Summer Fellowship Report

On

SubCircuit IC Design

Submitted by

Sai Akhil Cheruvu
B.E (Electronics and Communication)
Chaitanya Bharathi Institute of Technology, Hyderabad

Under the guidance of

Prof. Prabhu Ramachandran
Principal Investigator
Department of Aerospace Engineering
Indian Institute of Technology Bombay

July 7, 2026

Acknowledgment

I express my sincere gratitude to Prof. Prabhu Ramachandran for providing me with the opportunity to be part of the FOSSEE internship program and for his continued efforts in promoting open-source engineering tool development. His leadership and vision have been instrumental in fostering meaningful student participation in the open-source ecosystem.

I also acknowledge Prof. Kannan M. Moudgalya for his foundational role in establishing and strengthening the FOSSEE initiative. His contributions to open-source education and the development of the FOSSEE fellowship framework have been pivotal in creating the academic and organisational platform through which this internship was undertaken.

My sincere appreciation extends to my mentor, Sumanto Kar, for his continual support, technical guidance, and encouragement throughout the duration of this project. His insights and feedback played a key role in refining ideas, overcoming challenges, and ensuring timely completion of the tasks assigned to me.

I would also like to thank my internal mentors, Mr. Varad Patil and Ms. Shanthi Priya K for their valuable guidance, coordination, and technical inputs during the internship. Their mentorship contributed significantly to the clarity, progress, and successful execution of the work.

This internship has been an enriching learning experience, allowing me to work closely with open-source EDA tools, develop IC subcircuits in eSim, and gain exposure to realworld circuit modeling and simulation workflows. The knowledge acquired during this period will undoubtedly support my future academic and professional pursuits.

I would also like to thank the entire FOSSEE team for their coordination, assistance, and timely interactions at various stages of this work. Their collective efforts ensured smooth workflow, resource accessibility, and effective project execution.

Contents

Chapter 1

Introduction

1.1 Background

FOSSEE, which stands for **Free/Libre and Open Source Software for Education**, is a project initiated and funded by the Ministry of Education (formerly the Ministry of Human Resource Development), Government of India, under the National Mission on Education through Information and Communication Technology (NMEICT). The project is hosted and executed at the Indian Institute of Technology Bombay (IIT Bombay), and functions under the guidance of experienced faculty along with a dedicated team of developers, researchers, and student contributors.

The core philosophy behind FOSSEE stems from a simple but powerful observation: a large number of engineering colleges, polytechnics, and universities in India, and in many other developing countries, rely heavily on expensive, proprietary licensed software for teaching core engineering subjects. Tools such as MATLAB, OrCAD, Multisim, and ANSYS come with substantial licensing fees that many institutions, especially smaller colleges in semi-urban and rural areas, simply cannot afford. This creates a digital and educational divide, where students at well-funded institutions get hands-on exposure to industry-standard simulation and design tools, while students elsewhere are left with only theoretical knowledge, widening the gap between classroom learning and industry readiness.

This divide is not merely a matter of convenience. Practical, tool-based learning is essential in engineering disciplines such as electronics, where the ability to simulate, debug, and validate a circuit before physical implementation saves both time and resources. Without access to such tools, students often graduate having only seen circuit theory on paper, without ever having tested their understanding against real simulated behavior. FOSSEE's founders recognized this early on, and positioned open-source software not as a compromise, but as a genuinely capable, and in many cases equally rigorous, alternative to commercial tools.

FOSSEE was created to bridge this exact gap. Its mission is to:

1. Promote the use of FLOSS (Free/Libre and Open Source Software) in academic institutions across India, so that no student is denied practical, hands-on learning due to financial constraints.

2. Reduce the cost burden on educational institutions by providing free alter-

natives to costly proprietary tools, without compromising on functionality or learning outcomes.

3. Build a community-driven ecosystem of educators, students, and developers who continuously contribute to, improve, and support open-source tools tailored for engineering and science education.

4. Encourage indigenous software development, aligning with the broader national vision of self-reliance in the technology and education sectors.

5. Provide structured pathways, such as fellowships and internships, through which students can directly contribute to strengthening these open-source tools, gaining valuable hands-on experience in the process.

To achieve these goals, FOSSEE runs a wide array of initiatives, including the Textbook Companion Project, Spoken Tutorials, Lab Migration Projects, Online Workshops, Semester-Long and Summer Internships, and Teacher Fellowships. Each of these initiatives is designed to lower a specific barrier to adoption — Spoken Tutorials address the learning curve, Lab Migration Projects help institutions transition existing lab courses to open-source tools, and Fellowships like the one this report is based on allow students to actively expand the capabilities of these tools rather than simply using them.

Under its umbrella, FOSSEE promotes and develops several key open-source tools, each targeting a specific domain of engineering education, as summarized in Table 1.1.

Table 1.1: Key Open Source Tools Promoted by FOSSEE

Tool	**Domain**	**Proprietary Alternative Replaced**
Scilab	Numerical computing, control systems	MATLAB
OpenFOAM	Computational Fluid Dynamics (CFD)	ANSYS Fluent
Python	General-purpose scientific computing	MATLAB
eSim	Circuit design, simulation, PCB design	OrCAD, Multisim, PSpice
OpenModelica	Modeling of complex physical systems	Dymola

Among all these tools, eSim is of particular relevance to this report, as it forms the core software platform on which the entire project has been carried out. Its role in this fellowship is discussed in the following section.

1.2 Overview of eSim

eSim is a free and open-source Electronic Design Automation (EDA) tool developed under the FOSSEE project at IIT Bombay. It was created to provide students and professionals with a completely open-source, end-to-end circuit design environment, covering everything from schematic drawing, to simulation, to PCB layout, without requiring any proprietary software. In doing so, eSim addresses a gap that many individual open-source tools leave unfilled — while some tools specialize purely in simulation and others purely in PCB layout, eSim brings these stages together into a single continuous workflow.

eSim achieves this by integrating several mature, independently developed open-source projects under one unified graphical interface, most notably **KiCad**, used for schematic capture and PCB layout design, and **Ngspice**, used as the underlying circuit simulation engine. eSim acts as a bridging and orchestration layer between these tools — allowing a user to draw a circuit schematic, automatically convert it into a SPICE-compatible netlist, simulate it, and view the results, all from within a single interface. This tight integration significantly reduces the friction that typically comes with switching between separate schematic, simulation, and layout tools, and is one of the primary reasons eSim was chosen as the platform for this fellowship. A more detailed discussion of eSim's individual features, including its schematic, simulation, PCB design, and Subcircuit capabilities, is presented separately in Chapter 2.

One capability worth introducing briefly here, since it forms the technical foundation of this entire project, is eSim's **Subcircuit** feature. A subcircuit allows a complex or frequently used circuit block to be encapsulated as a single, reusable component with its own defined ports, similar to a function in programming. This project relied extensively on this feature, since many ICs used, such as the SN74xx logic family and the TLC274 op-amp, are not available by default in eSim's component library. Rather than treating this as a limitation, the fellowship approached it as an opportunity — each missing IC became a chance to model real device behavior from first principles and contribute a reusable component back to the eSim ecosystem. The detailed subcircuit design workflow followed throughout this project is described in the Methodology section below.

1.3 Objectives of the Project

The primary objectives of this project were as follows:

1. To gain a thorough understanding of the eSim tool, including its schematic capture, simulation, and subcircuit creation workflows.

2. To design and simulate a range of integrated circuits (ICs), including digital logic ICs from the SN74xx family and analog ICs such as the TLC274, using the subcircuit feature of eSim.

3. To validate the simulated behavior of each IC against its official datasheet specifications, ensuring functional accuracy.

4. To document the internal transistor-level or gate-level architecture and working principle of each IC studied.

5. To identify and overcome practical challenges encountered while modeling and simulating ICs that are not natively available in eSim's component library.

6. To build a personal understanding of how commercial ICs are internally structured, beyond their black-box datasheet behavior, by reconstructing them from fundamental components.

7. To contribute towards FOSSEE's broader mission of promoting open-source alternatives to proprietary EDA tools in academic institutions.

1.4 Methodology

The methodology adopted for carrying out this project followed a structured, step-by-step approach, ensuring that every IC modeled met a consistent standard of accuracy and validation before being finalized:

1. **Literature and Datasheet Study:** For each IC under consideration, the official manufacturer datasheet was studied in detail to understand its pin configuration, internal architecture, and expected electrical behavior. This step often involved cross-referencing multiple datasheet revisions or application notes to resolve ambiguities in timing or functional description.

2. **Internal Circuit Reconstruction:** The internal transistor-level (or gate-level) circuit of each IC was recreated using eSim's schematic editor, employing basic components such as BJTs, diodes, and resistors available in the standard library. Where a datasheet provided only a functional block diagram rather than a full transistor-level schematic, equivalent gate-level or macro-model representations were used instead.

3. **Subcircuit Encapsulation:** The reconstructed internal circuit was encapsulated as a subcircuit by defining appropriate external ports corresponding to the IC's real pin layout, and generating the associated `.SUBCKT` netlist definition.

4. **Symbol Creation:** A schematic symbol was created for each subcircuit so that it could be reused like any standard library component in subsequent test schematics, with pin placement matching the IC's real-world package orientation.

5. **Test Schematic Design:** A test schematic was designed for each IC, incorporating appropriate input sources (voltage sources, pulse generators, etc.) and measurement probes to observe outputs, closely mirroring the test conditions or application circuits shown in the datasheet.

6. **Simulation:** Each test schematic was simulated using Ngspice through eSim's integrated interface, employing suitable analyses (transient, DC, or AC) depending on the nature of the IC.

7. **Validation:** The simulation results (waveforms, truth tables, or characteristic curves) were compared against the expected behavior specified in the datasheet to confirm the correctness of the subcircuit model.

8. **Documentation:** Each IC's design process, internal architecture, simulation results, and any observed deviations or challenges were documented systematically, forming the basis of Chapter 4 of this report.

9. **Iteration and Troubleshooting:** Where discrepancies were found between simulated and expected behavior, the internal subcircuit model was revisited and refined iteratively until satisfactory agreement was achieved.

This methodology ensured a consistent, repeatable, and verifiable process across all ICs studied in this project, and forms the framework within which the detailed IC-by-IC results in Chapter 4 should be interpreted.

Chapter 2

Literature Survey

2.1 Introduction

This chapter reviews the body of literature relevant to the present project, with emphasis on three major themes: the development of open-source Electronic Design Automation (EDA) tools, the theoretical basis of SPICE-based subcircuit modeling, and the technical background of the eighteen integrated circuits modeled in this work. The purpose of this literature survey is to establish the academic and technical context in which the present project is situated, and to justify the importance of modeling additional ICs within the eSim ecosystem.

As electronic circuit design has evolved, simulation tools have become an indispensable part of both academic learning and professional design workflows. However, access to mature commercial EDA environments has historically been limited by licensing cost, motivating the development and adoption of open-source alternatives. At the same time, the usefulness of any circuit design environment depends not only on its simulation engine, but also on the breadth and practical utility of its component library. This makes the study of subcircuit modeling especially relevant, since it offers a scalable method for extending existing EDA tools with new IC models derived from manufacturer documentation.

2.2 Survey of Open-Source EDA Tools and SPICE Simulation

Circuit simulation using SPICE-based engines has remained the standard approach for analog and mixed-signal verification since the original SPICE program was developed at the University of California, Berkeley, in the early 1970s. Over time, commercial derivatives such as PSpice, HSPICE, and LTspice expanded this foundation into highly capable simulation environments, but most full-featured professional variants have remained costly and therefore less accessible to students and many academic institutions. This cost barrier has contributed to sustained interest in open-source alternatives that are sufficiently capable for teaching, experimentation, and research.

Among open-source SPICE engines, Ngspice has emerged as one of the most

actively maintained successors to the Berkeley SPICE lineage, supporting analog, digital, and mixed-signal simulation with modern device models and improved numerical methods. In parallel, KiCad has matured into a powerful open-source platform for schematic capture and PCB design, making it possible to create a nearly complete open-source EDA workflow when coupled with Ngspice. eSim is built around this exact integration strategy, combining schematic design, simulation, and related workflows into a unified environment suitable for electronics education and practical circuit development.

An important concept inherited from SPICE literature is the `.SUBCKT` based hierarchical modeling approach, where a complex circuit can be represented as a reusable module with defined input and output pins. This approach is especially significant in the context of IC modeling, because it allows complex transistor-level or gate-level implementations to be encapsulated as single reusable components. In practical terms, subcircuit modeling provides a bridge between low-level device representation and high-level design reuse, making it highly suitable for extending the component library of tools such as eSim.

2.3 Category-Wise Review of ICs Modeled in This Project

The integrated circuits selected for this project were chosen to represent a broad cross-section of commonly used analog and digital building blocks. Rather than focusing on a single narrow class of devices, the project spans fundamental logic gates, multiplexers, arithmetic circuits, counters, registers, bus interface devices, display drivers, and analog signal-conditioning blocks. This diversity ensures that the contribution is educationally useful across a wide range of digital electronics, analog electronics, and mixed-signal laboratory applications.

Table 2.1 summarizes the complete set of ICs modeled in this project.

2.3.1 Basic Logic Gates

The SN7401, SN7402, SN7409, SN7433, and 74HC386 represent the most fundamental building blocks of digital electronics, implementing NAND, NOR, AND, open-collector NOR-buffer, and XOR functionality respectively. Standard digital design literature treats such gates as the basis of all combinational and sequential digital systems, since larger logic structures are ultimately synthesized from these primitives. In the TTL and CMOS logic families, these devices are also historically important because they reveal how different logic technologies trade off switching speed, power dissipation, fan-out capability, and output drive characteristics.

From a datasheet perspective, these ICs are well documented in terms of truth tables, recommended operating conditions, and propagation characteristics. However, reproducing their behavior as subcircuits still requires careful modeling of their output structures, especially in cases involving open-collector outputs or family-specific characteristics. Their inclusion in this project therefore serves both an educational purpose and a practical one: they are simple enough to validate rigorously, yet

Table 2.1: List of Integrated Circuits Modeled in This Project

S.No.	IC	Description
1	SN7401	Quad 2-Input NAND Gate
2	SN7402	Quad 2-Input NOR Gate
3	SN7409	Quad 2-Input AND Gate
4	SN7433	Quad 2-Input NOR Buffer (Open-Collector Outputs)
5	74HC386	Quad 2-Input XOR Gate
6	74HC158	Quad 2-to-1 Multiplexer
7	74HC251	8-to-1 Multiplexer with Tri-State Output
8	74S137	3-to-8 Line Decoder/Demultiplexer
9	74164	8-Bit Serial-In Parallel-Out Shift Register
10	74LS393	Dual 4-Bit Binary Ripple Counter
11	74LS197	Presettable Binary Counter
12	SN74LS283	4-Bit Binary Full Adder
13	74HC280	9-Bit Parity Generator
14	74LS245	Octal Bus Transceiver with Tri-State Outputs
15	74HC4511	BCD-to-7-Segment Latch/Decoder
16	8212	8-Bit Input/Output Port
17	TLC274	Quad Low-Power CMOS Op-Amp
18	CA3098	Programmable Schmitt Trigger
19	ICL8038	Precision Waveform Generator

foundational enough to be reused in a wide variety of future circuits.

2.3.2 Data Selection Devices

The 74HC158, 74HC251, and 74S137 belong to the class of logic devices used for signal routing and selection. Multiplexers and decoders are central to data path design, memory addressing, and bus control, since they allow one among several input or output paths to be selected through a smaller set of control signals. These devices are frequently encountered in textbooks on digital systems, microprocessor interfacing, and logic design because they provide compact implementations of routing and control functions that would otherwise require multiple primitive gates.

