Summer Fellowship Report

On

Integrated Circuit Design using Subcircuit feature of eSim

Submitted by

Nishit Bayen
B.Tech (Electronics & Communication Engineering)
Dr. B. C. Roy Engineering College, Durgapur

Under the guidance of

Mr. Sumanto Kar
Assistant Project Manager, FOSSEE
IIT Bombay

Prof. Kannan M. Moudgalya
Chemical Engineering Department
IIT Bombay

August 1, 2025

Acknowledgment

I extend my sincere gratitude to the FOSSEE team at IIT Bombay for the invaluable opportunity to work on the design and integration of sub-circuits using eSim. This internship has greatly enhanced my understanding of open-source tools and their real-world applications in electronic design.

I am especially thankful to Prof. Kannan M. Moudgalya for his visionary leadership in promoting open-source software in education. His initiative has enabled students like me to gain meaningful, hands-on experience.

My heartfelt thanks to Mr. Sumanto Kar for his constant guidance and encouragement throughout the internship. His support and insights were instrumental in navigating challenges and achieving project goals.

I truly appreciate FOSSEE's mission to make advanced tools freely accessible, fostering innovation and inclusivity in engineering education. This internship has been a valuable and memorable step in my journey toward a career in the semiconductor and VLSI domain.

Finally, I wish to thank my parents for their support and encouragement throughout my study.

Contents

1 Introduction **5**
- 1.1 eSim . . . 5
- 1.2 NgSpice . . . 5
- 1.3 KiCad . . . 6
  - 1.3.1 Makerchip . . . 6

2 Features of eSim **7**

3 Problem Statement **9**
- 3.1 Subcircuit Design and Integration . . . 9
  - 3.1.1 Approach . . . 9
- 3.2 Mixed Signal Simulation using NgVeri . . . 11
  - 3.2.1 Approach . . . 11

4 Subcircuit Design **12**
- 4.1 **CD4044BMS:** CMOS Quad 3 State R/S Latches . . . 12
  - 4.1.1 **Description:** . . . 12
  - 4.1.2 **Truth Table** . . . 12
  - 4.1.3 **Pin Diagram** . . . 13
  - 4.1.4 **Subcircuit** . . . 13
  - 4.1.5 **Test Circuit** . . . 14
  - 4.1.6 **Waveform** . . . 14
- 4.2 **CD4042B:** Quad Clocked 'D' Latch . . . 15
  - 4.2.1 **Description:** . . . 15
  - 4.2.2 **Pin Diagram** . . . 15
  - 4.2.3 **Subcircuit** . . . 15
  - 4.2.4 **Subcircuit D Flipflop** . . . 16
  - 4.2.5 **Functional Block Diagram** . . . 16
  - 4.2.6 **Test Circuit** . . . 17
  - 4.2.7 **Waveform** . . . 17
- 4.3 **CD4068B:** CMOS 8 input NAND/AND Gate . . . 18
  - 4.3.1 **Description:** . . . 18
  - 4.3.2 **Pin Diagram** . . . 18
  - 4.3.3 **Subcircuit** . . . 18
  - 4.3.4 **Test Circuit** . . . 19
  - 4.3.5 **Waveform** . . . 19
- 4.4 **TC74HC4028AP:** BCD to Decimal Decoder . . . 20

4.4.1 **Description:** . . . 20
4.4.2 **Pin Diagram** . . . 20
4.4.3 **Subcircuit** . . . 21
4.4.4 **Test Circuit** . . . 21
4.4.5 **Waveform** . . . 21
4.5 **SN74HCS259:** Automotive 8-Bit Addressable Latches with Schmitt-Trigger Inputs . . . 22
4.5.1 **Description:** . . . 22
4.5.2 **Pin Diagram** . . . 22
4.5.3 **Subcircuit** . . . 22
4.5.4 **Test Circuit** . . . 23
4.5.5 **Waveform** . . . 23
4.6 **SN74ALS1011A:** TRIPLE 3-INPUT POSITIVE-AND BUFFERS . . . 24
4.6.1 **Description:** . . . 24
4.6.2 **Pin Diagram** . . . 24
4.6.3 **Subcircuit** . . . 24
4.6.4 **Test Circuit** . . . 25
4.6.5 **Waveform** . . . 25
4.7 **SN74LS548:** 8-Bit Two-Stage Pipelined Register . . . 26
4.7.1 **Description:** . . . 26
4.7.2 **Truth Table** . . . 26
4.7.3 **Pin Diagram** . . . 26
4.7.4 **Subcircuit** . . . 27
4.7.5 **Test Circuit** . . . 27
4.7.6 **Waveform** . . . 27
4.8 **SN74ABT337:** CLOCK DRIVER WITH 3-STATE OUTPUTS . . . 28
4.8.1 **Description:** . . . 28
4.8.2 **Pin Diagram** . . . 28
4.8.3 **Subcircuit** . . . 28
4.8.4 **Test Circuit** . . . 29
4.8.5 **Waveform** . . . 29
4.9 **SN54L99:** 4-BIT RIGHT-SHIFT LEFT-SHIFT REGISTERS . . . 30
4.9.1 **Description:** . . . 30
4.9.2 **Pin Diagram** . . . 30
4.9.3 **Subcircuit** . . . 30
4.9.4 **Test Circuit** . . . 31
4.9.5 **Waveform** . . . 31
4.10 **SN54L98:** 4-BIT DATA SELECTORS/STORAGE REGISTERS . . . 32
4.10.1 **Description:** . . . 32
4.10.2 **Pin Diagram** . . . 32
4.10.3 **Subcircuit** . . . 32
4.10.4 **Test Circuit** . . . 33
4.10.5 **Waveform** . . . 33

5 Mixed Signal Simulation **34**
5.1 **74HC181:** 4-bit arithmetic logic unit . . . 34
5.1.1 **Description:** . . . 34
5.1.2 **Verilog Code:** . . . 34
5.1.3 **Pin Diagram** . . . 36
5.1.4 **Test Circuit** . . . 36
5.1.5 **Makerchip Simulation** . . . 37
5.1.6 **NgSpice Waveform** . . . 37
5.2 **74HC153:** Dual 4-input multiplexer . . . 38
5.2.1 **Description:** . . . 38
5.2.2 **Verilog Code:** . . . 38
5.2.3 **Pin Diagram** . . . 39
5.2.4 **Test Circuit** . . . 39
5.2.5 **Makerchip Simulation** . . . 40
5.2.6 **NgSpice Waveform** . . . 40
5.3 **74HC148:** 8-Line to 3-Line Priority Encoder . . . 41
5.3.1 **Description:** . . . 41
5.3.2 **Pin Diagram** . . . 42
5.3.3 **Test Circuit** . . . 42
5.3.4 **Makerchip Simulation** . . . 43
5.3.5 **NgSpice Waveform** . . . 43
5.4 **74LS47:** BCD-TO-SEVEN-SEGMENT DECODERS . . . 44
5.4.1 **Description:** . . . 44
5.4.2 **Pin Diagram** . . . 45
5.4.3 **Test Circuit** . . . 45
5.4.4 **Makerchip Simulation** . . . 46
5.4.5 **NgSpice Waveform** . . . 46
5.5 **74HC161:** Presettable synchronous 4-bit binary counter; asynchronous reset . . . 47
5.5.1 **Description:** . . . 47
5.5.2 **Pin Diagram** . . . 48
5.5.3 **Test Circuit** . . . 48
5.5.4 **Makerchip Simulation** . . . 49
5.5.5 **NgSpice Waveform** . . . 49

6 Conclusion and Future Scope **50**

Bibliography **51**

Chapter 1

Introduction

1.1 eSim

eSim is an open-source Electronic Design Automation (EDA) tool developed by FOSSEE, IIT Bombay, aimed at providing a free alternative to commercial circuit design and simulation software for academic and research use.

It integrates several open-source tools to offer a comprehensive design and simulation environment:

- **KiCad** – for schematic and PCB design
- **NgSpice** – for analog circuit simulation
- **GHDL** – for VHDL-based digital simulation
- **Makerchip** – for online TL-Verilog-based digital design and verification

1.2 NgSpice

NgSpice is an open-source circuit simulator integrated into eSim for running analog and mixed-signal simulations. Built on the SPICE engine, it supports DC, AC, transient, and parametric analyses, making it versatile for a variety of circuit types.

It works with industry-standard SPICE models, supports subcircuits and behavioral modeling, and serves as the simulation backend for KiCad schematics in eSim. Users can view voltage, current, and frequency response waveforms, making NgSpice a reliable tool for testing and validating designs before moving to hardware.

1.3 KiCad

KiCad is an open-source tool for schematic capture and PCB design, fully integrated into eSim. It enables users to create multi-layer boards, define custom footprints, and run design rule checks to ensure accuracy before fabrication.

In eSim, KiCad serves as the main schematic editor, making it easy to build circuits using components from rich open-source libraries. These schematics can be directly used for simulation and PCB layout. Features like 3D PCB visualization, Gerber export, and electrical rule checking help streamline the design process. With its seamless integration, KiCad supports a complete open-source workflow—from schematic to simulation to final board design.

1.3.1 Makerchip

Makerchip is a cloud-based platform for digital circuit design using **TL-Verilog** (Transaction-Level Verilog), a powerful extension of traditional Verilog. Integrated within eSim, Makerchip allows users to write, simulate, and debug TL-Verilog code directly in the browser, without requiring local installation.

Its key features include real-time waveform visualization, instant simulation feedback, and support for modular, high-level hardware design. By abstracting away low-level timing details, TL-Verilog simplifies the creation of scalable and reusable digital systems. Makerchip's integration with eSim offers a modern and open-source environment for learning and exploring digital logic design.

Chapter 2

Features of eSim

eSim offers a complete open-source platform for circuit design, simulation, and PCB layout, making it a powerful alternative to commercial EDA tools.

Key Features

- **Open Source & Free:** Entirely free and open-source, eSim removes the cost barrier associated with commercial EDA tools—ideal for academia and research.
- **Integrated Toolchain:** Combines multiple open-source tools like:
  - **KiCad** – for schematic capture and PCB design
  - **NgSpice** – for analog circuit simulation
  - **GHDL** – for VHDL-based digital simulation
  - **Makerchip** – for TL-Verilog-based design and debugging
  - **NGVeri** - for Mixed Signal Simulation
- **Schematic Creation:** User-friendly interface with drag-and-drop components and editing tools for easy circuit building.
- **Circuit Simulation:** Supports DC, AC, and transient analysis using SPICE. Integrated waveform viewer aids in visualization and debugging.
- **Mixed-Signal Support:** Simulates both analog and digital circuits in one environment.
- **PCB Design:** Provides PCB layout editing, design rule checks (DRC), and Gerber file export for fabrication.
- **Subcircuit Feature:** Enables hierarchical design using reusable submodules, simplifying large circuit development.
- **Device Modeling:** Allows simulation of custom semiconductor devices using real-world parameters.

- **SkyWater SKY130 PDK Support:** Supports simulation using an open-source 130nm process design kit for IC research and design.
- **Cross-Platform Support:** Compatible with Linux and Windows operating systems.
- **Community & Documentation:** Well-documented with tutorials and strong community support from the FOSSEE initiative.

Chapter 3

Problem Statement

The objective of this project is to design and develop Digital Integrated Circuits using the Subcircuit feature of eSim & perform Mixed-Signal Simulation using NGVeri..The developed models enhance the eSim library and extend its functionality for future academic and industrial applications.

3.1 Subcircuit Design and Integration

Digital ICs are modeled as subcircuits based on datasheet specifications and integrated into eSim's subcircuit library.

3.1.1 Approach

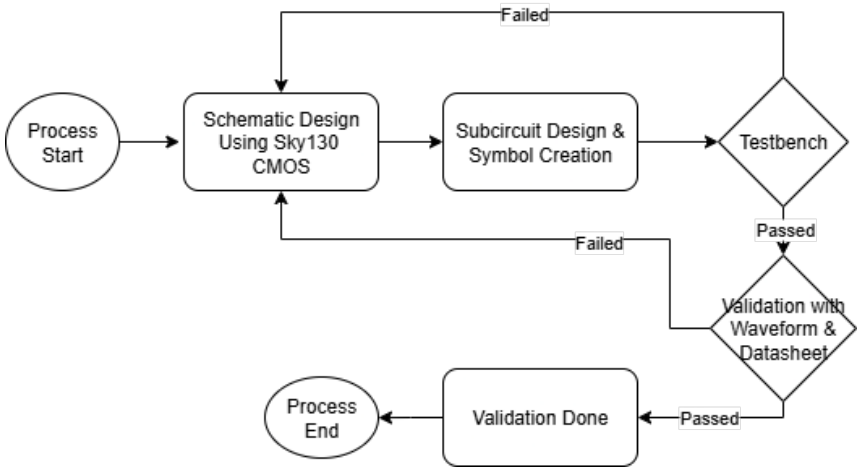

Figure 3.1: Flowchart for Subcircuit Design, Testing and Validation

- **Datasheet Analysis:** Datasheets from manufacturers like Texas Instruments, Analog Devices are studied to extract internal circuit diagrams, truth tables and functional blocks.

