of the eSim component library, thereby expanding its capabilities and applicability in both academic research and industrial design workflows.

## 3.1 Subcircuit Design

Digital and analog ICs are modeled as subcircuits in eSim based on manufacturer datasheets, incorporating internal schematics, electrical characteristics, and pin configurations. These subcircuits are integrated into eSim's library to enable accurate simulation and design of real-world components for academic and industrial applications.

### 3.1.1 Approach

- **Analysing Datsheets**: Comprehensive study of datasheets from multiple vendors is conducted to extract detailed internal circuit diagrams, truth tables, logic functionality, pin configurations, and functional block descriptions
- **Subcircuit Creation**: The extracted circuit logic is implemented using eSim's schematic editor, leveraging available components and device models from the internal library.
- **Symbol Creation**: Custom schematic symbols are created based on the IC package type and accurate pin mapping to ensure proper connectivity within user-designed circuits.
- **Test Circuit Design**:Validation circuits are developed to replicate standard test conditions specified in datasheets, enabling verification of subcircuit behavior.
- **Simulation and Validation**: Simulations are performed using NgSpice to verify the functional accuracy of each subcircuit. Output waveforms are analyzed and compared with reference data provided in the datasheets to ensure fidelity.

## 3.2 Mixed-Signal Simulation using NGVeri

This stage involves leveraging eSim's NGVeri feature to perform mixed-signal simulation by integrating analog circuits with digital designs written in Verilog. NGVeri enables co-simulation of analog (NgSpice) and digital (Verilator) domains within a unified environment, supporting accurate analysis of mixed-signal systems.

### 3.2.1 Approach

- **Verilog Design Implementaion**:Digital modules are developed using Verilog, SystemVerilog, or TL-Verilog, depending on the design requirements and complexity.
- **Functional Verification using Makerchip IDE:**: Makerchip IDE is used for writing, debugging, and verifying digital logic to ensure correctness prior to integration with analog circuits.
- **NGVeri Model Generation**: Verified Verilog or TL-Verilog designs are imported into NGVeri. If TL-Verilog is used, it is first converted into SystemVerilog using SandPiper SaaS. The design is then compiled with Verilator to generate a digital model compatible with mixed-signal simulation.
- **Analog Circuit Design**: The analog components are designed using eSim's schematic editor via the KiCad interface, utilizing existing component models and custom subcircuits.
- **Mixed-Signal Co-Simulation**: The analog and digital domains are integrated and simulated simultaneously using NGVeri. NgSpice handles the analog simulation while Verilator executes the digital logic. Simulation results are analyzed using eSim's plotting tools and NgSpice waveform viewer to validate system behavior.

## 3.3 Device Modeling using Model Editor

When SPICE models are not readily available, eSim's built-in Model Editor enables users to create custom device models. Users can manually define essential device parameters, either extracted from datasheets or calculated using standard semiconductor equations. The Model Editor generates XML-based model files that can be directly integrated into the simulation environment. This capability provides enhanced flexibility, making it possible to simulate and analyze components that are not supported by predefined models or standard libraries.

### 3.3.1 Approach

  - **Parameter Extraction**: Key device parameters such as saturation current, ideality factor, junction capacitances, and breakdown voltage are extracted from datasheets or computed using standard analytical methods.

- **Model File Creation**: The extracted parameters are manually input into eSim's Model Editor, which generates XML-based model files compatible with the simulation environment.
- **Test Circuit Design**: Representative test circuits are developed to evaluate the electrical behavior of the modeled device under typical operating conditions.
- **Simulation and Validation**: The test circuits are simulated using NgSpice, and the results are analyzed by comparing them against manufacturer datasheet specifications to ensure the model's accuracy and reliability.

# Chapter 4

# Digital IC's

## 4.1 SN74ALS29827:10-Bit Buffers and Bus Drivers With 3-State Outputs

### 4.1.1 Description

The SN74ALS29827 and SN74ALS29828 are 10-bit buffers and bus drivers that provide a high-performance bus interface for wide data paths or buses carrying parity. They are functionally equivalent to AMD's AM29827 and AM29828. These devices offer:

- 3-state outputs for driving bus lines or buffering memory address registers
- pnp inputs for reduced DC loading
- Data flow-through pinout with all inputs on one side and outputs on the opposite side
- High-impedance state at power-up
- Availability in plastic small-outline (DW) and standard plastic DIP (NT) packages

[8]

### 4.1.2 Pin Diagram

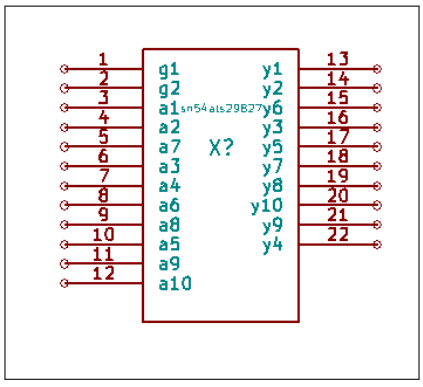


Figure 4.1: Pin Diagram of SN74ALS29827(Top View)

### 4.1.3 Subcircuit

The internal logic of the SN74ALS29827/SN74ALS29828 includes 10-bit tristate buffers. To represent this behavior in simulation and digital modeling, a Verilog-based subcircuit was developed. This subcircuit includes custom-designed tristate buffer modules to emulate the 3-state outputs.

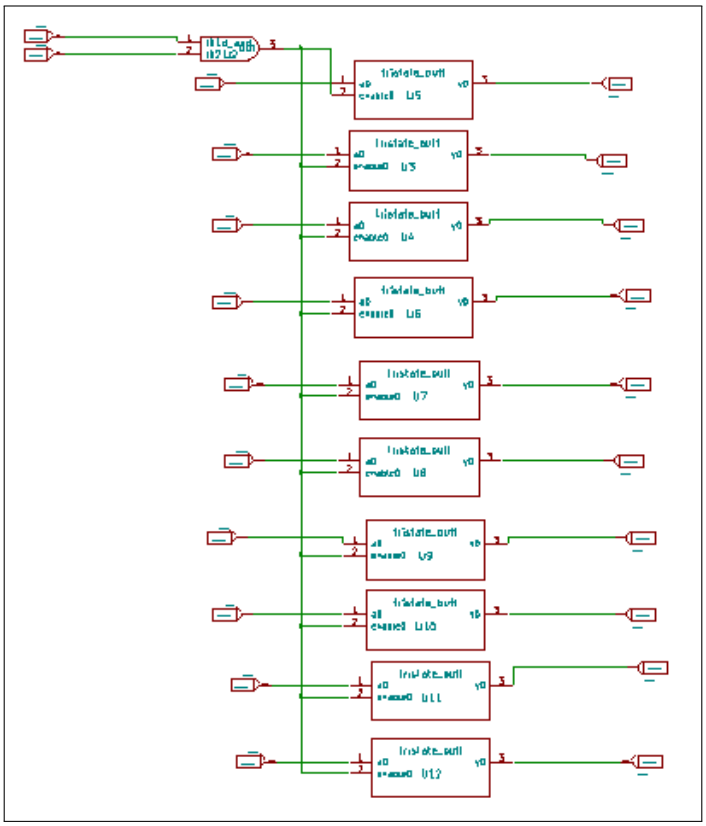

Figure 4.2: Subcircuit Diagram

#### Tristate Buffer Design

A key component of the subcircuit is the tristate buffer. The following Verilog module models a 1-bit tristate buffer where the output goes to high-impedance (‘z‘) when the enable signal is low.

```
module tristate_buff (
    input wire a,         // Input signal
    input wire en,        // Enable signal
    output wire y         // Output signal
);

assign y = en ? a : 1'bz;

endmodule
```

Listing 4.1: Tristate Buffer Verilog Code

This buffer allows controlled signal propagation. When EN = 1, the signal A is passed to Y. When EN = 0, output Y is disconnected from the bus (high-impedance). This is crucial for bus communication in ICs like the SN74ALS29827.

Here is the pin diagram of tristate buffer

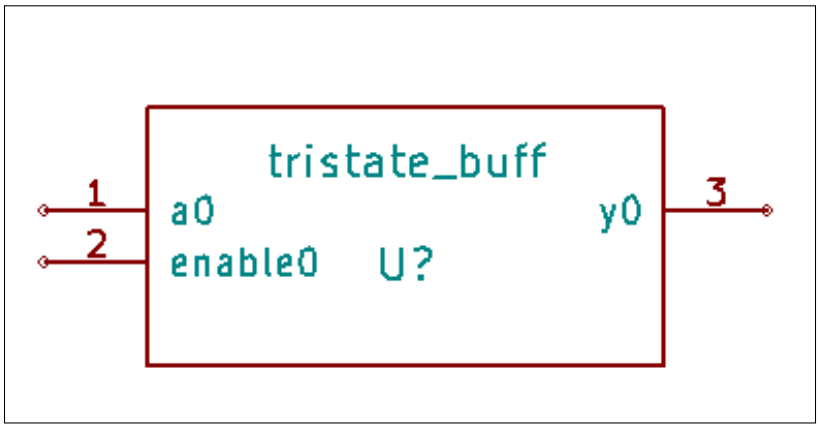


Figure 4.3: Pin diagram of Tristate Buffer

The following test circuit is used to verify the functionality of the Verilog-based tristate buffer. The testbench applies different combinations of input and enable signals to observe the behavior of the output under both enabled and disabled conditions.

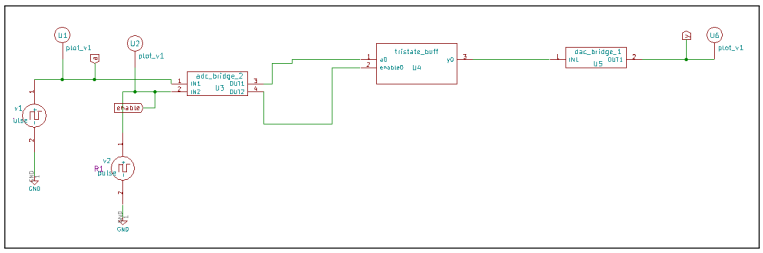

Figure 4.4: Test Circuit for Verilog-based Tristate Buffer

The output waveforms for the trisate buffer is as follows

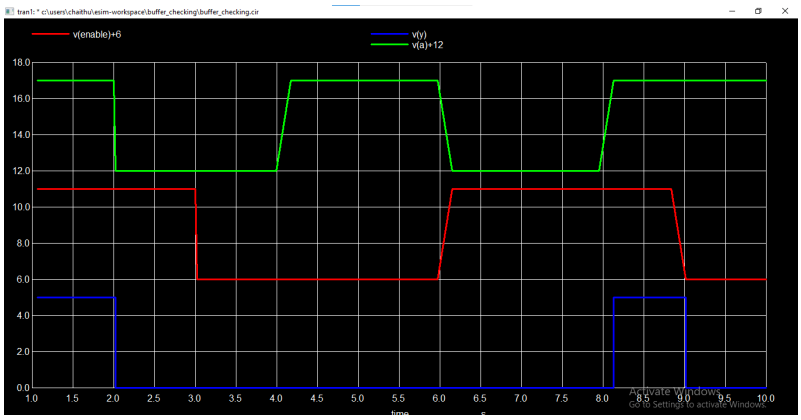


Figure 4.5: Simulation Output of Tristate Buffer

### 4.1.4 Test Circuit

The test circuit shown below is designed to verify the functional behavior of the SN54LS29827 10-bit 3-state buffer. It demonstrates the enabling and disabling of

the output through the Output Enable (OE) pin.

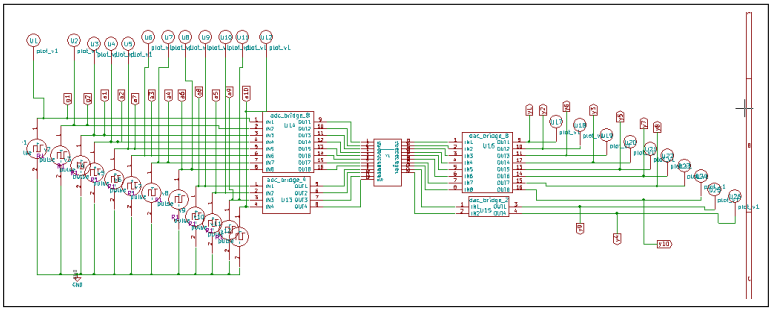

Figure 4.6: Test Circuit for SN54LS29827

In this test setup:

- A set of binary signals is applied to the input pins A1 to A10.
- The OE1 and OE2 signals are used to control whether the outputs are enabled or in high-impedance.
- When either OE1 or OE2 is high, all 10 outputs go to high-impedance state.
- When both are low, the output reflects the input data.

This configuration effectively demonstrates the use of SN54LS29827 in a bus interface scenario.

### 4.1.5 Output Waveforms

The following waveform captures the simulation or practical observation of the SN54LS29827. It clearly indicates the output response based on the OE control signals.

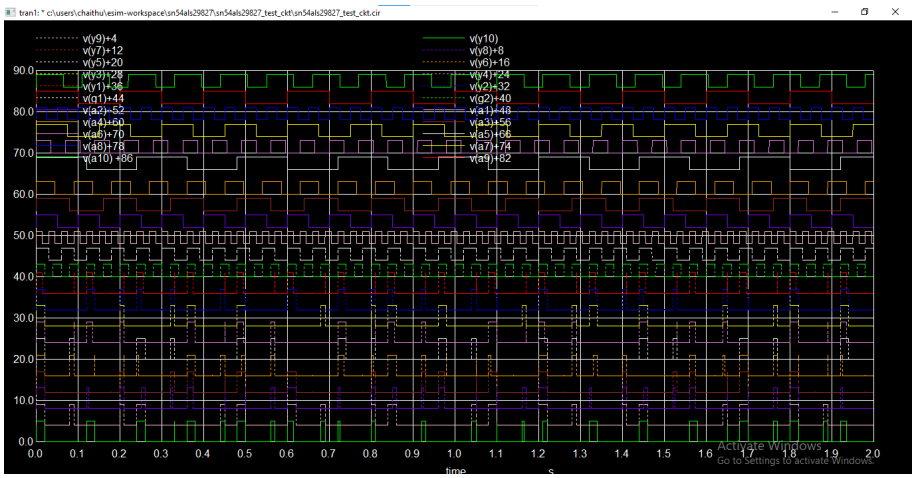


Figure 4.7: Output Waveform of SN54LS29827

## 4.2 SN54HC164 : 8-Bit Parallel-Out Serial Shift Registers

### 4.2.1 Description

The SN54HC164 is an 8-bit serial-in, parallel-out shift register. Data is entered serially through two inputs (A and B) and is clocked on the rising edge of the clock signal (CLK). It features a clear (CLR) input that asynchronously resets all flip-flops. This device is commonly used for expanding output lines or controlling LEDs and other loads.[9]

- Serial Data Input through A and B (ANDed internally)
- Asynchronous Master Reset (CLR)
- Outputs: Q0 to Q7 (Parallel outputs)
- Clock triggered on rising edge
- Package options: PDIP, SOIC, TSSOP

### 4.2.2 Pin Diagram

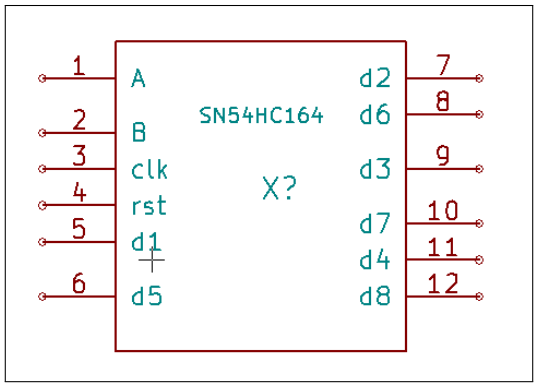


Figure 4.8: Pin Diagram of SN54HC164 (Top View)

### 4.2.3 Subcircuit

The subcircuit of SN54HC164 consists of eight edge-triggered D flip-flops connected in series to form a shift register.

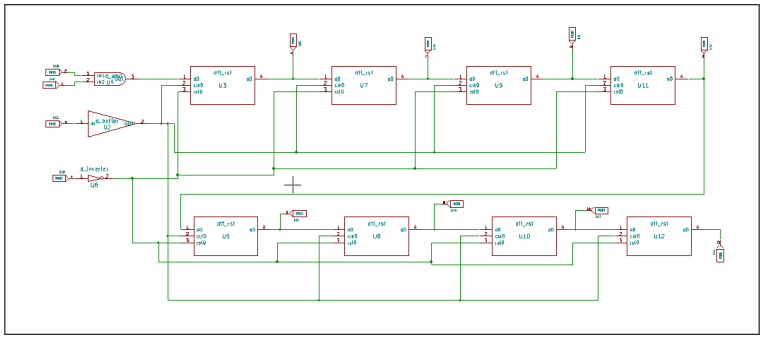

Figure 4.9: Subcircuit Diagram of SN54HC164

### D Flip-Flop Verilog Module

Each flip-flop in the SN54HC164 can be modeled using a D flip-flop. The following Verilog code represents a positive-edge-triggered D flip-flop with asynchronous reset.

```
module d_flip_flop (
    input wire clk,
    input wire reset,
    input wire d,
    output reg q
);

always @(posedge clk or posedge reset) begin
    if (reset)
        q <= 0;
    else
        q <= d;
end

endmodule
```

Listing 4.2: D Flip-Flop Verilog Code

## 4.2.4 Test Circuit

The following test circuit simulates the SN54HC164 behavior. In the test setup, a clock pulse is applied to the CLK pin to control data shifting. The serial data is fed through input A while input B is held high to enable the data flow. A low level at either A or B inhibits data entry and resets the first flip-flop to zero on the next clock rising edge. The asynchronous CLR input is used to reset all outputs (Q0–Q7) regardless of the clock.

This test circuit effectively validates the proper operation of the SN54HC164 by observing how input data is shifted and displayed at the output terminals in synchronization with the clock.

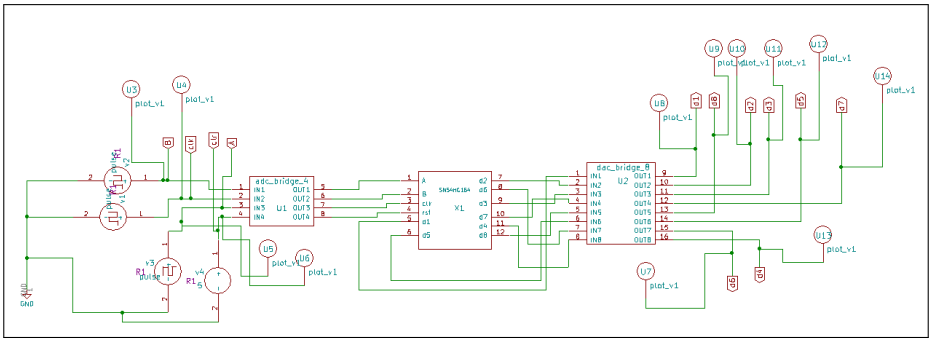

Figure 4.10: Test Circuit for SN54HC164

## 4.2.5 Waveforms

The waveform below shows the parallel outputs (Q0 to Q7) shifting data from the serial input A

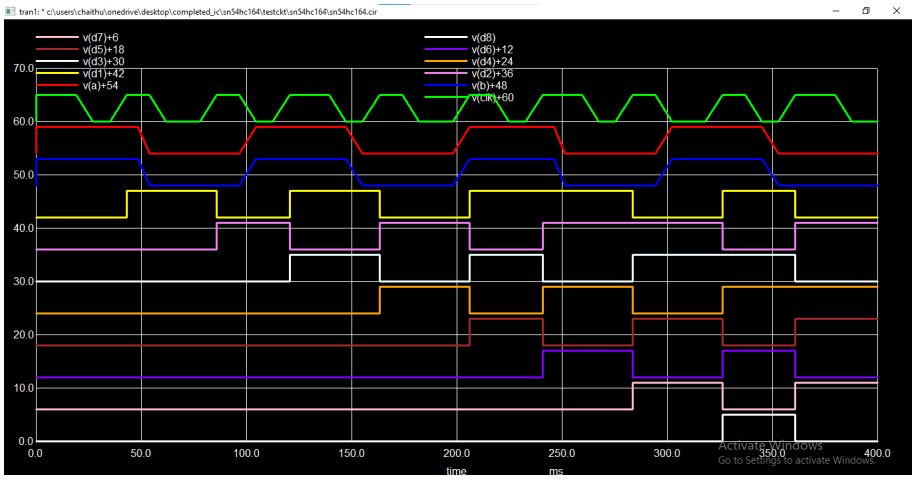


Figure 4.11: Output Waveform of SN54HC164

The waveform confirms:

- On each rising edge of the clock (CLK), the serial data at input A is latched into the first flip-flop (Q0).
- With each subsequent clock pulse, the data shifts one position to the right (Q0 to Q1, Q1 to Q2, ..., Q6 to Q7).
- The waveform shows a cascading pattern as the bits propagate through the register from Q0 to Q7.
- When the CLR (Clear) input is driven low, all outputs Q0 to Q7 immediately reset to logic low, regardless of the clock or data input.
- This output behavior verifies the SN54HC164's role as a serial-in, parallel-out register that captures and shifts data in sync with the clock signal.

## 4.3 74HC194: 4 Bit Bidirectional Universal shift Register

### 4.3.1 Description

A 4-bit bidirectional universal shift register is a sequential logic device that can perform:

- Parallel Load
- Shift Left
- Shift Right
- Hold (No Change)

The control operation is governed by two mode selection inputs (S1, S0), and a clock signal. These inputs select the desired function using an internal 4:1 multiplexer feeding each flip-flop.[10]

#### Mode Operations

The operations of the shift register are controlled by two mode control inputs (S1 and S0). The operations based on these modes are as follows:

| Mode (S1 S0) | Operation |
|---|---|
| 00 | No Change (Hold) |
| 01 | Shift Right |
| 10 | Shift Left |
| 11 | Parallel Load |

This waveform confirms correct sequential logic for all modes of operation.

### 4.3.2 Pin Diagram

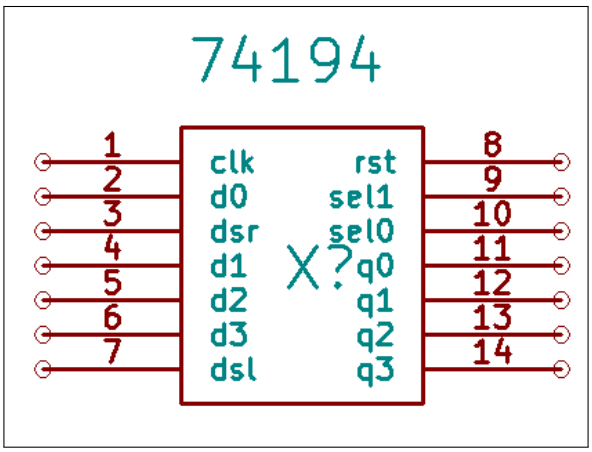


Figure 4.12: Pin Diagram of 4-Bit Universal Shift Register

### 4.3.3 Subcircuit

The subcircuit consists of four D flip-flops, each with a 4:1 multiplexer controlling its input based on the operation selected by S1 and S0.