Their datasheets emphasize not only their truth-functional behavior but also control-oriented features such as enable inputs and tri-state outputs. These details are especially important for realistic subcircuit modeling, because a multiplexer's behavior cannot be captured correctly unless both data selection and output enable

conditions are faithfully reproduced. The same applies to decoder and demultiplexer devices, where the active state of outputs and the interpretation of enable signals significantly affect overall system operation.

2.3.3 Storage and Counting Elements

The 74164, 74LS393, and 74LS197 represent sequential logic devices, which differ fundamentally from combinational logic in that their outputs depend not only on present inputs but also on prior circuit states. Shift registers and counters are extensively covered in standard digital electronics literature as basic memory and timing elements, used in data serialization, event counting, frequency division, and control sequencing. The shift from simple gate modeling to sequential device modeling introduces an additional layer of complexity, since correct timing behavior becomes essential.

Datasheet literature for these devices therefore places strong emphasis on clocking behavior, asynchronous clear or preset functions, and the sequence of state transitions. In simulation, such devices must be validated through time-domain analysis rather than static truth tables alone. Their inclusion in this project broadens the usefulness of the resulting eSim library by adding commonly required sequential building blocks for teaching and experimentation.

2.3.4 Arithmetic and Parity Circuits

The SN74LS283 and 74HC280 represent functional blocks related to arithmetic computation and data integrity checking. Full adders form the basis of binary arithmetic units, while parity generators are widely used for simple error-detection schemes in digital communication and storage systems. These devices are particularly interesting from a modeling perspective because they sit between simple logic gates and larger arithmetic subsystems, making them useful as mid-level digital building blocks.

Datasheets for such ICs typically provide detailed logical relationships, carry behavior, parity generation conventions, and recommended cascading arrangements. Modeling them accurately requires more than direct gate copying; it also requires ensuring that internal logical relationships produce correct multi-bit outputs under all tested input combinations. Their presence in this project strengthens the arithmetic and data-verification capabilities of the resulting subcircuit collection.

2.3.5 Bus Interfacing and Display Driving Devices

The 74LS245, 74HC4511, and 8212 are representative of devices designed to connect core logic systems with external buses, displays, and peripheral interfaces. Literature on digital system design highlights such ICs as essential support components in practical systems, especially in microprocessor-based architectures where buffering, latching, decoding, and display driving are all common requirements. Unlike basic gates, these devices are closely associated with system-level integration.

Their datasheets include control signals such as direction, latch enable, tri-state enable, and output decoding logic, all of which must be represented carefully in a

working subcircuit model. The 74HC4511, in particular, is useful in educational contexts because it links binary-coded decimal input logic to visible seven-segment display output behavior. Similarly, the 74LS245 and 8212 are relevant for bus-oriented data exchange and I/O interfacing, making them valuable additions for broader digital system simulations.

2.3.6 Analog and Mixed-Signal Devices

The TLC274 and CA3098 extend the scope of this project beyond purely digital logic into analog and mixed-signal modeling. Operational amplifiers and Schmitt trigger based devices are widely studied in analog electronics because they illustrate gain, feedback, thresholding, hysteresis, and signal conditioning. Unlike digital ICs, whose behavior can often be validated primarily against truth tables, analog devices require attention to continuous voltage behavior, transfer characteristics, and device-level operating regions.

The TLC274 is a quad low-power CMOS operational amplifier, making it relevant in low-power analog signal processing applications. The CA3098, as a programmable Schmitt trigger, is important for waveform shaping, noise immunity improvement, and threshold-based signal conditioning. Modeling such devices in eSim requires a more detailed transistor-level or macro-model approach than most logic ICs, and their inclusion makes the overall contribution of this project more balanced by covering analog as well as digital circuit design needs.

2.4 Summary and Identified Gap

The literature reviewed in this chapter indicates that open-source EDA tools have matured substantially, and SPICE-based simulation remains the accepted foundation for electronic circuit verification. It also shows that hierarchical subcircuit modeling is a technically established and scalable method for representing reusable IC-level functionality within SPICE-based environments. However, the practical value of such environments depends strongly on the availability of sufficiently broad and well-validated component libraries.

The ICs selected in this project occupy an important educational and practical middle ground: they are common enough to be highly useful in digital and analog electronics learning, yet specialized enough that they are not always readily available as reusable models in open-source design environments. By modeling devices spanning logic gates, routing logic, sequential circuits, arithmetic circuits, bus/display interfacing blocks, and analog components, this project addresses that gap and contributes toward a more functionally complete subcircuit library for future eSim users.

Chapter 3

Problem Statement

3.1 Problem Statement

eSim, despite being a functionally comprehensive and actively developed open-source EDA platform, relies on a component library whose breadth is determined by the range of devices modeled and integrated into it so far. Unlike commercial EDA tools such as OrCAD or Multisim, which are supported by dedicated commercial teams maintaining large and continuously updated component libraries, eSim expands primarily through community contributions and structured development efforts such as fellowship projects. As a result, a considerable number of commonly used integrated circuits, including standard TTL/CMOS logic devices as well as certain analog components, are not available as ready-to-use elements within the eSim library.

This limitation creates a practical difficulty for students, educators, and designers who wish to simulate circuits involving such devices in eSim. In the absence of a prebuilt model, a user must either search for and manually import an external SPICE model of uncertain compatibility, or avoid simulation of that device entirely and rely only on theoretical analysis. Neither option is ideal in an educational or design-oriented workflow, since the former may introduce reliability and integration issues, while the latter defeats the simulation-based learning and verification capability that eSim is intended to provide.

The problem addressed in this project is therefore the absence of accurate and reusable subcircuit models for a functionally diverse set of eighteen integrated circuits within the eSim environment. These devices span basic logic gates, data-selection circuits, storage and counting elements, arithmetic and parity circuits, bus and display interfacing devices, and analog components. Without such models, the scope of circuits that can be realistically designed and simulated in eSim remains unnecessarily restricted.

3.2 Approach

The approach adopted to address this problem was based on modeling each selected IC as a SPICE subcircuit using eSim's Subcircuit Builder and Model Builder utilities, together with device models already available within the eSim library. This avoided reliance on externally sourced or unverified model files and ensured compatibility

with the existing eSim simulation workflow. The overall objective was not merely to reproduce symbolic functionality, but to create reusable models that could be integrated into practical circuit simulations within the platform.

For each of the eighteen ICs considered in this work, a consistent methodology was followed:

1. **Datasheet-based analysis** — The manufacturer's datasheet for each IC was treated as the primary technical reference for understanding its internal structure, pin configuration, truth tables, operating conditions, and functional characteristics.

2. **Subcircuit development** — Based on the available datasheet information, the internal behavior of the IC was reconstructed as a subcircuit and encapsulated using the `.subckt` representation. External terminals were mapped according to the actual pin arrangement of the device, and a corresponding schematic symbol was created for use within eSim.

3. **Test circuit construction** — A dedicated test circuit was designed for each IC in order to verify the behavior of the developed subcircuit model. Wherever possible, the testing strategy was aligned with the expected operating behavior described in the datasheet.

4. **Simulation and verification** — The test circuits were simulated through eSim's KiCad-to-Ngspice workflow, and the resulting output waveforms or logic states were compared against the functional behavior expected from the datasheet specifications.

5. **Iterative correction** — Whenever the simulated results deviated from the expected behavior, the subcircuit design was re-examined, corrected, and simulated again until satisfactory agreement was obtained.

This methodology was applied uniformly across all eighteen ICs, irrespective of whether the device belonged to a digital logic family or to the analog and mixed-signal category. The primary variation lay only in the nature of the internal implementation, with digital ICs generally requiring logic-level or gate-based reconstruction, while analog devices required more detailed transistor-level or macro-model based realization. The resulting verified subcircuits, together with their pin diagrams, internal schematics, test circuits, and simulation outputs, are presented in the following chapter.

Chapter 4

Implementation

4.1 SN7401 - Quad 2-Input NAND Gate

4.1.1 Description

The SN7401 is a quad 2-input NAND gate IC belonging to the standard TTL family. It contains four independent NAND gates in a single package, and each gate uses an open-collector output stage. Because of this output configuration, an external pull-up resistor is required to obtain a valid logic-high level. The device is commonly used in logic gating, wired-AND style interconnections, and bus-oriented digital circuits.

float

4.1.2 Pin Diagram

The SN7401 is available in a 14-pin dual in-line package. It includes eight input pins, four output pins, one `VCC` pin, and one `GND` pin, arranged so that each NAND gate has two inputs and one output.

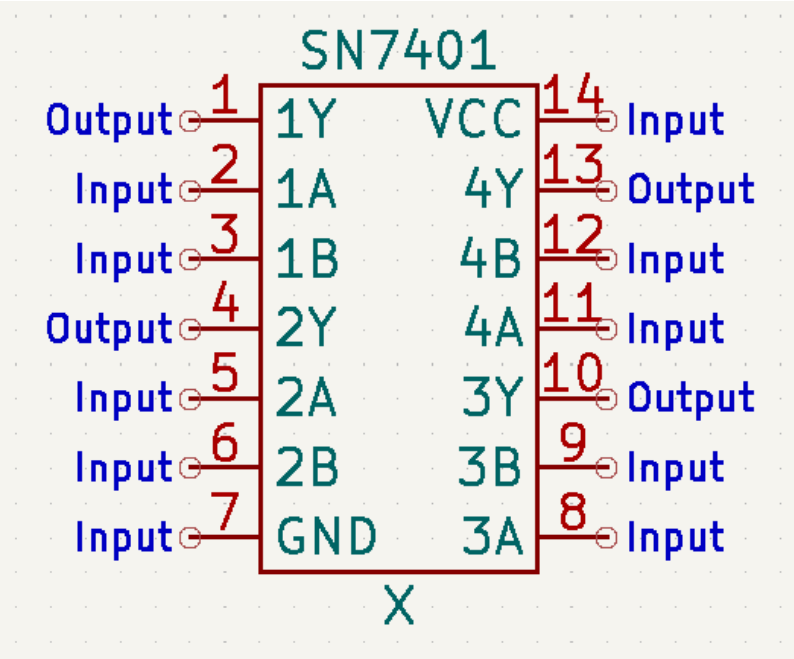

Figure 4.1: Pin Diagram of SN7401

4.1.3 Features

- Four independent 2-input NAND gates in a single package.
- Open-collector outputs, allowing wired-AND style connection of multiple outputs.
- Standard TTL input and output logic levels.
- Compatible with other standard TTL family devices.

4.1.4 Applications

- Basic Boolean logic implementation in digital systems.
- Wired-AND bus logic where multiple open-collector outputs are combined.
- Driving external loads through open-collector outputs with a pull-up resistor.
- Building block for larger combinational logic circuits.

4.1.5 Subcircuit Design

The internal architecture of the SN7401 was reconstructed at the transistor level using BJT device models available in eSim. Each NAND gate was implemented using the appropriate TTL structure, and the open-collector output stage was preserved in the design. The four gates were then mapped to the corresponding package pins and encapsulated as a reusable subcircuit.

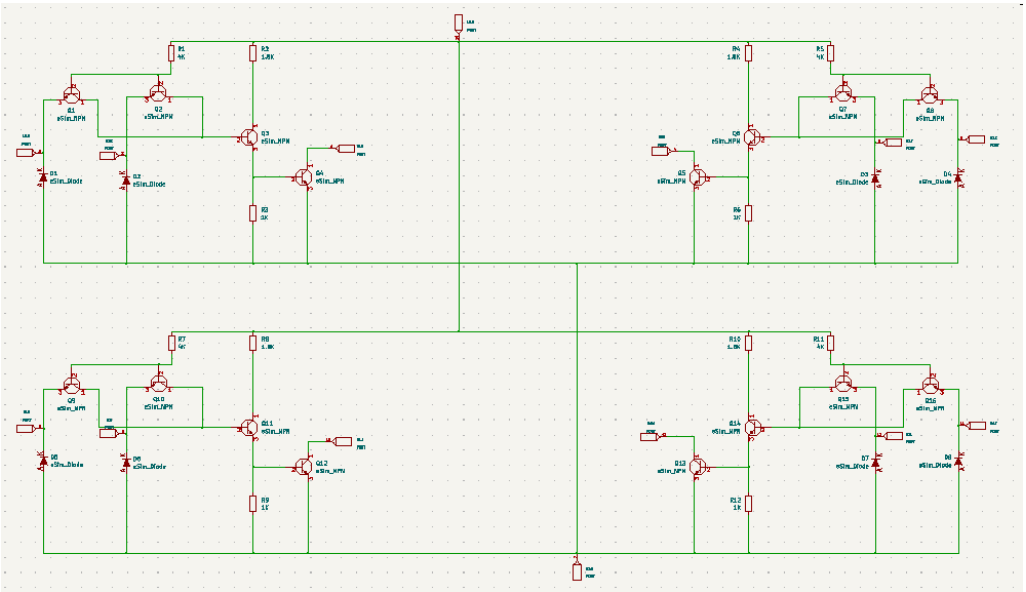

Figure 4.2: Subcircuit Schematic of SN7401

4.1.6 Test Circuit

A test circuit was designed for one NAND gate of the SN7401 to verify its operation under different input combinations. Two pulse sources were applied to the inputs, and a pull-up resistor was connected at the output to represent the open-collector behaviour correctly. The circuit was simulated to confirm the expected NAND response.

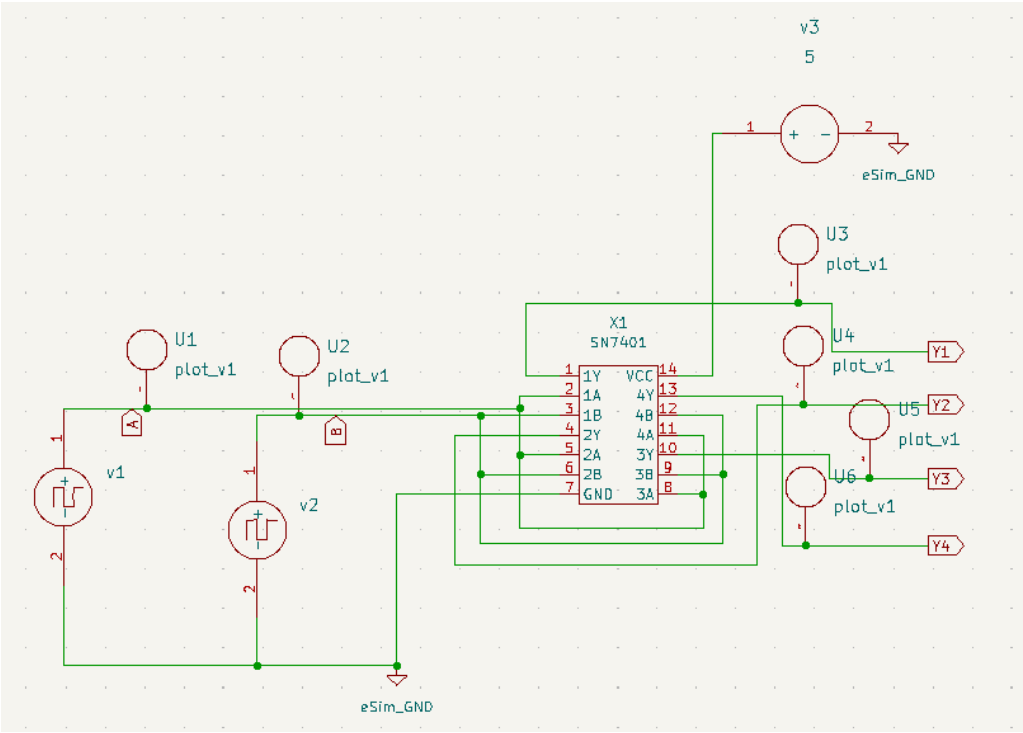

Figure 4.3: Test Circuit of SN7401

4.1.7 Output

The input waveform was chosen to generate the required logic combinations at the two NAND inputs. The output waveform showed the expected NAND behavior, with the output remaining high for all input combinations except when both inputs were high, in which case the output went low. This confirmed that the developed subcircuit model matched the expected device functionality.