- **Subcircuit Creation:** Using eSim's Kicad schematic editor and Sky130 library, IC subcircuits are created in compliance with the datasheet's specifications.
- **Symbol Creation:** Custom symbols for ICs are designed according to their pin layout to ensure accurate integration.
- **Test Circuit Design:** Standard test circuits are constructed based on the application circuits and test conditions provided in the datasheet.
- **Simulation and Validation:** The designed test circuits are simulated using NgSpice in eSim. Simulation results such as waveforms are compared with expected outcomes from the datasheet.
- **Iteration if Needed:** If the output does not match expectations, the design is reviewed and modified, and the test process is repeated.
- **Finalization:** Once the subcircuit passes validation, it is considered verified and added to the eSim subcircuit library.

3.2 Mixed Signal Simulation using NgVeri

This part involves using eSim's NGVeri feature for mixed-signal simulation by integrating analog circuits with Verilog digital designs. NGVeri allows simultaneous simulation of analog (NgSpice) and digital (Verilator) domains.

3.2.1 Approach

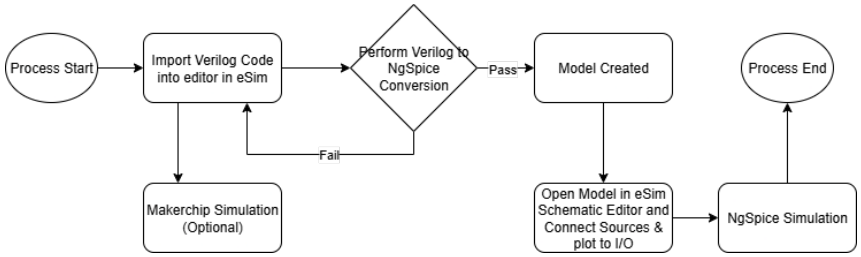

Figure 3.2: Flowchart for Mixed Signal Simulation using NgVeri

- **Verilog Code Development:** The digital design was first written using Verilog, SystemVerilog, or TL-Verilog depending on the requirement.
- **Code Verification using Icarus Verilog:** The Verilog code was initially verified using Icarus Verilog to ensure functionality before integration.
- **Makerchip IDE (Optional):** Additional verification was carried out using Makerchip IDE for debugging and simulation.
- **NGVeri Model Creation:** Once verified, the Verilog files were imported into NGVeri. If TL-Verilog was used, it was converted into SystemVerilog using SandPiper SaaS. The final compilation was done using Verilator to generate the NGVeri model.
- **Schematic Design:** The analog portion of the circuit was designed using eSim's schematic editor with the KiCad interface.
- **Mixed-Signal Simulation:** The complete mixed-signal simulation was performed using NGVeri, which integrates NgSpice and Verilator. The outputs were verified using eSim's plotting tools and NgSpice.

Chapter 4

Subcircuit Design

The Subcircuit feature of eSim allows hierarchical modeling of complex Integrated Circuits by combining multiple basic components into a single reusable block. By utilizing datasheets and internal schematics of various ICs, accurate subcircuit models are created using eSim's schematic editor.

4.1 CD4044BMS: CMOS Quad 3 State R/S Latches

4.1.1 Description:

CD4044BMS types are quad cross-coupled 3-state CMOS NAND latches. Each latch has a separate Q outputand individual SET and RESET inputs. The Q outputs are con-trolled by a common ENABLE input. A logic "1" or high on theENABLE input connects the latch states to the Q outputs. A logic"0" or low on the ENABLE input disconnects the latch states fromthe Q outputs, results in an open circuit feature allows commonbusing of the outputs.

4.1.2 Truth Table

S	R	E	Q
X	X	0	OC*
1	1	1	NC**
0	1	1	1
1	0	1	0
0	0	1	ΔΔ

* Open Circuit
** No Change
ΔΔ Dominated by R = 0 input

4.1.3 Pin Diagram

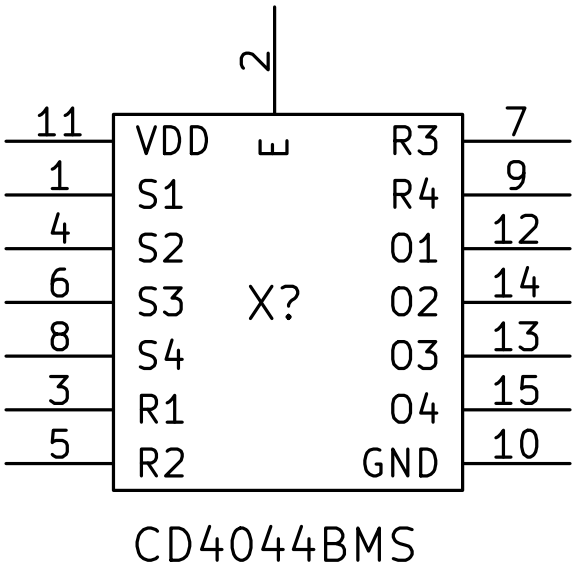

Figure 4.1: Pin Diagram of CD4044BMS

4.1.4 Subcircuit

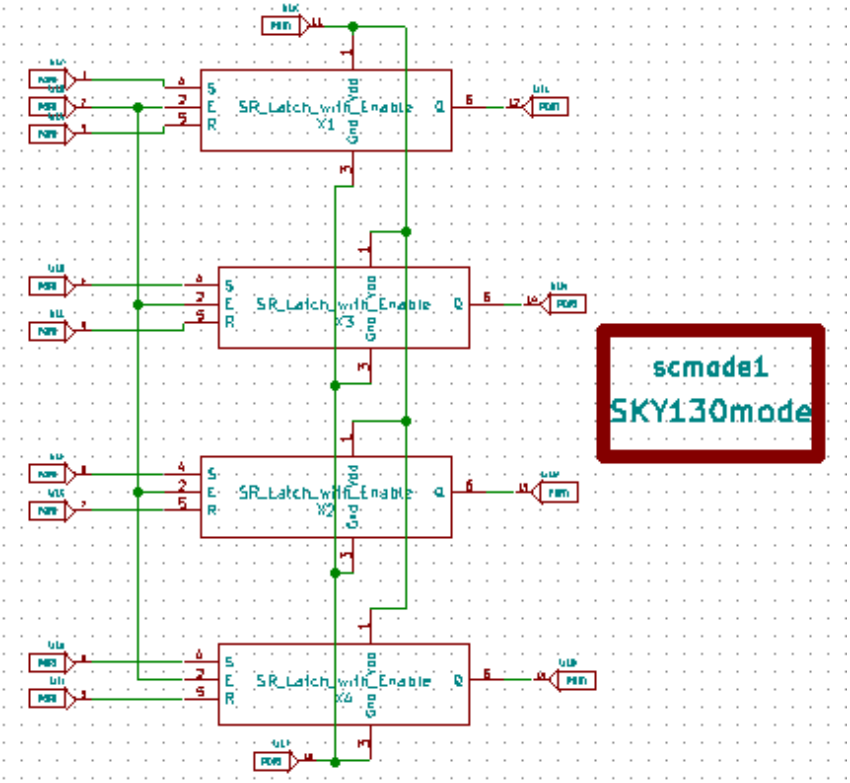

Figure 4.2: Subcircuit for CD4044BMS

4.1.5 Test Circuit

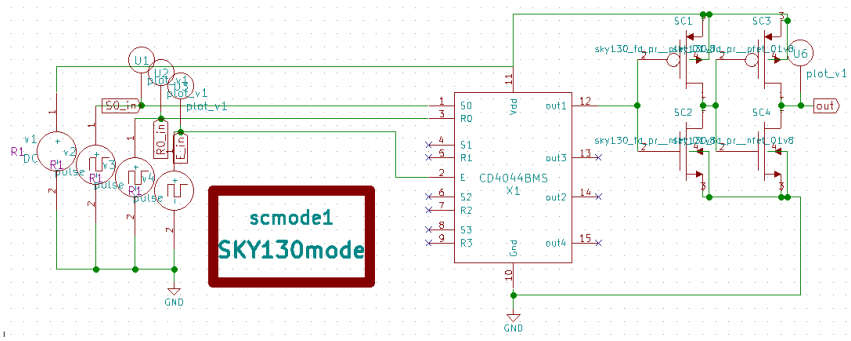

Figure 4.3: Testbench for CD4044BMS Subcircuit

4.1.6 Waveform

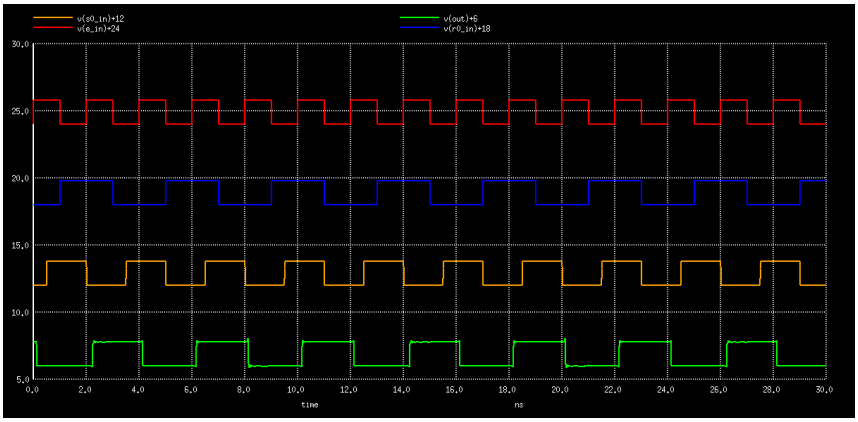

Figure 4.4: Input Output Waveform for CD4044BMS Subcircuit

4.2 CD4042B: Quad Clocked 'D' Latch

4.2.1 Description:

CD4042B types contain four latch circuits, each strobed by a common clock. Complementary buffered outputs are available from each circuit. The impedance of the n- and p-channel output devices is balanced and all outputs are electrically identical. Information present at the data input is transferred to outputs Q and Q‘ during the CLOCK level which is programmed by the POLARITY input. For POLARITY = 0 the transfer occurs during the 0 CLOCK level and for POLARITY = 1 the transfer occurs during the 1 CLOCK level. The outputs follow the data input providing the CLOCK and POLARITY levels defined above are present. When a CLOCK transition occurs (positive for POLARITY = 0 and negative for POLARITY = 1) the information present at the input during the CLOCK transition is retained at the outputs until an opposite CLOCK transition occurs.