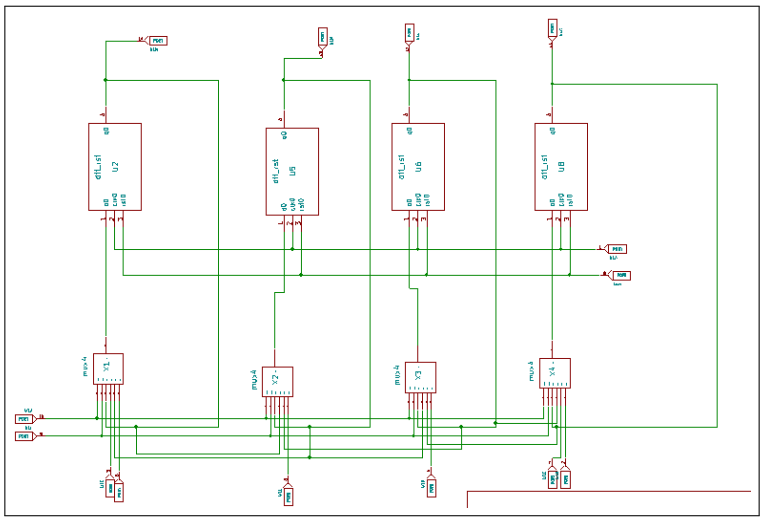

Figure 4.13: Subcircuit Diagram of 4-Bit Shift Register (with MUX 4:1)

#### MUX 4:1 Block

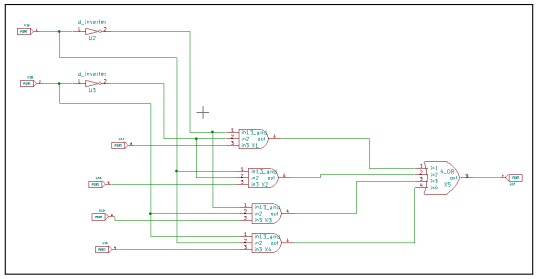

Figure 4.14: 4:1 Multiplexer used in Subcircuit

#### Test Circuit for MUX 4:1

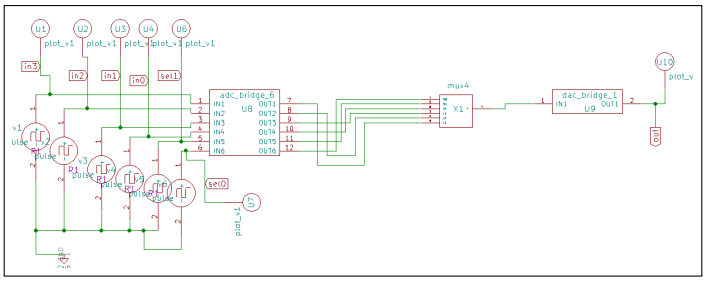

Figure 4.15: Test Circuit for 4:1 MUX

**Output Waveforms of MUX 4:1**

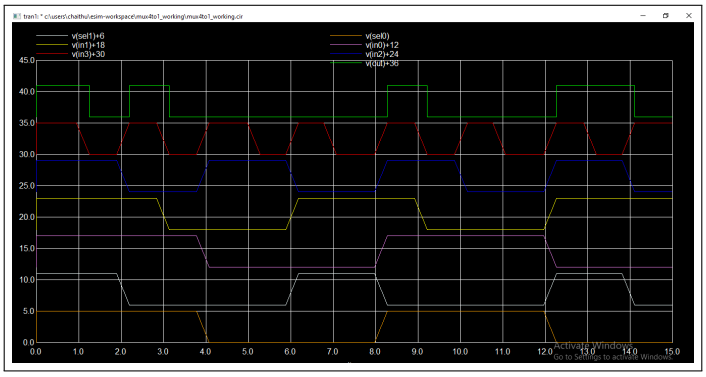

Figure 4.16: Output Waveform of 4:1 MUX

### 4.3.4 Test Circuit

This test setup applies various combinations of inputs to test the operation of the full shift register, including parallel load, shift-left, shift-right, and hold.

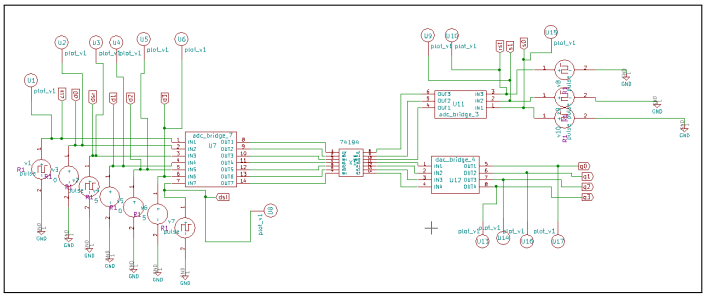

Figure 4.17: Test Circuit for 4-Bit Bidirectional Universal Shift Register

### 4.3.5 Waveforms

The waveform below shows the behavior of the shift register for all four operations. Each clock cycle demonstrates either a shift-left, shift-right, parallel load, or no change.

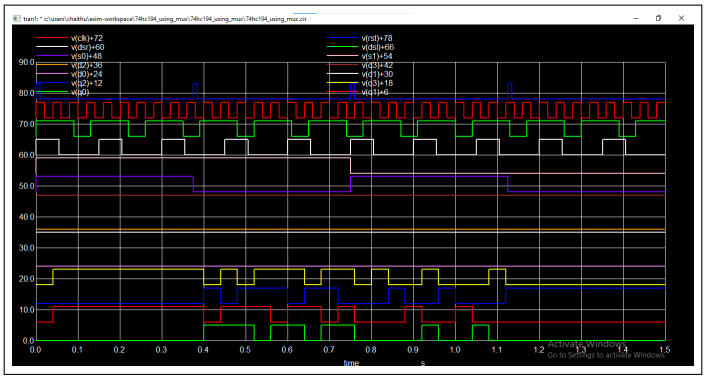

Figure 4.18: Output Waveform of the 4-Bit Bidirectional Shift Register

This waveform confirms correct sequential logic for all modes of operation.

## 4.4 IXDD414PI: 14 Amp Low-Side Ultrafast MOSFET Driver

### 4.4.1 Description

The **IXDD414PI** is a high-speed, high-current gate driver specifically designed to drive large MOSFETs and IGBTs with optimal switching performance. Some of its notable features include:

- Capable of sourcing and sinking up to **14A peak current**
- **Voltage rise and fall times less than 30ns**
- **TTL and CMOS compatible input** with latch-up immunity
- Very small internal delays to minimize **cross conduction / current shoot-through**
- **Tristate output disable mode** via ENABLE pin

These features make the IXDD414PI ideal for high-frequency, high-power applications requiring fast switching and protection mechanisms.[11]

### 4.4.2 Pin Diagram

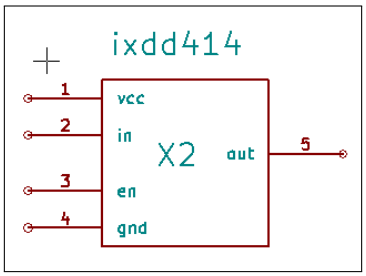


Figure 4.19: Pin Diagram of IXDD414PI Gate Driver

### 4.4.3 Subcircuit

The subcircuit of IXDD414PI includes an input buffer stage, enable control logic, and high-current output stage using push-pull configuration. This enables fast transitions and protection features.

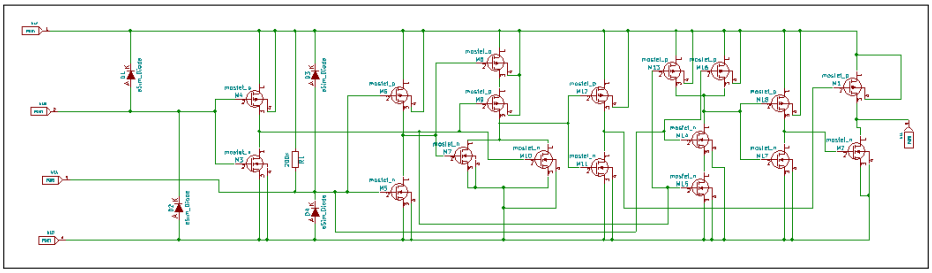

Figure 4.20: Subcircuit of IXDD414PI (Functional Block)

### 4.4.4 Test Circuit

The test circuit validates IXDD414's ability to switch a MOSFET or IGBT under various logic inputs and enable/disable conditions.

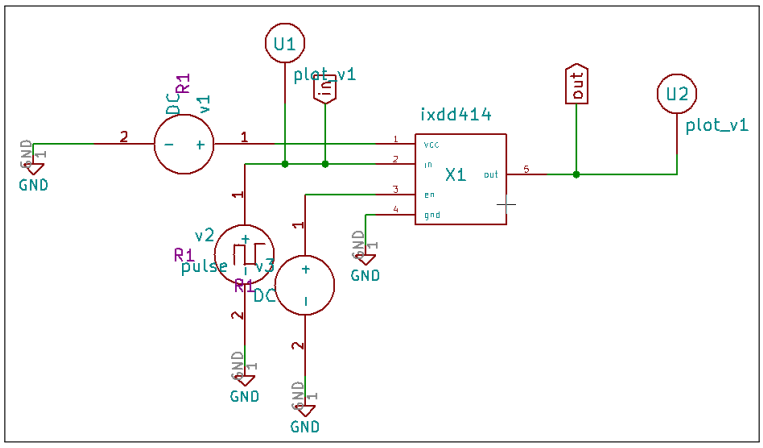


Figure 4.21: Test Circuit for IXDD414PI

### 4.4.5 Waveforms

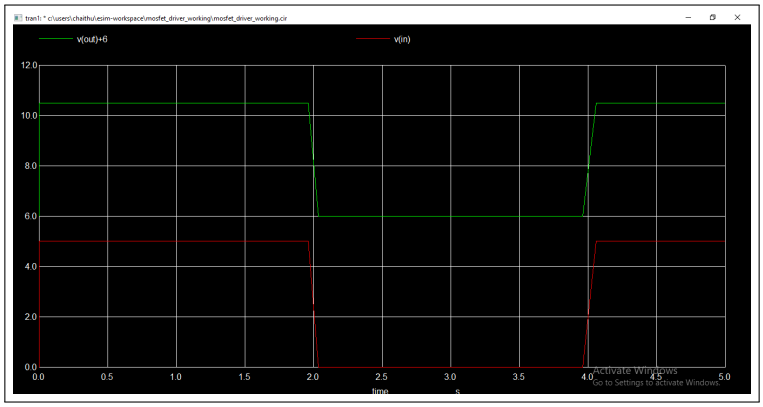


Figure 4.22: Output Waveform of IXDD414PI showing fast switching and soft turn-off

The waveform illustrates:

- High-speed transitions at the output
- Soft turn-off behavior when Enable is pulled low
- Minimal delay between input logic and output response

## 4.5 74FCT163501:Registered Transciever

### 4.5.1 Description:

The **FCT163501** 18-bit registered transceivers are high-speed, low-power bus transceivers built with advanced dual-metal CMOS technology. These devices are optimized for high-performance memory and I/O subsystems. [12]

**Key features**

- Combines **D-type latches and D-type flip-flops**
- Supports **transparent, latched, and clocked data modes**
- **Output-enable (OEAB/OEBA), latch-enable (LEAB/LEBA), and clock (CLKAB/CLKBA)** for both directions
- Flow-through pin organization simplifies PCB layout
- **Built-in current-limiting resistors** reduce undershoot and ground bounce
- All inputs have **hysteresis for improved noise immunity**

**1-Bit Registered Transceiver: Pin Diagram**

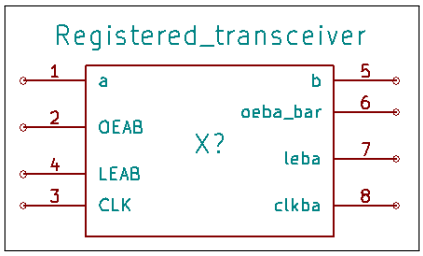


Figure 4.23: Pin Diagram of 1-Bit Registered Transceiver

**1-Bit Subcircuit**

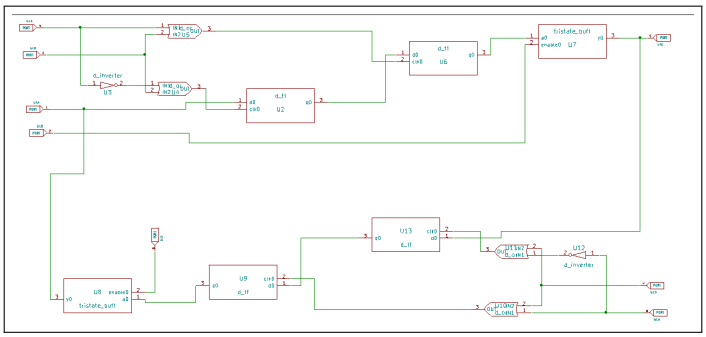

Figure 4.24: Subcircuit of 1-Bit Registered Transceiver

### 1-Bit Test Circuit

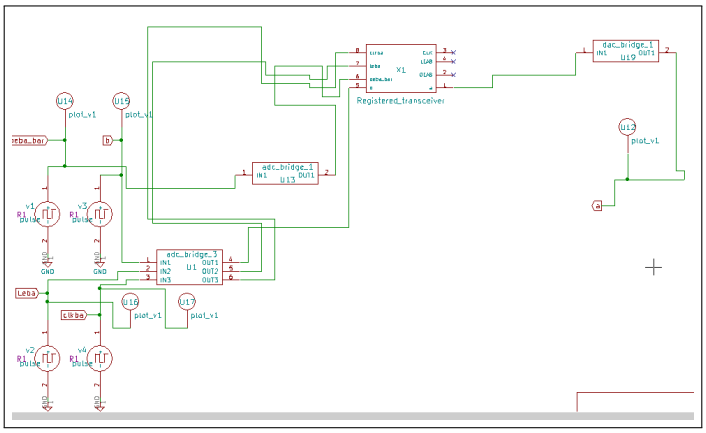


Figure 4.25: Test Circuit for 1-Bit Registered Transceiver

### 1-Bit Output Waveforms

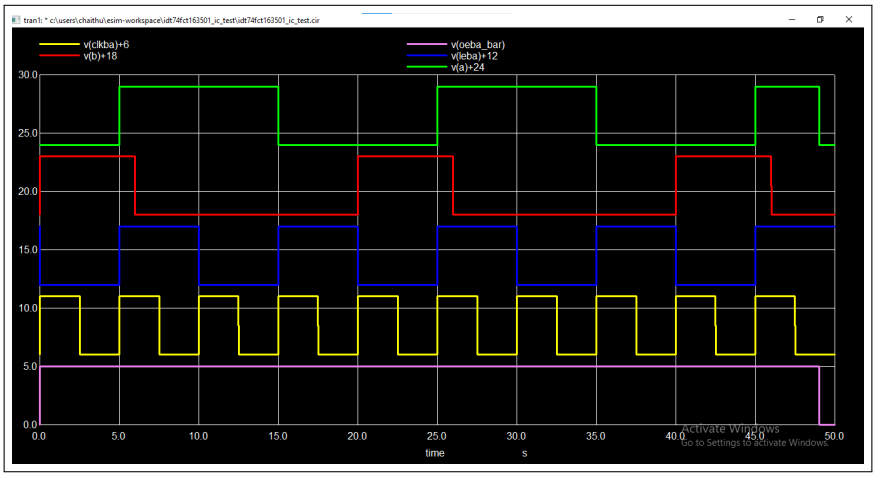


Figure 4.26: Output Waveform of 1-Bit Registered Transceiver

## 4.5.2 Pin Diagram

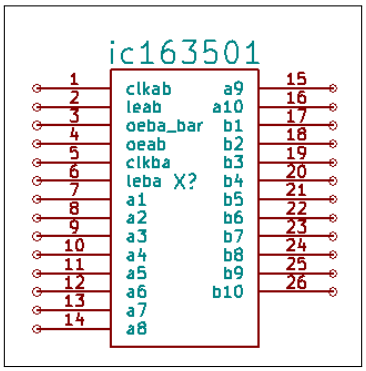


Figure 4.27: Pin Diagram of 10-Bit Registered Transceiver

### 4.5.3 Subcircuit

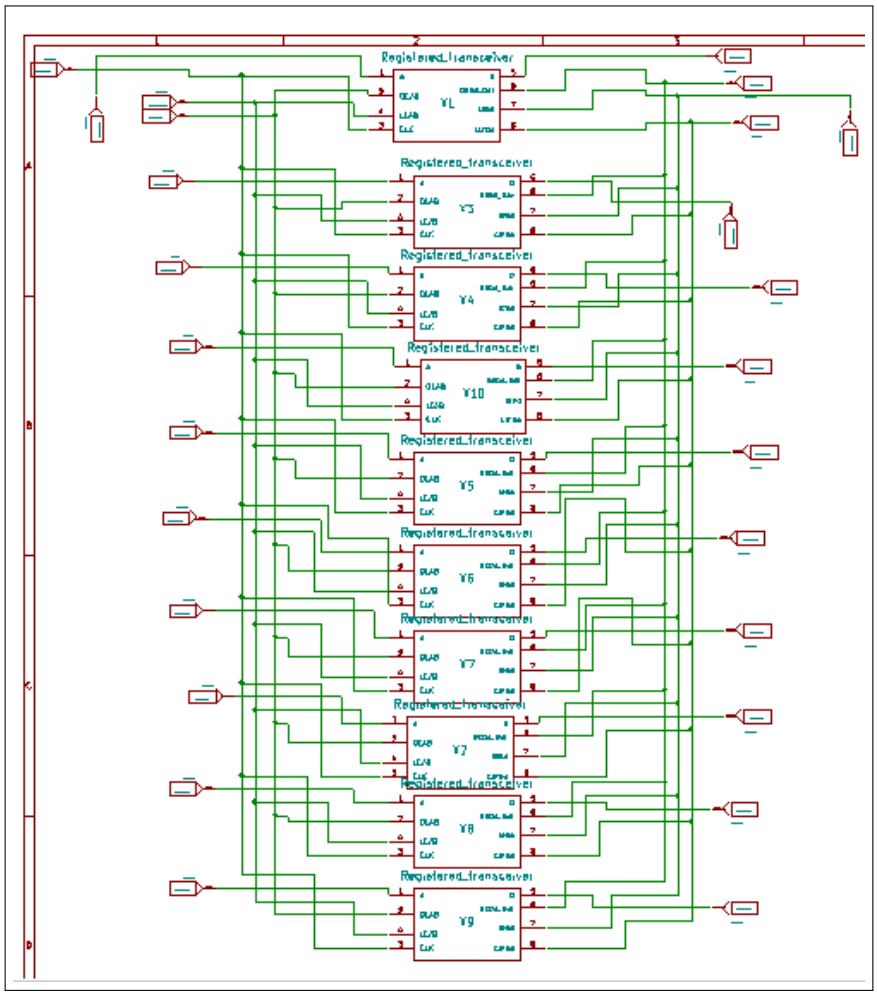


Figure 4.28: Subcircuit of 10-Bit Registered Transceiver

### 4.5.4 Test Circuit

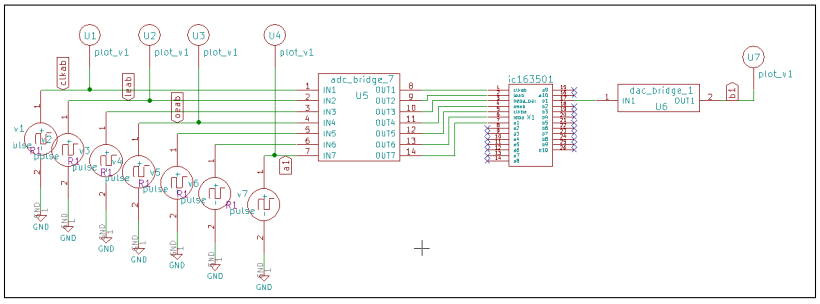


Figure 4.29: Test Circuit for 10-Bit Registered Transceiver

### 4.5.5 Waveforms

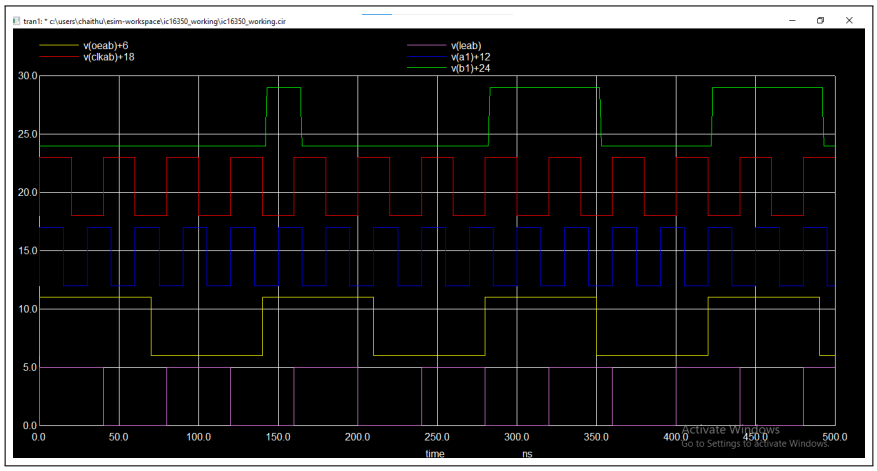


Figure 4.30: Output Waveform of 10-Bit Registered Transceiver

The output waveforms of the IC 163501 registered transceiver depend on the clock (CLKAB/CLKBA), latch enable (LEAB/LEBA), and output enable (OEAB/OEBA) signals. Data is latched on the rising edge of the clock when the corresponding latch enable is high. The outputs follow the latched data only when the output enable is active (logic low). When OE is high, the output enters a high-impedance (Hi-Z) state. Thus, the waveform shows synchronized data transitions or flat regions depending on the control signals.

# Chapter 5

# Analog IC's

## 5.1 LM13700MX:Operational Transconductance Amplifier

### 5.1.1 Description

The **LM13700MX** is a dual OTA (Operational Transconductance Amplifier) featuring differential inputs, high-impedance outputs, linearizing diodes, and buffer stages. It's widely used in analog signal processing such as voltage-controlled amplifiers, filters, and oscillators.[13]

- Dual OTA with differential inputs
- Linearizing diodes for distortion reduction
- Includes Darlington buffer output
- Suitable for voltage-controlled signal processing

### 5.1.2 Pin Diagram

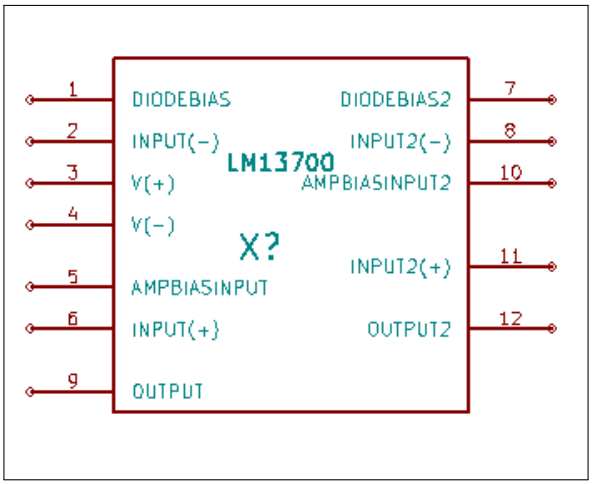


Figure 5.1: Pin Diagram of LM13700MX

Shows input, output, diode, and bias pins for both OTAs in the dual package.