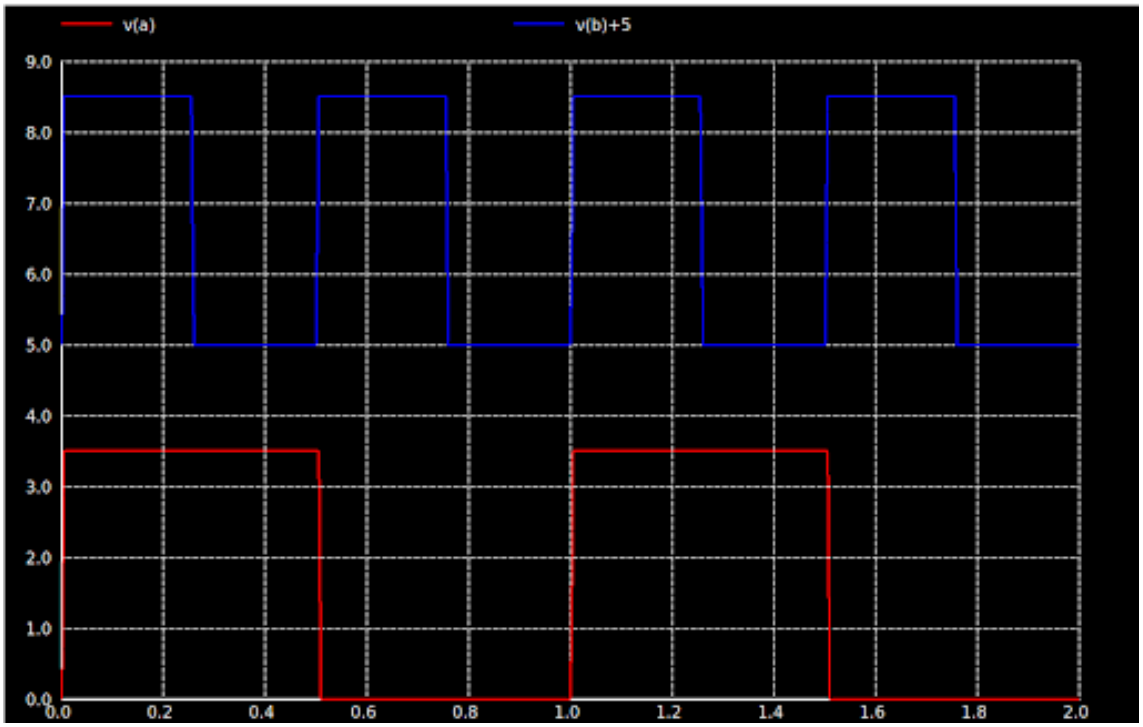

Figure 4.4: Input Waveform of SN7401

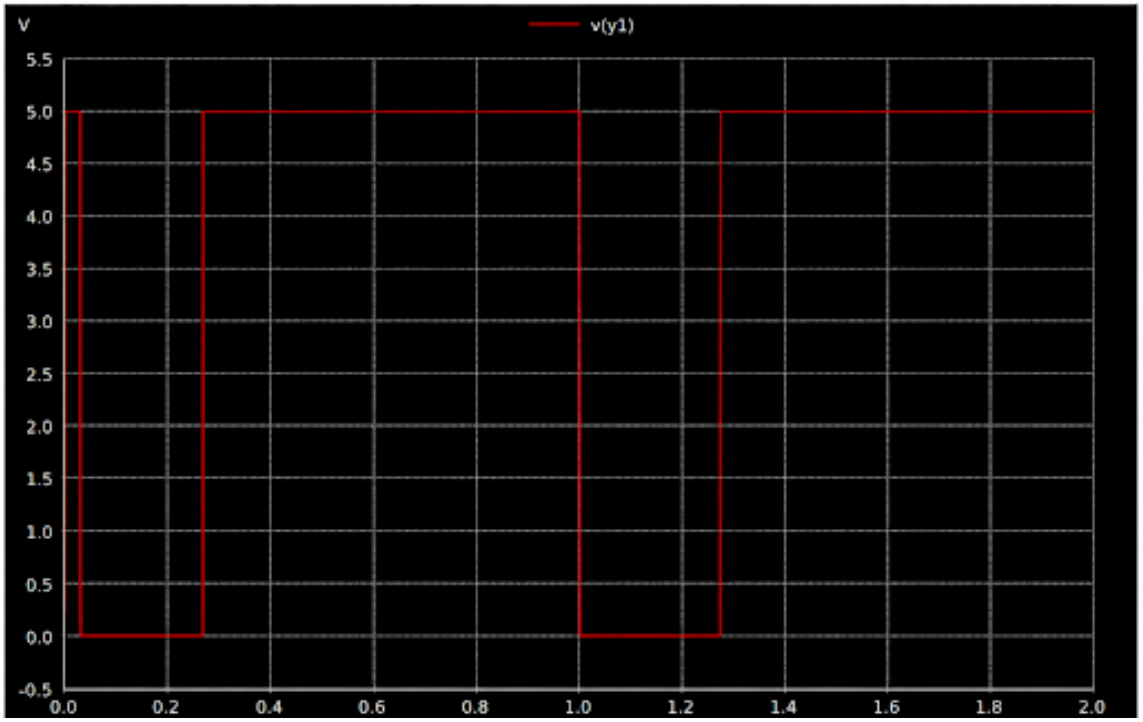

Figure 4.5: Output Waveform of SN7401

4.2 SN7402 - Quad 2-Input NOR Gate

4.2.1 Description

The SN7402 is a quad 2-input NOR gate IC belonging to the standard TTL family. It contains four independent NOR gates in a single package, each with a standard totem-pole output stage capable of driving TTL-compatible loads directly without requiring an external pull-up resistor. The device is commonly used wherever basic NOR logic is required, including in combinational logic design and as a building block for more complex digital functions such as flip-flops and oscillators.

4.2.2 Pin Diagram

The SN7402 is available in a 14-pin dual in-line package. Unlike the SN7401, the output of each NOR gate is placed adjacent to its corresponding inputs on the same side of the package (e.g., pin 1 is the output 1Y, with inputs 1A and 1B on pins 2 and 3), with `VCC` on pin 14 and `GND` on pin 7.

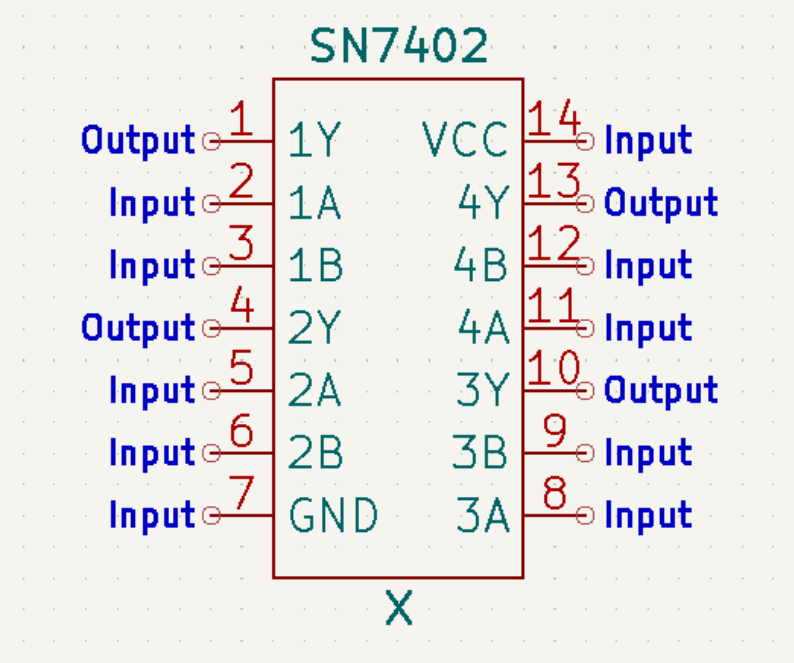

Figure 4.6: Pin Diagram of SN7402

4.2.3 Features

- Four independent 2-input NOR gates in a single package.
- Standard totem-pole output stage, no external pull-up required.
- Standard TTL input and output logic levels.
- Compatible with other standard TTL family devices.

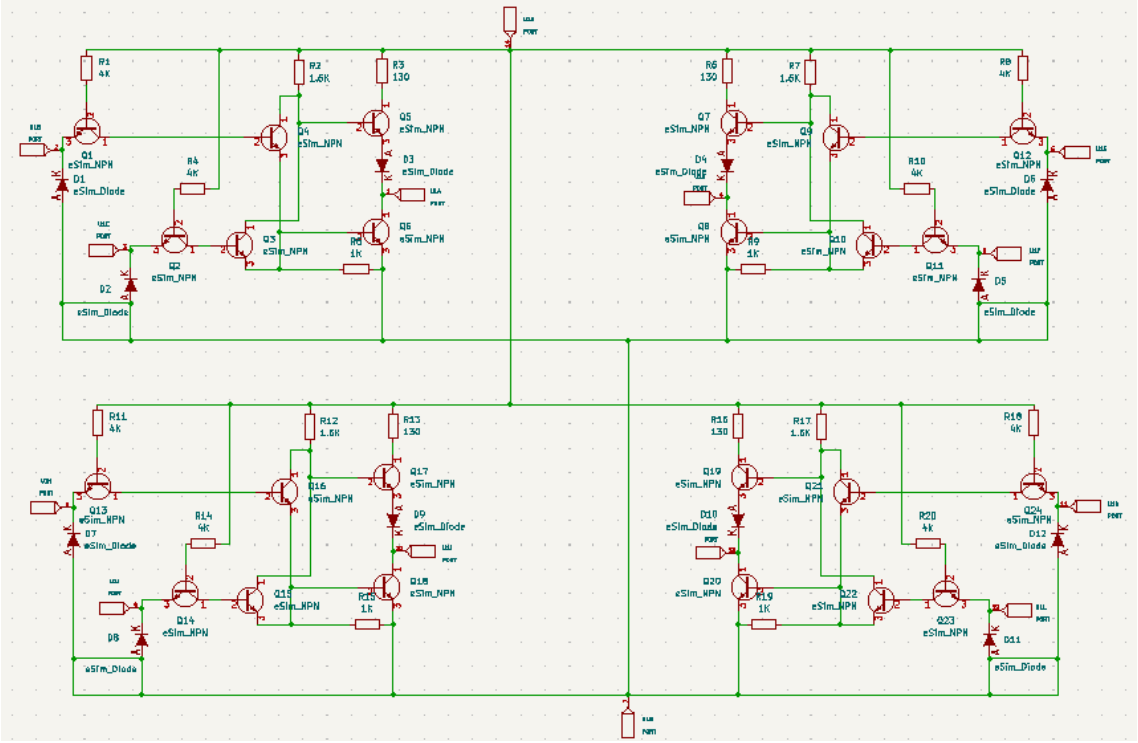

Figure 4.7: Subcircuit Schematic of SN7402

4.2.4 Applications

- Basic Boolean logic implementation in digital systems.
- Building block for latches, flip-flops, and oscillator circuits.
- Combinational logic design requiring NOR functionality.
- General-purpose gating in TTL-based digital systems.

4.2.5 Subcircuit Design

The internal architecture of the SN7402 was reconstructed at the transistor level using BJT and diode device models available in eSim, following the standard TTL NOR gate structure. Each of the four gates consists of a multi-transistor input and phase-splitter arrangement followed by a totem-pole output stage, with diodes used for input clamping. The four gates were mapped to the corresponding package pins and encapsulated as a reusable subcircuit.

4.2.6 Test Circuit

A test circuit was designed using the SN7402 subcircuit, with two independent voltage sources connected to inputs A and B, and the outputs 1Y, 2Y, 3Y, and 4Y probed to observe the NOR response. The output of gate 1 (1Y) was analyzed in detail to verify correct switching behavior across all four input combinations.

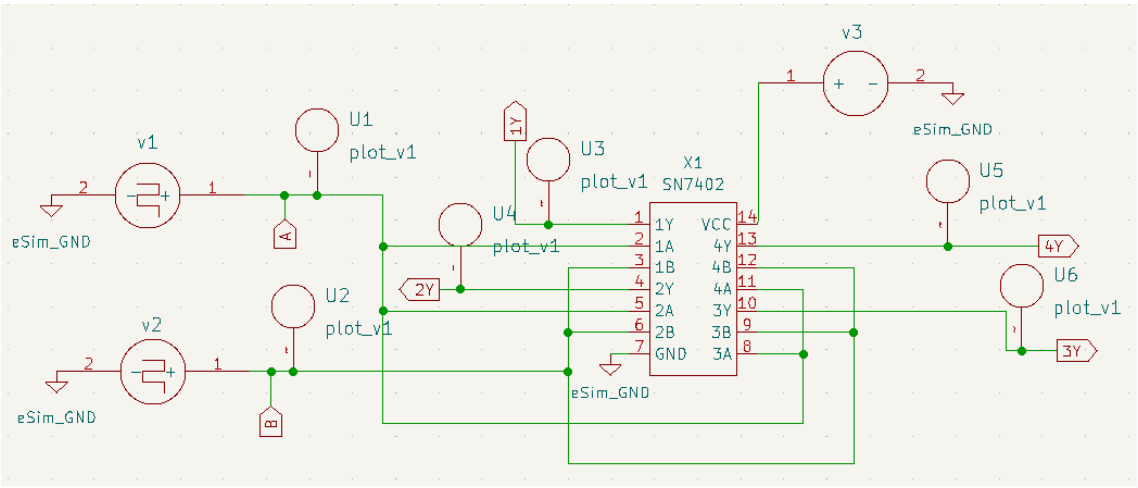

Figure 4.8: Test Circuit of SN7402

4.2.7 Output

The input waveform shows the two input signals applied to the gate: v(a) (offset by +5 V for clarity of display) and v(b), each toggling between logic low and logic high at different intervals to generate all four possible input combinations.

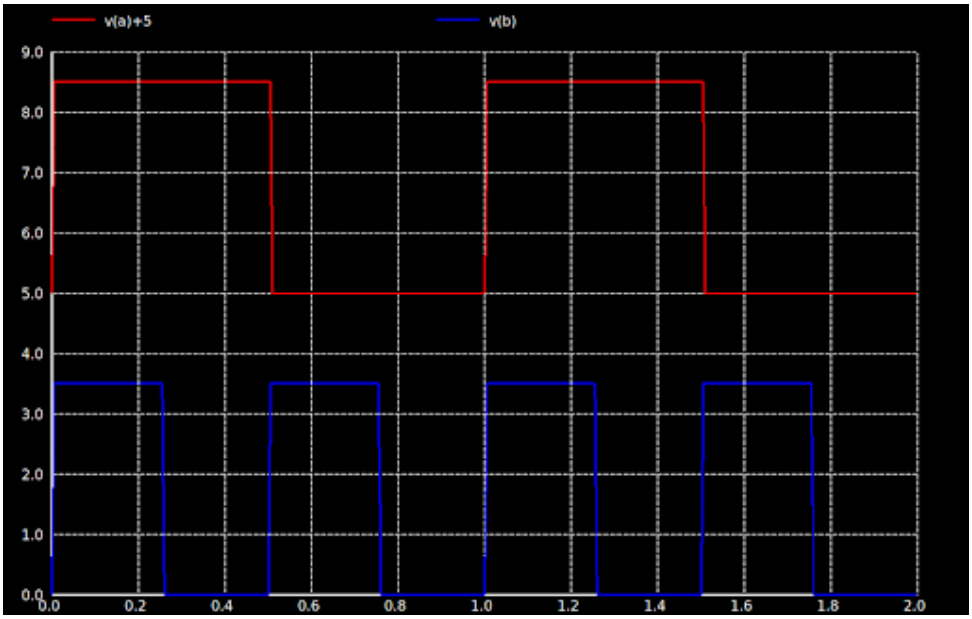

Figure 4.9: Input Waveform of SN7402

The resulting output waveform, v(1y), remained high only during the interval when both inputs were low, and dropped to logic low whenever either or both inputs were high. This matched the expected NOR truth table behavior, confirming that the developed subcircuit model correctly replicated the functionality of the SN7402.