4.2.2 Pin Diagram

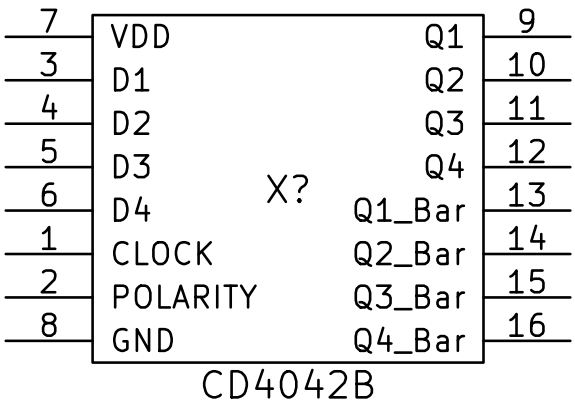

Figure 4.5: Pin Diagram of CD4042B

4.2.3 Subcircuit

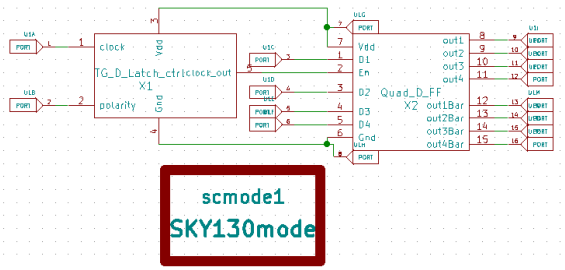

Figure 4.6: Subcircuit for CD4042B

4.2.4 Subcircuit D Flipflop

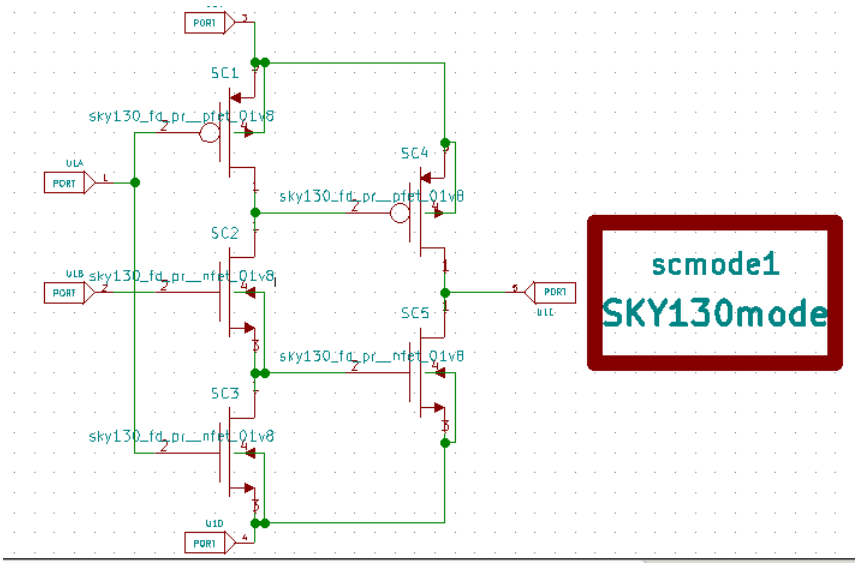

Figure 4.7: D Flipflop of Subcircuit CD4042B

4.2.5 Functional Block Diagram

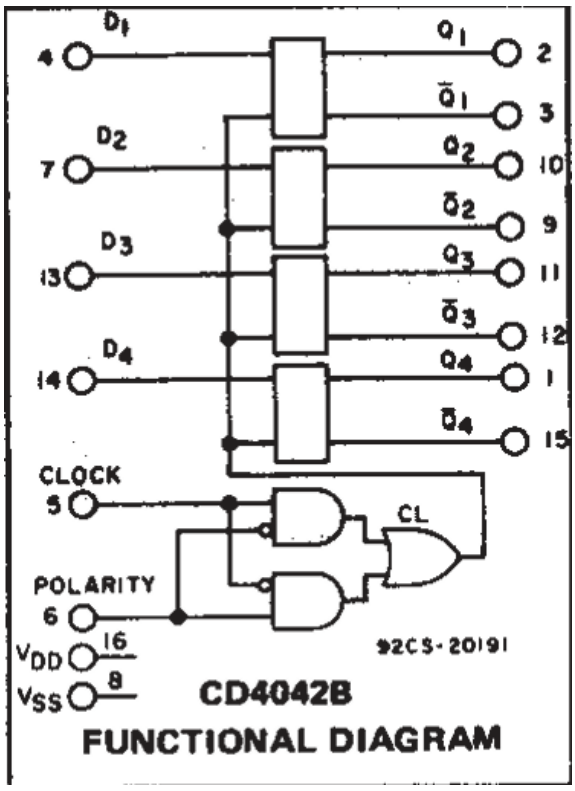

Figure 4.8: Functional Block Diagram of CD4042B

4.2.6 Test Circuit

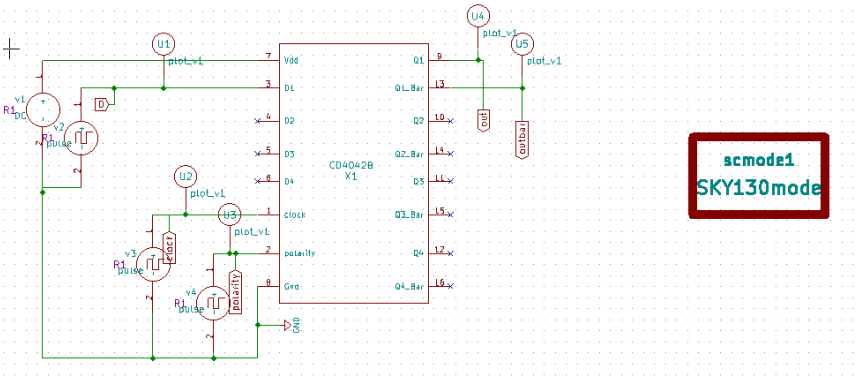

Figure 4.9: Testbench for CD4042B Subcircuit

4.2.7 Waveform

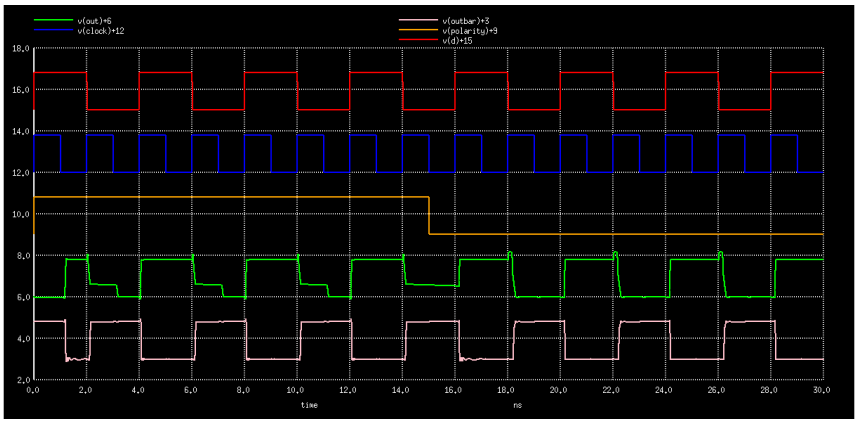

Figure 4.10: Input Output Waveform for CD4042B Subcircuit

4.3 CD4068B: CMOS 8 input NAND/AND Gate

4.3.1 Description:

CD4068B NAND/AND gate provides the system designer with direct imple- mentation of the positive-logic 8-input NAND and AND functions and supplements the existing family of CMOS gates.

4.3.2 Pin Diagram

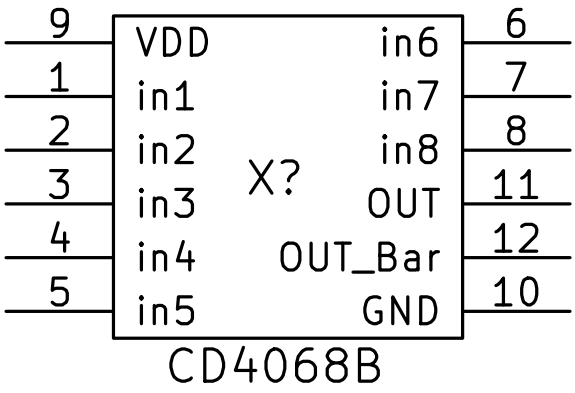

Figure 4.11: Pin Diagram of CD4068B

4.3.3 Subcircuit

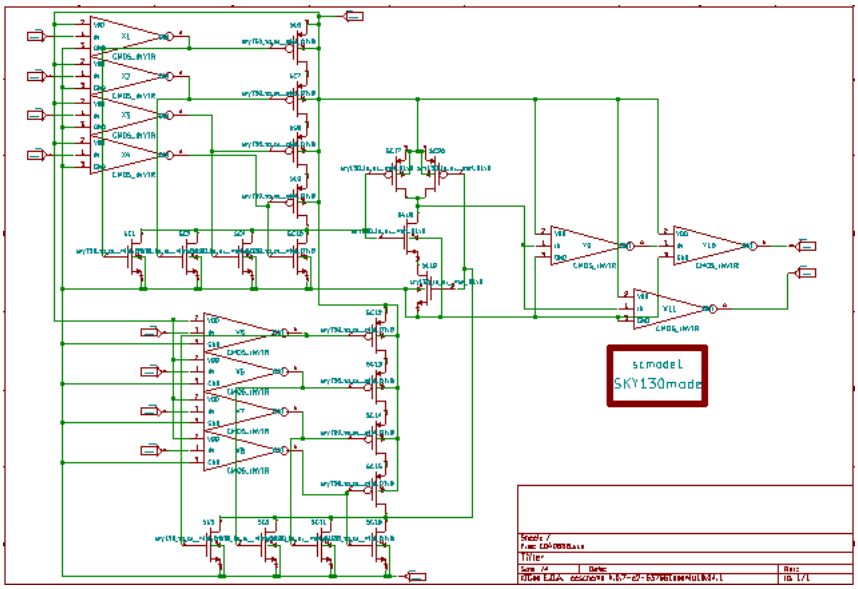

Figure 4.12: Subcircuit for CD4068B

4.3.4 Test Circuit

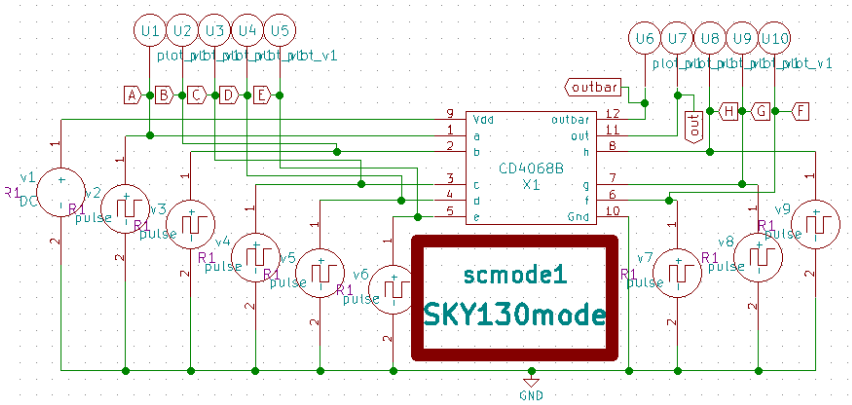

Figure 4.13: Testbench for CD4068B Subcircuit

4.3.5 Waveform

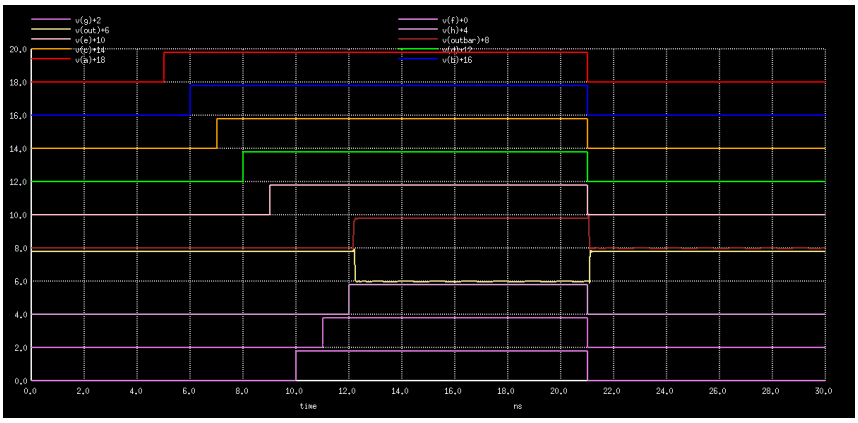

Figure 4.14: Input Output Waveform for CD4068B Subcircuit

4.4 TC74HC4028AP: BCD to Decimal Decoder

4.4.1 Description:

The TC74HC4028A is a high speed CMOS BCD-to-DECIMAL DECODER, It achieves high speed operation similar to equivalent LSTTL while maintaining CMOS low power dissipation. A BCD code applied to the four inputs (A - D) sets a high level at one of ten decoded outputs. An illegal BCD code such as eleven thru fifteen sets all outputs low. This device can be used as 3 - to - 8 LINE DECODER when input D is held high. This device is useful for code conversion, address decoding, memory selection, multiplexing, or readout decoding. All inputs are equipped with protection circuits against static discharge or transient excess voltage.