### 5.1.3 Subcircuit

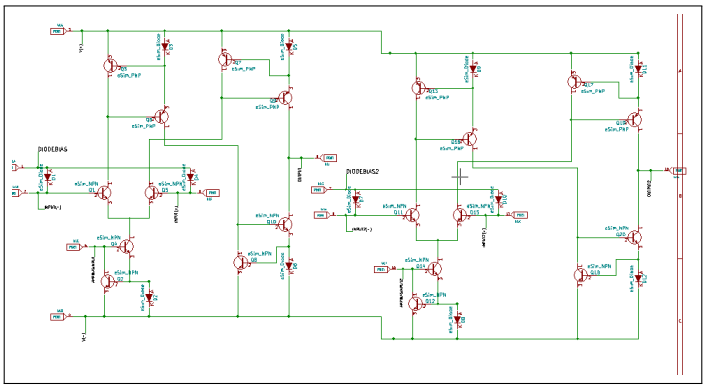

Figure 5.2: Subcircuit Representation of LM13700 OTA Core

Illustrates the internal analog structure (e.g., transconductance stage, current source, and output buffer).

### 5.1.4 Test Circuit

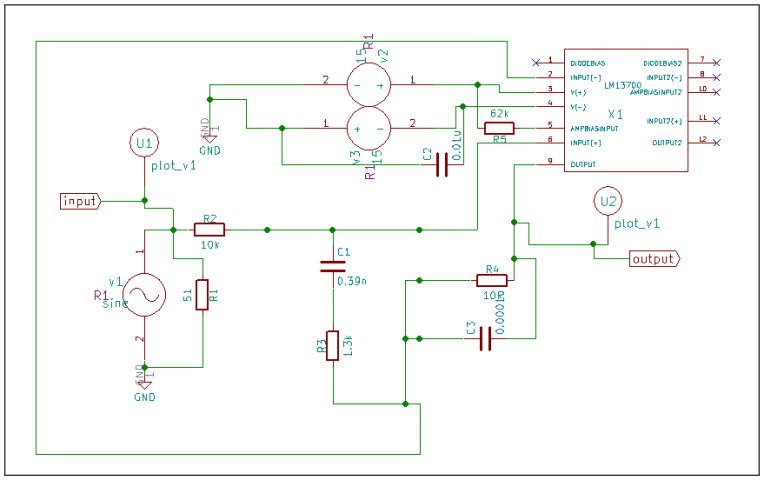


Figure 5.3: LM13700 Configured as Unity Gain Follower

This configuration uses the internal buffer to implement a voltage follower behavior with high linearity.

### 5.1.5 Waveforms

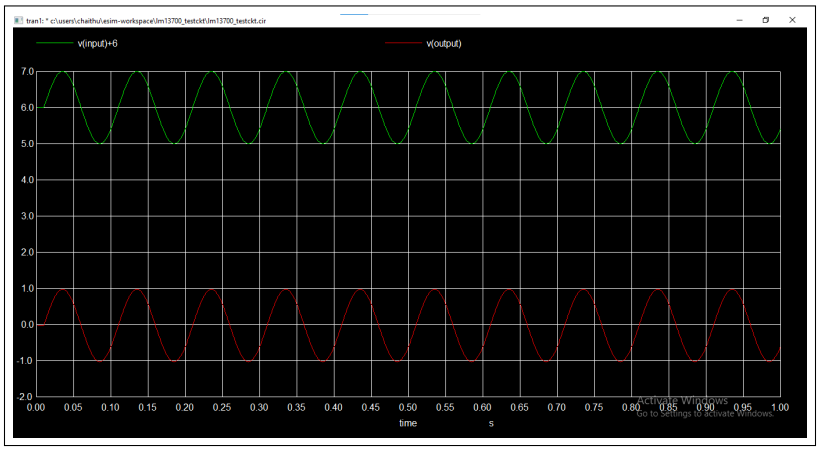


Figure 5.4: Input and Output Signals of Unity Gain Follower

When configured as a unity gain follower, the LM13700MX outputs a waveform that closely follows the input signal with minimal distortion and no amplification. The output voltage mirrors the input voltage, showing the same amplitude and phase. Any delay or deviation is minimal, depending on the bias current and load conditions. The waveform confirms linear tracking, making it suitable for buffering and impedance matching in analog signal paths.

## 5.2 CD4007UB : CMOS Dual Complementary Pair Plus Inverter

### 5.2.1 Description

The CD4007UB types are comprised of three n-channel and three p-channel enhancement-type MOS transistors. The transistor elements are accessible through the package terminals to provide a convenient means for constructing various typical circuits. This IC is widely used in education, prototyping, and low-power analog/digital circuit applications due to its ease of use and availability. It is particularly valuable for understanding the behavior of discrete MOSFETs and how they are used in CMOS logic. The CD4007UB types are supplied in 14-lead hermetic dual-in-line ceramic packages (F3A suffix), 14-lead dual-in-line plastic packages (E suffix), 14-lead small-outline packages (M, MT, M96, and NSR suffixes), and 14-lead thin shrink small-outline packages (PW and PWR suffixes)[7]

**Features**

- Standardized symmetrical output characteristics
- Medium speed operation: $t_{PHL}, t_{PLH}$ 30 ns (typ.) at 10 V
- 100% tested for quiescent current at 20 V
- Meets all JEDEC Tentative Standard No. 138 for "B" series CMOS
- Maximum input current:
  - 1 $\mu$A at 18 V (over full temperature range)
  - 100 nA at 18 V and 25°C

### 5.2.2 Pin Diagram

The following figure shows the pin configuration of the CD4007UB IC.

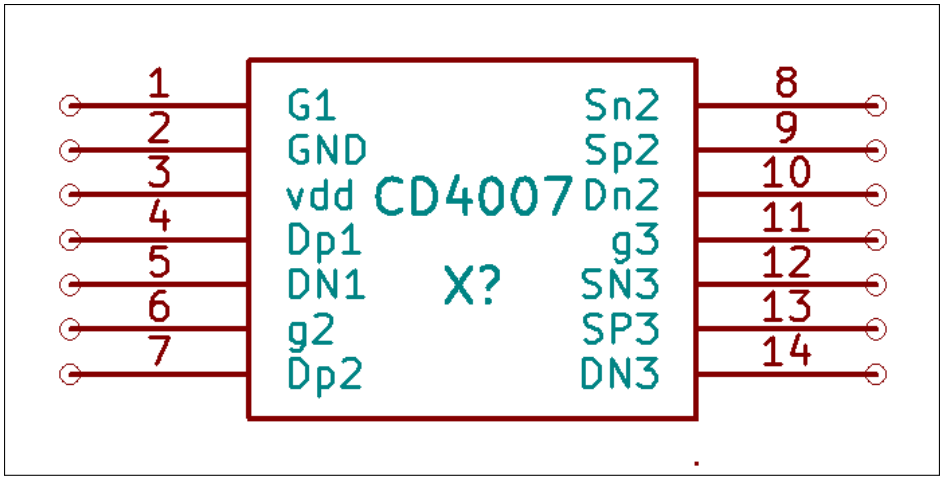


Figure 5.5: Pin Diagram of CD4007UB

### 5.2.3 Subcircuit

The internal configuration of the CD4007UB can be represented as a subcircuit as shown below.

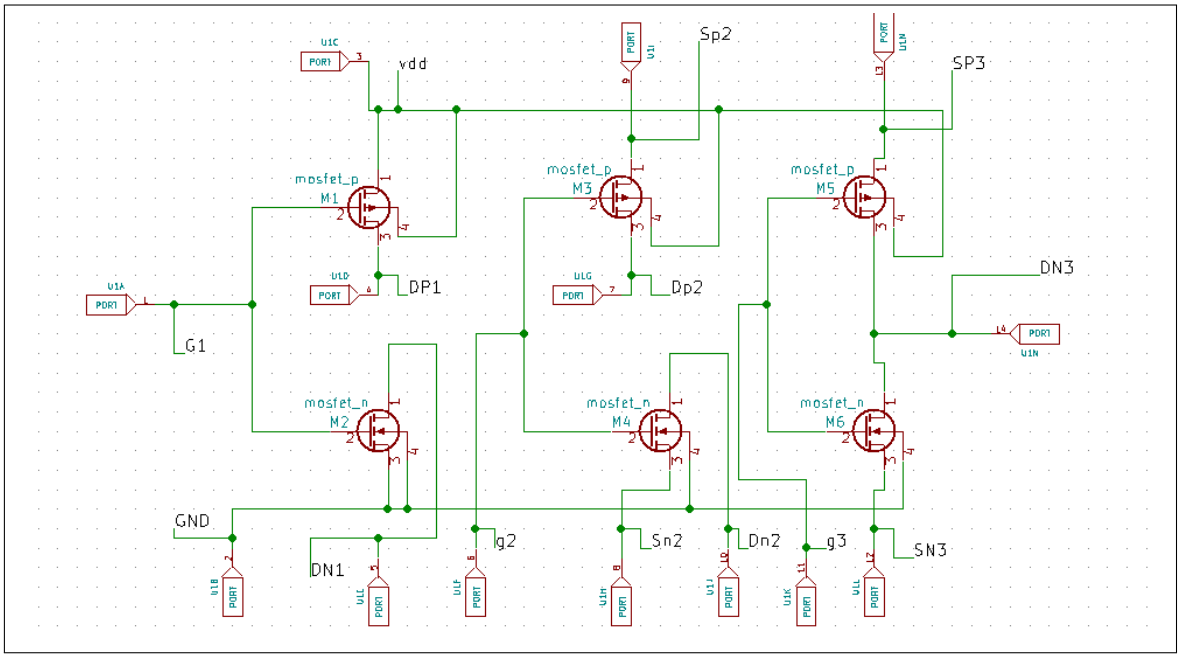


Figure 5.6: Subcircuit Diagram of CD4007UB

### 5.2.4 Test Circuit

The following test setup demonstrates the working of CD4007UB in a practical configuration.

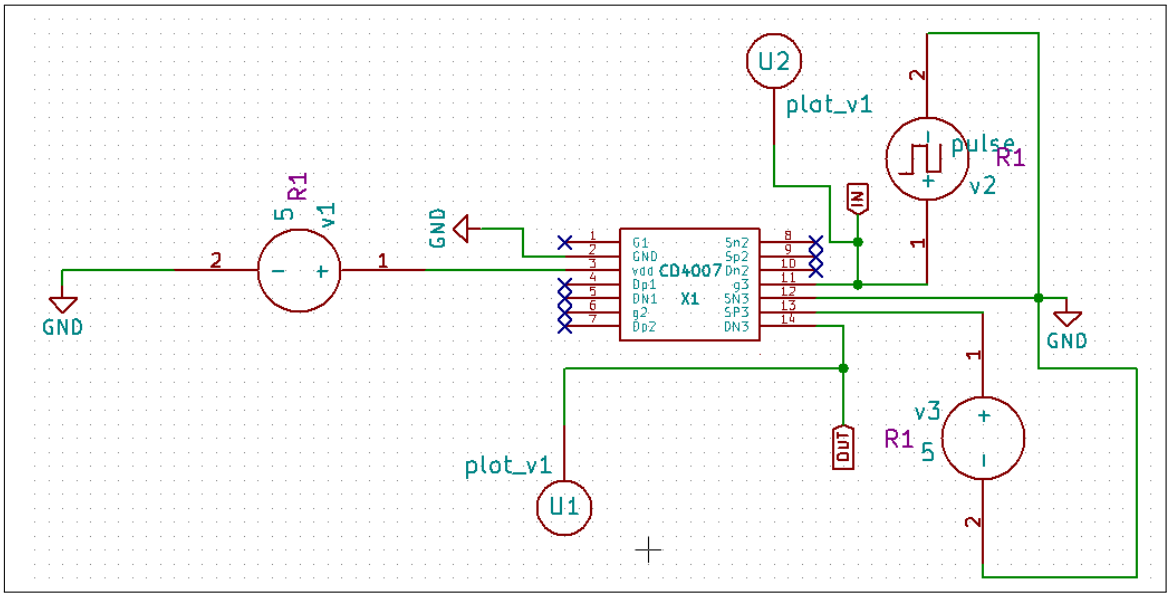


Figure 5.7: Test Circuit for CD4007UB

### 5.2.5 Waveforms

The input waveform corresponding to the test circuit is shown below.

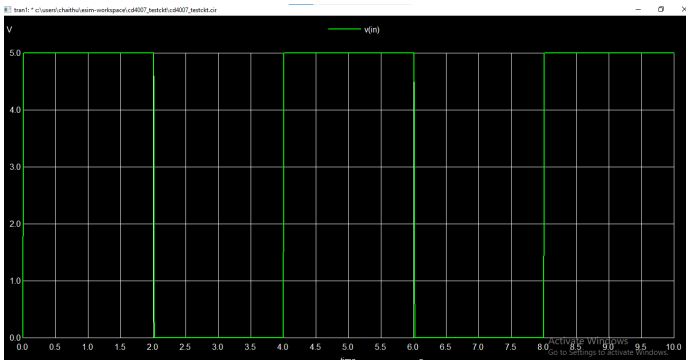


Figure 5.8: input Waveforms of the Test Circuit

The waveform corresponding to the test circuit is shown below.

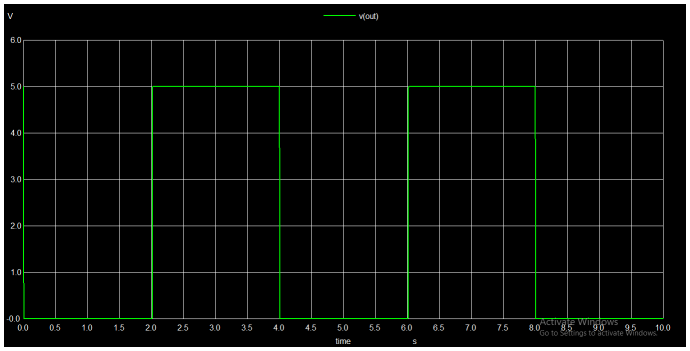


Figure 5.9: output Waveforms of the Test Circuit

# Chapter 6

# Mixed-Signal Simulation

The Mixed-Signal Simulation capability in eSim enables the co-simulation of analog and digital domains by integrating SPICE-based analog circuits with Verilog-based digital logic. This is facilitated through the NgVeri interface, which combines the simulation strengths of NgSpice for analog behavior and Verilator for digital logic. By allowing simultaneous simulation of both domains, mixed-signal co-simulation supports comprehensive analysis of systems where analog and digital components interact—such as ADCs, DACs, and sensor interfaces—ensuring accurate and efficient system-level verification.

## 6.1 82s126 :1k bit prom

### 6.1.1 Description

The **82S126** is a bipolar 1K-bit (1024-bit) Programmable Read-Only Memory (PROM), typically organized as 256 words × 4 bits. It is a high-speed memory device used for lookup tables, logic replacement, microcode, and address decoding. [14]

**Key features :**

- Organized as **256 × 4 bits** (total 1024 bits)
- **Bipolar technology** offering fast access times
- Used for implementing **logic functions and memory tables**
- Inputs: **A0 to A7**, Outputs: **O0 to O3**
- Chip Enable (CE) and Output Enable (OE) signals for memory control

### Block Diagram

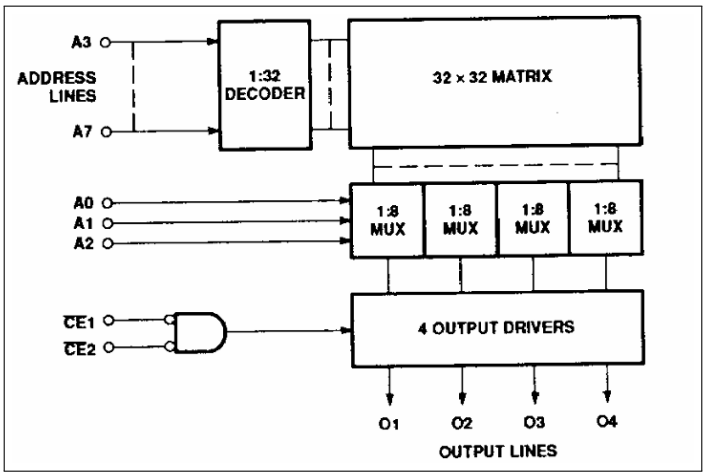


Figure 6.1: Block Diagram of 82S126 PROM

## 6.1.2 Verilog Code

The Verilog code below implements a 256 × 4 PROM model based on the 82S126 logic.

```
module decoder_5to32 (
    input [4:0] addr,
    output reg [31:0] decode_out
);
    always @(*) begin
        decode_out = 32'b0;
        decode_out[addr] = 1'b1;
    end
endmodule
module mux_1to8 (
    input [7:0] in,
    input [2:0] sel,
    output out
);
    assign out = in[sel];
endmodule
module matrix_32x32 (
    input [31:0] row_select, // From decoder
    output [31:0] row_data   // Selected row output
);
    reg [31:0] mem [0:31]; // 32 rows, 32 bits each

    integer i;
    assign row_data = (
        row_select[0]  ? mem[0]  :
        row_select[1]  ? mem[1]  :
        row_select[2]  ? mem[2]  :
        row_select[3]  ? mem[3]  :
        row_select[4]  ? mem[4]  :
        row_select[5]  ? mem[5]  :
        row_select[6]  ? mem[6]  :
        row_select[7]  ? mem[7]  :
```

```
33          row_select[8]  ? mem[8]  :
34          row_select[9]  ? mem[9]  :
35          row_select[10] ? mem[10] :
36          row_select[11] ? mem[11] :
37          row_select[12] ? mem[12] :
38          row_select[13] ? mem[13] :
39          row_select[14] ? mem[14] :
40          row_select[15] ? mem[15] :
41          row_select[16] ? mem[16] :
42          row_select[17] ? mem[17] :
43          row_select[18] ? mem[18] :
44          row_select[19] ? mem[19] :
45          row_select[20] ? mem[20] :
46          row_select[21] ? mem[21] :
47          row_select[22] ? mem[22] :
48          row_select[23] ? mem[23] :
49          row_select[24] ? mem[24] :
50          row_select[25] ? mem[25] :
51          row_select[26] ? mem[26] :
52          row_select[27] ? mem[27] :
53          row_select[28] ? mem[28] :
54          row_select[29] ? mem[29] :
55          row_select[30] ? mem[30] :
56          mem[31]
57      );
58
59      initial begin
60          for (i = 0; i < 32; i = i + 1)
61              mem[i] = 32'h00000000; // Initialize with zeros
62        mem[0] = 32'b10101010100000000010101010101010;
63        mem[1] = 32'b10101010100011000010101010101111;
64
65      mem[3] = 32'h6F20574F; // ASCII 'o WO'
66      mem[4] = 32'h524C4421; // ASCII 'RLD!'
67      mem[5] = 32'b00000000_00000000_11111111_00000000;  // One byte
            high
68      mem[6] = 32'b11111111_00000000_00000000_00000000;  // MSB byte
            high
69      mem[7] = 32'b11110000_00001111_10101010_01010101;  // Bit
            pattern
70      mem[8] = 32'b00001111_11110000_11001100_00110011;  // Bit
            pattern
71      mem[9] = 32'hDEADBEEF;  // Debugging constant
72
73      end
74  endmodule
75  module bipolar_prom (
76      input [7:0] address, // A0-A7
77      input CE1, CE2,
78      output [3:0] data     // O1-O4
79  );
80      wire [4:0] row_addr = address[7:3];  //  A 3 A7
81      wire [2:0] col_sel = address[2:0];   //  A 0 A2
82
83      wire [31:0] row_select;
84      wire [31:0] row_data;
85      wire [7:0] col_group0, col_group1, col_group2, col_group3;
86
```

```
87      // Enable signal
88      wire enable = CE1 & CE2;
89
90      // Decoder
91      decoder_5to32 dcd (.addr(row_addr), .decode_out(row_select));
92
93      // Matrix
94      matrix_32x32 mem32x32 (.row_select(row_select), .row_data(
            row_data));
95
96      // Divide 32-bit row into 4 groups of 8
97      assign col_group0 = row_data[7:0];
98      assign col_group1 = row_data[15:8];
99      assign col_group2 = row_data[23:16];
100     assign col_group3 = row_data[31:24];
101
102     // 4 MUXes for outputs
103     wire o1, o2, o3, o4;
104
105     mux_1to8 mux0 (.in(col_group0), .sel(col_sel), .out(o1));
106     mux_1to8 mux1 (.in(col_group1), .sel(col_sel), .out(o2));
107     mux_1to8 mux2 (.in(col_group2), .sel(col_sel), .out(o3));
108     mux_1to8 mux3 (.in(col_group3), .sel(col_sel), .out(o4));
109
110     assign data = enable ? {o4, o3, o2, o1} : 4'bzzzz; // Tri-state
            when disabled
111 endmodule
```

Listing 6.1: Behavioral Verilog Model of 82S126 PROM

### EDA Playground Output Waveform

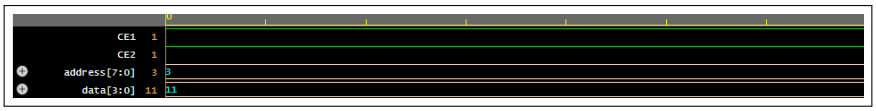


Figure 6.2: Output Waveform of 82S126 PROM Simulated in EDA Playground

The above waveform was generated using Verilog simulation on EDA Playground, validating address-to-data mapping behavior of the PROM module.

### 6.1.3 Pin Diagram

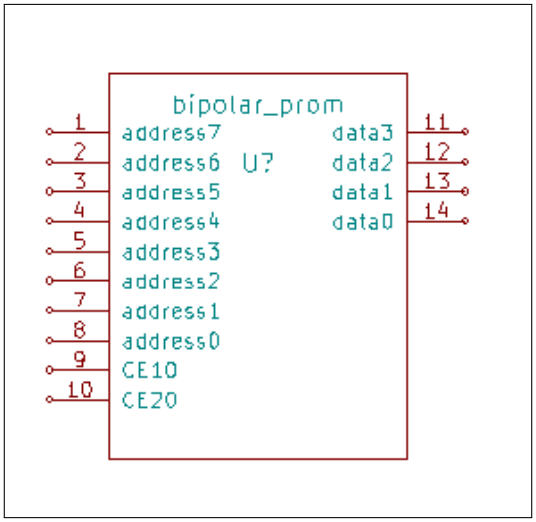


Figure 6.3: Pin Diagram of 82S126 PROM

The pin diagram shows address inputs (A0–A7), data outputs (O0–O3), and control signals (CE, OE) of the PROM.

### 6.1.4 Test Circuit

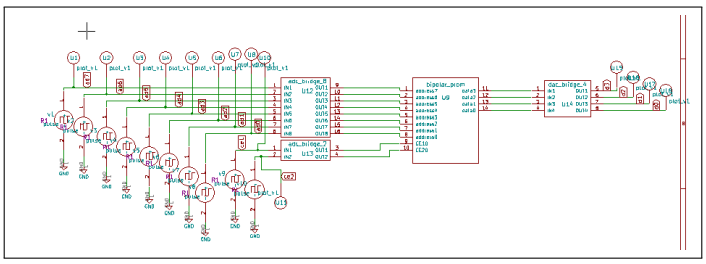

Figure 6.4: Test Circuit for 82S126 PROM

The test circuit verifies the PROM's read operation by varying address and control signals while observing the output.