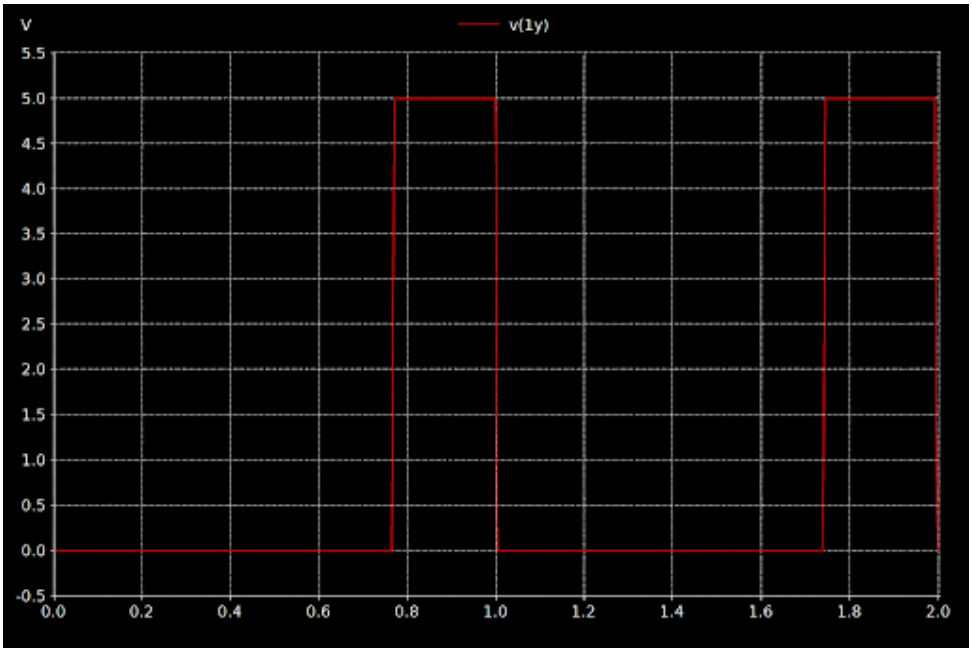

Figure 4.10: Output Waveform of SN7402

4.3 SN7409 – Quad 2-Input AND Gate

4.3.1 Description

The SN7409 is a quad 2-input AND gate IC belonging to the standard TTL family. It contains four independent AND gates in a single package, and each gate uses an open-collector output stage. Because of this output configuration, an external pull-up resistor is required to obtain a valid logic-high level, and the rise time of the output is governed by the RC time constant formed by the pull-up resistor and any load capacitance. The device is commonly used in logic gating applications and wired-logic style digital interconnections where open-collector outputs are advantageous.

4.3.2 Pin Diagram

The SN7409 is available in a 14-pin dual in-line package. Unlike the SN7401, the output of each AND gate is placed adjacent to its corresponding inputs on the same side of the package (e.g., pins 1 and 2 are inputs 1A and 1B, with output 1Y on pin 3), with `VCC` on pin 14 and `GND` on pin 7.

4.3.3 Features

- Four independent 2-input AND gates in a single package.
- Open-collector outputs, allowing wired-AND style connection of multiple outputs.

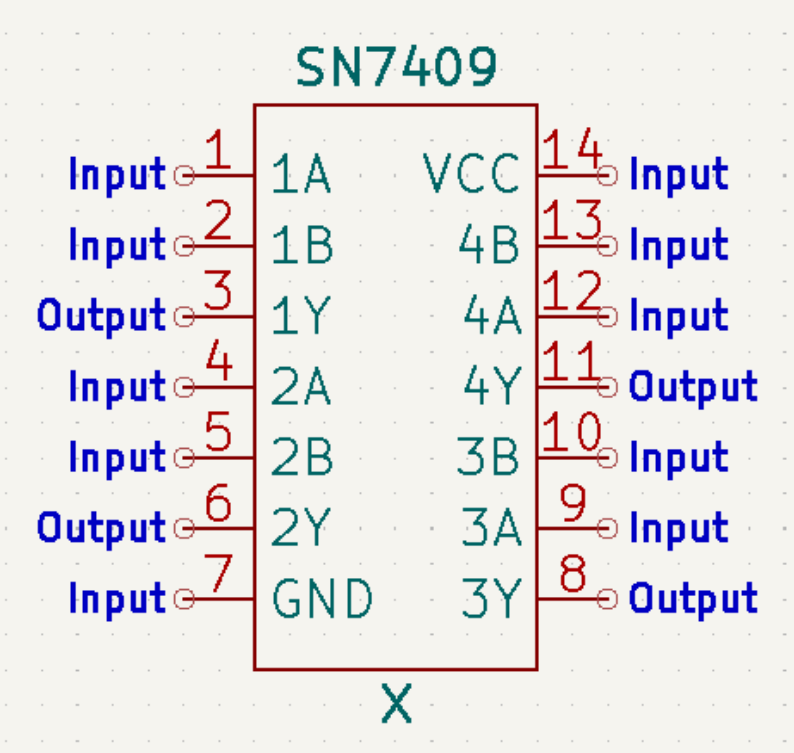

Figure 4.11: Pin Diagram of SN7409

- Standard TTL input logic levels.
- Compatible with other standard TTL family devices.

4.3.4 Applications

- Basic Boolean logic implementation in digital systems.
- Wired-AND bus logic where multiple open-collector outputs are combined.
- Driving external loads through open-collector outputs with a pull-up resistor.
- Building block for larger combinational logic circuits.

4.3.5 Subcircuit Design

The internal architecture of the SN7409 was reconstructed at the transistor level using BJT and diode device models available in eSim, following the standard TTL AND gate structure. Each of the four gates consists of a multi-transistor input and phase-splitter arrangement followed by an open-collector output transistor, with diodes used for input clamping. The four gates were mapped to the corresponding package pins and encapsulated as a reusable subcircuit.

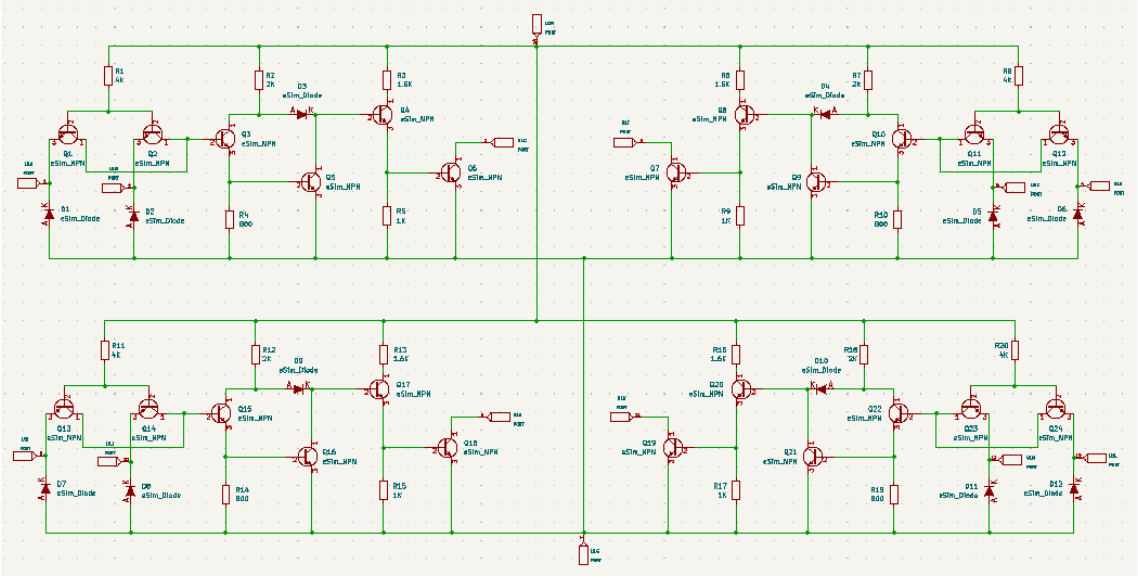

Figure 4.12: Subcircuit Schematic of SN7409

4.3.6 Test Circuit

A test circuit was designed using the SN7409 subcircuit, with two independent voltage sources connected to inputs A and B, and a pull-up resistor connected to the output of gate 1 (1Y) to represent the open-collector behaviour correctly. The outputs 1Y, 2Y, 3Y, and 4Y were probed to observe the AND response, with the output of gate 1 (1Y) analyzed in detail to verify correct switching behavior across all four input combinations.

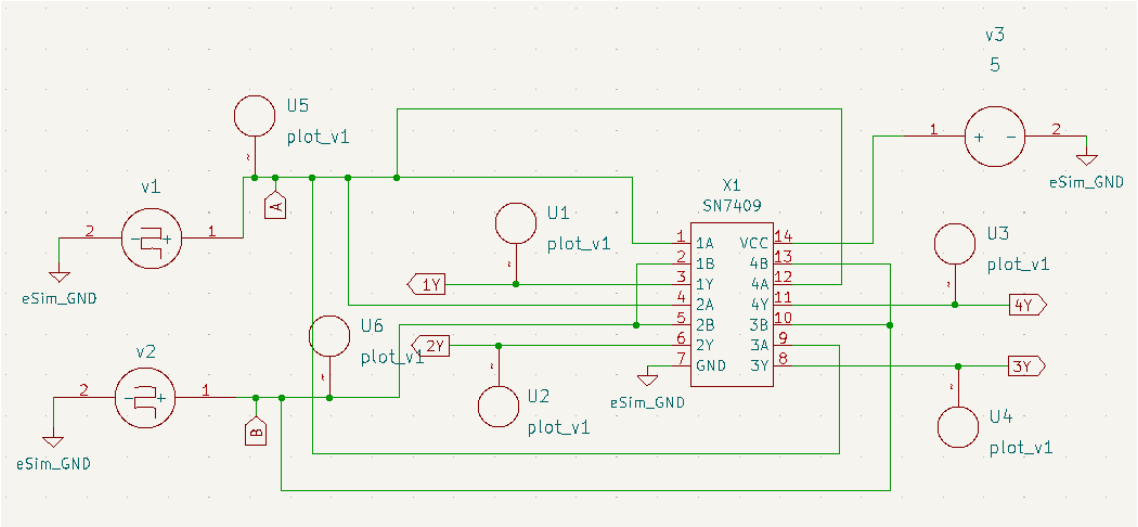

Figure 4.13: Test Circuit of SN7409

4.3.7 Output

The input waveform shows the two input signals applied to the gate: v(a) and v(b)+6 (offset for clarity of display), each toggling between logic low and logic high at different intervals to generate all four possible input combinations.

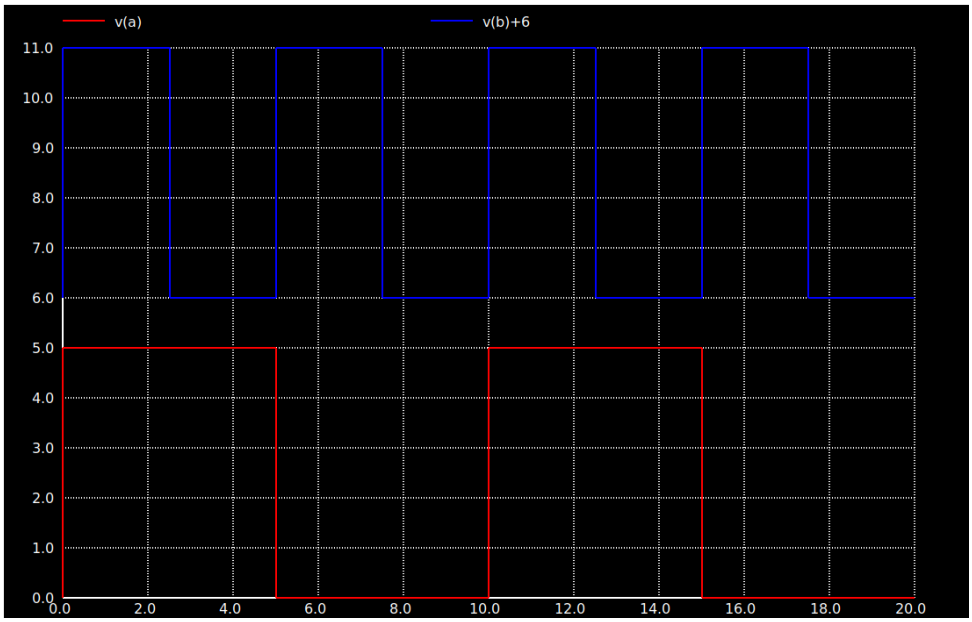

Figure 4.14: Input Waveform of SN7409

The resulting output waveform, v(y1), remained high only during the intervals when both inputs A and B were simultaneously high, and dropped to logic low whenever either or both inputs went low. The visibly rounded rising edges of the output are characteristic of the open-collector output stage charging through the external pull-up resistor. This matched the expected AND truth table behavior, confirming that the developed subcircuit model correctly replicated the functionality of the SN7409.

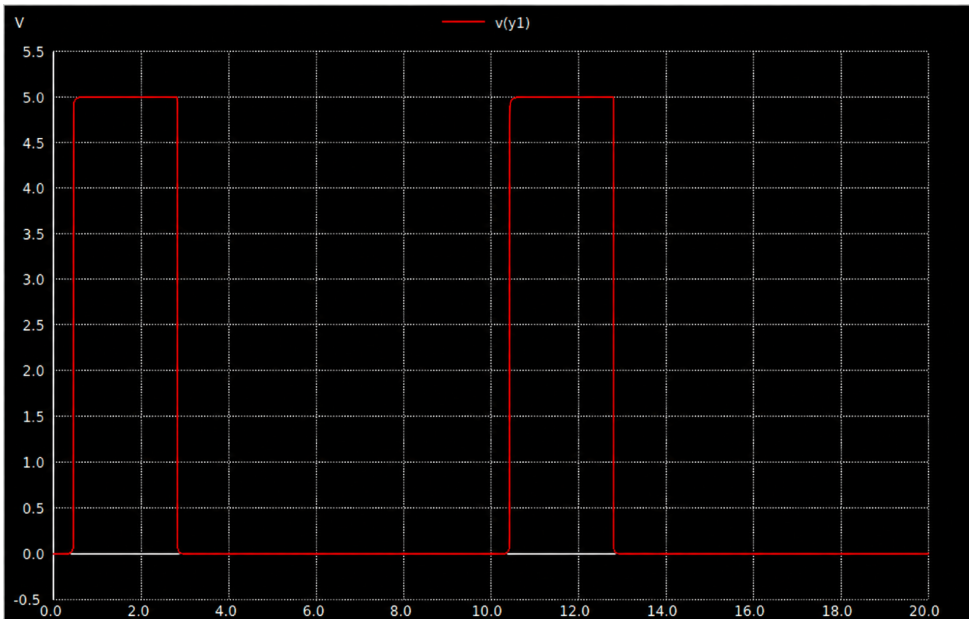

Figure 4.15: Output Waveform of SN7409

4.4 SN7433 – Quad 2-Input NOR Buffer

4.4.1 Description

The SN7433 is a quad 2-input NOR buffer IC belonging to the standard TTL family. It contains four independent NOR gates in a single package, each providing higher output drive capability than a standard NOR gate, and each using an open-collector output stage. Because of this output configuration, the SN7433 is well suited for driving heavier loads or interfacing with external circuitry operating at a different voltage level, though an external pull-up resistor is typically required at the output to obtain a valid logic-high level. The device is commonly used wherever NOR logic combined with buffered, high-drive output capability is required.