4.4.2 Pin Diagram

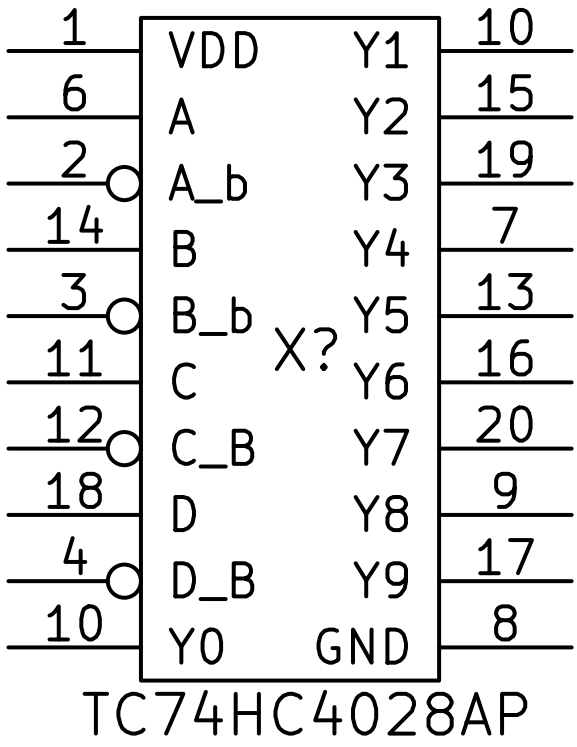

Figure 4.15: Pin Diagram of TC74HC4028AP

4.4.3 Subcircuit

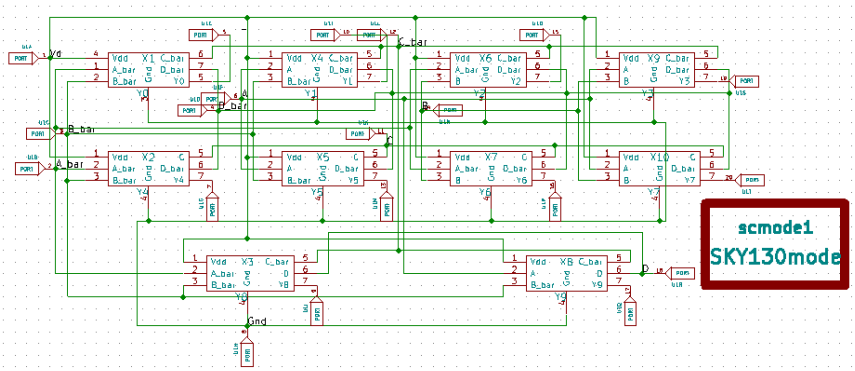

Figure 4.16: Subcircuit for TC74HC4028AP

4.4.4 Test Circuit

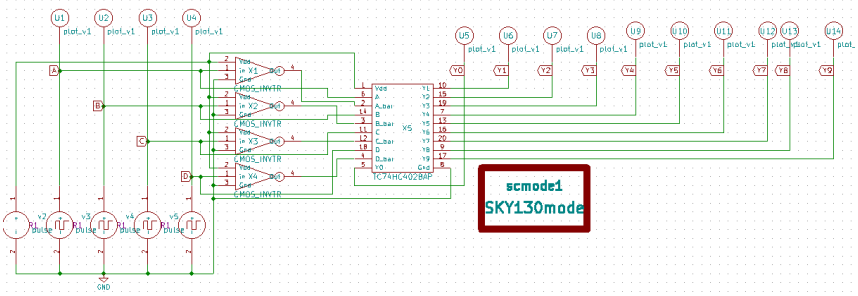

Figure 4.17: Testbench for TC74HC4028AP Subcircuit

4.4.5 Waveform

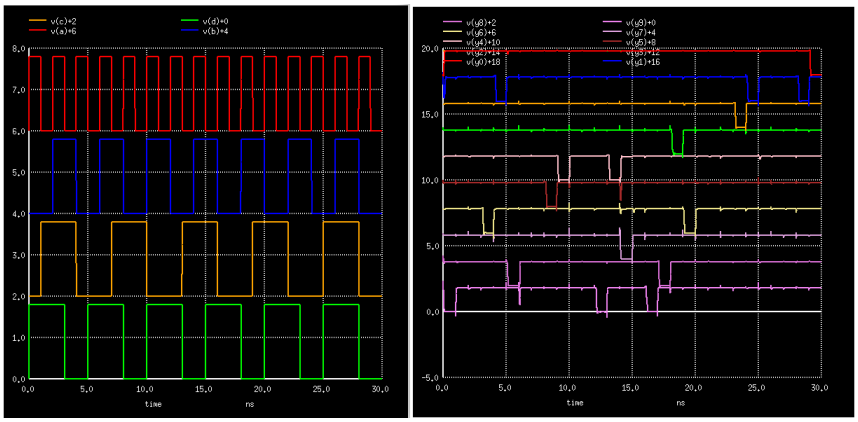

Figure 4.18: Input Output Waveform for TC74HC4028AP Subcircuit

4.5 SN74HCS259: Automotive 8-Bit Addressable Latches with Schmitt-Trigger Inputs

4.5.1 Description:

The SN74HCS259 8-bit addressable latches are designed for general-purpose storage applications in digital systems. Specific uses include working registers, serial-holding registers, and active-high decoders or demultiplexers. They are multifunctional devices capable of storing single-line data in eight addressable latches and being a 1-of-8 decoder or demultiplexer with active-high outputs.

4.5.2 Pin Diagram

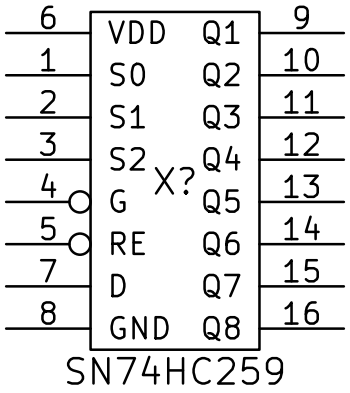

Figure 4.19: Pin Diagram of SN74HCS259

4.5.3 Subcircuit

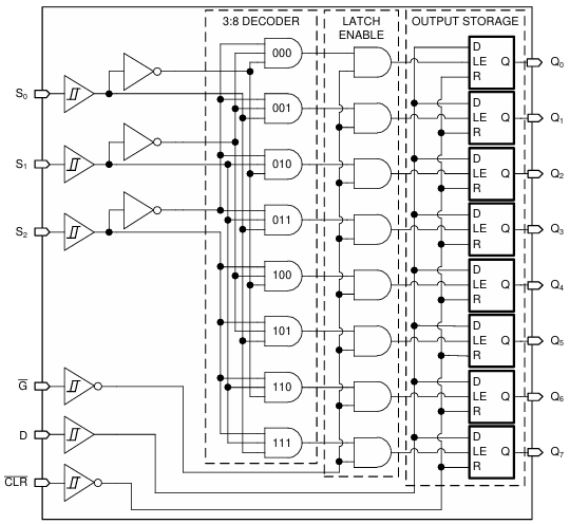

Figure 4.20: Subcircuit for SN74HCS259

4.5.4 Test Circuit

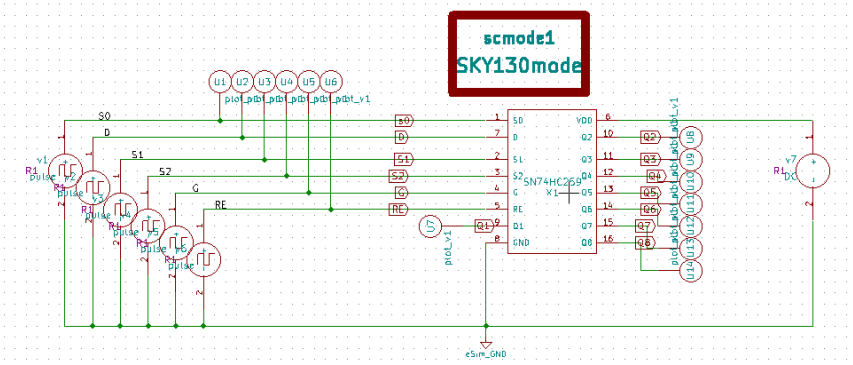

Figure 4.21: Testbench for SN74HCS259 Subcircuit

4.5.5 Waveform

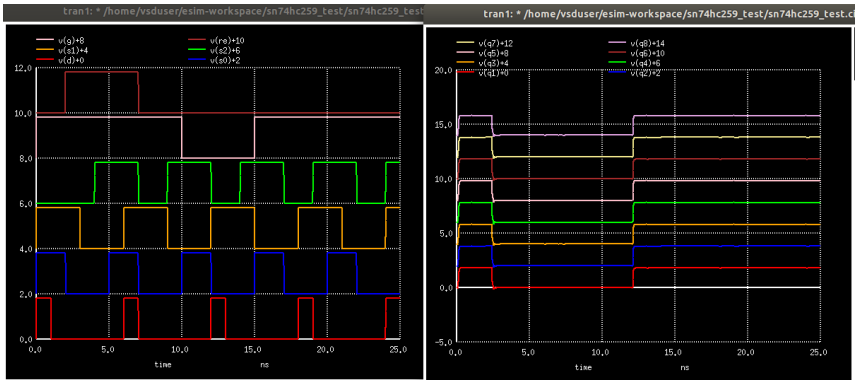

Figure 4.22: Input Output Waveform for SN74HCS259 Subcircuit

4.6 SN74ALS1011A: TRIPLE 3-INPUT POSITIVE-AND BUFFERS

4.6.1 Description:

These devices contain three independent 3-input AND buffers. They perform the Boolean functions

$$Y = A \cdot B \cdot C \quad \text{or} \quad Y = \overline{A + B + C}$$

in positive logic.

4.6.2 Pin Diagram

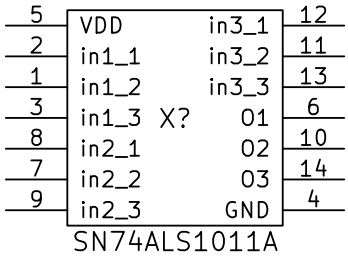

Figure 4.23: Pin Diagram of SN74ALS1011A

4.6.3 Subcircuit

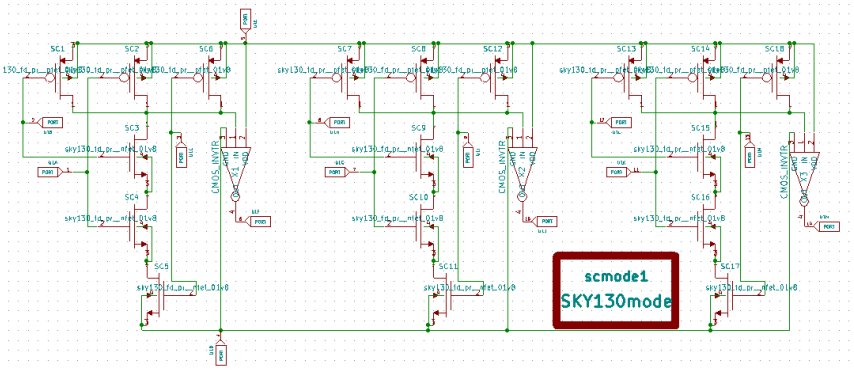

Figure 4.24: Subcircuit for SN74ALS1011A

4.6.4 Test Circuit

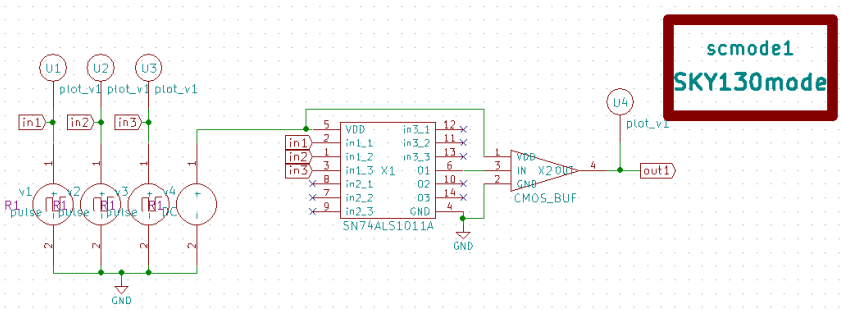

Figure 4.25: Testbench for SN74ALS1011A Subcircuit

4.6.5 Waveform

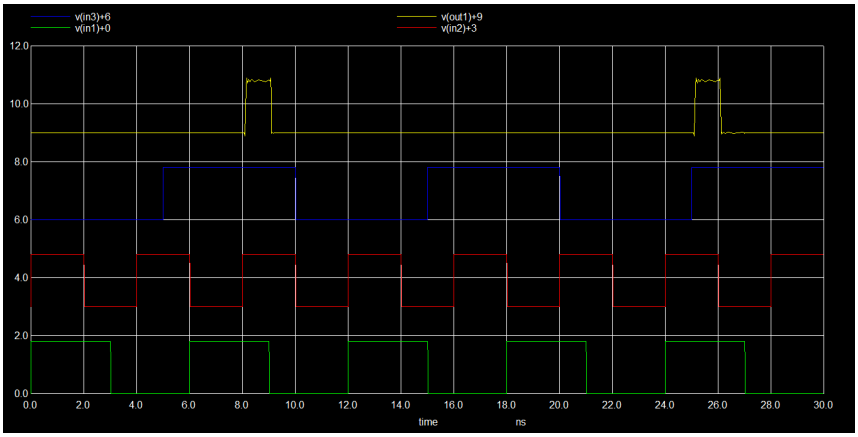

Figure 4.26: Input Output Waveform for SN74ALS1011A Subcircuit

4.7 SN74LS548: 8-Bit Two-Stage Pipelined Register

4.7.1 Description:

The 74LS548 is an 8-bit storage register with two internal ranks of D-type edge-triggered flip-flops. Data is loaded into rank 1 on the rising edge of the clock (`CK`) when both clock enables (`CKE1` and `CKE2`) are active. The input select signal (`INSEL`) determines whether rank 2 receives data from rank 1 or directly from the inputs (`D7--D0`). The output select signal (`OUTSEL`) chooses whether the outputs are driven by rank 1 or rank 2. When output enable (`OE`) is active, the corresponding 3-state outputs are enabled.