### 6.1.5 Waveforms

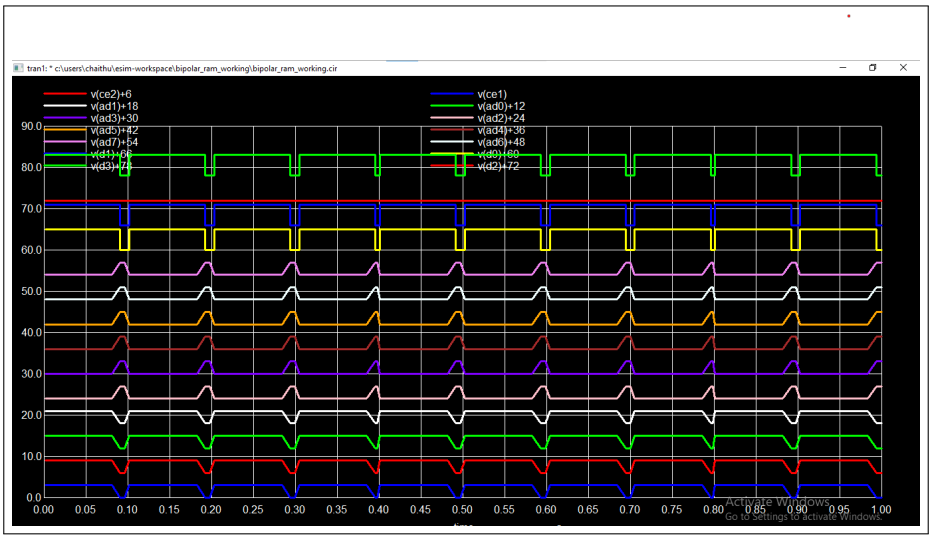


Figure 6.5: Output Waveform showing PROM behavior

The waveform confirms correct data output for different address values when CE and OE are active.

#### NgSpice Output

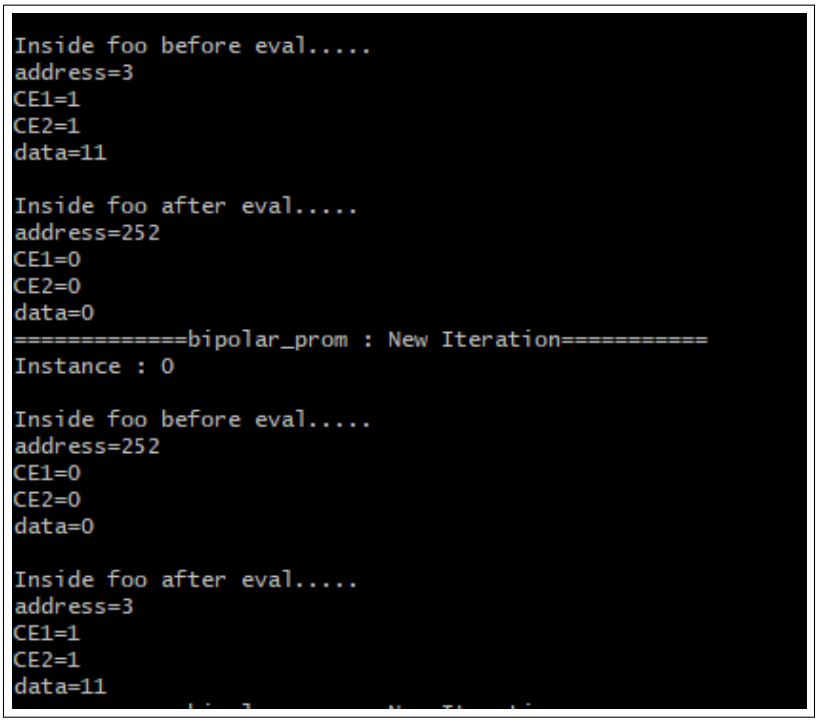


Figure 6.6: NgSpice Simulation Output for 82S126 PROM

The simulation plot from NgSpice demonstrates voltage-level behavior and validates the PROM's timing and logic operation.

## 6.2 AM2901 : 4-bit Microprocessor Slice

### 6.2.1 Description

The **AMD AM2901** is a 4-bit bit-slice ALU designed in the 1970s. It features a register file, arithmetic logic unit (ALU), and control logic. It was widely used in the design of CPUs, controllers, and high-speed logic systems.[15]

**Key features:**

- 4-bit wide register file and ALU
- 16 general-purpose registers
- Supports arithmetic, logic, shift, and register transfer operations
- Expandable to n-bit data width using multiple slices
- High performance and modular design for microprogrammed control units

### 6.2.2 Verilog Code

```
module am2901(
    input wire [3:0] A_addr,
    input wire [3:0] B_addr,
    input wire [8:0] I,
    input wire [3:0] D_in,
    input wire CP,
    input wire Cn,
    input wire OE,

    output wire [3:0] Y,
    output wire F0,
    output wire F3,
    output wire Cn4,
    output wire G,
    output wire P,
    output wire OVR
);

// Register file and special registers
reg [3:0] regfile [0:15];
reg [3:0] Q;
reg [3:0] RAM;

// ALU signals
reg [3:0] aluA, aluB;
wire [4:0] aluResult;

// Instruction decoding
wire [2:0] src_select = I[2:0];
wire [2:0] func_select = I[5:3];
wire [2:0] dest_select = I[8:6];

// ALU operand selection
always @(*) begin
```

```
35      case (src_select)
36          3'b000: aluA = regfile[A_addr];
37          3'b001: aluA = regfile[B_addr];
38          3'b010: aluA = Q;
39          3'b011: aluA = D_in;
40          3'b100: aluA = 4'b0000;
41          3'b101: aluA = 4'b0000;
42          3'b110: aluA = 4'b0000;
43          3'b111: aluA = 4'b1111;
44      endcase
45
46      case (src_select)
47          3'b000: aluB = regfile[B_addr];
48          3'b001: aluB = regfile[A_addr];
49          3'b010: aluB = D_in;
50          3'b011: aluB = Q;
51          3'b100: aluB = regfile[A_addr];
52          3'b101: aluB = regfile[B_addr];
53          3'b110: aluB = 4'b1111;
54          3'b111: aluB = 4'b1111;
55      endcase
56  end
57
58  // ALU operation
59  assign aluResult =
60      (func_select == 3'b000) ? (aluA + aluB + Cn) :
61      (func_select == 3'b001) ? (aluA - aluB - ~Cn) :
62      (func_select == 3'b010) ? {1'b0, aluA & aluB} :
63      (func_select == 3'b011) ? {1'b0, aluA | aluB} :
64      (func_select == 3'b100) ? {1'b0, aluA ^ aluB} :
65      (func_select == 3'b101) ? {1'b0, ~aluA & aluB} :
66      (func_select == 3'b110) ? {1'b0, aluA} :
67      {1'b0, aluB};
68
69  // Rising edge: write Q and RAM
70  always @(posedge CP) begin
71      case (dest_select)
72          3'b000: Q <= aluResult[3:0];
73          3'b001: RAM <= aluResult[3:0];
74          3'b010: begin Q <= aluResult[3:0]; RAM <= aluResult[3:0];
                end
75          3'b110: begin // Right shift
76              Q <= {Cn, Q[3:1]};
77              RAM <= {Q[0], RAM[3:1]};  // Corrected: Q[0] feeds RAM
                    MSB
78          end
79          3'b111: begin // Left shift
80              Q <= {Q[2:0], Cn};
81              RAM <= {RAM[2:0], RAM[0]};  // Corrected: RAM[0] feeds
                    Q LSB
82          end
83          default: ;
84      endcase
85  end
86
87  // Falling edge: write to register file
88  always @(negedge CP) begin
89      case (dest_select)
```

```
90          3'b011: regfile[A_addr] <= D_in;  // Direct load A
91          3'b100: regfile[B_addr] <= D_in;  // Direct load B
92          3'b101: begin                     // Load both
93              regfile[A_addr] <= D_in;
94              regfile[B_addr] <= D_in;
95          end
96          3'b000,3'b001,3'b010: begin       // ALU result to regfile
97              if (I[5:3] != 3'b110) begin  // Don't write on PASS A/B
98                  regfile[A_addr] <= aluResult[3:0];
99              end
100         end
101         default: ;
102     endcase
103 end
104
105 // Output control
106 reg [3:0] Y_reg;
107 always @(*) begin
108     case (dest_select)
109         3'b110, 3'b111: Y_reg = Q;
110         default: Y_reg = aluResult[3:0];
111     endcase
112 end
113
114 assign Y = (OE == 1'b0) ? Y_reg : 4'bz;
115 assign F3 = aluResult[3];
116 assign F0 = (aluResult[3:0] == 4'b0000) ? 1'b0 : 1'bz;
117 assign Cn4 = aluResult[4];
118 assign OVR = (func_select == 3'b000 || func_select == 3'b001) ? (Cn
        ^ Cn4) : 1'b0;
119 assign P = (func_select == 3'b000 || func_select == 3'b001) ? &(
    aluA ^ aluB) : 1'b0;
120 assign G = (func_select == 3'b000 || func_select == 3'b001) ? |(
    aluA & aluB) : 1'b0;
121
122 endmodule
```

Listing 6.2: Simplified Verilog Model of AM2901 (Core ALU)

### 6.2.3 Pin Diagram

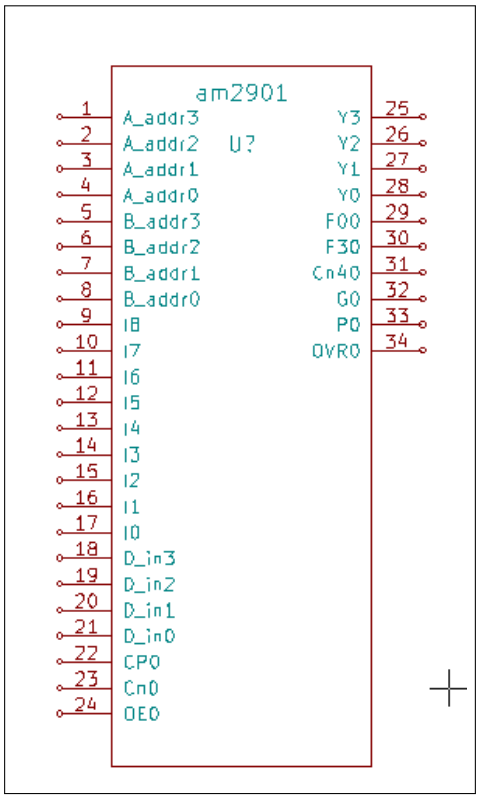


Figure 6.7: Pin Diagram of AM2901 ALU

Pinout includes A/B register selects, ALU inputs/outputs, status flags, clock, and function control lines.

### 6.2.4 Test Circuit

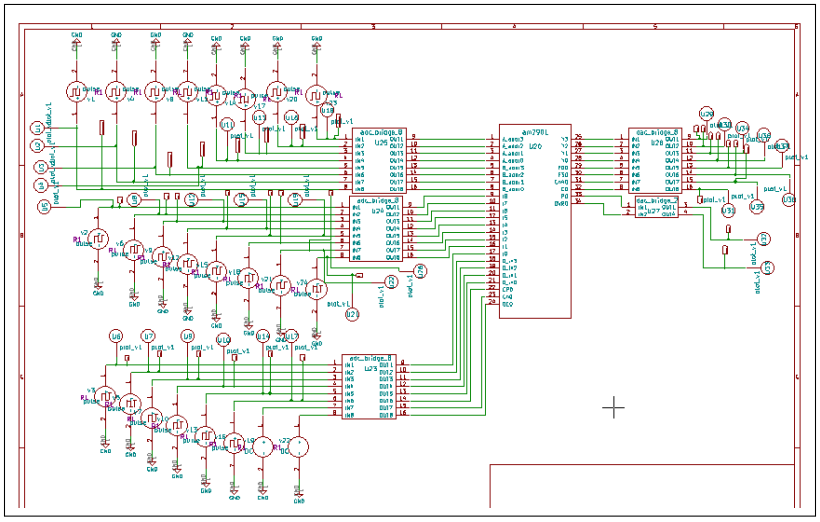

Figure 6.8: Test Circuit for AM2901 Verilog ALU

The test circuit drives different A, B, and op values to simulate ALU operations.

### 6.2.5 Waveforms

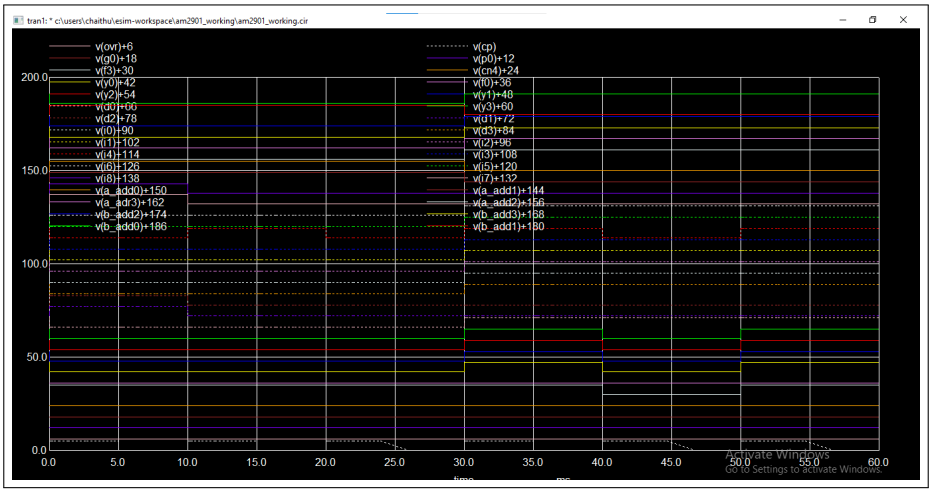


Figure 6.9: Output Waveforms of AM2901 ALU from Verilog Simulation

Simulation confirms the correct behavior for ADD, SUB, AND, OR, and shift operations.

#### NgSpice Output

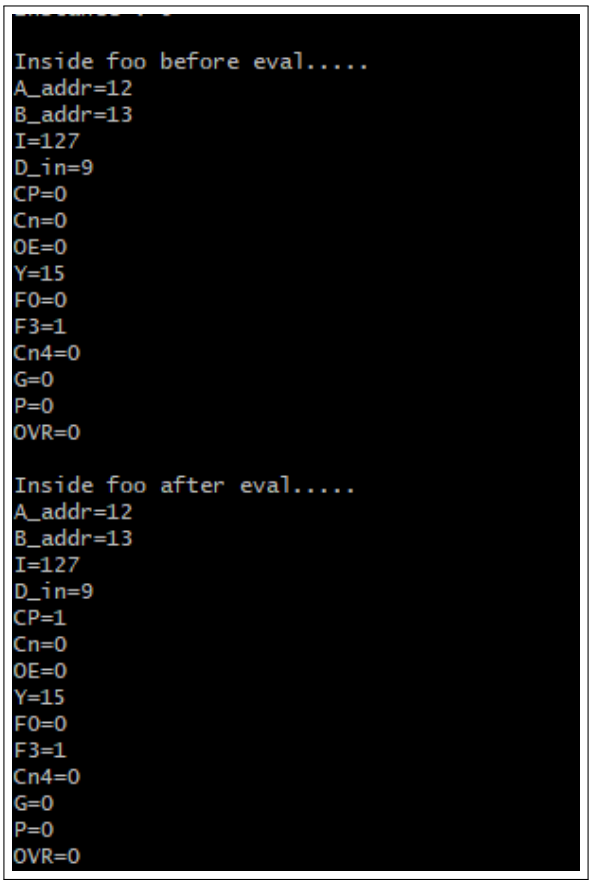


Figure 6.10: NgSpice Waveform for AM2901 Logic Operation

## 6.3 TDC1008 :8×8 Multiplier with Accumulator

### 6.3.1 Description

The **TDC1008** is a high-performance, bit-slice multiplier designed for fast arithmetic operations. It supports parallel 8-bit by 8-bit multiplication, making it ideal for digital signal processing (DSP), image processing, and VLSI implementations. By combining the TDC1008 with an accumulator unit, a multiply-accumulate (MAC) function can be realized.[16]

- Performs **8-bit × 8-bit multiplication**
- Integrated **accumulator unit** for successive MAC operations
- Fast carry-save addition mechanism
- Suitable for **pipelined or parallel** VLSI designs
- Capable of signed and unsigned arithmetic

#### Block Diagram

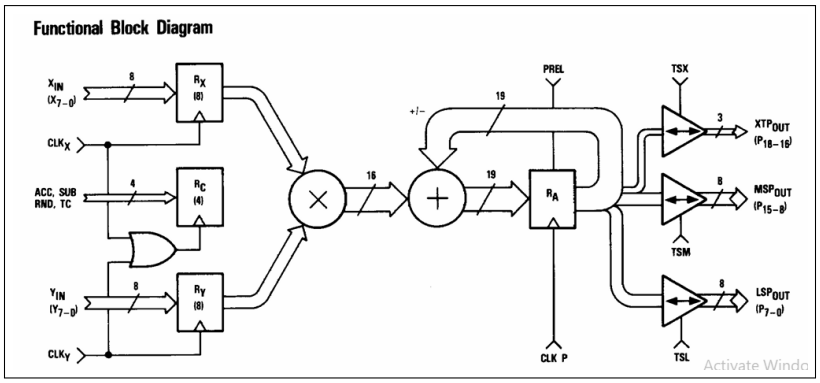


Figure 6.11: Block Diagram of TDC1008-Based 8×8 Multiplier with Accumulator

- The block diagram illustrates the data flow from input operands (X, Y) through the multiplier core and into the accumulator register.
- Control signals such as `clk`, `rst`, and `load` manage synchronous operation and accumulation cycles.

### 6.3.2 Verilog Code

```
module TDC1008 (
    input [7:0] X, Y,
    input CLK_X, CLK_Y, CLK_P,
    input RND, TC, ACC, SUB,
    input PREL,
    input TSX, TSM, TSL,
    output reg [2:0] XTP,     // Bits [18:16]
    output reg [7:0] MSP,     // Bits [15:8]
```

```
9       output reg [7:0] LSP
10
11  );
12
13      // Internal input registers (positive-edge triggered)
14      reg [7:0] RX, RY;
15
16      // Control register RC (positive-edge triggered)
17      reg [3:0] RC; // {ACC, SUB, RND, TC}
18
19      // Accumulator register RA (positive-edge triggered)
20      reg signed [18:0] RA;
21
22      // Intermediate wires
23      wire signed [7:0] X_signed, Y_signed;
24      wire signed [15:0] Product_signed;
25      wire [15:0] Product_unsigned;
26      wire [15:0] Product_base;
27      reg  [15:0] Product_rounded;
28      wire signed [18:0] Product_ext;
29      wire signed [18:0] Accum_in;
30
31      // Latch inputs on clock edges
32      always @(posedge CLK_X) RX <= X;
33      always @(posedge CLK_Y) RY <= Y;
34
35      // Store control inputs in RC (latched on CLK_P)
36      always @(posedge CLK_P) RC <= {ACC, SUB, RND, TC};
37
38      // Signed/unsigned multiplication
39      assign X_signed = RX;
40      assign Y_signed = RY;
41      assign Product_signed   = X_signed * Y_signed;
42      assign Product_unsigned = RX * RY;
43
44        // Keep multiplication logic based on TC
45      assign Product_base = RC[0] ? Product_signed : Product_unsigned
           ;
46
47      // Apply rounding if RND is set (RC[2])
48      always @(*) begin
49          if (RC[2])
50              Product_rounded = Product_base + 16'h0040;
51          else
52              Product_rounded = Product_base;
53      end
54
55      // Always sign-extend Product_rounded regardless of TC
56      assign Product_ext = $signed({{3{Product_rounded[15]}},
           Product_rounded});
57
58
59      // Accumulation/subtraction/load logic
60      assign Accum_in = (RC[3] && RC[1]) ? RA - Product_ext :  // ACC
            & SUB
61                        (RC[3])          ? RA + Product_ext :  // ACC
                              only
```

```
62                                          Product_ext;         //
                                               Load new product
                                               directly
63
64      // Update accumulator on CLK_P
65      always @(posedge CLK_P)
66          RA <= Accum_in;
67
68      // Output register with tri-state control
69      always @(*) begin
70          XTP = 3'bz;
71          MSP = 8'bz;
72          LSP = 8'bz;
73
74          if (PREL) begin
75              if (TSX) XTP = RA[18:16];
76              if (TSM) MSP = RA[15:8];
77              if (TSL) LSP = RA[7:0];
78          end
79      end
80
81  endmodule
```

Listing 6.3: 8×8 Multiplier with Accumulator (TDC1008 Model)

### 6.3.3 Pin Diagram

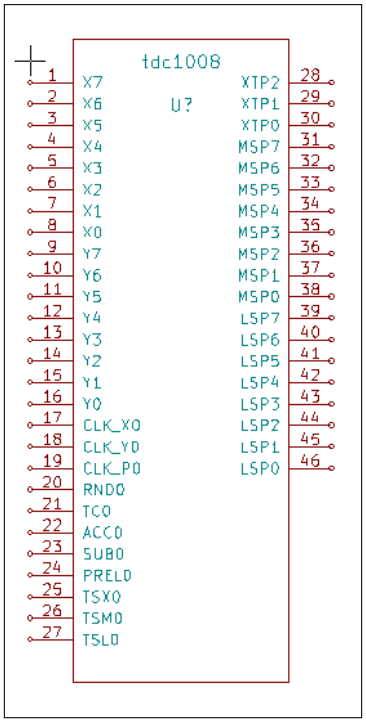


Figure 6.12: Pin Diagram of TDC1008 Multiplier

Shows 8-bit inputs (X, Y), accumulator control, CLK, RST, and 16-bit output lines.

### 6.3.4 Test Circuit

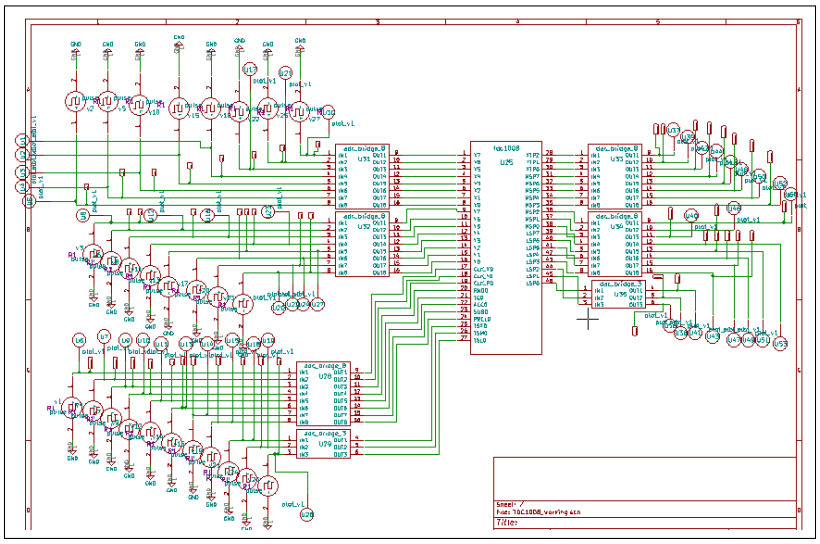

Figure 6.13: Test Circuit for TDC1008 8×8 Multiplier

Applies multiple X, Y values and observes the accumulation of their product in P.