4.4.2 Pin Diagram

The SN7433 is available in a 14-pin dual in-line package. Each of the four NOR gates has its output placed adjacent to its two corresponding inputs (e.g., pin 1 is the output 1Y, with inputs 1A and 1B on pins 2 and 3), with `VCC` on pin 14 and `GND` on pin 7.

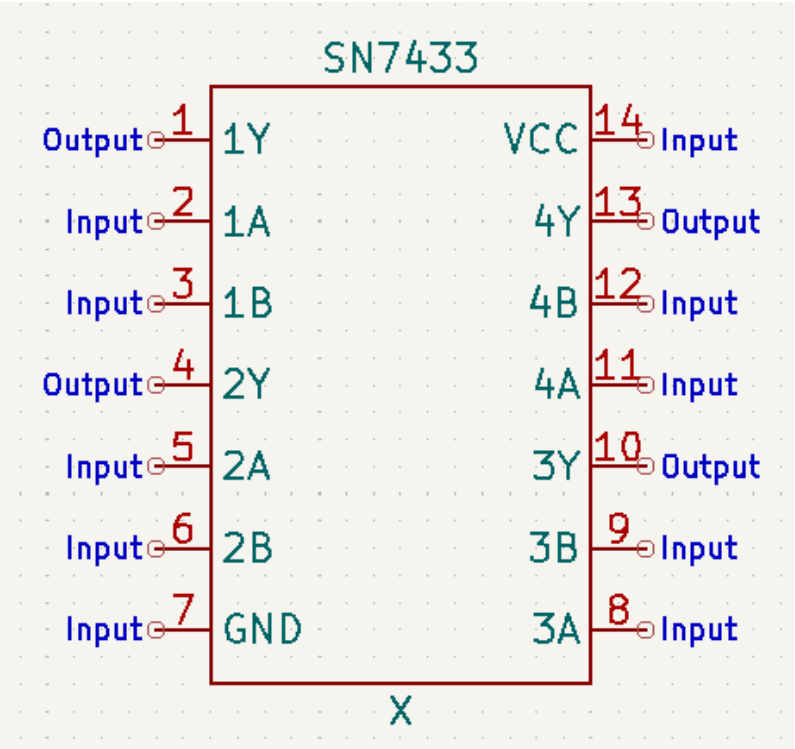

Figure 4.16: Pin Diagram of SN7433

4.4.3 Features

- Four independent 2-input NOR buffers in a single package.
- Open-collector outputs with higher drive capability than standard NOR gates.
- Standard TTL input logic levels.
- Compatible with other standard TTL family devices.

4.4.4 Applications

- NOR logic implementation requiring higher output drive.
- Driving external loads or interfacing circuits through open-collector outputs.
- Wired-logic style digital interconnections.
- General-purpose buffering in TTL-based digital systems.

4.4.5 Subcircuit Design

The internal architecture of the SN7433 was reconstructed at the transistor level using BJT and diode device models available in eSim, following the standard TTL NOR buffer structure. Each of the four gates consists of a multi-transistor input and phase-splitter arrangement followed by an open-collector output transistor, with diodes used for input clamping. The four gates were mapped to the corresponding package pins and encapsulated as a reusable subcircuit.

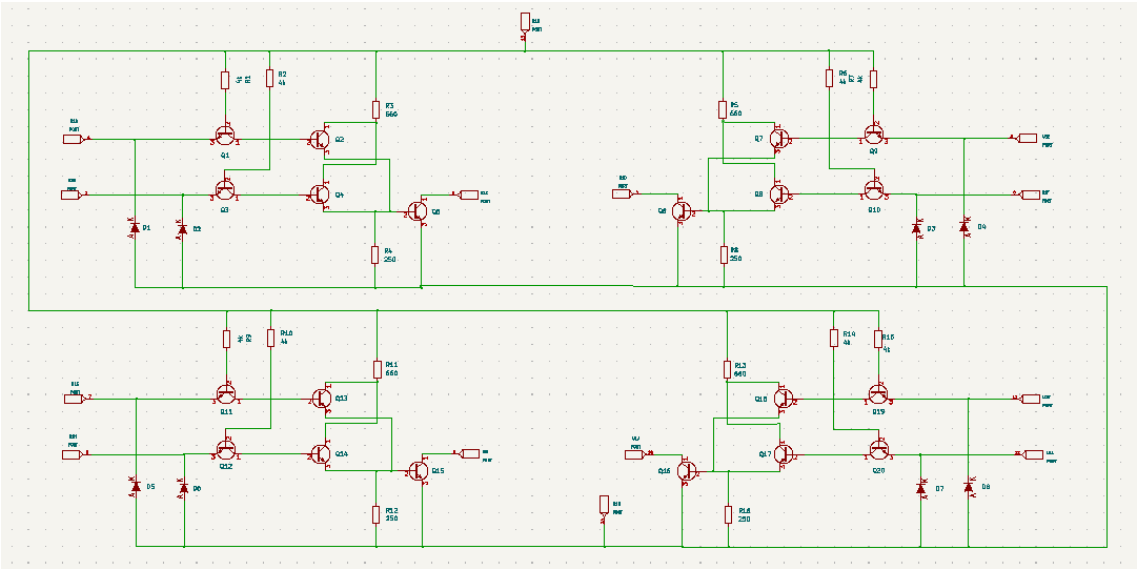

Figure 4.17: Subcircuit Schematic of SN7433

4.4.6 Test Circuit

A test circuit was designed using gate 1 of the SN7433 subcircuit, with two independent voltage sources connected to inputs 1A and 1B, and the output 1Y directly probed to observe the NOR response across all four input combinations.

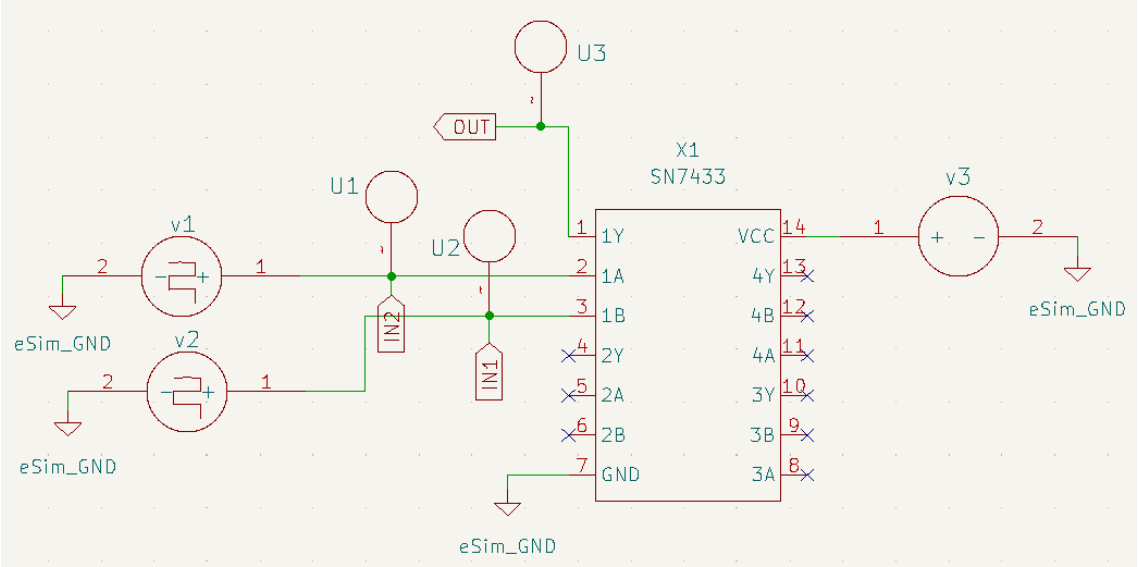

Figure 4.18: Test Circuit of SN7433

4.4.7 Output

The input waveform shows the two input signals applied to the gate: v(in1) and v(in2)+6 (offset for clarity of display), each toggling between logic low and logic high at different intervals to generate all four possible input combinations.

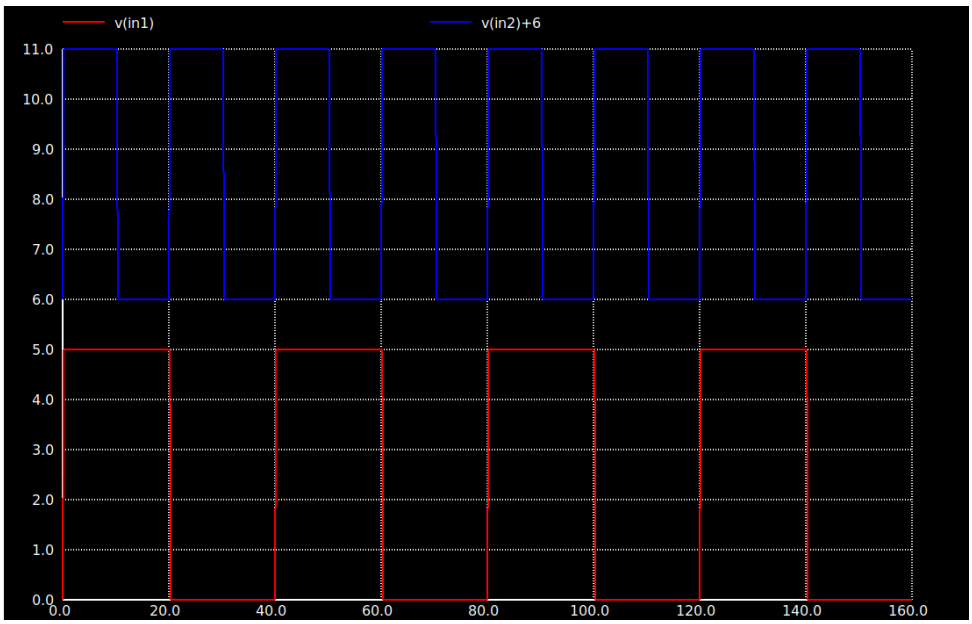

Figure 4.19: Input Waveform of SN7433

The resulting output waveform, v(out), remained high only during the intervals when both inputs (in1 and in2) were simultaneously low, and dropped to logic low whenever either or both inputs went high. This matched the expected NOR truth table behavior, confirming that the developed subcircuit model correctly replicated the functionality of the SN7433.

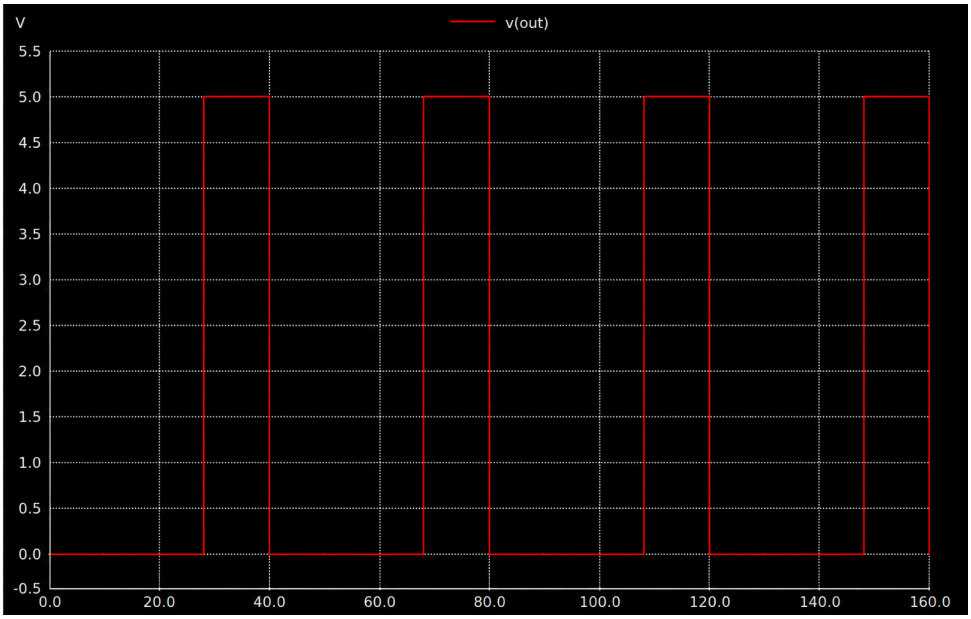

Figure 4.20: Output Waveform of SN7433

4.5 74HC386 - Quad 2-Input XOR Gate

4.5.1 Description

The 74HC386 is a quad 2-input XOR (Exclusive-OR) gate IC belonging to the high-speed CMOS (HC) logic family. It contains four independent XOR gates in a single package, each producing a logic-high output only when its two inputs differ in logic state. Being part of the HC family, the device offers CMOS-level power efficiency while maintaining speed performance comparable to standard TTL logic, and is commonly used wherever bitwise comparison, parity, or difference-detection logic is required.

4.5.2 Pin Diagram

The 74HC386 is available in a 14-pin dual in-line package, with eight input pins and four output pins arranged so that each XOR gate has two inputs and one output.

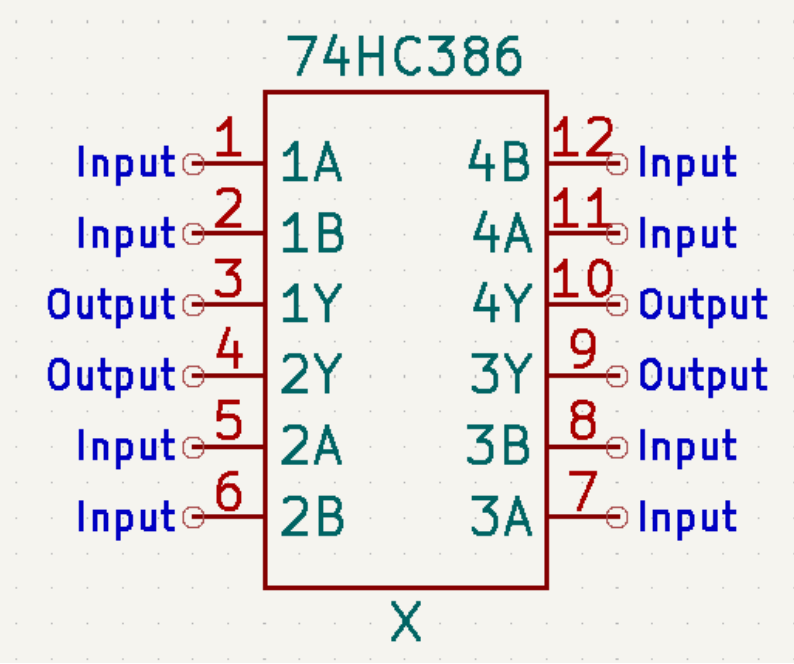

Figure 4.21: Pin Diagram of 74HC386

4.5.3 Features

- Four independent 2-input XOR gates in a single package.
- High-speed CMOS (HC) logic family with low static power consumption.
- Standard totem-pole output stage, no external pull-up required.
- Output is logic-high only when the two inputs differ.
- Compatible with other standard CMOS/HC family devices.

4.5.4 Applications

- Bitwise comparison and difference-detection logic.
- Parity generation and checking circuits.
- Controlled inverter stages (using one input as a control signal).
- Building block for adders, comparators, and error-detection circuits.