4.7.2 Truth Table

CK	CKE1	CKE2	RANK 1	RANK 2
L or H or ↓	X	X	Disabled	Disabled
↑	L	L	Enabled	Enabled
↑	L	H	Enabled	Disabled
↑	H	L	Disabled	Enabled
X	H	H	Disabled	Disabled

Table 4.1: Truth Table for 74LS548 Register Operation

4.7.3 Pin Diagram

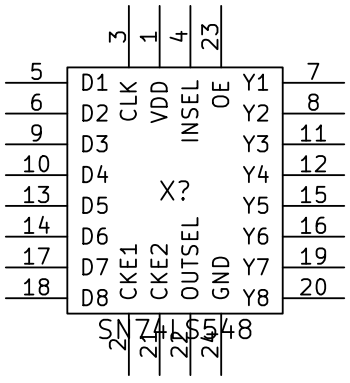

Figure 4.27: Pin Diagram of SN74LS548

4.7.4 Subcircuit

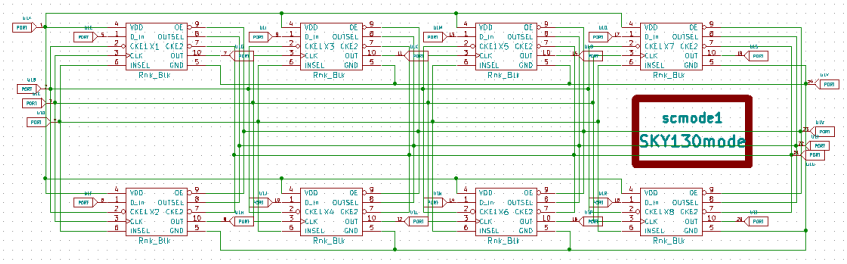

Figure 4.28: Subcircuit for SN74LS548

4.7.5 Test Circuit

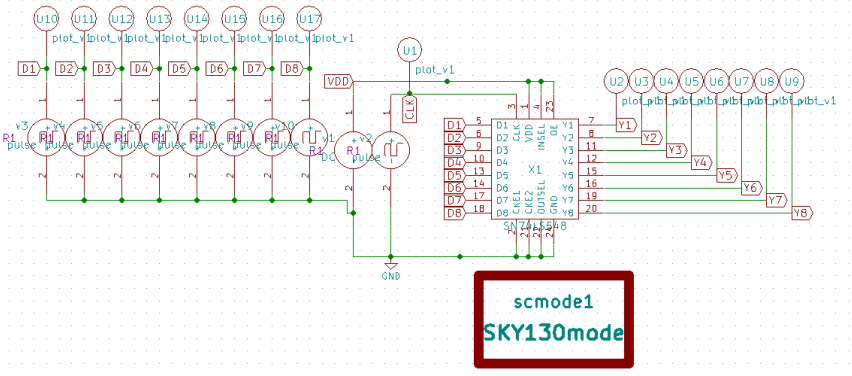

Figure 4.29: Testbench for SN74LS548 Subcircuit

4.7.6 Waveform

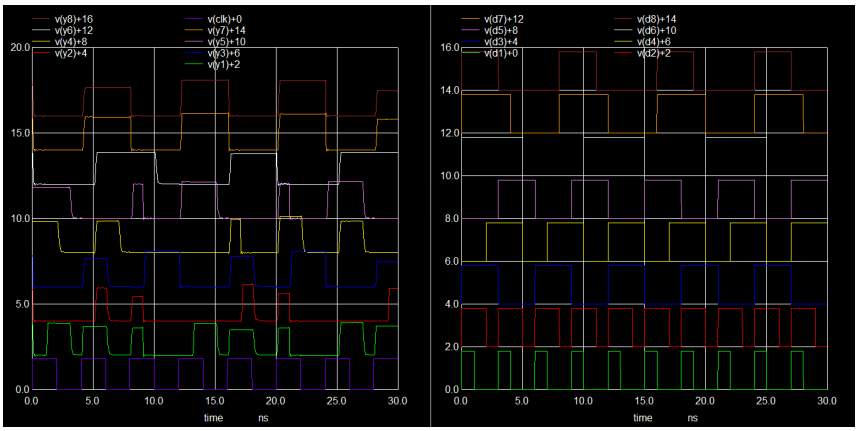

Figure 4.30: Input Output Waveform for SN74LS548 Subcircuit

4.8 SN74ABT337: CLOCK DRIVER WITH 3-STATE OUTPUTS

4.8.1 Description:

The SN74ABT337 is a high-performance clock driver IC designed to distribute synchronized clock signals with minimal timing error. It uniquely provides outputs at both the original input frequency and at half the frequency. Its key feature is the 3-state outputs, which allow the chip to be electrically disconnected from a shared bus, preventing signal conflicts in complex digital systems.

4.8.2 Pin Diagram

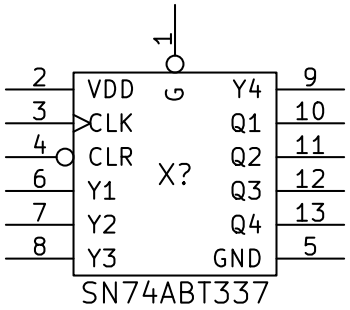

Figure 4.31: Pin Diagram of SN74ABT337

4.8.3 Subcircuit

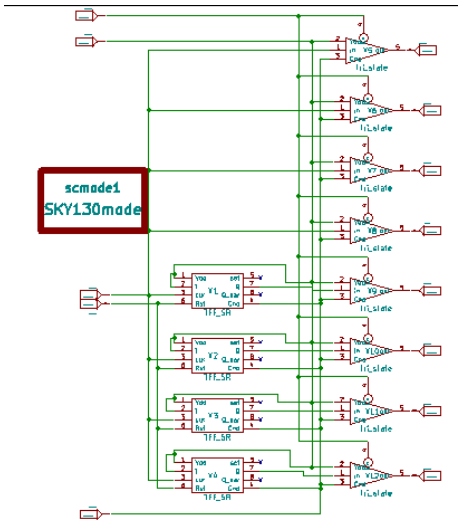

Figure 4.32: Subcircuit for SN74ABT337

4.8.4 Test Circuit

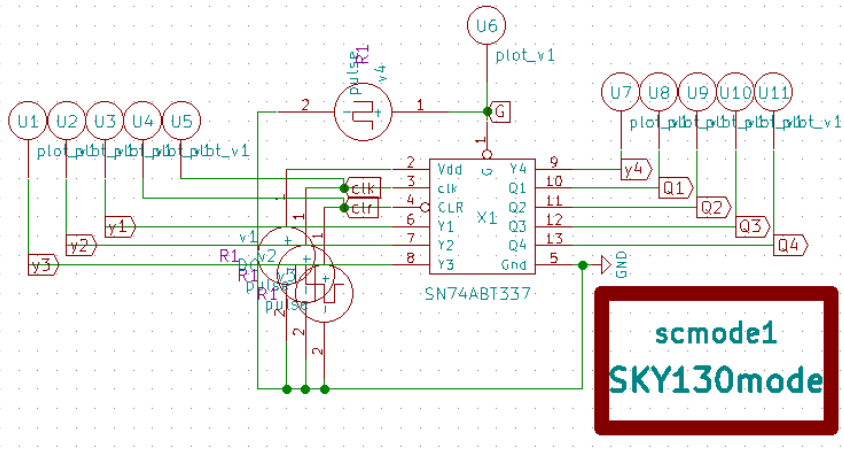

Figure 4.33: Testbench for SN74ABT337 Subcircuit

4.8.5 Waveform

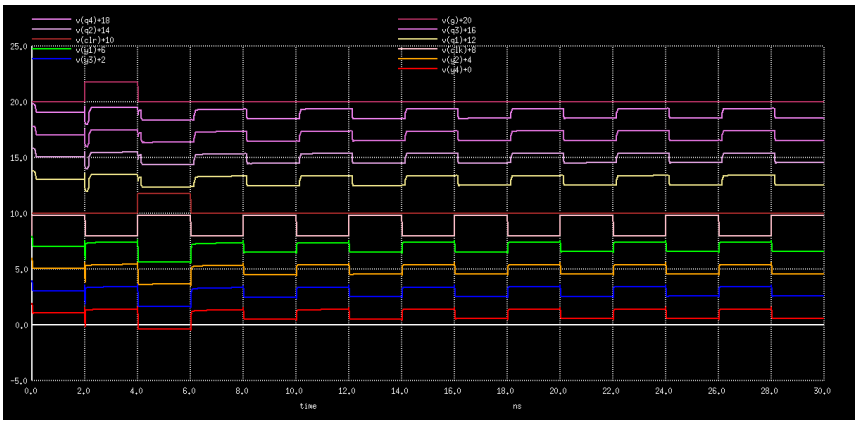

Figure 4.34: Input Output Waveform for SN74ABT337 Subcircuit

4.9 SN54L99: 4-BIT RIGHT-SHIFT LEFT-SHIFT REGISTERS

4.9.1 Description:

These 4-bit registers support parallel load, shift right, and shift left operations. Data can be loaded in parallel using clock 2 when mode control is high. Shift right occurs on clock 1 with mode low and uses J-K serial inputs. Shift left occurs on clock 2 with mode high, using the D input. The device features two clocks and mode control to select between operations. Serial input modes allow flexible flip-flop configurations (J-K, D, or T).

4.9.2 Pin Diagram

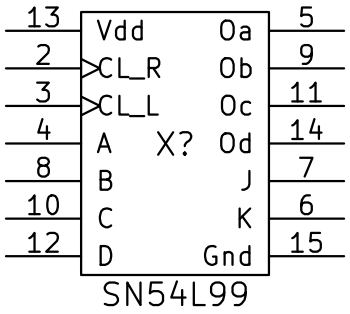

Figure 4.35: Pin Diagram of SN54L99

4.9.3 Subcircuit

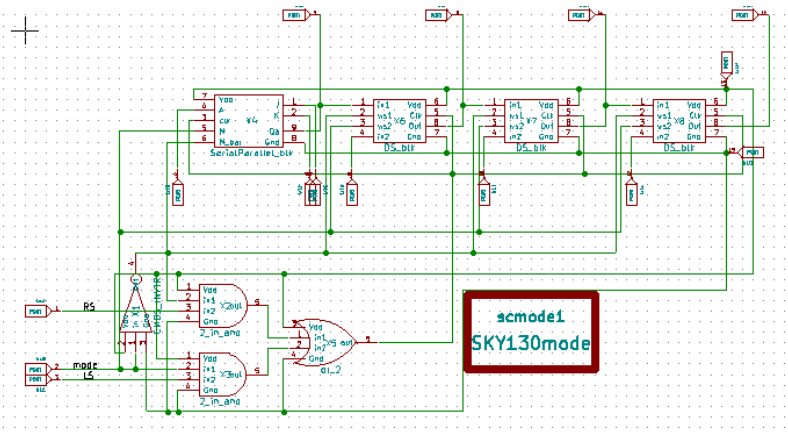

Figure 4.36: Subcircuit for SN54L99

4.9.4 Test Circuit

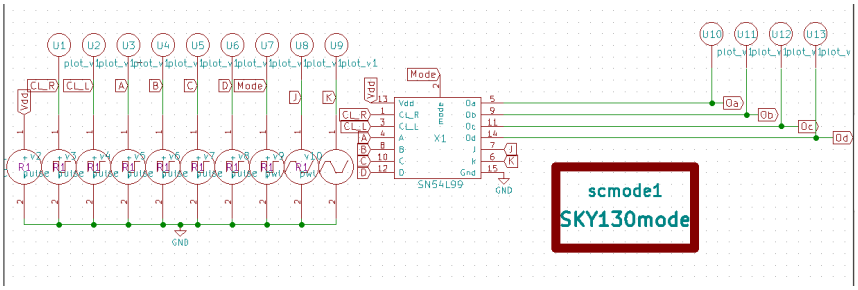

Figure 4.37: Testbench for SN54L99 Subcircuit

4.9.5 Waveform

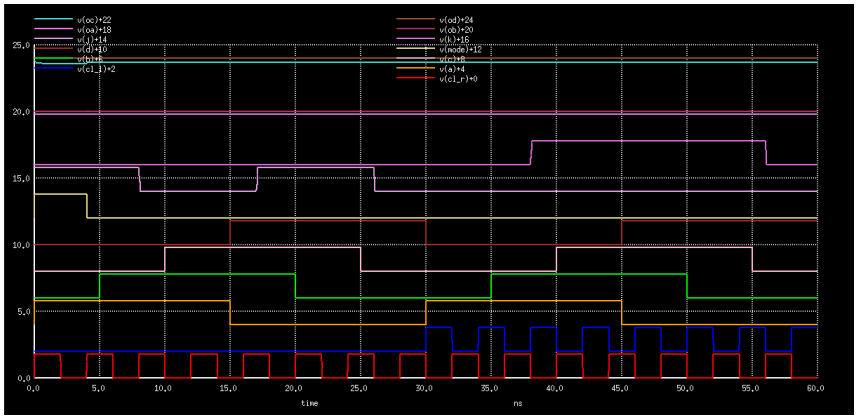

Figure 4.38: Input Output Waveform for SN54L99 Subcircuit

4.10 SN54L98: 4-BIT DATA SELECTORS/STORAGE REGISTERS

4.10.1 Description:

These monolithic data selectors/storage registers are composed of four S-R master-slave flip-flops, four AND-OR-INVERT gates, one buffer, and six inverter/ drivers. When the word select input is low, word 1 (A1, B1, CI, D1) is applied to the flip-flops. A high input to word select will cause the selection of word 2 (A2, B2, C2, D2). The selected word is shifted to the output terminals on the negative-going edge of the clock pulse.