### 6.3.5 Waveforms

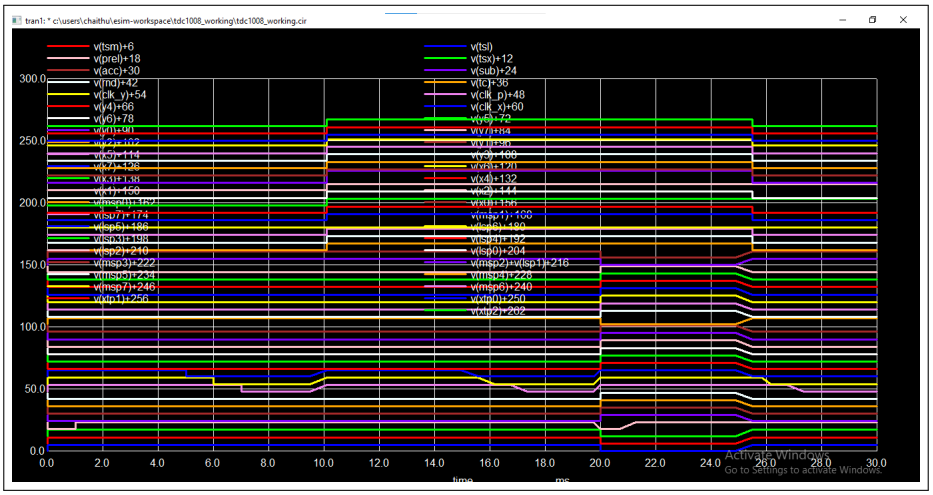


Figure 6.14: Output Waveform showing Multiply-Accumulate Operation

Shows accumulation of X × Y over multiple clock cycles.

**NgSpice Output**

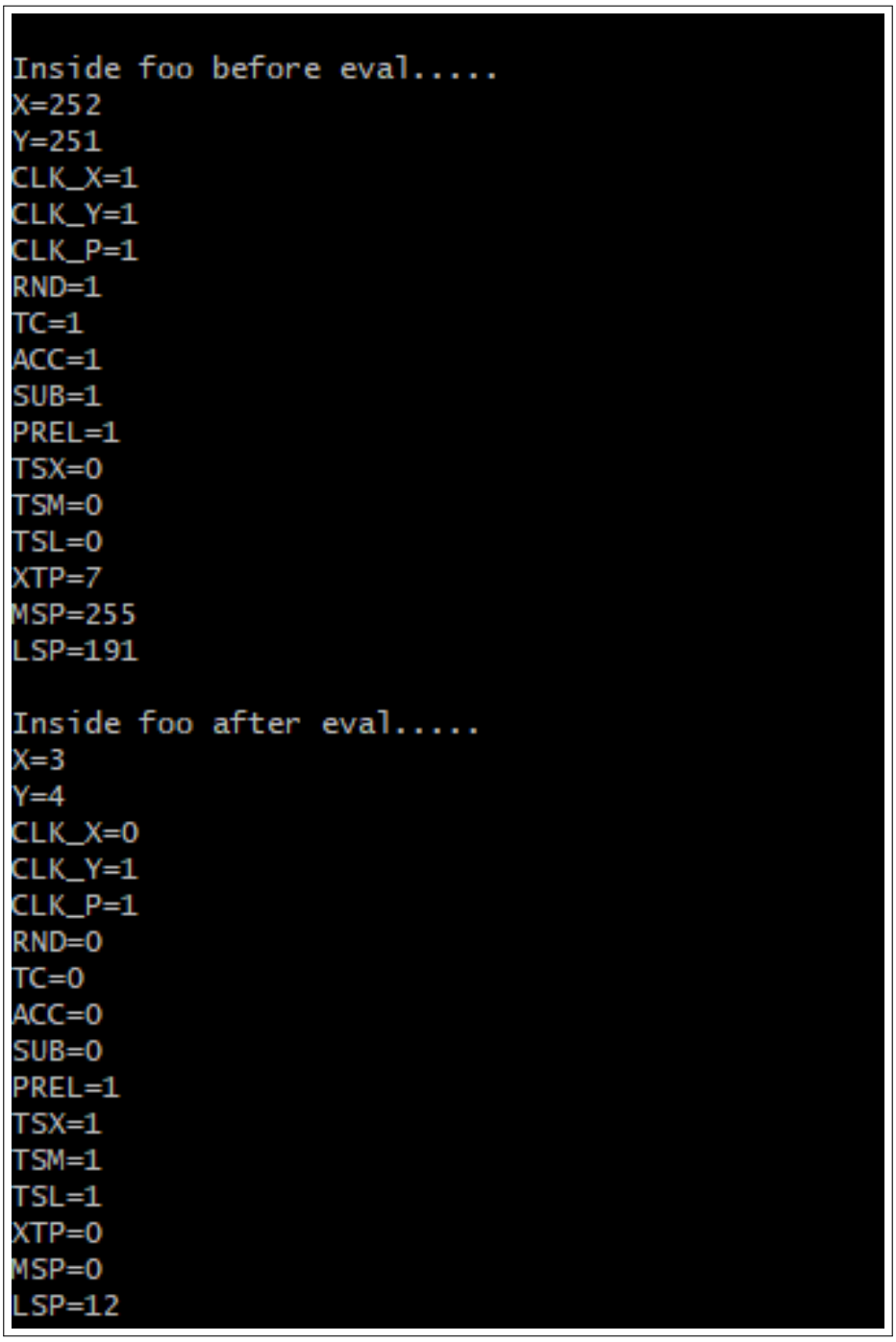


Figure 6.15: NgSpice Output for Gate-Level Multiplier Simulation

Validates propagation delay, switching activity, and timing for key nets.

## 6.4 CRC16 Generator

### 6.4.1 Description

The **CRC16 Generator** is used to detect errors in digital data transmission. It calculates a 16-bit Cyclic Redundancy Check (CRC) code based on the input data and a predefined polynomial (commonly 0x8005 or 0x1021). This design implements a serial-input CRC16 generator with reset, seed, and polynomial configuration.[17]

#### Block Diagram

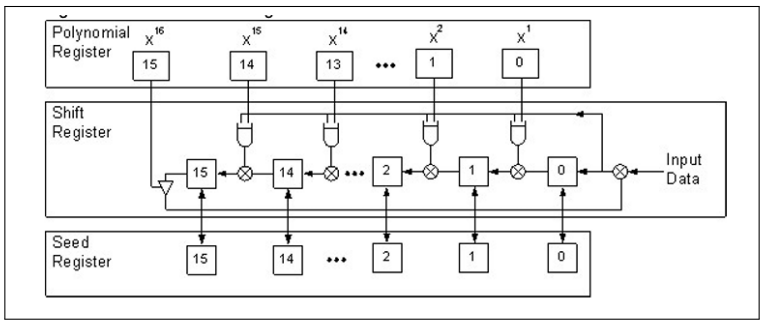


Figure 6.16: Block Diagram of CRC16 Generator

The block diagram shows the shift register-based architecture with XOR feedback paths for CRC computation.

### 6.4.2 Verilog Code

```
module CRC_Generator #(
    parameter WIDTH = 16
) (
    input clk,
    input  reset_n,
    input start,
    input data_in,
    input  [15:0] seed,
    input [15:0] poly,
    output reg [15:0] crc_out
);

reg [WIDTH-1:0] crc_reg;

// LFSR logic (generic for any polynomial)
always @(posedge clk or negedge reset_n) begin
    if (!reset_n) begin
        crc_reg <= seed;  // Initialize with seed
    end
    else if (start) begin
        crc_reg <= {crc_reg[WIDTH-2:0], 1'b0} ^ (poly & {WIDTH{(
           data_in ^ crc_reg[WIDTH-1])}});
    end
end

```

```
25 // Output the CRC
26 always @(posedge clk) begin
27     crc_out <= crc_reg;
28 end
29
30 endmodule
```

Listing 6.4: CRC16 Generator (Serial Input)

### 6.4.3 Pin Diagram

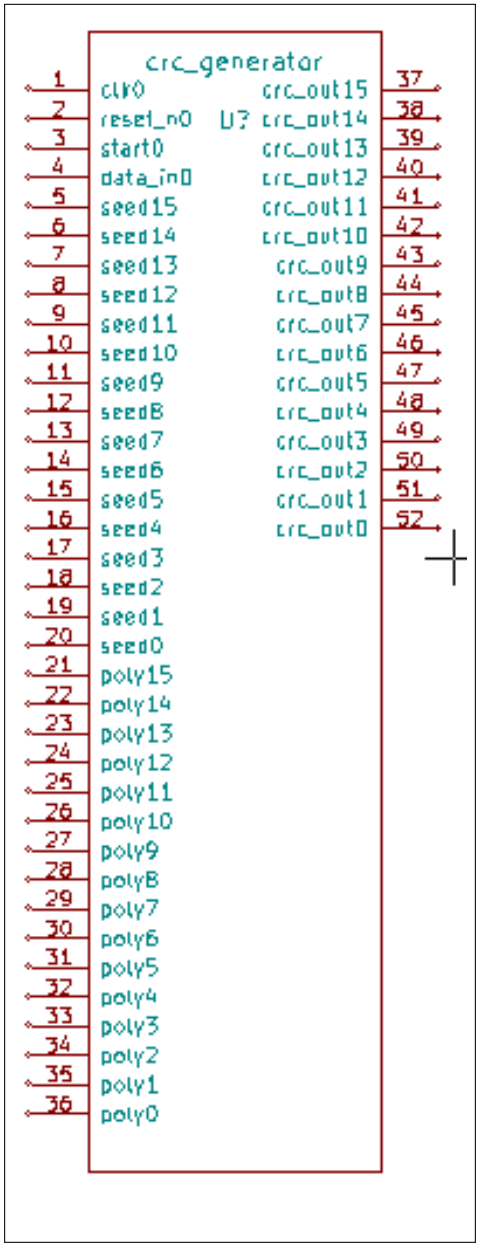


Figure 6.17: Pin Diagram of CRC16 Generator

### 6.4.4 Test Circuit

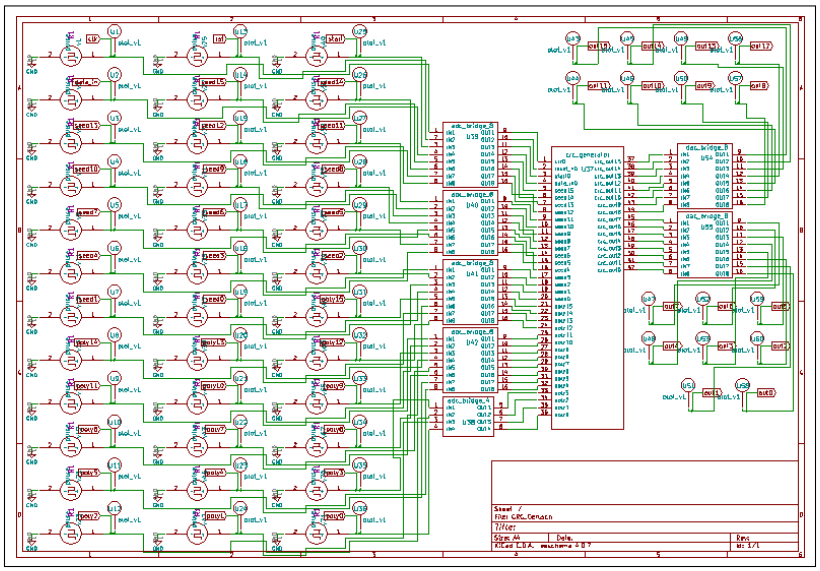

Figure 6.18: Test Circuit for CRC16 Generator

The test circuit provides serial input data to validate CRC output against known values.

### 6.4.5 Waveforms

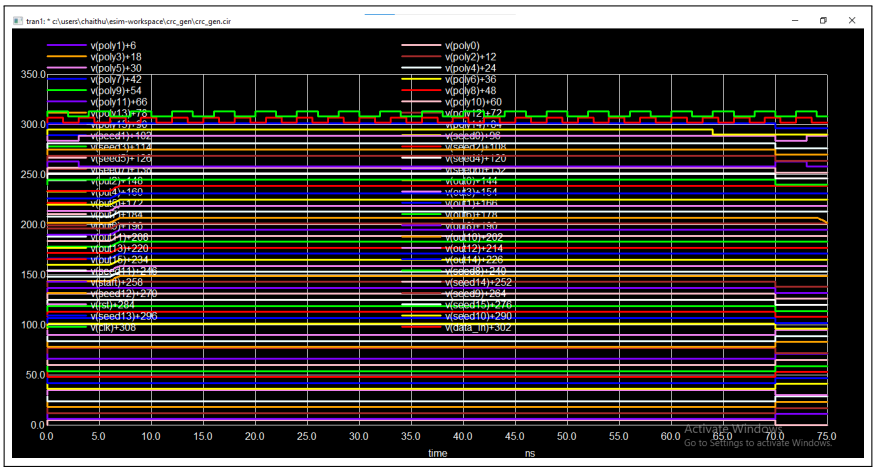


Figure 6.19: Simulation Output Waveform for CRC16 Generator

This waveform shows how CRC is calculated bit-by-bit using the given polynomial and seed value.

### NgSpice Output

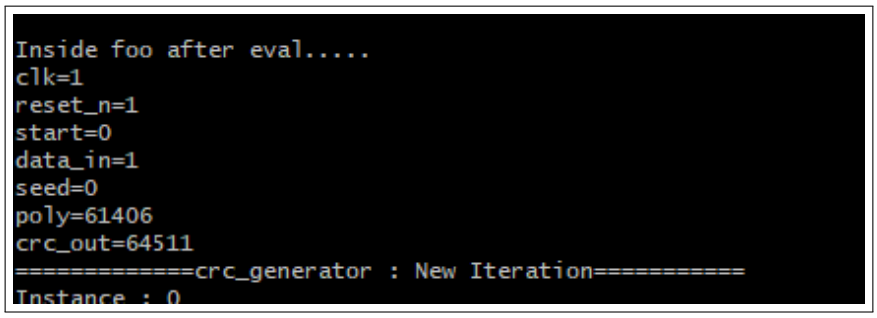


Figure 6.20: NgSpice Simulation Output of CRC16 Generator

- The figure shows the voltage waveforms at key nodes obtained using NgSpice simulation.
- It verifies the behavior of the circuit under test by comparing expected timing and logical transitions.
- The output confirms proper CRC shift register behavior and polynomial feedback implementation.

# Chapter 7

# Device Modeling

Device modelling in eSim refers to the process of designing fundamental semiconductor devices such as BJTs, MOSFETs, and diodes through physics-based SPICE-level modelling.This approach involves formulating accurate .model statements that encapsulate key parameters—such as threshold voltage, saturation current, mobility factors, and junction capacitances—governing the device's operation. These models are grounded in semiconductor physics and are essential for reproducing real-world phenomena like switching behavior, biasing characteristics, and nonlinearities under various operating conditions

## 7.1 SST406 : N-Channel JFET Monolithic Dual

### 7.1.1 Description

The **SST406** is a high-performance monolithic dual N-Channel JFET designed for analog signal processing applications, such as low-noise amplifiers, buffer stages, and active filters. Its matched pair structure ensures excellent thermal tracking and offset characteristics.[18]

- N-Channel JFET with low noise
- Ideal for differential amplifier configurations
- High input impedance and low leakage
- Matched characteristics for precision analog use

### 7.1.2 SPICE Model

```
.MODEL SST406 NJF ( Beta=1.577m Betatce=-0.5 Vto=-1.316 Vtotc=-2.5m Lambda=10m Is=19.73f Xti=3 Isr=191.3f Nr=2 Alpha=68.56u N=1 Rd=1 Rs=1 Cgd=5.6p Cgs=6.044p Fc=0.5 Vk=212.2 M=391.6m Pb=0.5 Kf=45920f Af=1 Mfg=Linear Systems )
```

### 7.1.3 Amplifier Configuration

**Amplifier Circuit:**

The following circuit demonstrates the use of the SST406 JFET configured as a common-source amplifier to achieve voltage gain.

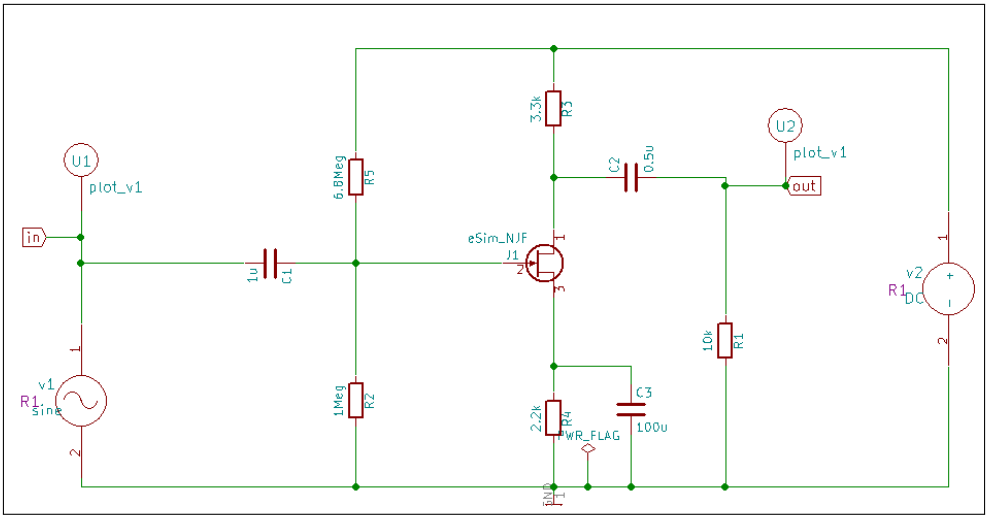


Figure 7.1: SST406 JFET Amplifier Circuit

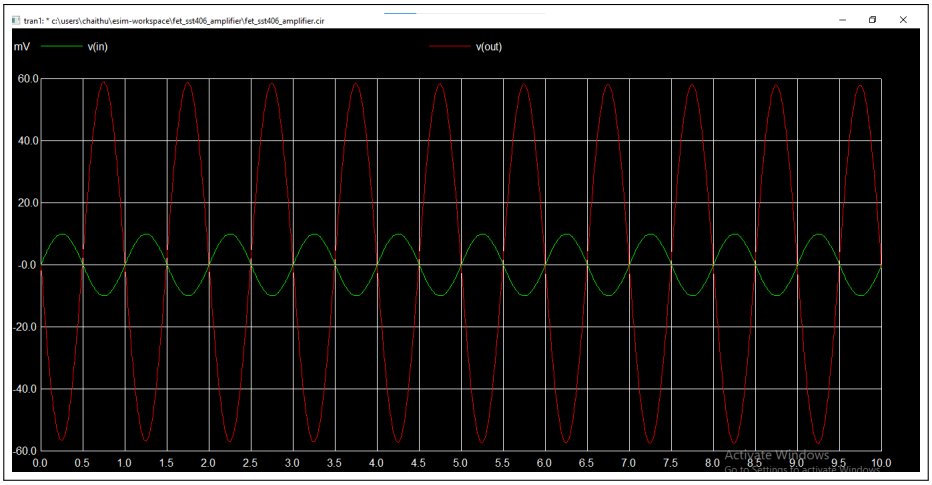


Figure 7.2: Amplifier Characteristics of SST406

The transient simulation of the SST406 NJFET amplifier clearly demonstrates its amplification capability. This confirms the proper operation of the NJFET amplifier in the active region, where it amplifies small input variations into larger output swings. The gain is consistent, and the output remains undistorted, indicating good linearity within the frequency range used. The phase of the output also follows the input with minimal delay, suitable for analog signal processing applications.

### 7.1.4 Frequency Response Analysis

#### Frequency Response Circuit

This circuit is used to evaluate how the SST406 JFET amplifier responds over a wide range of input frequencies, helping to determine the bandwidth and cutoff frequencies.

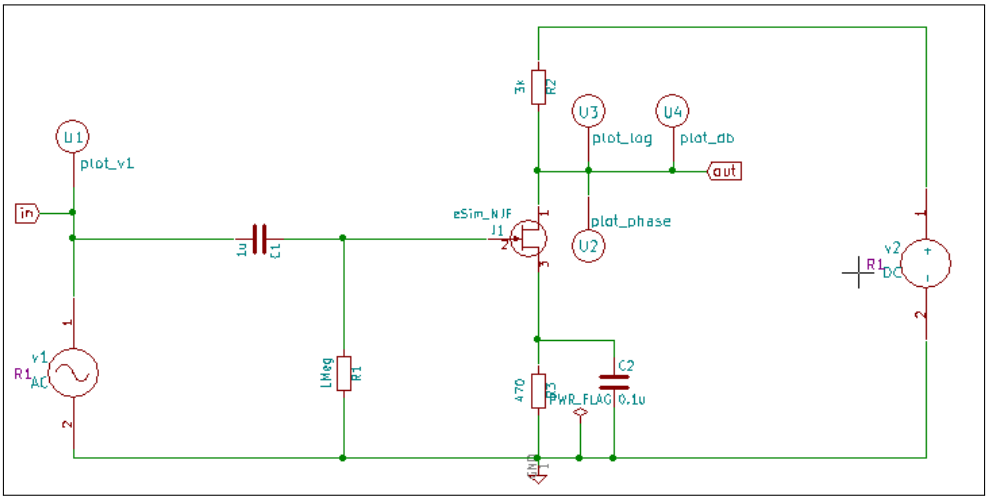


Figure 7.3: Frequency Response Measurement Circuit

The SST406-based JFET amplifier exhibits typical low-pass frequency response behavior. At lower frequencies, the gain rises steadily and stabilizes into a flat mid-band region, demonstrating a wide bandwidth of stable amplification.

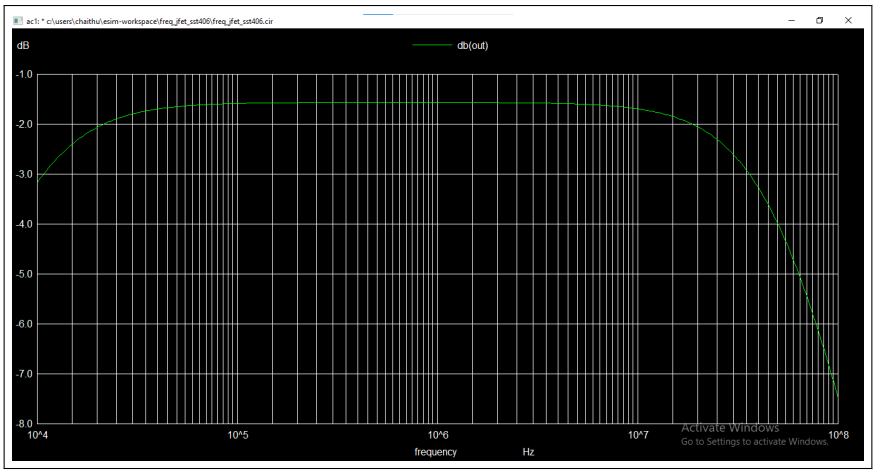


Figure 7.4: Frequency Response – Magnitude in dB (db(out))

From the dB magnitude response, the gain remains nearly constant up to the high-frequency region where it begins to roll off. The -3 dB point, which defines the upper cutoff frequency, occurs slightly above $10^7$ Hz, beyond which capacitive effects and parasitic elements reduce the amplifier's gain. .