4.5.5 Subcircuit Design

The internal architecture of the 74HC386 was implemented in eSim using digital gate primitives rather than transistor-level modeling. Each of the four XOR gates was realized by combining a `d_xnor` gate with a `d_inverter` at its output, since an XOR function can be equivalently obtained by inverting the output of an XNOR gate. Additionally, `d_inverter` stages were placed at each input line to condition the signals before they reached the XNOR gate. The four such gate arrangements were mapped to their respective input/output ports (1A/1B/1Y, 2A/2B/2Y, 3A/3B/3Y, and 4A/4B/4Y) and encapsulated together as a single reusable subcircuit representing the complete 74HC386 package.

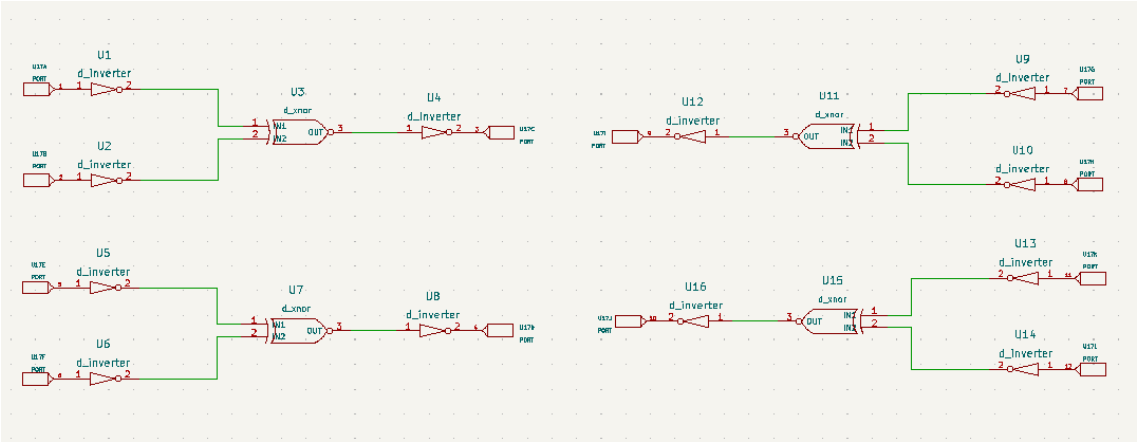

Figure 4.22: Subcircuit Schematic of 74HC386

4.5.6 Test Circuit

Since the 74HC386 subcircuit was built using digital gate primitives, an `adc_bridge_8` was used at the input stage to convert the analog voltage signals from two pulse sources into digital logic levels compatible with the subcircuit's inputs, and a `dac_bridge_4` was used at the output stage to convert the digital outputs (1Y, 2Y, 3Y, 4Y) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

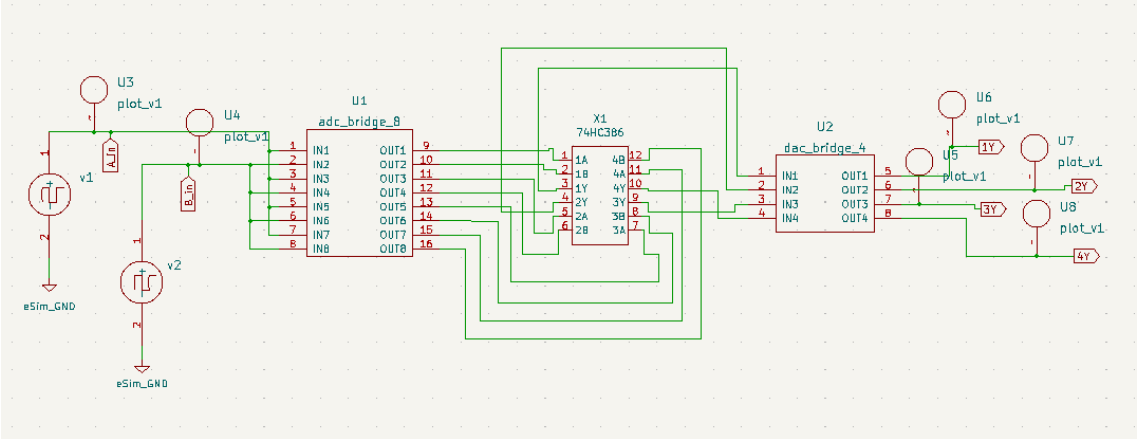

Figure 4.23: Test Circuit of 74HC386

4.5.7 Output

The input waveform shows the two input signals applied through the ADC bridge: v("a_in") and v("b_in") (offset by +5 V for clarity of display), each toggling between logic low and logic high at different intervals to generate all four possible input combinations.

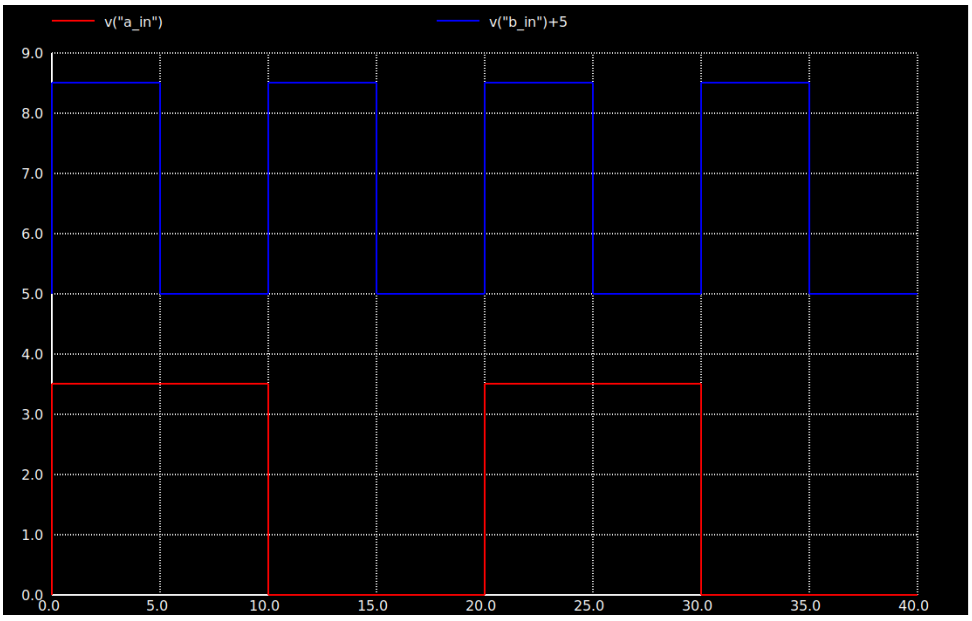

Figure 4.24: Input Waveform of 74HC386

The resulting output waveform, v("1y"), remained high whenever the two inputs differed in logic state, and dropped to logic low whenever both inputs were equal (either both low or both high). This matched the expected XOR truth table behavior, confirming that the developed subcircuit model correctly replicated the functionality of the 74HC386.

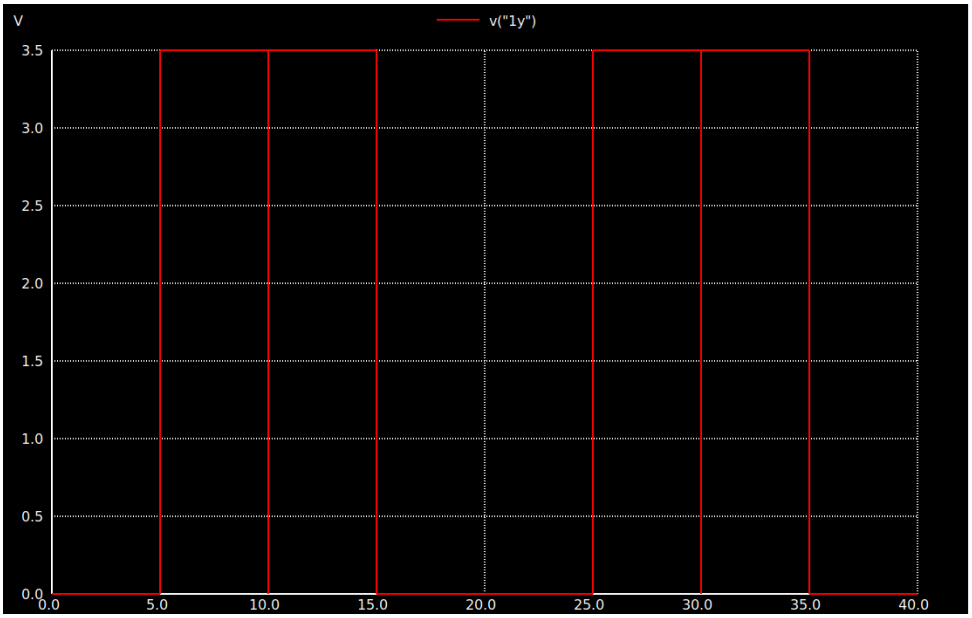

Figure 4.25: Output Waveform of 74HC386

4.6 74HC158 - Quad 2-to-1 Multiplexer

4.6.1 Description

The 74HC158 is a quad 2-input inverting multiplexer IC belonging to the high-speed CMOS (HC) logic family. It contains four independent 2:1 multiplexers sharing a common select line (S) and a common active-low enable (E_bar), each producing an inverted version of the selected data input at its output. Being part of the HC family, the device offers CMOS-level power efficiency while maintaining speed performance comparable to standard TTL logic, and is commonly used wherever data selection, routing, or boolean function generation is required.

4.6.2 Pin Diagram

The 74HC158 is available in a 14-pin dual in-line package, with a common select input (S), a common active-low enable (E_bar), eight data input pins, and four inverted output pins arranged so that each multiplexer channel has two data inputs and one output.

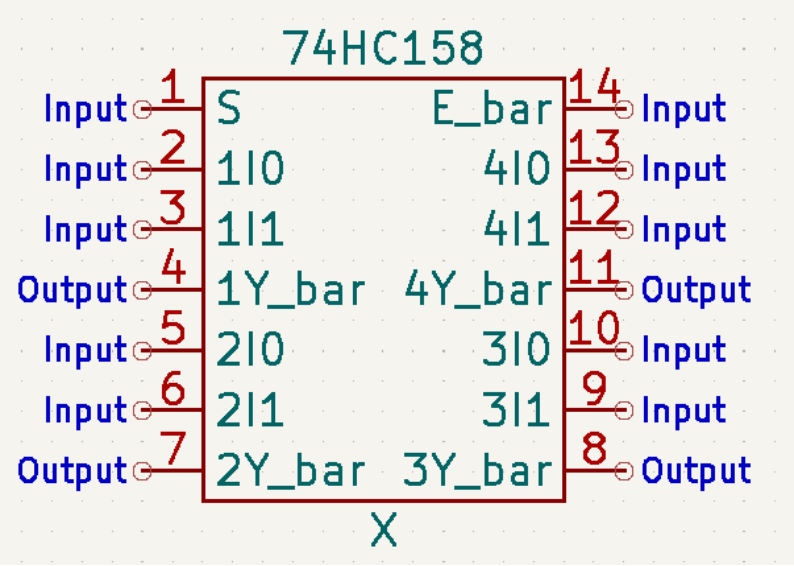

Figure 4.26: Pin Diagram of 74HC158

4.6.3 Features

- Four independent 2-input multiplexers with a common select line.
- Active-low enable (E_bar) forces all four outputs high when disabled.
- High-speed CMOS (HC) logic family with low static power consumption.
- Inverted (active-low) outputs at each of the four channels.
- Compatible with other standard CMOS/HC family devices.

4.6.4 Applications

- Data selection and routing in bit-parallel buses.
- Boolean function generation using the select/enable control.
- Bit-slice input selection in ALUs and data path logic.
- Multiplexed addressing in memory and I/O interfacing.

4.6.5 Subcircuit Design

The internal architecture of the 74HC158 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The common select line (S) and active-low enable (E_bar), along with each channel's data inputs, were first conditioned through `d_inverter` stages to generate the complementary signals needed for selection. These complemented signals were then combined using `d_and` gates

to realize the AND-OR select logic for each channel in NAND-NAND (De Morgan equivalent) form, which directly produces the inverted output required by the device without an additional output inverter. Each channel's combined logic was passed through a `d_buffer` stage before reaching its output pin. The four such arrangements were mapped to their respective input/output ports (1I0/1I1/1Y_bar, 2I0/2I1/2Y_bar, 3I0/3I1/3Y_bar, and 4I0/4I1/4Y_bar) and encapsulated together, along with the shared S and E_bar lines, as a single reusable subcircuit representing the complete 74HC158 package.

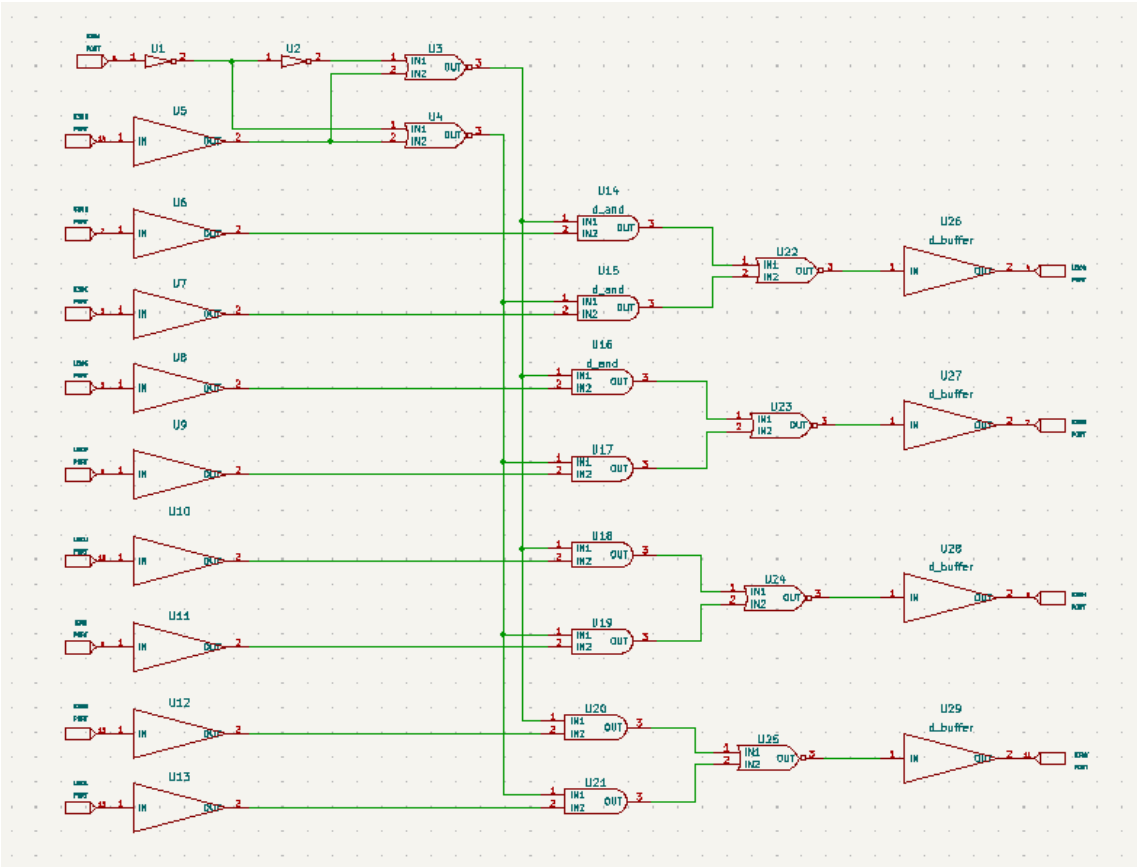

Figure 4.27: Subcircuit Schematic of 74HC158

4.6.6 Test Circuit

Since the 74HC158 subcircuit was built using digital gate primitives, an `adc_bridge_4` was used at the input stage to convert the analog voltage signals representing the select line (S), the two data inputs of channel 1 (1I0, 1I1), and the enable input (E_bar) into digital logic levels compatible with the subcircuit's inputs, and a `dac_bridge_1` was used at the output stage to convert the digital output (1Y_bar) back into an analog voltage signal for observation. Since all four channels operate identically and independently, only channel 1 was tested to verify multiplexer functionality. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

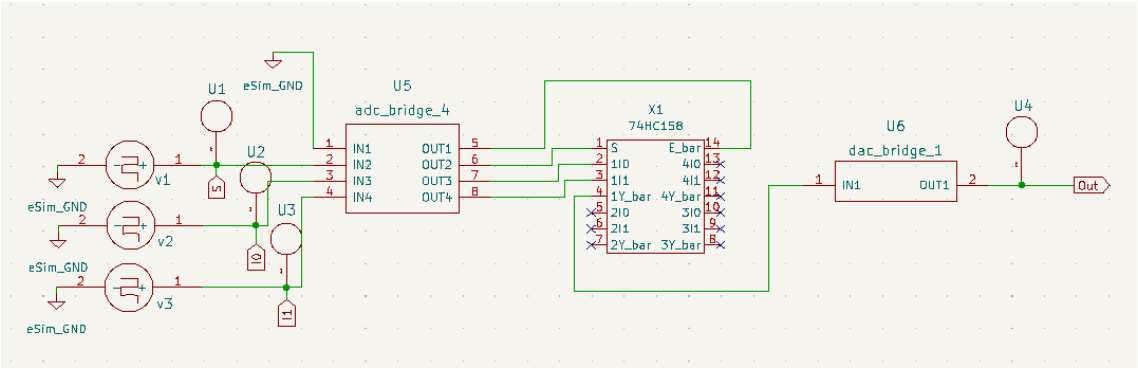

Figure 4.28: Test Circuit of 74HC158

4.6.7 Output

The input waveform shows the select and data signals applied through the ADC bridge: v("i0") and v("i1") (offset by +2 V for clarity of display), each toggling at different intervals along with the select line to generate all combinations of channel-1 select and data states.