4.10.2 Pin Diagram

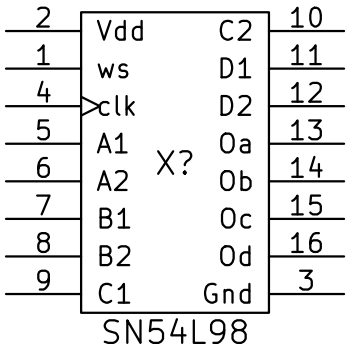

Figure 4.39: Pin Diagram of SN54L98

4.10.3 Subcircuit

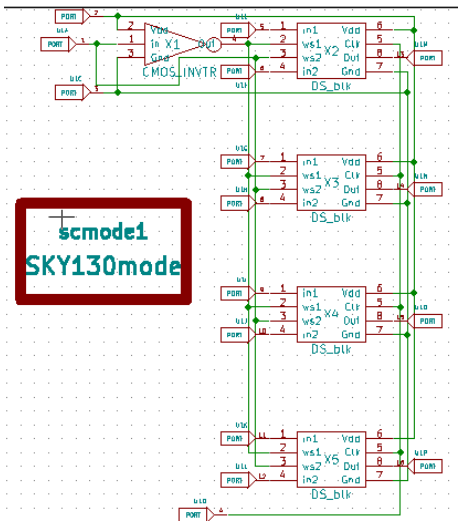

Figure 4.40: Subcircuit for SN54L98

4.10.4 Test Circuit

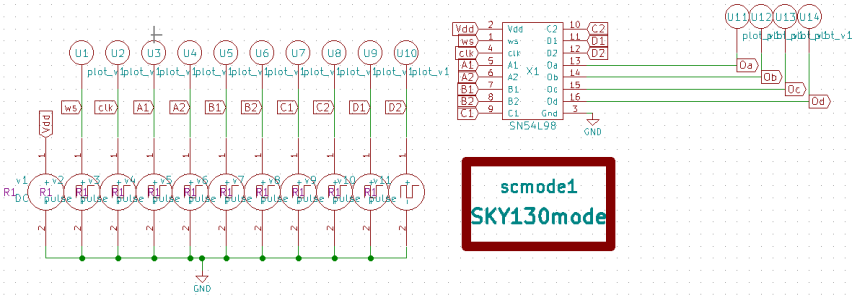

Figure 4.41: Testbench for SN54L98 Subcircuit

4.10.5 Waveform

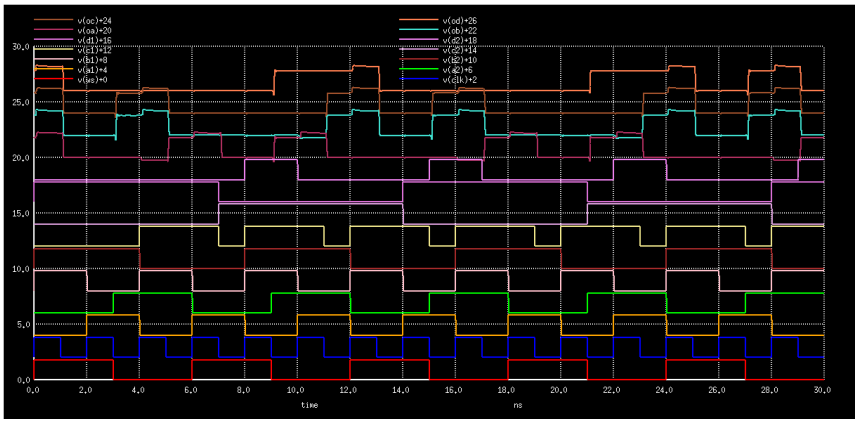

Figure 4.42: Input Output Waveform for SN54L98 Subcircuit

Chapter 5

Mixed Signal Simulation

5.1 74HC181: 4-bit arithmetic logic unit

5.1.1 Description:

The 74HC181 is a 4-bit high-speed parallel Arithmetic Logic Units (ALU). Controlled by the four function select inputs (S0 to S3) and the mode control input (M), they can perform all the 16 possible logic operations or 16 different arithmetic operations on active HIGH or active LOW operands

5.1.2 Verilog Code:

```
// 74HC181: 4-bit Arithmetic Logic Unit (ALU)
module alu_74hc181 (
    input  wire [3:0] a,     // 4-bit Input A
    input  wire [3:0] b,     // 4-bit Input B
    input  wire       cn,    // Carry In
    input  wire       m,     // Mode Select (0=Arithmetic, 1=
       Logic)
    input  wire [3:0] s,     // 4-bit Function Select
    output reg  [3:0] f,     // 4-bit Function Output
    output reg        cn4    // Carry Out
);

    reg [4:0] temp_result;

    always @(*) begin
        // Use a case statement based on the mode and select
           lines {m, s}
        case ({m, s})
            // LOGIC OPERATIONS (M=1)
            5'b1_0000: temp_result = ~a;
            5'b1_0001: temp_result = ~(a | b);
            5'b1_0010: temp_result = ~a & b;
            5'b1_0011: temp_result = 5'b0;
```

```
22              5'b1_0100: temp_result = ~(a & b);
23              5'b1_0101: temp_result = ~b;
24              5'b1_0110: temp_result = a ^ b;
25              5'b1_0111: temp_result = a & ~b;
26              5'b1_1000: temp_result = ~a | b;
27              5'b1_1001: temp_result = ~(a ^ b);
28              5'b1_1010: temp_result = b;
29              5'b1_1011: temp_result = a & b;
30              5'b1_1100: temp_result = 5'b11111; // Logic '1'
31              5'b1_1101: temp_result = a | ~b;
32              5'b1_1110: temp_result = a | b;
33              5'b1_1111: temp_result = a;
34
35              // ARITHMETIC OPERATIONS (M=0)
36              5'b0_0000: temp_result = a - 1 + {4'b0, cn};
37              5'b0_0001: temp_result = (a | b) - 1 + {4'b0, cn
                    };
38              5'b0_0010: temp_result = (a | ~b) - 1 + {4'b0, cn
                    };
39              5'b0_0011: temp_result = {4'b0, cn} - 1;
40              5'b0_0100: temp_result = a + (a & ~b) + {4'b0, cn
                    };
41              5'b0_0101: temp_result = (a | b) + (a & ~b) + {4'
                    b0, cn};
42              5'b0_0110: temp_result = a - b - 1 + {4'b0, cn};
43              5'b0_0111: temp_result = (a & ~b) - 1 + {4'b0, cn
                    };
44              5'b0_1000: temp_result = a + (a | b) + {4'b0, cn
                    };
45              5'b0_1001: temp_result = a + b + {4'b0, cn};
46              5'b0_1010: temp_result = (a | ~b) + (a | b) + {4'
                    b0, cn};
47              5'b0_1011: temp_result = (a | b) - 1 + {4'b0, cn
                    };
48              5'b0_1100: temp_result = a + a + {4'b0, cn};
49              5'b0_1101: temp_result = (a | b) + a + {4'b0, cn
                    };
50              5'b0_1110: temp_result = (a | ~b) + a + {4'b0, cn
                    };
51              5'b0_1111: temp_result = a - 1 + {4'b0, cn};
52
53              default: temp_result = 5'bxxxxx;
54          endcase
55
56          f = temp_result[3:0];
57          cn4 = (m) ? cn : temp_result[4];
58      end
59
60  endmodule
```

5.1.3 Pin Diagram

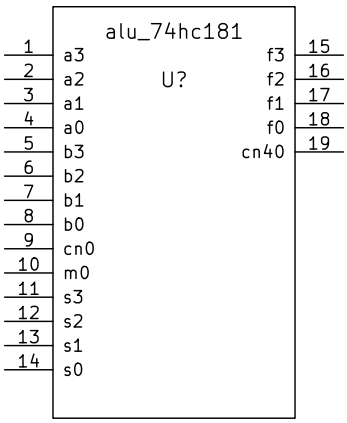

Figure 5.1: Pin Diagram of 74HC181

5.1.4 Test Circuit

Test input: a = 4'h6, b = 4'h2

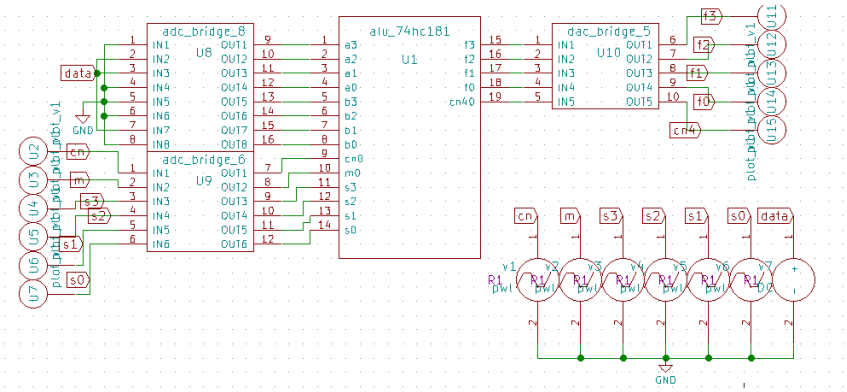

Figure 5.2: Mixed Signal Model of 74HC181

5.1.5 Makerchip Simulation

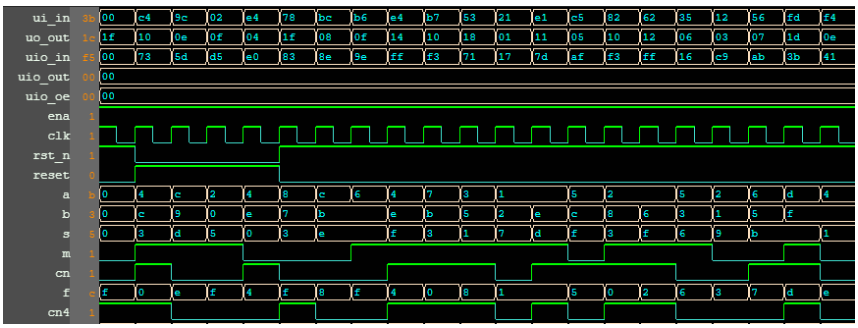

Figure 5.3: Makerchip Simulation for 74HC181

5.1.6 NgSpice Waveform

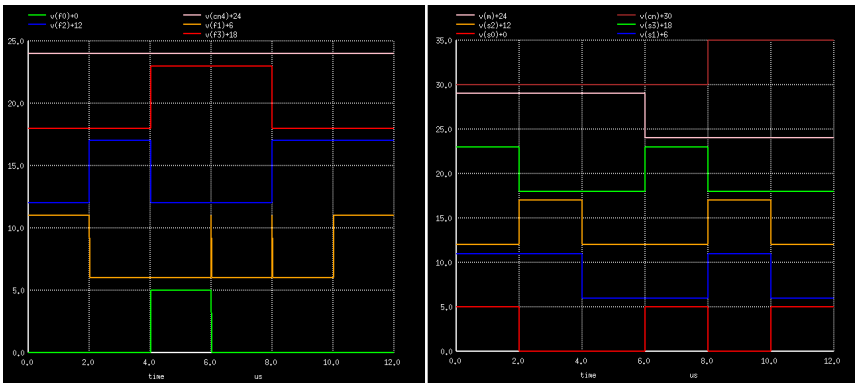

Figure 5.4: Input Output Waveform for 74HC181

5.2 74HC153: Dual 4-input multiplexer

5.2.1 Description:

The 74HC153 is a dual 4-input multiplexer. The device features independent enable inputs (nE) and common data select inputs (S0 and S1). For each multiplexer, the select inputs select one of the four binary inputs and routes it to the multiplexer output (nY). A HIGH on E forces the corresponding multiplexer outputs LOW.