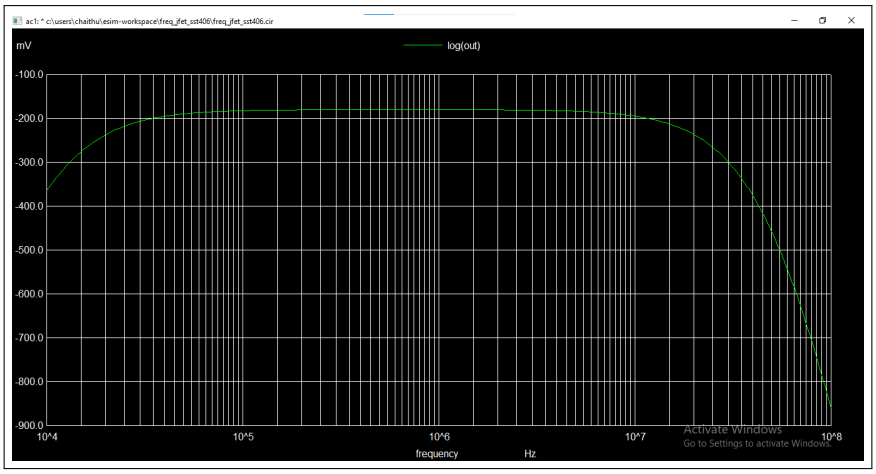


Figure 7.5: Frequency Response – Linear Magnitude (log(out))

This log-magnitude plot shows a bandpass-like frequency response characteristic of JFET amplifier circuits. The gain is flat in the mid-frequency range and drops at both ends, indicating the frequency-dependent nature of the amplifier's gain.

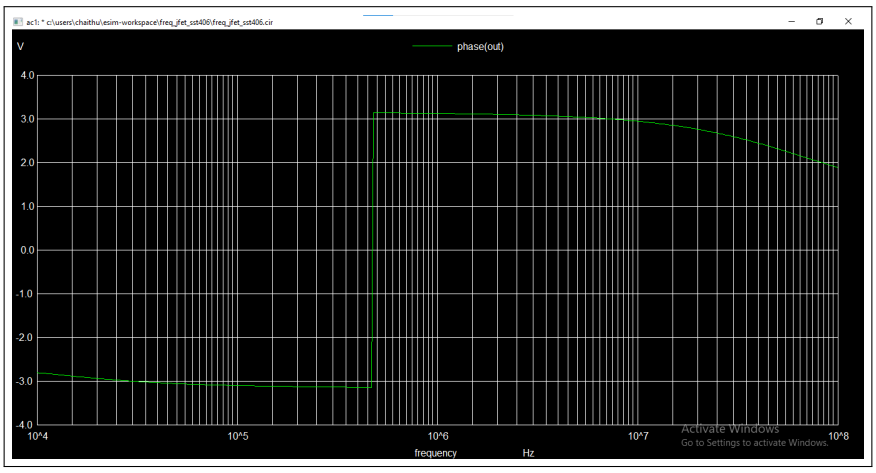


Figure 7.6: Frequency Response – Phase Plot (phase(out))

The phase plot reveals a stable phase in the mid-band, followed by a significant lag at higher frequencies, consistent with the amplifier's low-pass nature. Together, these graphs provide clear evidence of the amplifier's bandwidth and frequency limitations.

### 7.1.5 Characteristics Graphs

#### Characteristics Circuit

The characteristics circuit is used to analyze the output current-voltage (I-V) behavior of the SST406 JFET, enabling extraction of key parameters like pinch-off voltage and saturation current.

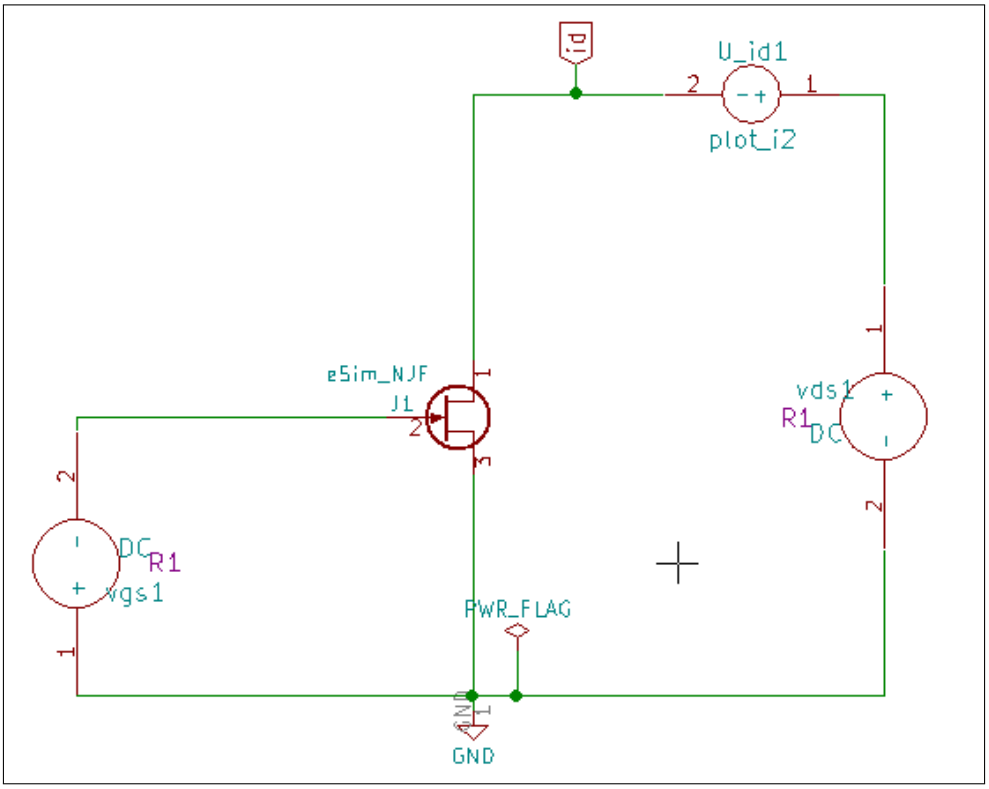


Figure 7.7: Fig. 3: SST406 JFET Characteristics Circuit

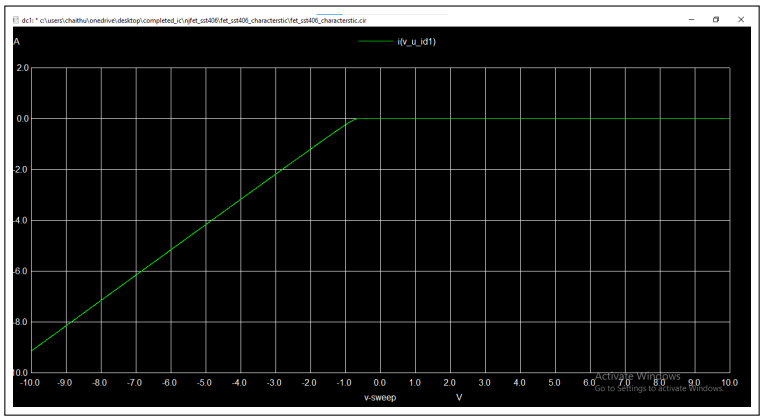


Figure 7.8: Output Characteristics of SST406 ($I_D$ vs. $V_{DS}$)

This graph illustrates the drain current $I_D$ versus drain-source voltage $V_{DS}$ for various gate-source voltages $V_{GS}$. It highlights the SST406's saturation region, pinch-off behavior, and the effect of gate biasing on channel conductivity.

## 7.2 2N4124 : NPN General Purpose Transistor

### 7.2.1 Description

The **2N4124** is an NPN bipolar junction transistor commonly used for low-power switching and general-purpose amplification. It offers high current gain, low saturation voltage, and good frequency response for analog applications.[19]

- General-purpose NPN BJT for analog and digital switching
- High DC gain and transition frequency

### 7.2.2 SPICE Model

.MODEL 2N4124 NPN(Is=6.734f Xti=3 Eg=1.11 Vaf=74.03 Bf=495 Ne=1.28 Ise=6.734f Ikf=69.35m Xtb=1.5 Br=.7214 Nc=2 Isc=0 Ikr=0 Rc=1 Cjc=3.638p Mjc=.3085 Vjc=.75 Fc=.5 Cje=4.493p Mje=.2593 Vje=.75 Tr=238.3n Tf=301.3p Itf=.4 Vtf=4 Xtf=2 Rb=10 Vceo=25 Icrating=200m mfg=Fairchild)

### 7.2.3 Amplifier Configuration

**Description:**
The following test circuit demonstrates the use of the 2N4124 NPN BJT as a common-emitter amplifier. A sinusoidal input signal is applied to the base via a coupling capacitor, and biasing resistors are used to set the quiescent operating point. The transistor operates in the active region, allowing it to amplify the small input AC signal superimposed on the DC bias. The amplified output is taken from the collector and observed across a load resistor.

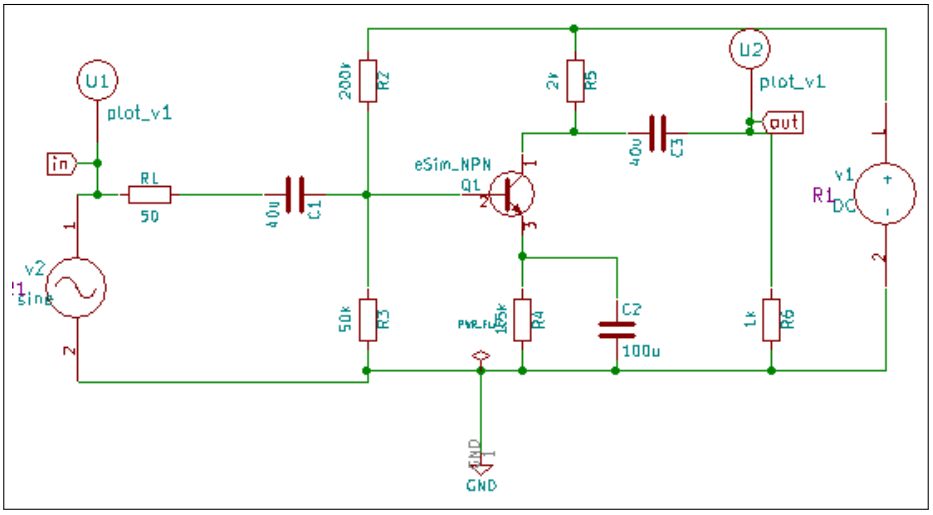


Figure 7.9: Common-Emitter Amplifier Test Circuit using 2N4124

**Output Waveform**

**Description:**
The output waveform clearly shows the amplified version of the input sinusoidal signal. The phase is inverted, which is characteristic of a common-emitter amplifier. The gain is determined by the ratio of collector resistance to emitter resistance , and the waveform indicates faithful amplification without clipping, proving proper biasing and linear operation.

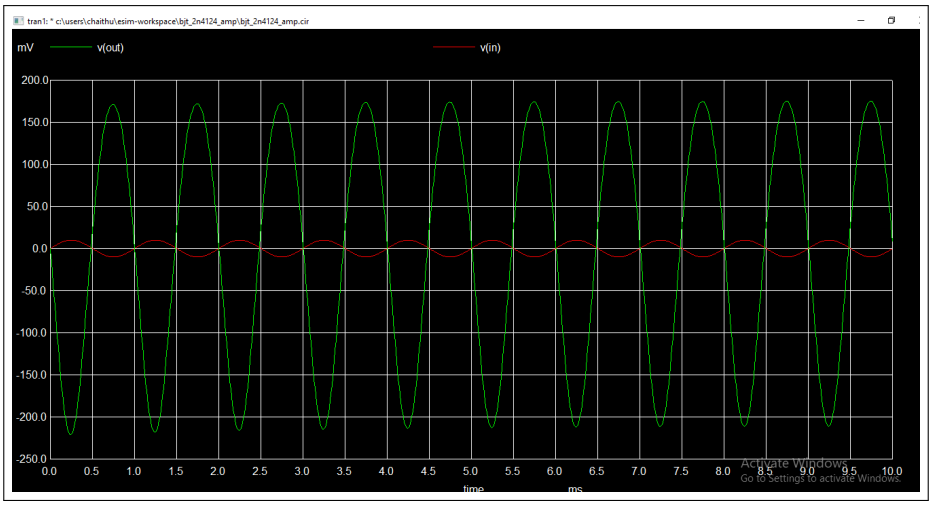


Figure 7.10: Output Waveform of the 2N4124 Amplifier

### 7.2.4 Frequency Response Analysis

**Description:**
The frequency response of the 2N4124 amplifier is measured by sweeping the input signal across a wide frequency range and analyzing how the amplifier's gain and phase shift change. The Bode plots generated below (log-magnitude and phase) help determine the -3dB cutoff frequency, gain bandwidth, and phase behavior of the amplifier. These characteristics are essential for understanding the stability and frequency performance of the design.

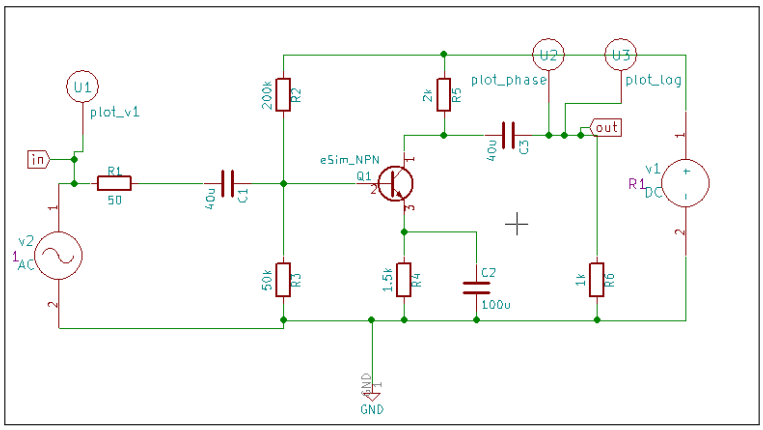


Figure 7.11: Frequency Response Test Circuit using 2N4124

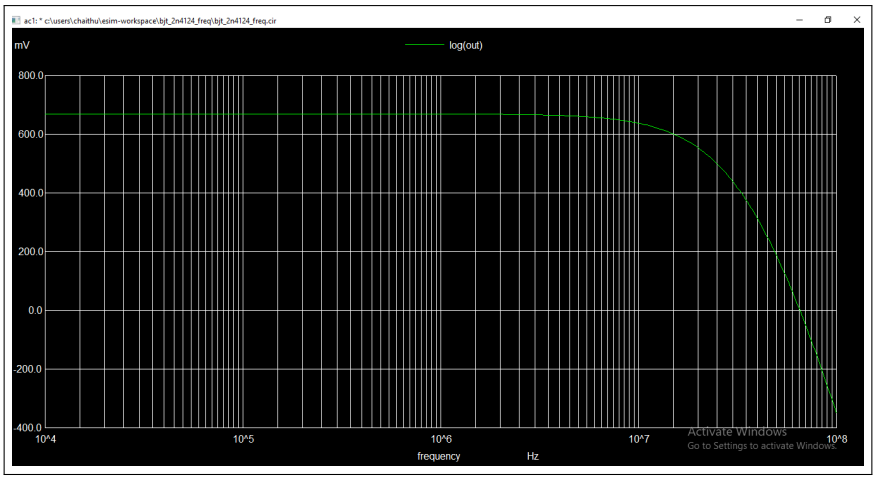


Figure 7.12: Log-Magnitude Plot of Amplifier Output (Gain vs Frequency)

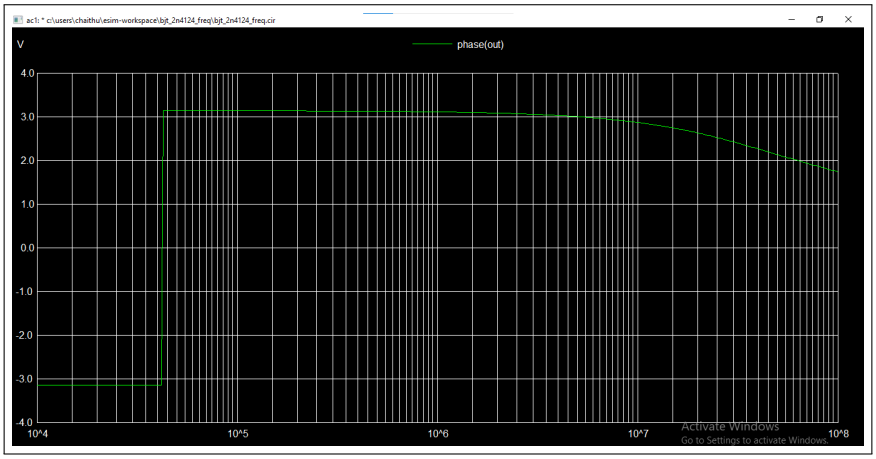


Figure 7.13: Phase Plot of Amplifier Output (Phase vs Frequency)

- **Log Plot (Gain vs Frequency)**: This plot shows the amplifier gain (in dB) on a logarithmic frequency scale. It remains flat in the mid-frequency range and decreases at low and high frequencies due to coupling capacitors and internal capacitances. The points where the gain drops by 3 dB are known as the cutoff frequencies.
- **Phase Plot (Phase vs Frequency)**: This plot shows the phase shift between input and output signals as a function of frequency. At low frequencies, the phase starts near 0°, shifts significantly through midband, and approaches –90° to –180° at higher frequencies due to the reactive components.
- Together, these plots help determine the amplifier's bandwidth, stability, and frequency-dependent behavior.

### 7.2.5 Common-Base (CB) Configuration

The CB configuration is used for high-frequency gain and low input impedance.

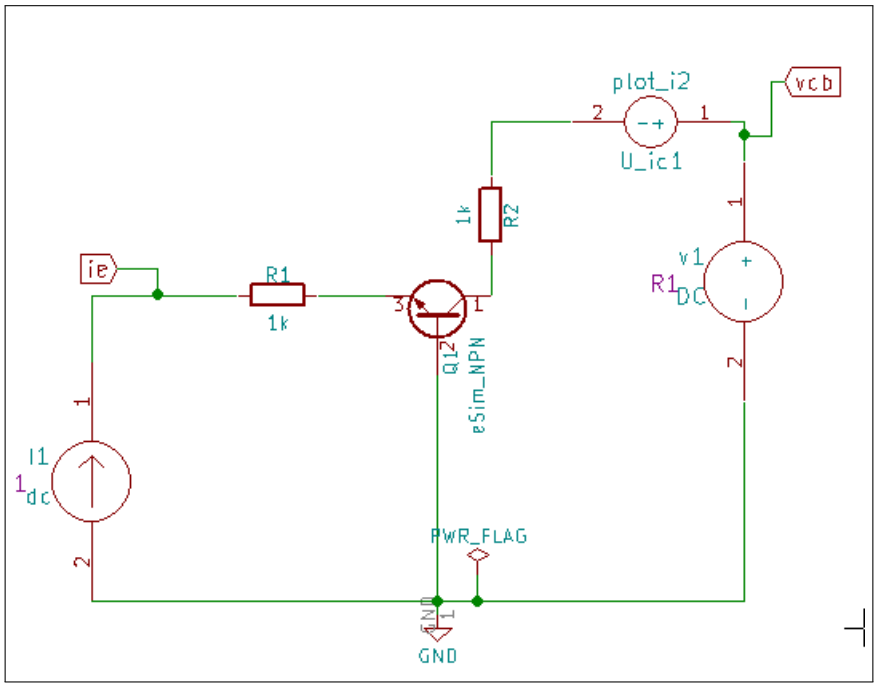


Figure 7.14: Common-Base Configuration

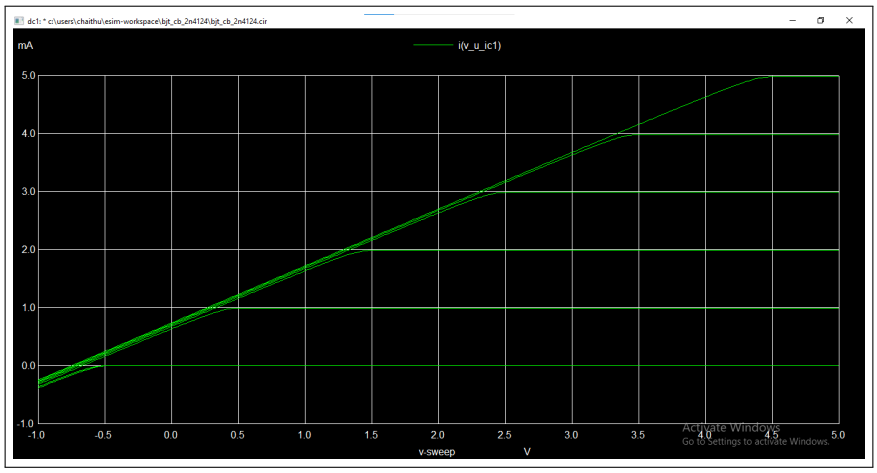


Figure 7.15: CB Input and Output Waveforms

In the common-base configuration, the output characteristics show the relationship between collector current ($I_C$) and collector-base voltage ($V_{CB}$) for different constant emitter currents ($I_E$). The curves are almost flat, indicating high output resistance. Collector current remains nearly constant with increasing $V_{CB}$, showing good current source behavior.

### 7.2.6 Common-Emitter (CE) Configuration

The CE configuration provides voltage gain and is widely used in amplifiers.

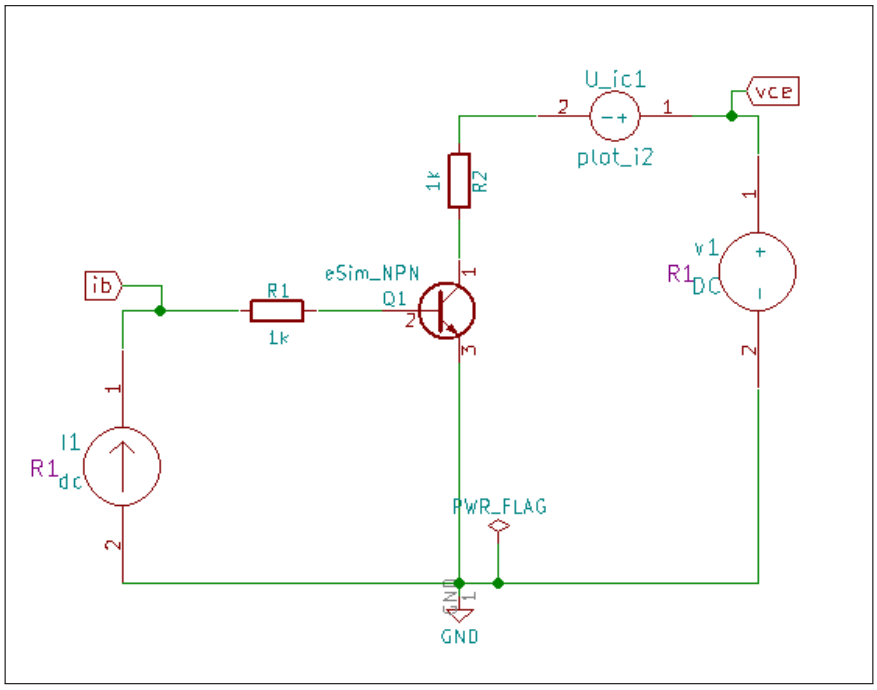


Figure 7.16: Common-Emitter Configuration

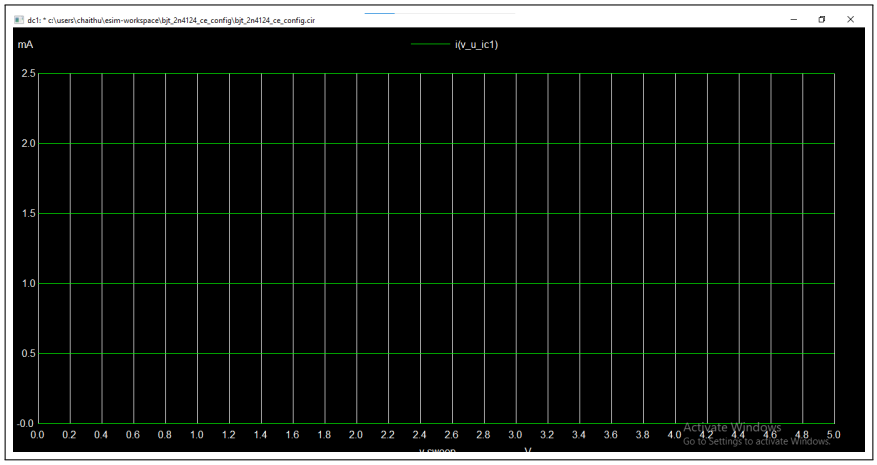


Figure 7.17: CE Input and Output Waveforms

In the common-emitter configuration, the output characteristics plot collector current ($I_C$) versus collector-emitter voltage ($V_{CE}$) for different base currents ($I_B$). The curves rise sharply at first and then flatten, indicating active region operation. This configuration provides significant current and voltage gain, making it widely used in amplification.