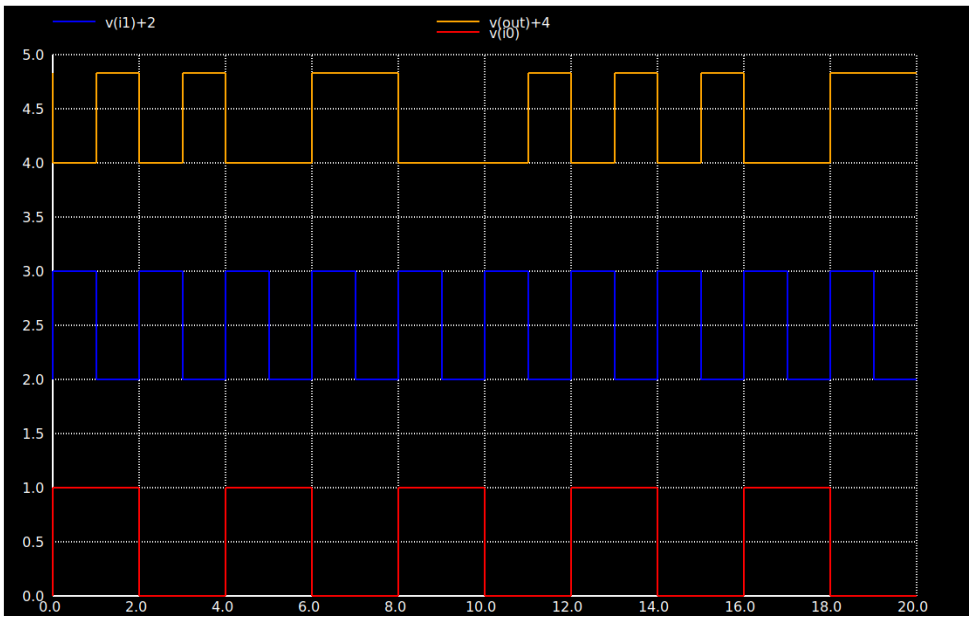

Figure 4.29: Input Waveform of 74HC158

The resulting output waveform, v("out"), shows the inverting-multiplexer behavior clearly: the output goes low whenever the selected data input is high, and high whenever the selected data input is low, confirming correct 2:1 multiplexing and inversion functionality. The output swing observed through the `dac_bridge_1` spans approximately 0–830 mV, consistent with the DAC bridge's default output scaling for this configuration.

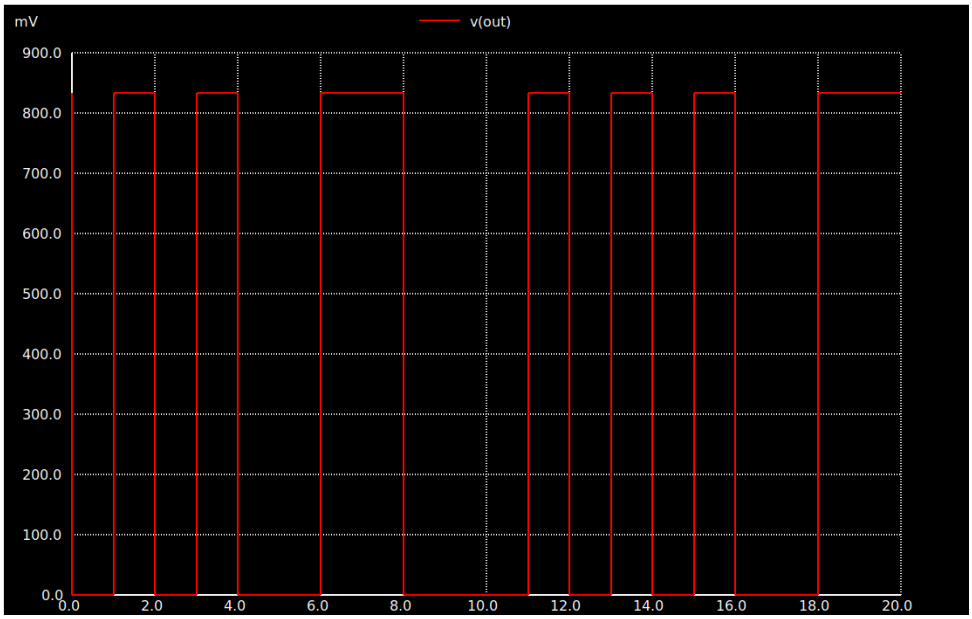

Figure 4.30: Output Waveform of 74HC158

4.7 74HC251 – 8-to-1 Multiplexer

4.7.1 Description

The 74HC251 is an 8-to-1 multiplexer IC belonging to the high-speed CMOS (HC) logic family. It selects one of eight data inputs (I0-I7) based on a 3-bit select code (S0, S1, S2) and routes it to both a true output (Y) and a complementary output (Y_bar), with an active-low output enable (OE_bar) that tri-states/disables the outputs when high. Being part of the HC family, the device offers CMOS-level power efficiency while maintaining speed performance comparable to standard TTL logic, and is commonly used wherever wide data selection or bus-routing logic is required.

4.7.2 Pin Diagram

The 74HC251 is available in a 16-pin dual in-line package, with eight data input pins (I0-I7), three select input pins (S0, S1, S2), one active-low output enable pin (OE_bar), and two complementary output pins (Y, Y_bar).

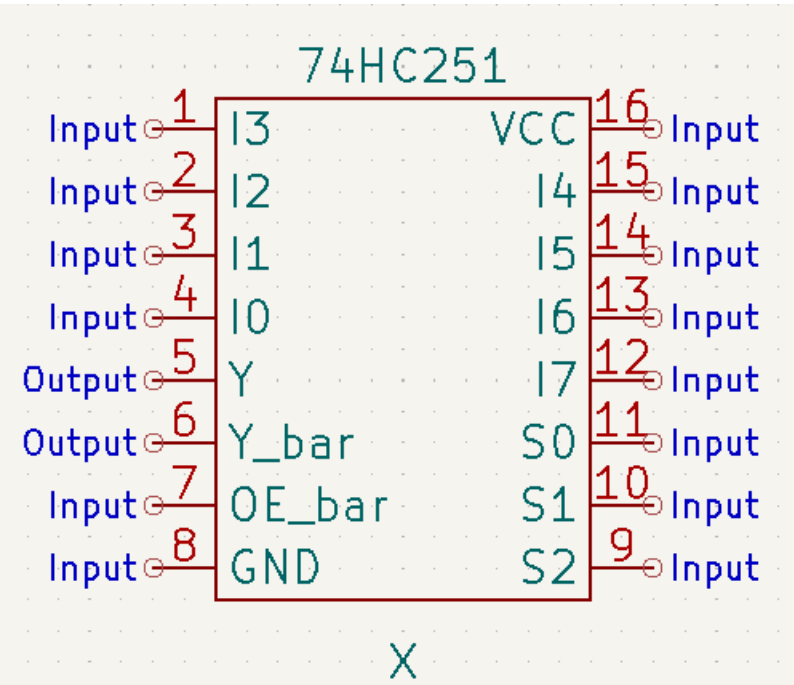

Figure 4.31: Pin Diagram of 74HC251

4.7.3 Features

- Eight-input, single-output multiplexing on one shared 3-bit select code.
- Active-low output enable (OE_bar) for tri-state/disable control.
- Both true (Y) and complementary (Y_bar) outputs provided.
- High-speed CMOS (HC) logic family with low static power consumption.
- Compatible with other standard CMOS/HC family devices.

4.7.4 Applications

- Wide data selection in bit-parallel buses.
- Parallel-to-serial data conversion.
- Boolean function generation using select lines as address inputs.
- Bus routing and channel selection in data path logic.

4.7.5 Subcircuit Design

The internal architecture of the 74HC251 was implemented in eSim using digital gate primitives rather than transistor-level modeling. Each of the eight data inputs (I0-I7) was first conditioned through a `d_inverter` stage. The three select lines (S0, S1, S2) were also passed through `d_inverter` stages to generate their complementary signals. Each data input was then combined with the appropriate combination of select/select-complement signals using AND gates, forming eight AND terms corresponding to each of the eight select codes. These eight AND terms were combined in pairs using OR gates, and the resulting partial sums were further combined down to a single line representing the selected data value. This combined signal was passed through a final inverter/buffer stage to generate the true output (Y) and a separate buffer stage to generate the complementary output (Y_bar), with the output-enable (OE_bar) logic gating both output stages. The complete arrangement was mapped to the I0-I7, S0-S2, OE_bar, Y, and Y_bar ports and encapsulated as a single reusable subcircuit representing the complete 74HC251 package.

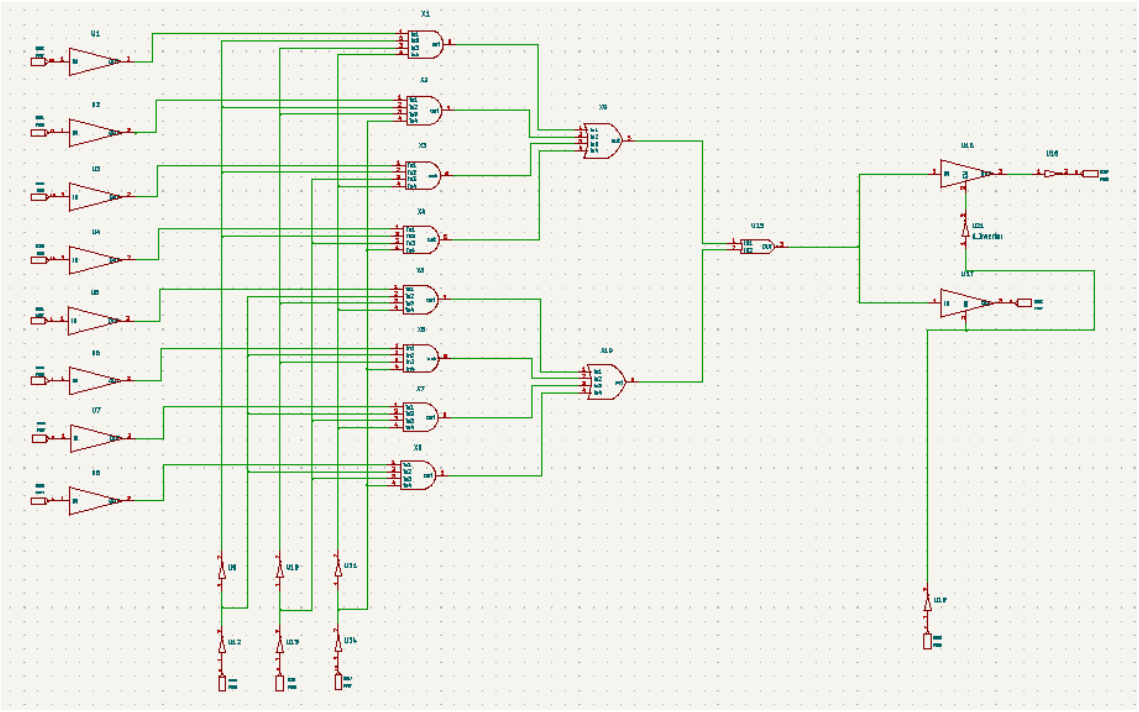

Figure 4.32: Subcircuit Schematic of 74HC251

4.7.6 Test Circuit

Since the 74HC251 subcircuit was built using digital gate primitives, an `adc_bridge_2` was used at the input stage to convert the analog pulse signals from two voltage sources (v1, v2) into digital logic levels for two of the data inputs (I0, I1), while the remaining data inputs (I2-I7) were tied to fixed logic levels. The select lines (S0, S1, S2) were driven directly by stepped voltage sources (v3, v4, v5) to sequence through the select codes, and the output-enable (OE_bar) line was tied active. The Y and Y_bar outputs were probed directly from the device pins.

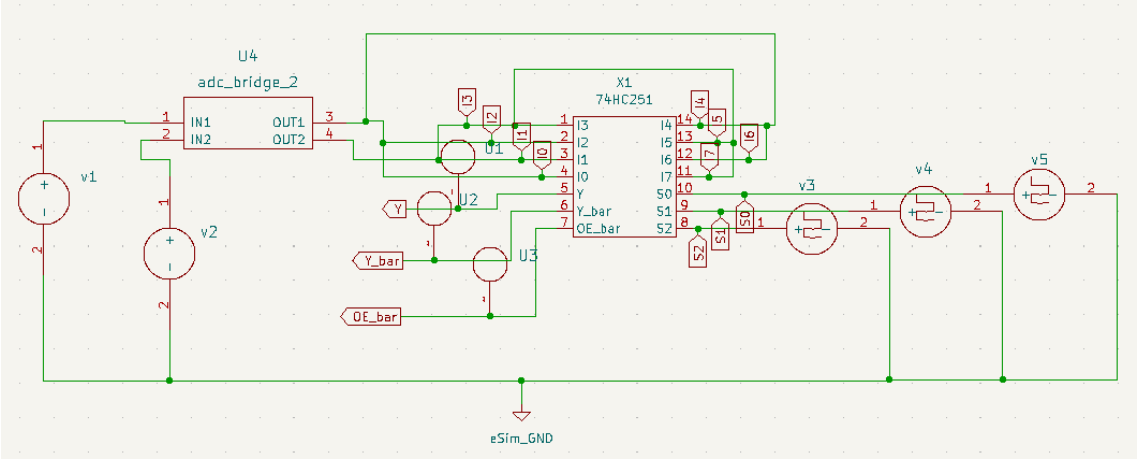

Figure 4.33: Test Circuit of 74HC251

4.7.7 Output

4.7.8 Output

The input waveform capture shows the data inputs and select lines used to exercise the multiplexer. The even-indexed inputs (I0, I2, I4, I6) are held at one fixed logic level, and the odd-indexed inputs (I1, I3, I5, I7) are held at the other fixed logic level, so alternating channels present opposite logic states to the multiplexer. As the select lines step through their combinations over the simulation window to sequence through all eight select codes, v("out")+4 tracks the corresponding output response, moving between the two fixed levels in sync with the select code parity.