5.2.2 Verilog Code:

```
// 74HC153: Dual 4-to-1 Multiplexer
module mux_74hc153 (
    input  wire [1:0] s,      // Select lines S0, S1 (shared
       by both muxes)
    input  wire       e1_n,   // Enable for MUX 1 (active low
       )
    input  wire       e2_n,   // Enable for MUX 2 (active low
       )
    input  wire [3:0] d1,     // Data inputs for MUX 1
    input  wire [3:0] d2,     // Data inputs for MUX 2
    output wire       y1,     // Output of MUX 1
    output wire       y2      // Output of MUX 2
);

    // MUX 1: Output is low when enable is high, otherwise
       selects input
    assign y1 = (!e1_n) ? d1[s] : 1'b0;

    // MUX 2: Output is low when enable is high, otherwise
       selects input
    assign y2 = (!e2_n) ? d2[s] : 1'b0;

endmodule
```

5.2.3 Pin Diagram

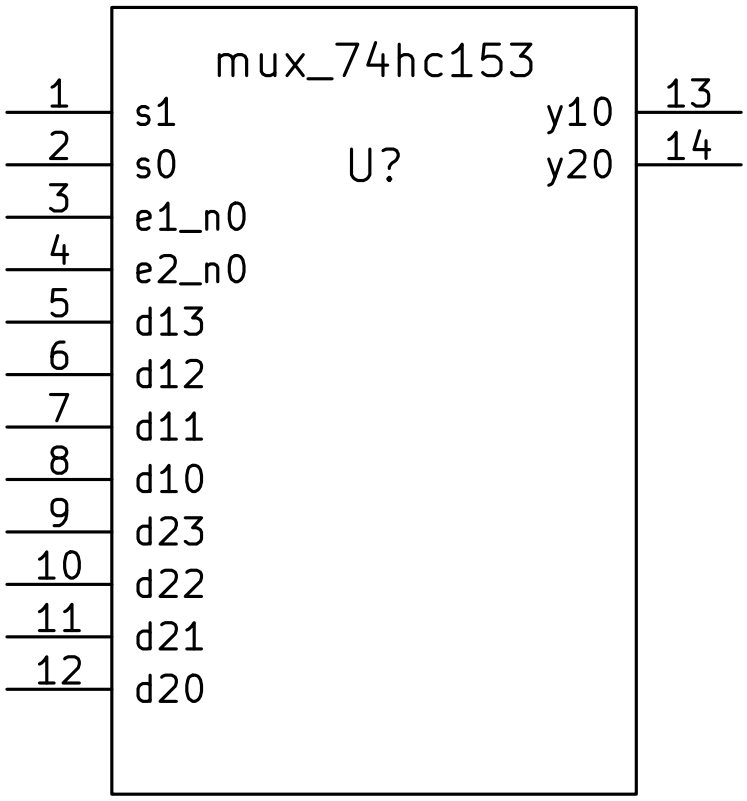

Figure 5.5: Pin Diagram of 74HC153

5.2.4 Test Circuit

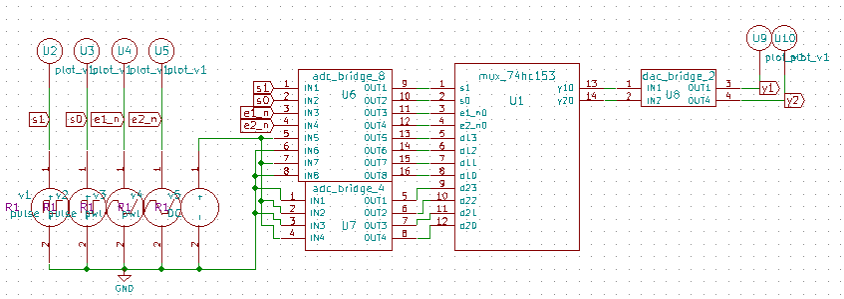

Figure 5.6: Mixed Signal Model of 74HC153

5.2.5 Makerchip Simulation

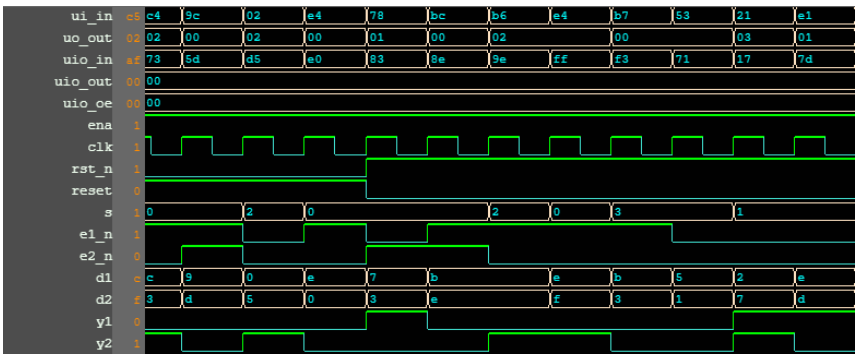

Figure 5.7: Makerchip Simulation for 74HC153

5.2.6 NgSpice Waveform

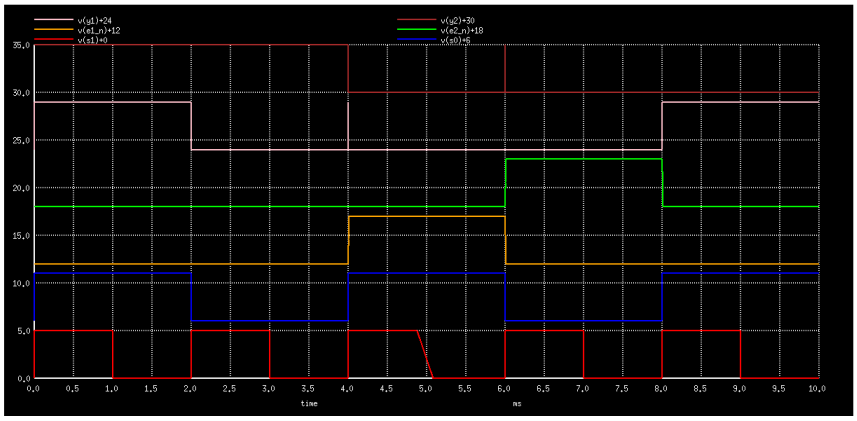

Figure 5.8: Input Output Waveform for 74HC153

5.3 74HC148: 8-Line to 3-Line Priority Encoder

5.3.1 Description:

The 74HC148 is an 8-input priority encoder. Added input enable (EI) and output enable (EO) signals allow for cascading multiple stages without added external circuitry.

```
// 74HC148: 8-to-3 Priority Encoder
module priority_encoder_74hc148 (
    input  wire [7:0] i_n,   // 8 active-low inputs
    output reg  [2:0] a_n,   // 3-bit active-low address output
    output wire       gs_n,  // Group Signal (indicates any input is active)
    output wire       eo_n   // Enable Output (indicates no inputs are active)
);

    // Combinational block for priority logic
    always @(*) begin
        // Default output if no inputs are active
        a_n = 3'b111; // Inactive state
        if      (!i_n[7]) a_n = 3'b000; // Corresponds to 7
        else if (!i_n[6]) a_n = 3'b001; // Corresponds to 6
        else if (!i_n[5]) a_n = 3'b010; // Corresponds to 5
        else if (!i_n[4]) a_n = 3'b011; // Corresponds to 4
        else if (!i_n[3]) a_n = 3'b100; // Corresponds to 3
        else if (!i_n[2]) a_n = 3'b101; // Corresponds to 2
        else if (!i_n[1]) a_n = 3'b110; // Corresponds to 1
        else if (!i_n[0]) a_n = 3'b111; // Corresponds to 0
    end

    // Group Select (GS) is active (low) if any input is active (low).
    // This is the NOR of the inputs.
    assign gs_n = |i_n;

    // Enable Out (EO) is active (low) if no inputs are active.
    // This is the AND of the inputs.
    assign eo_n = &i_n;

endmodule
```

5.3.2 Pin Diagram

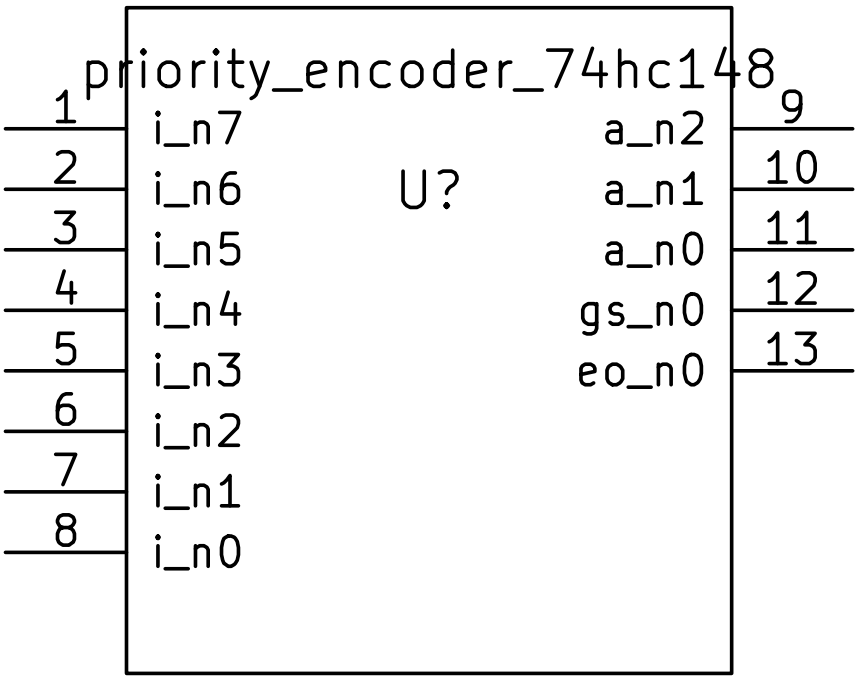

Figure 5.9: Pin Diagram of 74HC148

5.3.3 Test Circuit

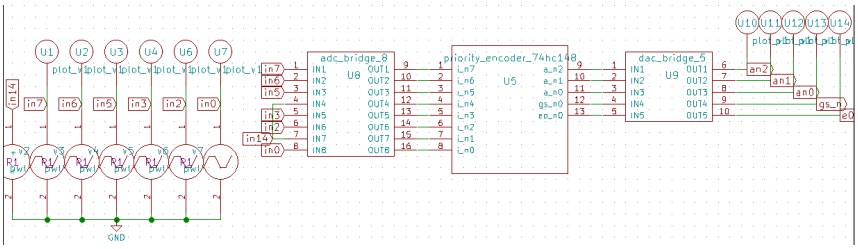

Figure 5.10: Mixed Signal Model of 74HC148

5.3.4 Makerchip Simulation

Figure 5.11: Makerchip Simulation for 74HC148

5.3.5 NgSpice Waveform

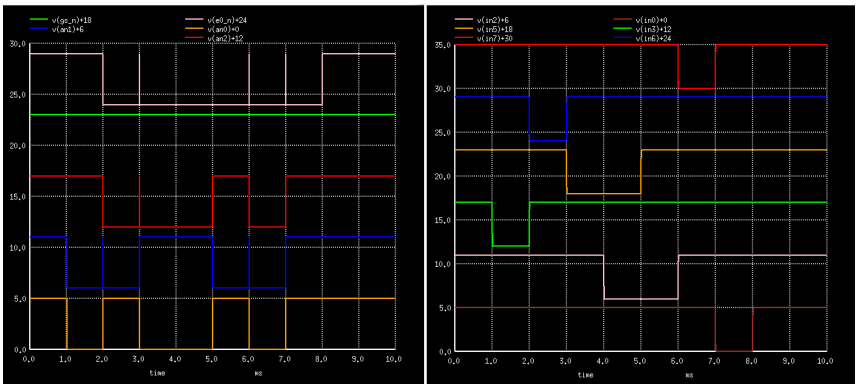

Figure 5.12: Input Output Waveform for 74HC148

5.4 74LS47: BCD-TO-SEVEN-SEGMENT DECODERS

5.4.1 Description:

The 74LS47 feature active-low outputs designed for driving common-anode LEDs or incandescent indicators directly. The ‘48, ‘LS48, and ‘LS49 feature active-high outputs for driving lamp buffers or common-cathode LEDs. All of the circuits except ‘LS49 have full ripple-blanking input/output controls and a lamp test input. The ‘LS49 circuit incorporates a direct blanking input. Segment identification and resultant displays are shown below. Display patterns for BCD input counts above 9 are unique symbols to authenticate input conditions.

```
// 74LS47: BCD to 7-Segment Decoder
module decoder_74ls47 (
    input  wire [3:0] bcd, // BCD input [D, C, B, A]
    input  wire       l_t,  // Lamp Test (active low)
    input  wire       bi,  // Blanking Input (active low)
    output reg  [6:0] seg  // 7-segment output [g,f,e,d,c,b,a
       ] (active low)
);

    // Combinational logic block
    always @(*) begin
        // Lamp Test has the highest priority: turns all
           segments ON (low)
        if (!l_t) begin
            seg = 7'b000_0000;
        // Blanking Input is next: turns all segments OFF (
           high)
        end else if (!bi) begin
            seg = 7'b111_1111;
        // Use a case statement to decode the BCD input
        end else begin
            case (bcd)
                //      gfedcba
                4'h0: seg = 7'b100_0000; // 0
                4'h1: seg = 7'b111_1001; // 1
                4'h2: seg = 7'b010_0100; // 2
                4'h3: seg = 7'b011_0000; // 3
                4'h4: seg = 7'b001_1001; // 4
                4'h5: seg = 7'b001_0010; // 5
                4'h6: seg = 7'b000_0010; // 6
                4'h7: seg = 7'b111_1000; // 7
                4'h8: seg = 7'b000_0000; // 8
                4'h9: seg = 7'b001_0000; // 9
                default: seg = 7'b111_1111; // Blank for
                   invalid BCD inputs > 9
```

```
32                endcase
33            end
34        end
35
36    endmodule
```