## 7.3 MUR460 : Switch-Mode Power Rectifier

### 7.3.1 Description

The MUR460 is a high-speed ultra fast recovery rectifier designed for use in switch-mode power supplies, freewheeling diodes, and other high-frequency rectification applications. This diode is commonly used in high-speed switching power circuits to improve efficiency and reduce electromagnetic interference (EMI).[20]

- Ultrafast 25 ns, 50 ns, and 75 ns Recovery Times
- 175°C Operating Junction Temperature
- Low Forward Voltage
- Low Leakage Current
- High Temperature Glass Passivated Junction

### 7.3.2 SPICE Model

```
.MODEL MUR460 D( Is=10n Rs=.0384 N=2.5 tt=44.4n Cjo=126.4p M=.52
Vj=1.34 Bv=1600 Ibv=1e-05 EG=1.285 XTI=.5 Iave=4 Vpk=600 )
```

### 7.3.3 Forward Bias Characteristics

The following test circuit is used to observe the forward bias behavior of the MUR406 diode. A DC voltage source is applied in series with a current-limiting resistor and the diode under test.

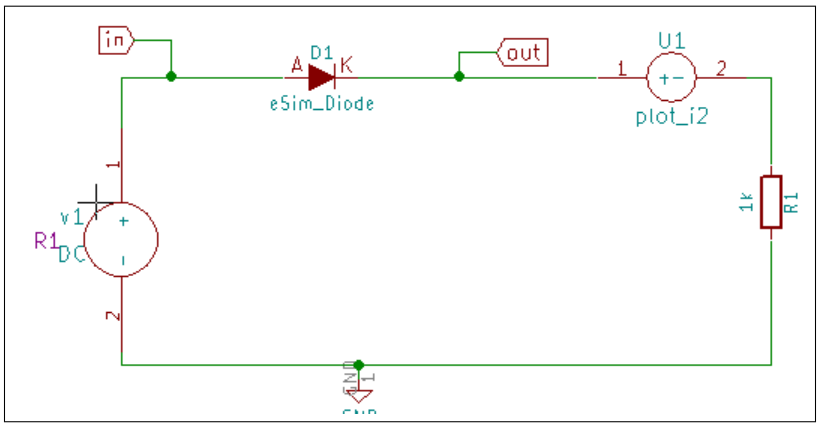


Figure 7.18: Forward Bias Test Circuit for MUR406

**Forward Bias Characteristics – Waveform**

The waveform below shows the forward voltage ($V_F$) across the diode versus the forward current ($I_F$). As the current increases, the diode conducts with a small voltage drop, typically around 0.8–1.2V for MUR406.

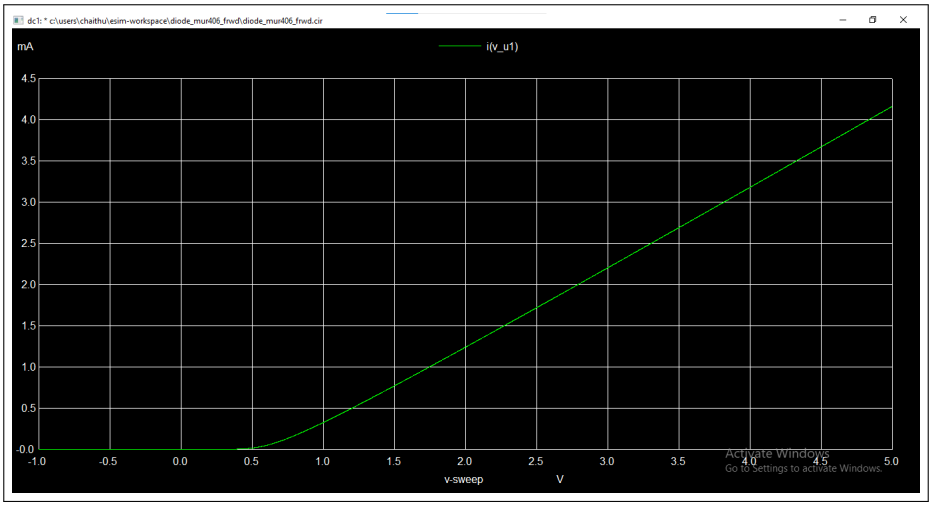


Figure 7.19: Forward Bias Characteristics ($V_F$ vs $I_F$)

### 7.3.4 Reverse Bias Characteristics

To test the reverse bias response, a DC source is connected with polarity reversed across the diode along with a current measurement setup. The reverse current observed is typically very small,indicating high resistance to reverse conduction and confirming the diode's rectifying behavior.

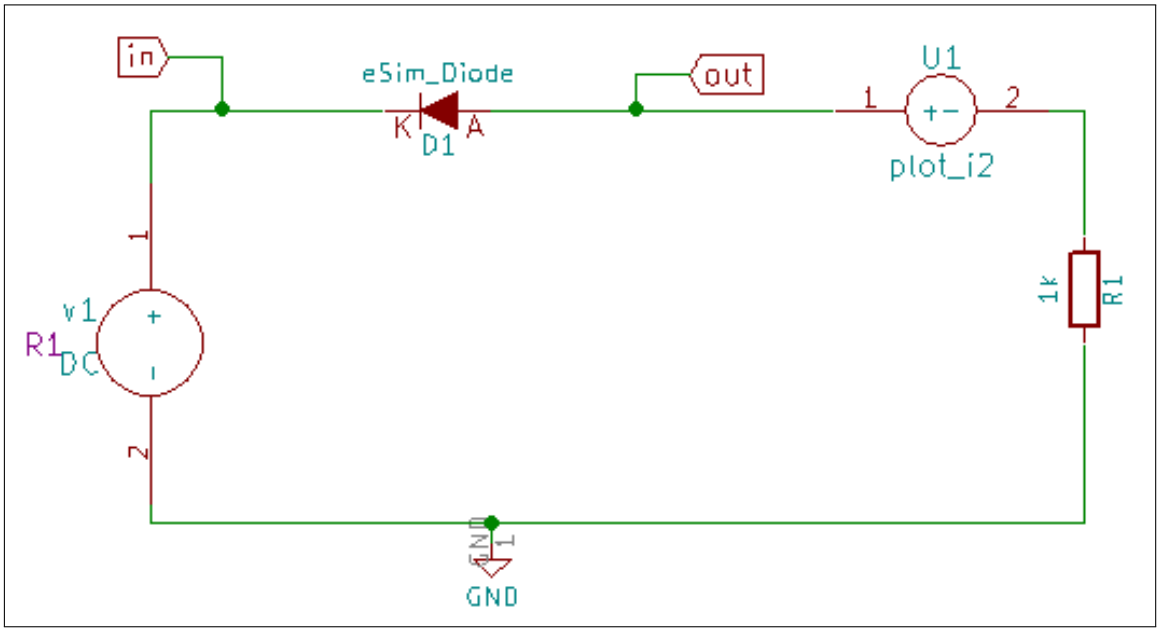


Figure 7.20: Reverse Bias Test Circuit for MUR406

### Reverse Bias Characteristics Waveform

The graph below plots the reverse current ($I_R$) versus reverse voltage ($V_R$). The reverse leakage current remains very small until the breakdown voltage is approached, typically beyond 600V for MUR460.

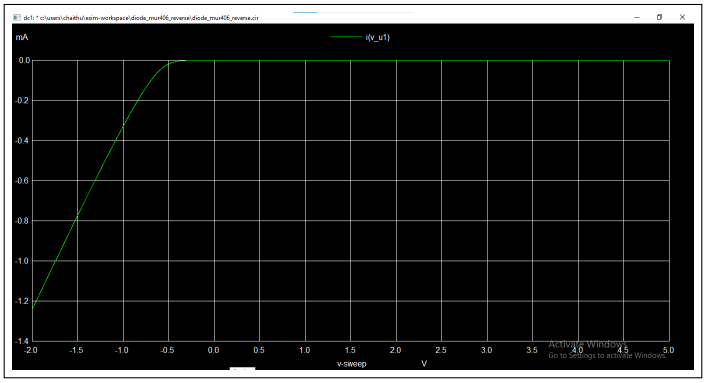


Figure 7.21: Reverse Bias Characteristics ($I_R$ vs $V_R$)

## 7.3.5 Half-Wave Rectifier using MUR460 (Ultrafast Recovery Diode)

The MUR460 is an ultrafast recovery diode with higher voltage handling capability and low reverse recovery time. It is commonly used in high-frequency power rectifiers and switching regulators.

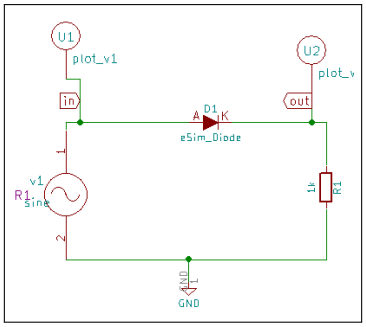


Figure 7.22: Half-Wave Rectifier Circuit using MUR460

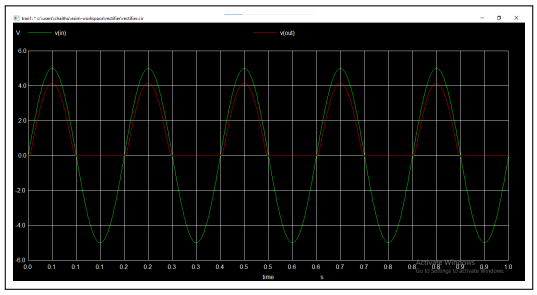


Figure 7.23: Output Waveform of Half-Wave Rectifier using MUR460

**Observation:** The MUR460 has a higher forward voltage drop ( 0.7–1.0 V), which slightly reduces the output voltage compared to Schottky diodes.

## 7.4 2S2088U3 : Medium power transistor

### 7.4.1 Description

The 2S2088U3 is a medium power PNP transistor commonly used in switching and linear amplifier applications. It features high collector current capability and moderate gain, making it suitable for audio amplifier stages, motor control, and power supply circuits.[21]

### 7.4.2 SPICE Model

.MODEL 2SA2088U3 PNP(Is=65.000E-15 Bf=249.50 Vaf=100 Ikf=.69106 Ise=65.001E-15 Ne=1.4594 Br=10.162 Var=100 Ikr=11.394 Isc=5.0270E-12 Nc=1.3515 Nk=.79748 Re=.5 Rb=.81834 Rc=.20879 Cje=96.030E-12 Mje=.77026 Cjc=32.647E-12 Mjc=.57832 Tf=304.61E-12 Xtf=11.555 Vtf=13.588 Itf=8.1887 Tr=35.022E-9 Xtb=1.5000 Vceo=60 Icrating=0.5 mfg=Rohm)

### 7.4.3 Amplifier Configuration

The amplifier circuit shown below uses the 2S2088U3 in a common-emitter configuration, suitable for inverting signal amplification.

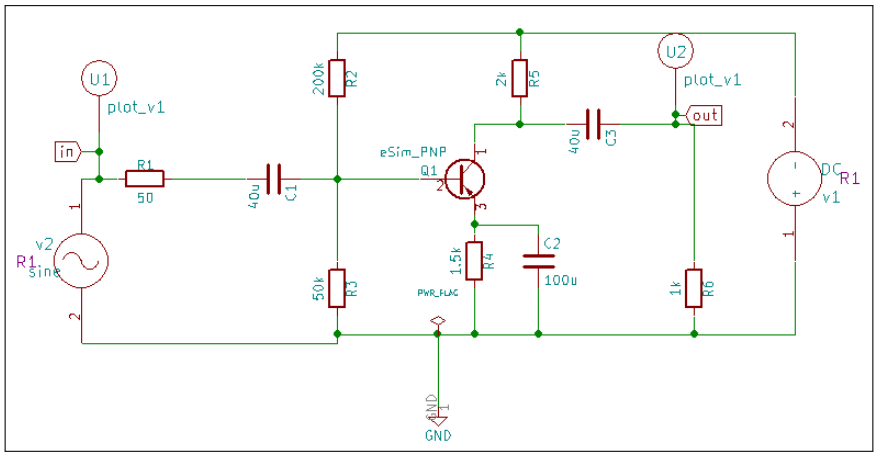


Figure 7.24: Amplifier Circuit Using 2S2088U3 in Common-Emitter Configuration

#### Amplifier Output Waveform

The following waveform shows the output of the 2S2088U3-based amplifier when driven by a sinusoidal input signal. As expected from a common-emitter configuration, the output is **inverted** with respect to the input and **amplified** in magnitude.

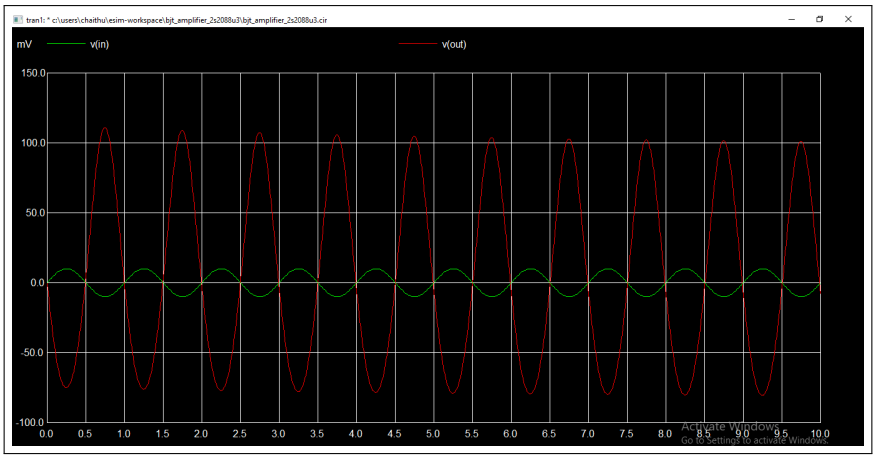


Figure 7.25: Output Waveform of 2S2088U3 Amplifier

### 7.4.4 Common-Base Configuration

#### Common-Base Input Characteristics

In the common-base configuration for a PNP transistor, the emitter current ($I_E$) is plotted against emitter-base voltage ($V_{EB}$) for various collector-base voltages ($V_{CB}$). Since it's PNP, the emitter-base junction is forward-biased (negative $V_{EB}$).

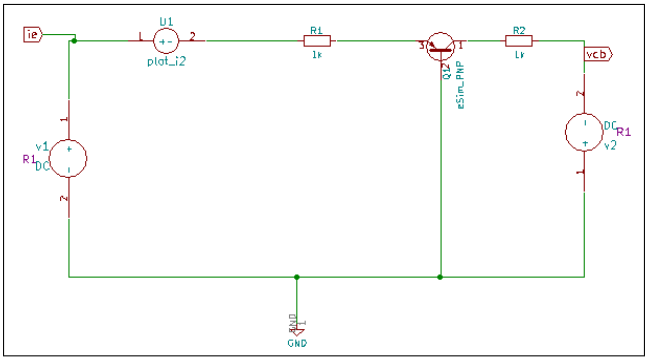


Figure 7.26: Common-Base Input Characteristic Circuit

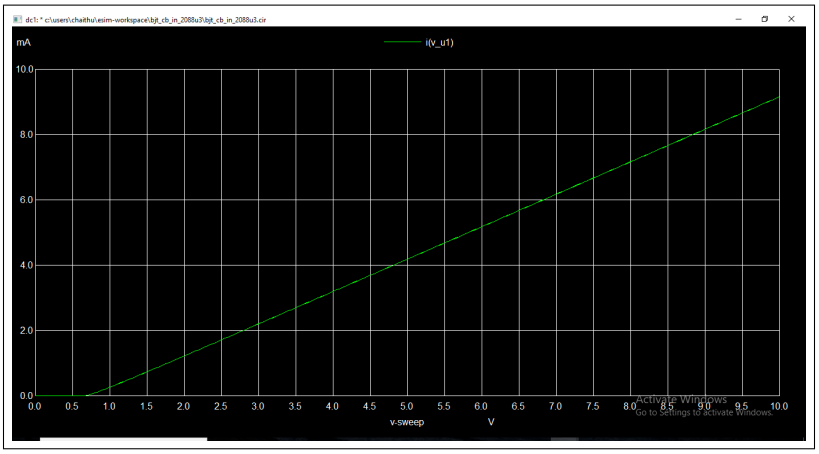


Figure 7.27: Common-Base Input Characteristics ($I_E$ vs $V_{EB}$)

### Common-Base Output Characteristics

This graph illustrates the relationship between collector current ($I_C$) and collector-base voltage ($V_{CB}$) at different constant emitter currents for a PNP transistor. Below is the testcircuit which reveals the output characterstis of common base configuration

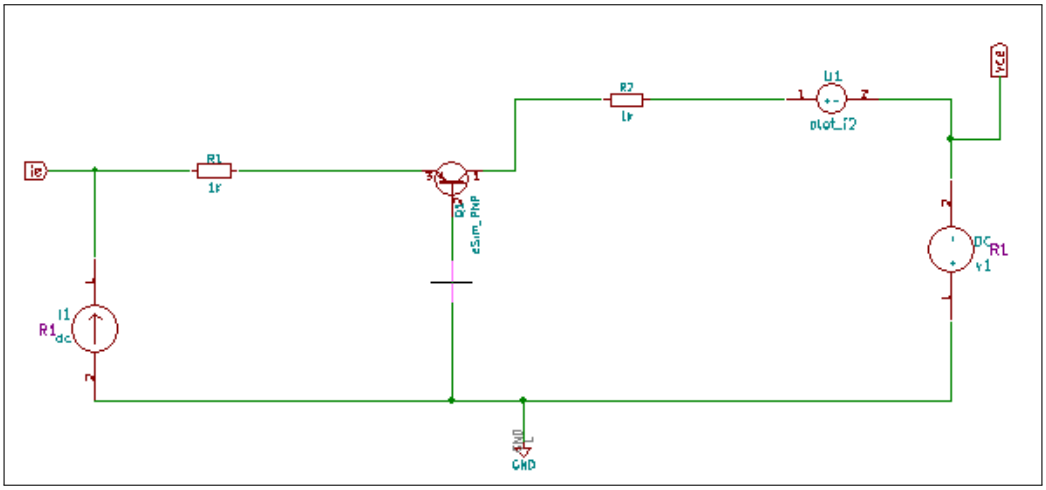


Figure 7.28: Common-Base Output Characteristic Circuit

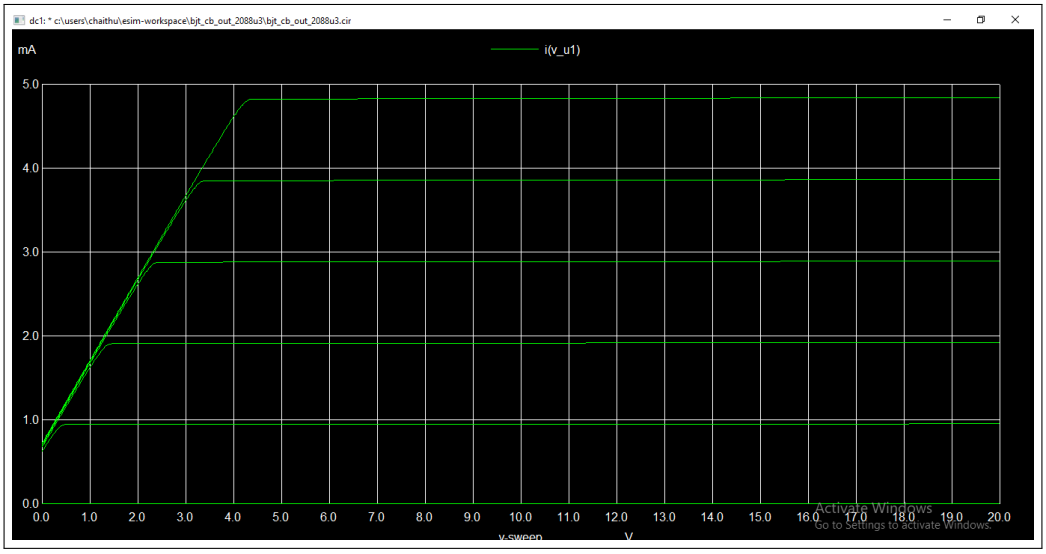


Figure 7.29: Common-Base Output Characteristics ($I_C$ vs $V_{CB}$)

The common-base output characteristics show the relationship between the collector current ($I_C$) and the collector-base voltage ($V_{CB}$) for different constant emitter currents ($I_E$). The curves are nearly flat, indicating that $I_C$ remains mostly constant as $V_{CB}$ increases. This reflects a high output resistance and makes the configuration suitable for current amplification.

### 7.4.5 Common-Emitter Configuration

#### Common-Emitter Input Characteristics

The input characteristic curve for the common-emitter configuration of a PNP transistor shows base current ($I_B$) versus base-emitter voltage ($V_{BE}$) at constant collector-emitter voltages ($V_{CE}$).