5.4.2 Pin Diagram

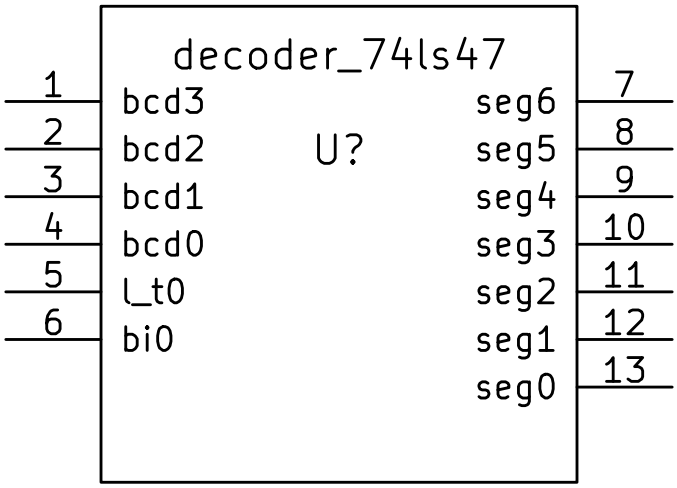

Figure 5.13: Pin Diagram of 74LS47

5.4.3 Test Circuit

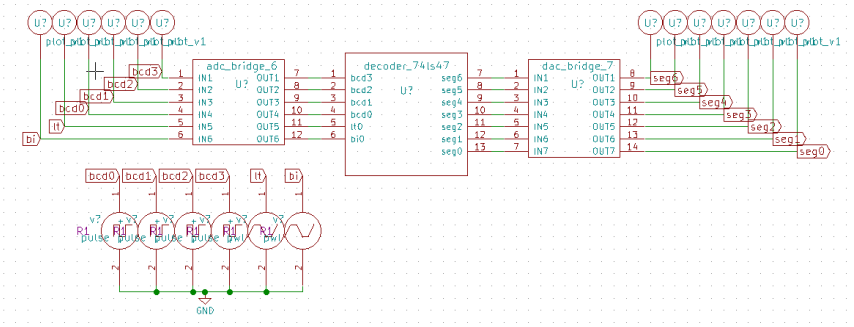

Figure 5.14: Mixed Signal Model of 74LS47

5.4.4 Makerchip Simulation

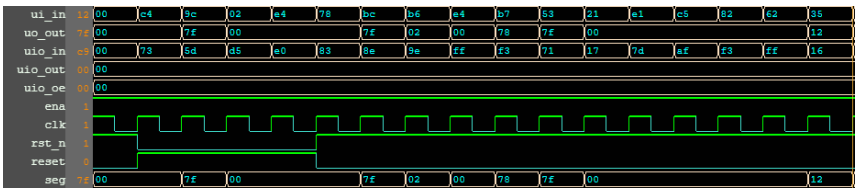

Figure 5.15: Makerchip Simulation for 74LS47

5.4.5 NgSpice Waveform

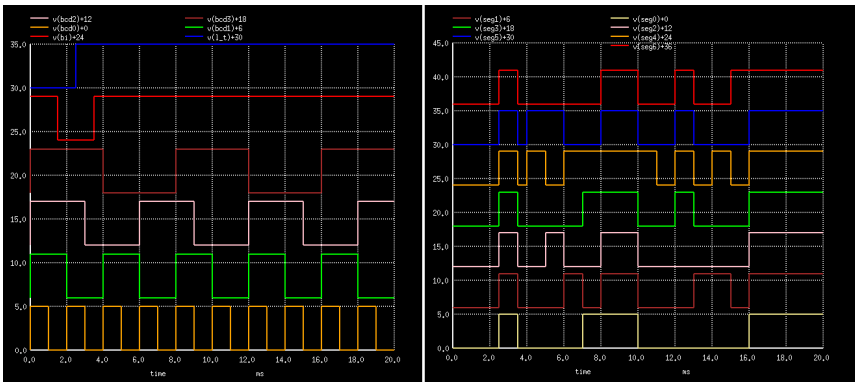

Figure 5.16: Input Output Waveform for 74LS47

5.5 74HC161: Presettable synchronous 4-bit binary counter; asynchronous reset

5.5.1 Description:

The 74HC161 is a synchronous presettable binary counter with an internal look-head carry. Synchronous operation is provided by having all flip-flops clocked simultaneously on the positive going edge of the clock (CP). The outputs (Q0 to Q3) of the counters may be preset HIGH or LOW. A LOW at the parallel enable input (PE) disables the counting action and causes the data at the data inputs (D0 to D3) to be loaded into the counter on the positive-going edge of the clock. Preset takes place regardless of the levels at count enable inputs (CEP and CET).

```
// 74HC161: 4-bit Synchronous Binary Counter with Asynchronous Reset
module counter_74hc161 (
    input  wire       clk,   // Clock input
    input  wire       mr,    // Master Reset (active low, asynchronous)
    input  wire       pe,    // Parallel Enable (active low, synchronous)
    input  wire       cep,   // Count Enable P
    input  wire       cet,   // Count Enable T
    input  wire [3:0] d,     // Parallel Data input
    output reg  [3:0] q,     // Counter output
    output wire       tc     // Terminal Count output
);

    // The counter logic is described in a single clocked block.
    always @(posedge clk or negedge mr) begin
        // Asynchronous reset has the highest priority
        if (!mr) begin
            q <= 4'b0000;
        end
        // All other operations are synchronous to the clock edge
        else begin
            // Synchronous parallel load
            if (!pe) begin
                q <= d;
            // Count operation is enabled by PE, CEP, and CET
            end else if (cep && cet) begin
                q <= q + 1;
            end
            // If not reset, loaded, or enabled, the counter holds its value.
        end
```

```
30      end
31
32      // Terminal Count (TC) is high when the count is 15 AND
            CET is high.
33      assign tc = (q == 4'b1111) && cet;
34
35  endmodule
```

5.5.2 Pin Diagram

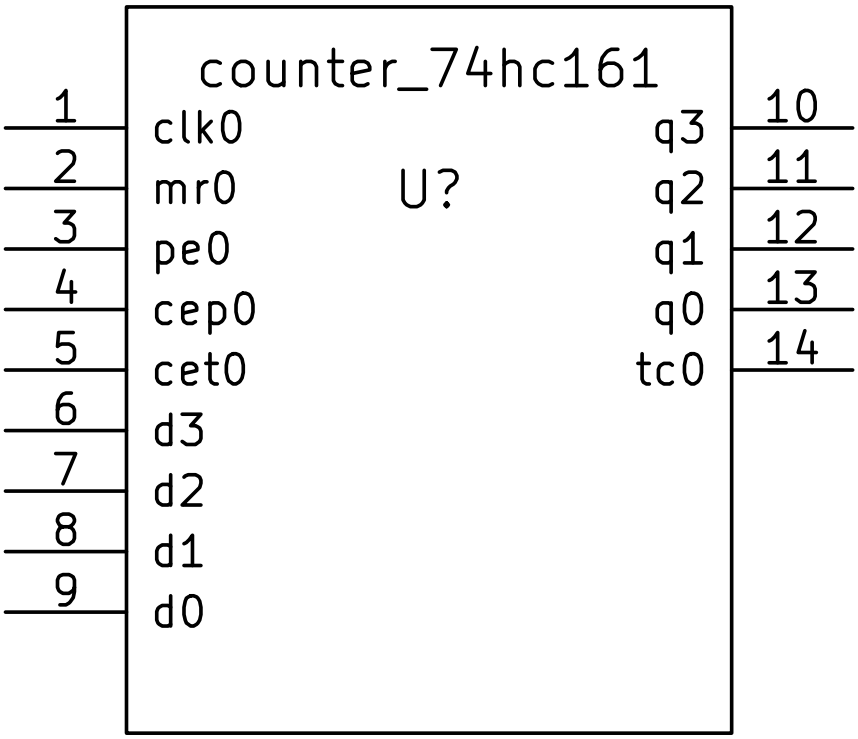

Figure 5.17: Pin Diagram of 74HC161

5.5.3 Test Circuit

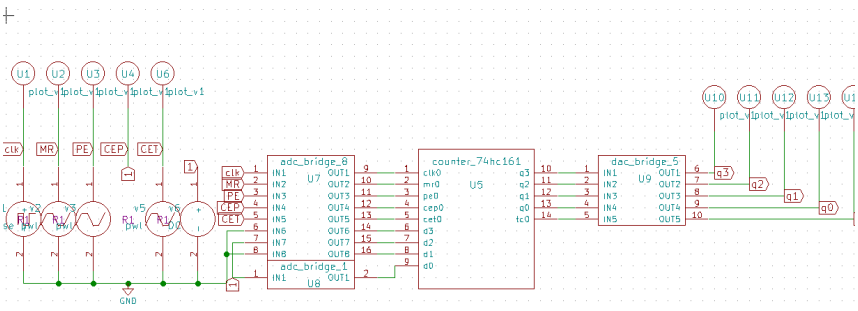

Figure 5.18: Mixed Signal Model of 74HC161

5.5.4 Makerchip Simulation

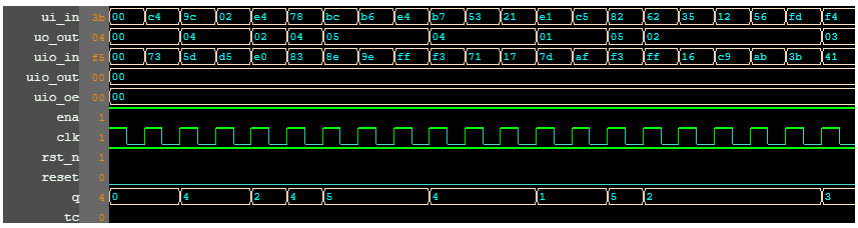

Figure 5.19: Makerchip Simulation for 74HC161

5.5.5 NgSpice Waveform

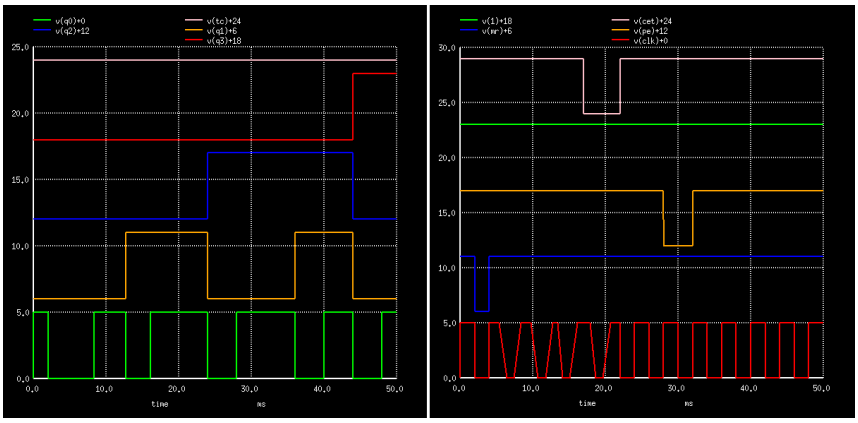

Figure 5.20: Input Output Waveform for 74HC161

Chapter 6

Conclusion and Future Scope

During this fellowship, I explored and modelled a variety of digital IC subcircuits, closely adhering to their datasheet specifications. Each design was thoroughly tested using eSim, with validation carried out through realistic simulation setups to ensure reliability and correctness. These verified modules are now prepared for integration into the official esim subckt library, where they can serve as reusable, drop-in components for academic projects, electronic design training, and open-source circuit development. This contribution helps broaden the reach of eSim as a practical tool for learning and prototyping in digital and mixed-signal domains.

Bibliography

[1] SN74HCS259 (TI)
URL: https://www.alldatasheet.com/datasheet-pdf/view/1360429/TI/SN74HCS259.html

[2] CD4044 (InterSil)
URL: https://www.alldatasheet.com/datasheet-pdf/view/66431/INTERSIL/CD4044.html

[3] CD4042B (TI)
URL: https://www.alldatasheet.com/datasheet-pdf/view/114733/TI/CD4042B.html

[4] CD4068B (TI)
URL: https://www.alldatasheet.com/datasheet-pdf/pdf/26883/TI/CD4068B.html

[5] TC74HC4028AP (Toshiba)
URL: https://www.alldatasheet.com/datasheet-pdf/view/31745/TOSHIBA/TC74HC4028AP.html

[6] SN54ALS1011A (TI)
URL: https://archive.org/details/bitsavers_tidataBookVol3_25840031/page/n715/mode/1up?view=theater

[7] SN74LS548 (TI)
URL: https://archive.org/details/bitsavers_mmidataBook6ed_79579213/page/n638/mode/1up?view=theater

[8] SN74ABT337 (TI)
URL: https://archive.org/details/TexasInstruments-TI-Data-AdvancedLogicandBusInt
page/n486/mode/1up?view=theater

[9] SN54L99 (TI)
URL: https://archive.org/details/bitsavers_tidataBookVol2_45945352/page/n452/mode/1up?view=theater

[10] SN54L98 (TI)
URL: https://archive.org/details/bitsavers_tidataBookVol2_45945352/page/n450/mode/1up?view=theater

[11] SN74HC148 (TI)
URL: https://www.ti.com/lit/ds/scls109h/scls109h.pdf?ts=1753875988661&ref_url=https%253A%252F%252Fwww.google.com%252F

[12] 74HC153 (NXP)
URL: https://www.mouser.com/datasheet/2/302/74HC_HCT153-353677.pdf

[13] 74HC181 (PHILIPS)
URL: https://mil.ufl.edu/3701/docs/74HC181_philips.pdf

[14] SN74LS47 (TI)
URL: https://www.ti.com/lit/ds/symlink/sn54ls47.pdf

[15] 74HC161 (NEXPERIA)
URL: https://assets.nexperia.com/documents/data-sheet/74HC161.pdf