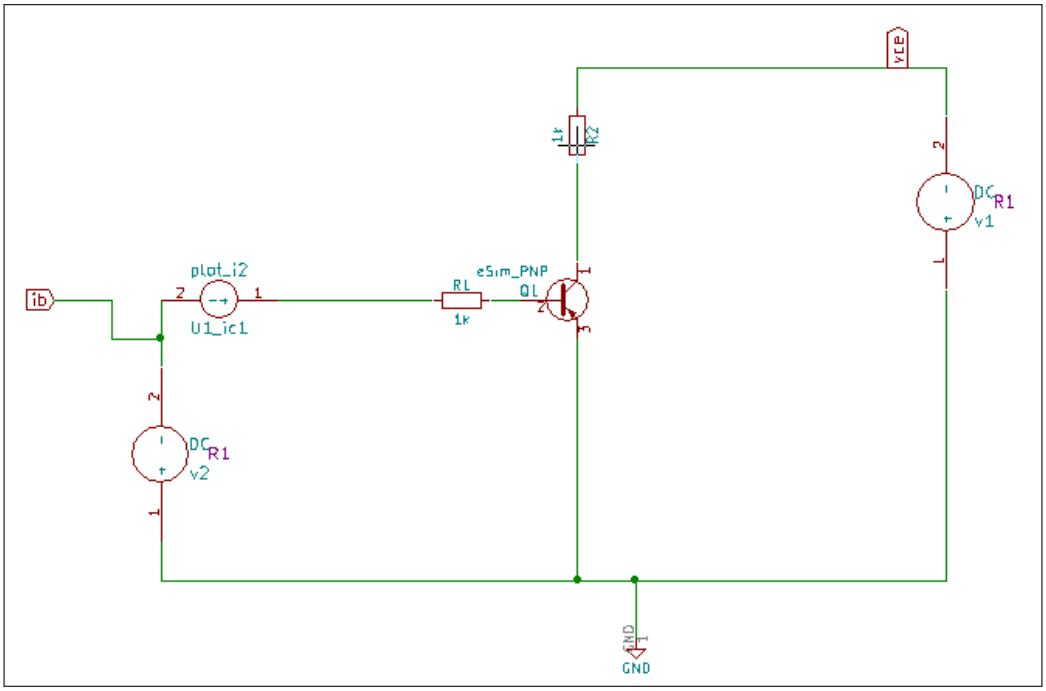


Figure 7.30: Common-Emitter Input Characteristic Circuit

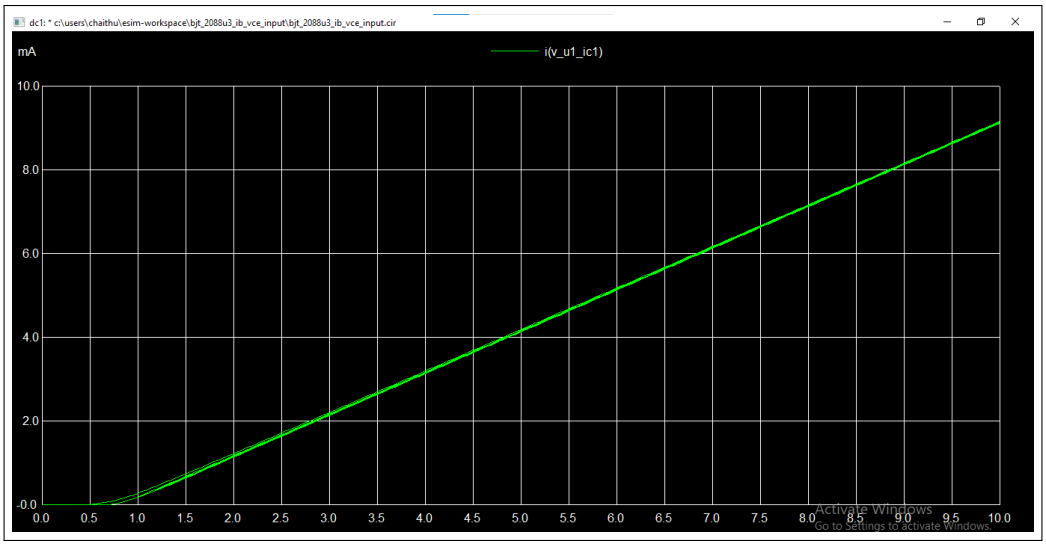


Figure 7.31: Common-Emitter Input Characteristics ($I_B$ vs $V_{BE}$)

The waveform resembles a forward-biased diode curve, where $I_B$ remains very small until $V_{BE}$ reaches approximately 0.6–0.7 V, after which it increases rapidly. This non-linear curve reflects the exponential behavior of the base-emitter junction.

### Common-Emitter Output Characteristics

The output characteristics plot collector current ($I_C$) versus collector-emitter voltage ($V_{CE}$) at various base current levels ($I_B$). Since it's a PNP device, current directions and voltage polarities are reversed compared to NPN.

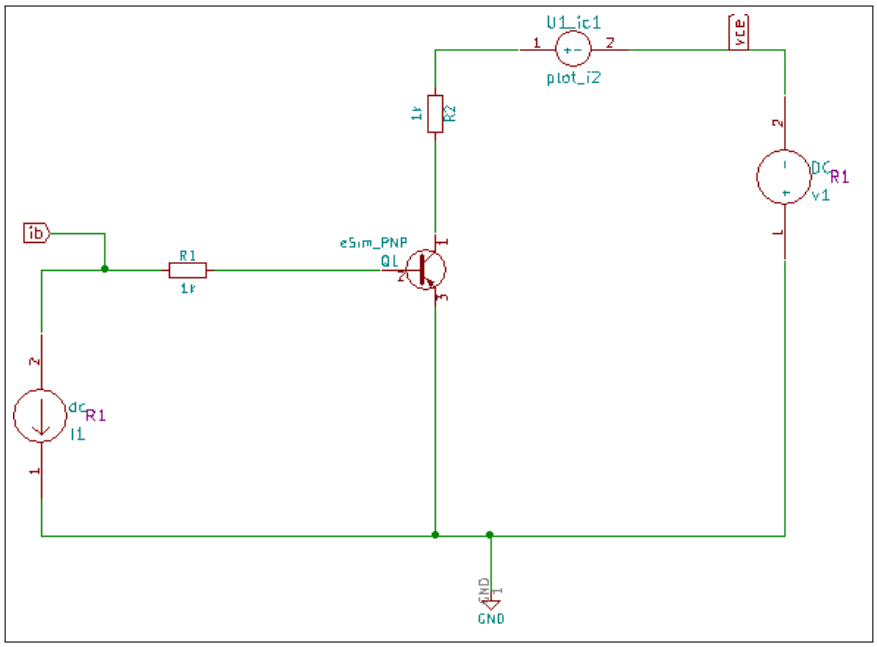


Figure 7.32: Common-Emitter Output Characteristic Circuit

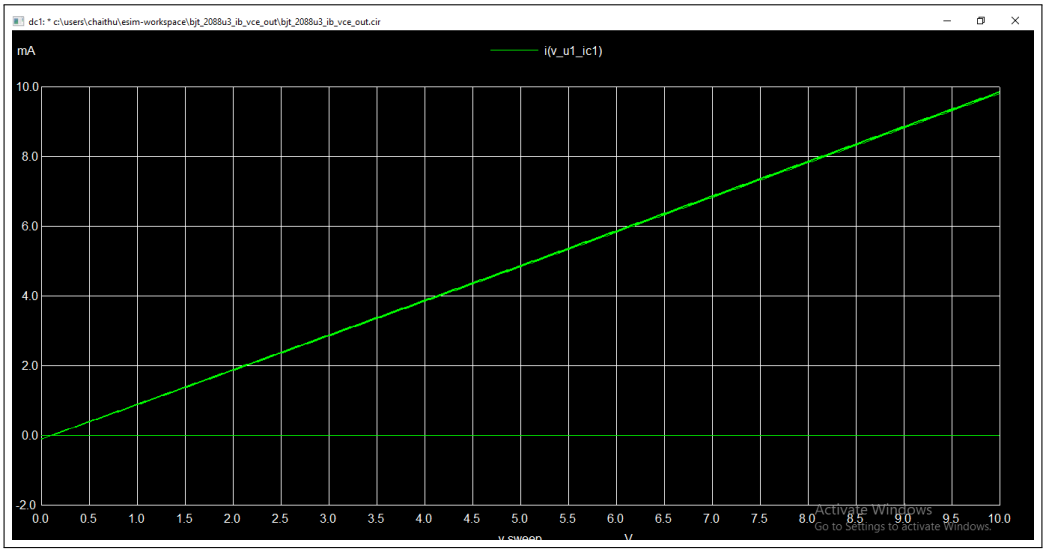


Figure 7.33: Common-Emitter Output Characteristics ($I_C$ vs $V_{CE}$)

In the common-emitter configuration, the output characteristics show the variation of collector current ($I_C$) with respect to collector-emitter voltage ($V_{CE}$) for different fixed base current levels ($I_B$). Each curve begins with a steep rise in $I_C$ and then flattens out, indicating the active region. The output waveform clearly displays three regions: cutoff (no conduction), active (amplification), and saturation (maximum conduction), making it suitable for amplification and switching applications.

### 7.4.6 Frequency Response Analysis

To evaluate high-frequency performance, the amplifier is subjected to a frequency sweep. Gain and phase shift are analyzed to determine bandwidth and stability.

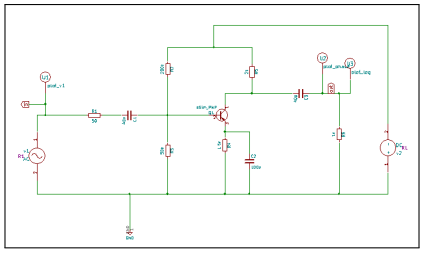

Figure 7.34: Frequency Response Test Circuit for Common-Emitter Amplifier

**Log-Magnitude Plot:**
This plot shows amplifier gain (in dB) versus frequency on a logarithmic scale. It reveals the cutoff frequency and bandwidth. The roll-off after the -3dB point helps determine the dominant pole behavior.

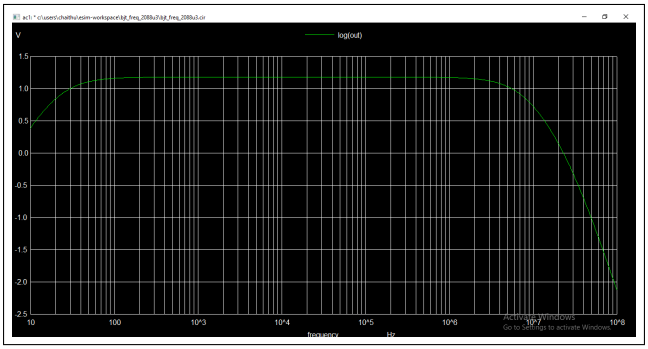


Figure 7.35: Log-Magnitude Plot (Gain vs Frequency)

**Phase Plot:**
The phase plot shows the variation of output signal phase relative to the input signal as the frequency increases. It helps assess phase margin and feedback system stability.

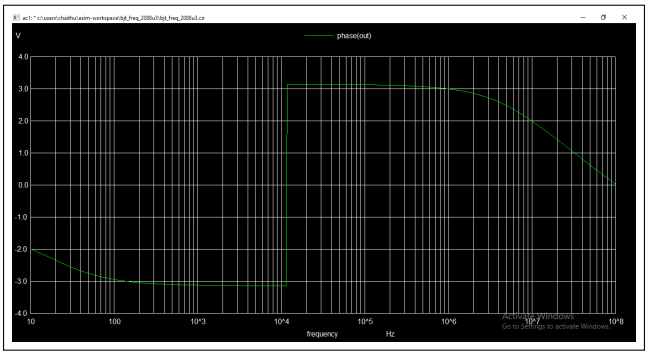


Figure 7.36: Phase Plot (Phase vs Frequency)

## 7.5 1N5819 : Schottky Barrier plastic rectifier

### 7.5.1 Description

The 1N5819 is a Schottky barrier rectifier designed for high-speed switching and low forward voltage drop applications. It is commonly used in low-voltage, high-frequency inverters, freewheeling diodes, and power management circuits. Due to its typical forward voltage drop of around 0.2 to 0.4 V, it offers higher efficiency compared to standard silicon diodes, especially in low-voltage DC applications.

Its fast switching capability makes it suitable for use in high-speed switching regulators, converters, and reverse polarity protection circuits. The diode also features low reverse leakage current and good thermal stability, enhancing its reliability in continuous operation. The 1N5819's compact packaging and low conduction loss make it an ideal choice for space-constrained designs such as battery-operated and portable devices.[22]

### 7.5.2 SPICE Model

```
.MODEL 1N5189 D( Is=0.2u Rs=0.05 N=1.373 Cjo=110p M=.35 Eg=.69
Xti=0 Iave=4 Vpk=40 )
```

### 7.5.3 Forward Bias Characteristics

The following circuit is used to study the forward bias behavior of the 1N5819. A DC source is applied in series with a current-limiting resistor and the diode.

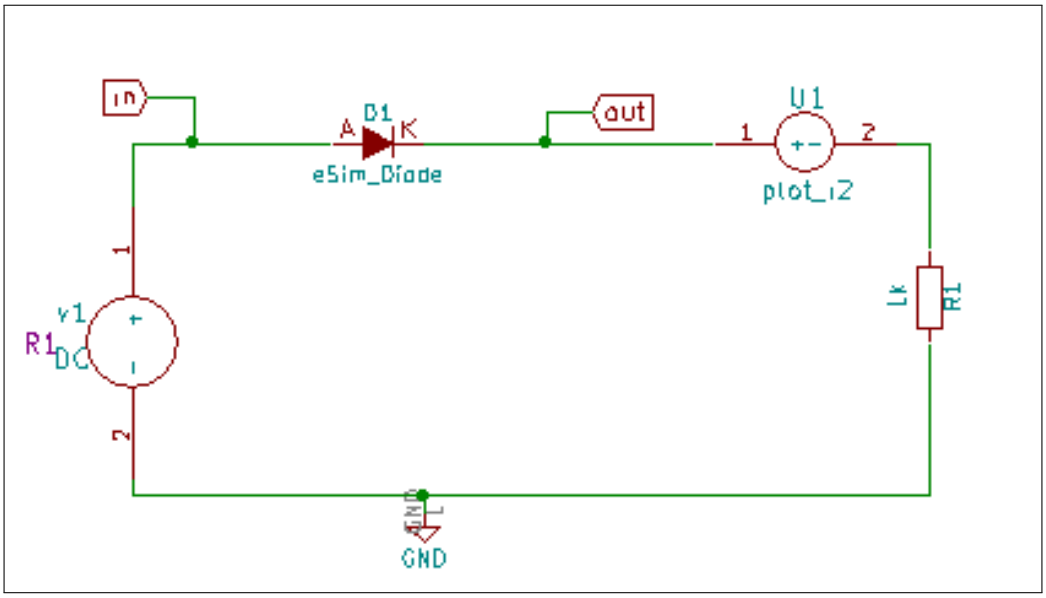


Figure 7.37: Forward Bias Test Circuit for 1N5819

### Forward Bias Characteristics – Waveform

The forward characteristics curve shows the diode's forward voltage ($V_F$) versus forward current ($I_F$). The 1N5819 starts conducting with a very low threshold, around 0.2–0.3V, rising linearly afterward.

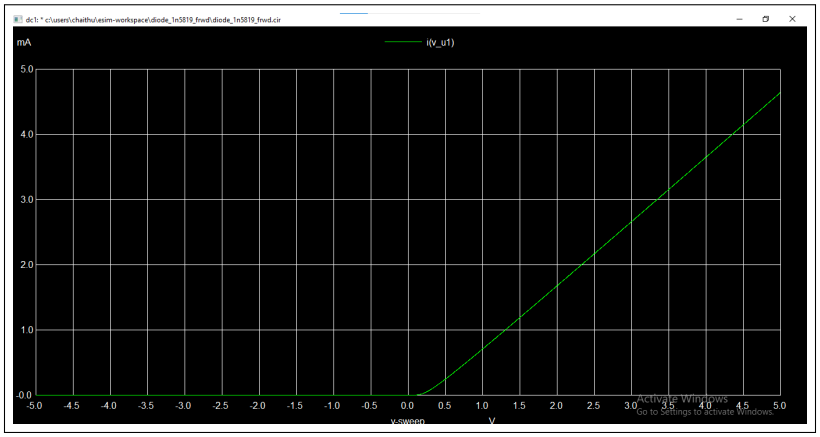


Figure 7.38: Forward Bias Characteristics ($V_F$ vs $I_F$)

## 7.5.4 Reverse Bias Characteristics

This test setup applies reverse voltage across the diode to analyze leakage current and breakdown behavior. A voltage source and ammeter are used in reverse configuration.

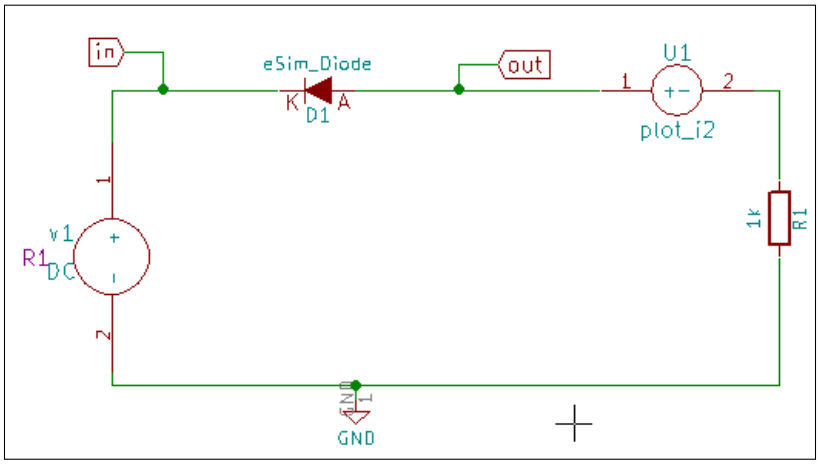


Figure 7.39: Reverse Bias Test Circuit for 1N5819

### Reverse Bias Characteristics – Waveform

The reverse characteristics plot reverse current ($I_R$) versus reverse voltage ($V_R$). The diode exhibits very low leakage current up to the rated reverse voltage of 40V.

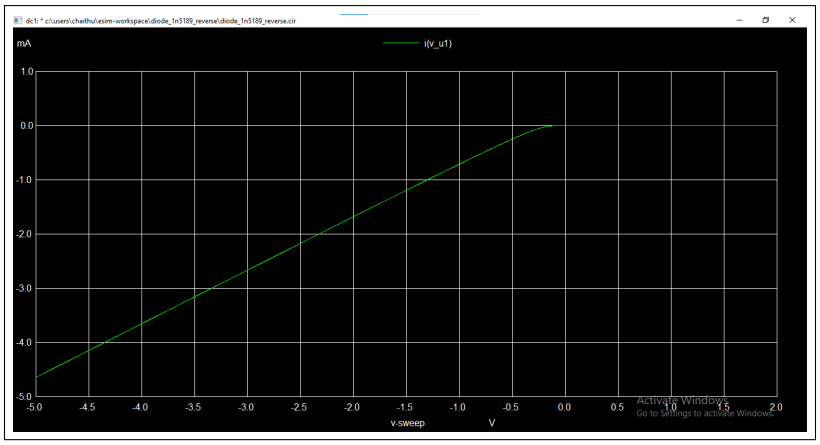


Figure 7.40: Reverse Bias Characteristics ($I_R$ vs $V_R$)

### 7.5.5 Half-Wave Rectifier using 1N5819 (Schottky Diode)

The 1N5819 is a Schottky diode with low forward voltage drop ( 0.2–0.3 V) and fast switching characteristics, making it suitable for low-voltage, high-efficiency rectification.

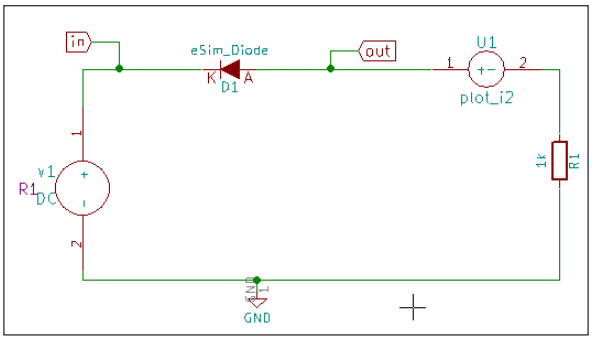


Figure 7.41: Half-Wave Rectifier Circuit using 1N5819

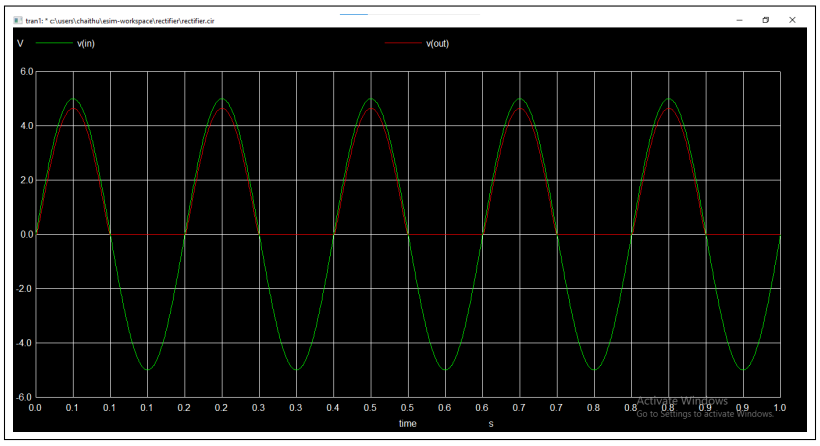


Figure 7.42: Output Waveform of Half-Wave Rectifier using 1N5819

**Observation:** Due to the low forward voltage of 1N5819, more of the input AC waveform is passed through, improving efficiency in low-voltage applications. However, its reverse leakage is higher compared to standard PN diodes.

# Bibliography


[1] eSim Official Website.
https://esim.fossee.in/

[2] NgSpice Official Website.
https://ngspice.sourceforge.io/

[3] Makerchip Official Website.
https://makerchip.com/

[4] KiCad Official Website.
https://www.kicad.org/

[5] GHDL Official Website.
http://ghdl.free.fr/

[6] eSim Manual v2.3.
https://static.fossee.in/esim/manuals/eSim_Manual_2.3.pdf

[7] CD4007UB Datasheet, Texas Instruments.
https://www.ti.com/lit/ds/symlink/cd4007ub.pdf

[8] SN74ALS29827 Datasheet, Texas Instruments.
https://www.alldatasheet.com/datasheet-pdf/download/838950/TI1/SN74ALS29827.html

[9] SN54HC164 Datasheet, Texas Instruments.
https://www.ti.com/lit/ds/symlink/sn54hc164.pdf

[10] 74HC194 Datasheet, Nexperia.
https://assets.nexperia.com/documents/data-sheet/74HC194.pdf

[11] IXDD414PI Datasheet, IXYS.
https://www.alldatasheet.com/datasheet-pdf/pdf/87433/IXYS/IXDD414PI.html

[12] 74FCT163501 Datasheet, DigChip.
https://www.digchip.com/datasheets/parts/datasheet/222/74FCT163501-pdf.php

[13] LM13700 Datasheet, Texas Instruments.
https://www.ti.com/lit/ds/symlink/lm13700.pdf

[14] 82S126 Datasheet, Philips.
https://www.alldatasheet.com/datasheet-pdf/pdf/91517/PHILIPS/82S126.html

[15] AM2901 Datasheet, AMD.
https://www.alldatasheet.com/datasheet-pdf/pdf/140842/AMD/AM2901.html

[16] TDC1008 Datasheet, TRW LSI Products.
https://datasheet4u.com/pdf-down/T/D/C/TDC1008_TRWLSIProducts.pdf

[17] CRC16 Software Module Datasheet, Infineon.
https://www.infineon.com/dgdl/Infineon-CRC16_V_3.2_13268-Software%20Module%20Datasheets-v03_02-EN.pdf?fileId=8ac78c8c7d0d8da4017d0fa99edb1a07

[18] SST406 Datasheet, Calogic.
https://www.alldatasheet.com/datasheet-pdf/download/57214/CALOGIC/SST406.html

[19] 2N4124 Datasheet, Philips.
https://www.alldatasheet.com/datasheet-pdf/pdf/15080/PHILIPS/2N4124.html

[20] MUR460 Datasheet, ON Semiconductor.
https://www.alldatasheet.com/datasheet-pdf/pdf/1923909/ONSEMI/MUR460.html

[21] 2S2088U3 Datasheet, ROHM Semiconductor.
https://fscdn.rohm.com/en/products/databook/datasheet/discrete/transistor/bipolar/2sa2088u3t106q-e.pdf

[22] 1N5189 Datasheet, Microsemi.
https://www.vishay.com/docs/88525/1n5817.pdf