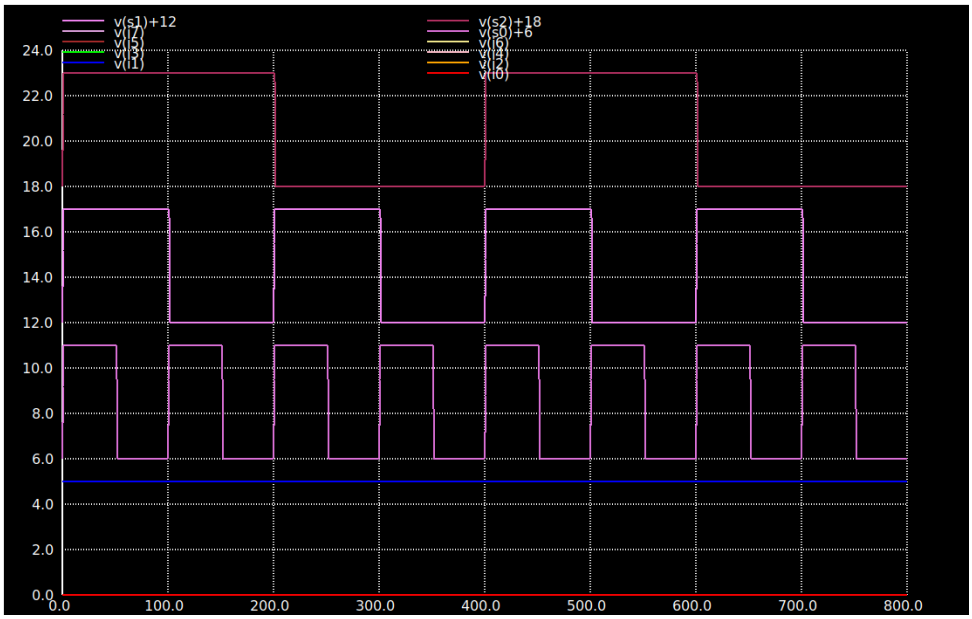

Figure 4.34: Input Waveform of 74HC251

The resulting output waveforms, v("y")+5 and v("y_bar"), track each other as complements throughout the simulation, with v("y") going high and v("y_bar") go-

ing low whenever the multiplexer selects a high data input, and the reverse whenever it selects a low data input.

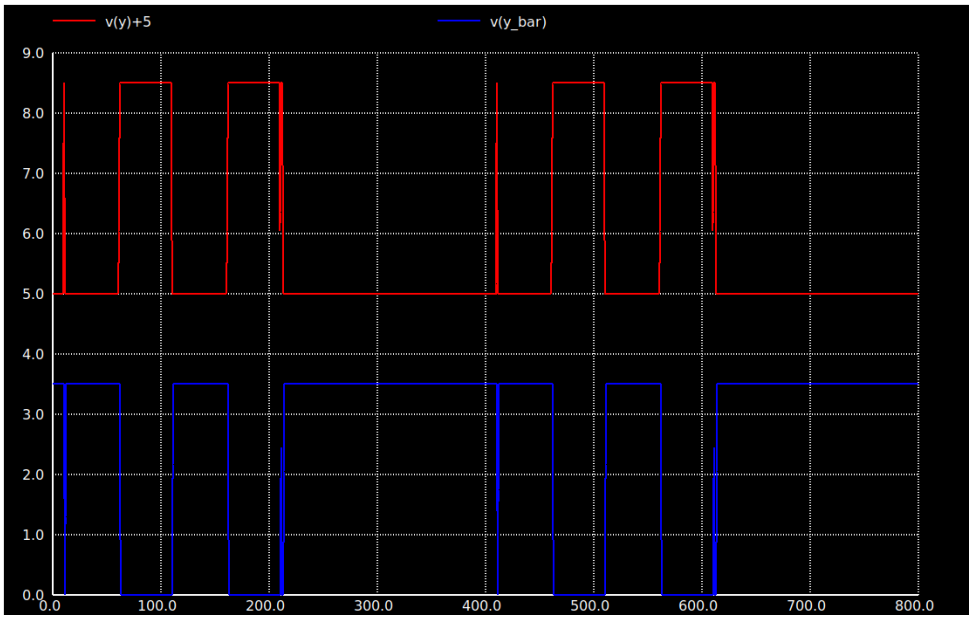

Figure 4.35: Output Waveform of 74HC251

4.8 74S137 – 3-to-8 Decoder/Demultiplexer

4.8.1 Description

The 74S137 is a 3-to-8 line decoder/demultiplexer IC belonging to the Schottky TTL (S) logic family. It decodes a 3-bit binary address (A0, A1, A2) into one of eight mutually exclusive, active-low outputs (O0'-O7'), with three enable inputs (LE', E1', E2) that together control whether the decoder is active and also allow the device to double as a demultiplexer with the address lines acting as a latch when disabled. Being part of the Schottky TTL family, the device offers fast switching speeds suited to address decoding and data routing in high-speed digital systems.

4.8.2 Pin Diagram

The 74S137 is available in a 16-pin dual in-line package, with three address input pins (A0, A1, A2), three enable input pins (LE', E1', E2), and eight active-low output pins (O0'-O7').

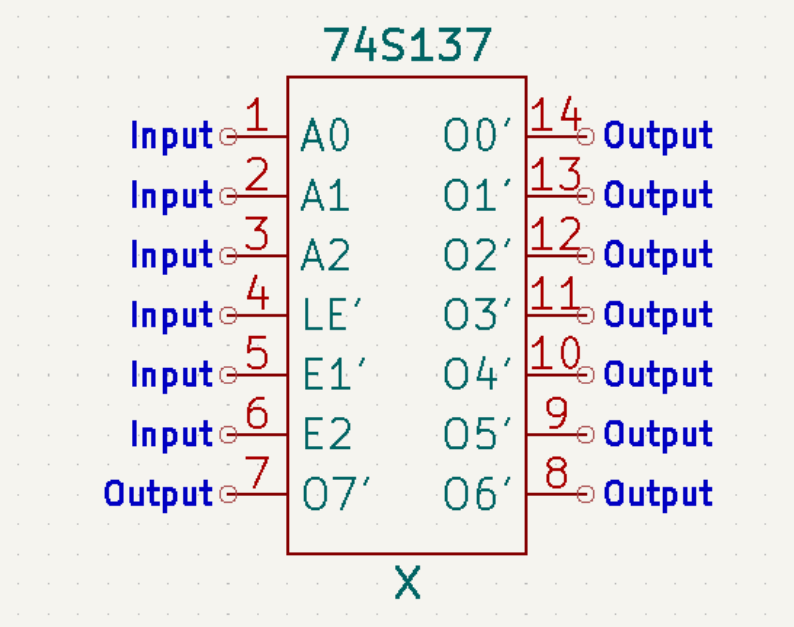

Figure 4.36: Pin Diagram of 74S137

4.8.3 Features

- Decodes a 3-bit binary address into one of eight active-low outputs.
- Three enable inputs (LE', E1', E2) for cascading and demultiplexing.
- Schottky TTL (S) logic family for high-speed switching.
- Only the selected output goes low; all others remain high.
- Compatible with other standard TTL/S family devices.

4.8.4 Applications

- Address decoding in memory and I/O systems.
- Data demultiplexing from a single line to multiple destinations.
- Chip-select generation in multi-device bus systems.
- Boolean function generation using the address lines.

4.8.5 Subcircuit Design

The internal architecture of the 74S137 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The three address lines (A0, A1, A2) and the enable inputs were each passed through `d_inverter` stages to generate their complementary signals. The enable inputs were first combined together using

a `d_and` gate to form a single internal enable term. This enable term was then combined with each of the eight possible combinations of the address lines and their complements using `d_and` gates, producing eight AND terms, one for each output address code. Each of these AND terms was passed through a final `d_nand`-based inverting stage to produce the active-low output behavior, so that only the AND term matching the current address and enable condition drives its corresponding output low while all others remain high. The eight such stages were mapped to the O0'-O7' output ports and, together with the A0, A1, A2, LE', E1', and E2 input ports, encapsulated as a single reusable subcircuit representing the complete 74S137 package.

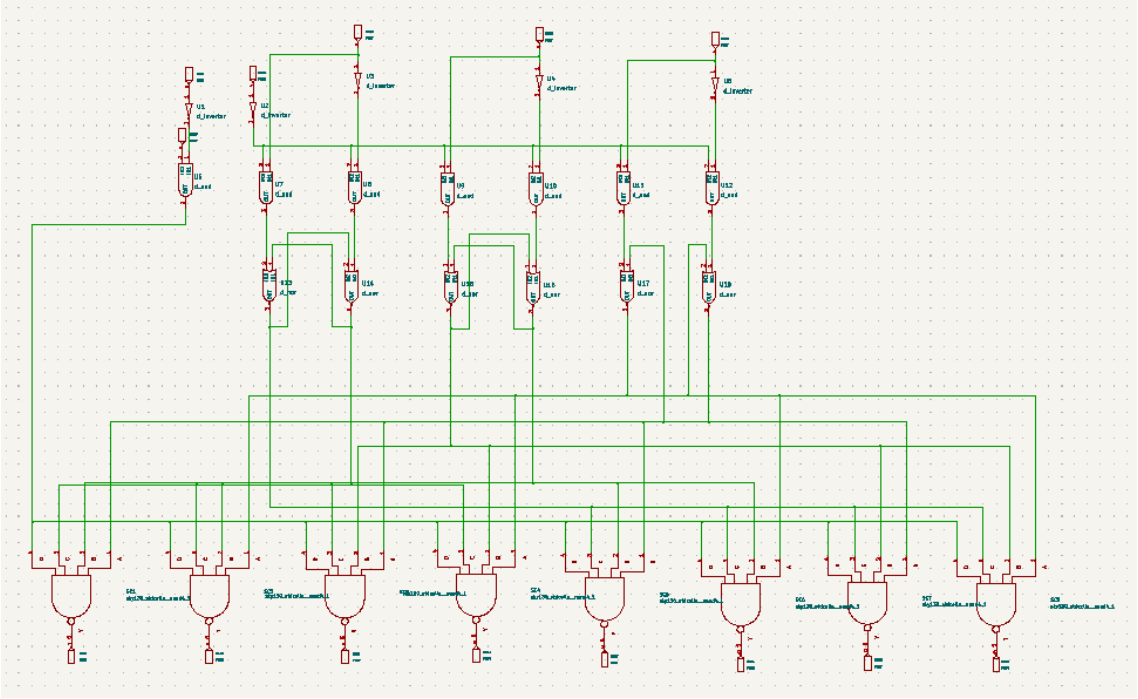

Figure 4.37: Subcircuit Schematic of 74S137

4.8.6 Test Circuit

Since the 74S137 subcircuit was built using digital gate primitives, an `adc_bridge_3` was used at the input stage to convert the analog voltage signals from three pulse sources (v2, v3, v4) into digital logic levels for the address lines A0, A1, and A2, with the enable inputs (LE', E1', E2) tied active through a fixed voltage source (v1). A `dac_bridge_8` was used at the output stage to convert all eight digital outputs (O0'-O7') back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

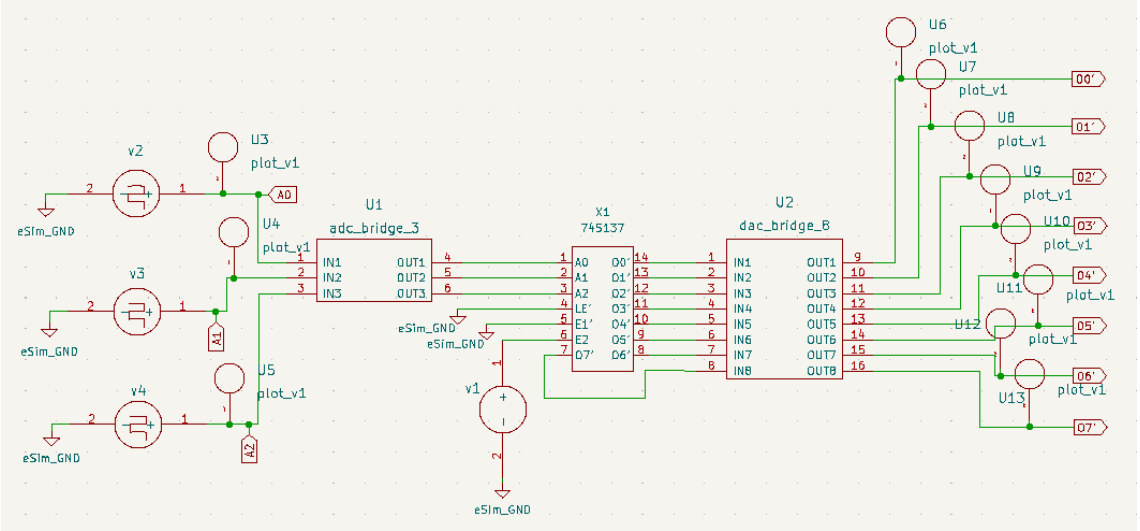

Figure 4.38: Test Circuit of 74S137

4.8.7 Output

The input waveform shows the three address signals applied through the ADC bridge: v("a2"), v("a1")+6, and v("a0")+12, together stepping through a binary count sequence so that all eight address combinations are exercised in turn over the simulation window.

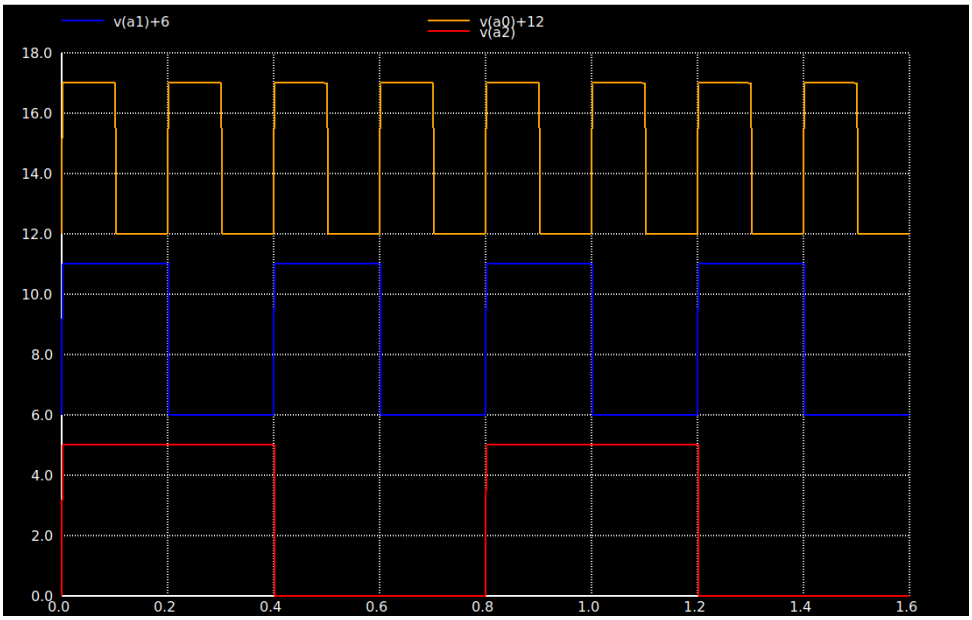

Figure 4.39: Input Waveform of 74S137

The resulting output waveforms, v("o0") through v("o7")+42 (each offset for clarity of display), remain high by default and each dip low only during the single time slice corresponding to its own address code, while all other outputs stay high during that interval. This one-hot active-low behavior confirms that the developed subcircuit model correctly replicates the decoding functionality of the 74S137.

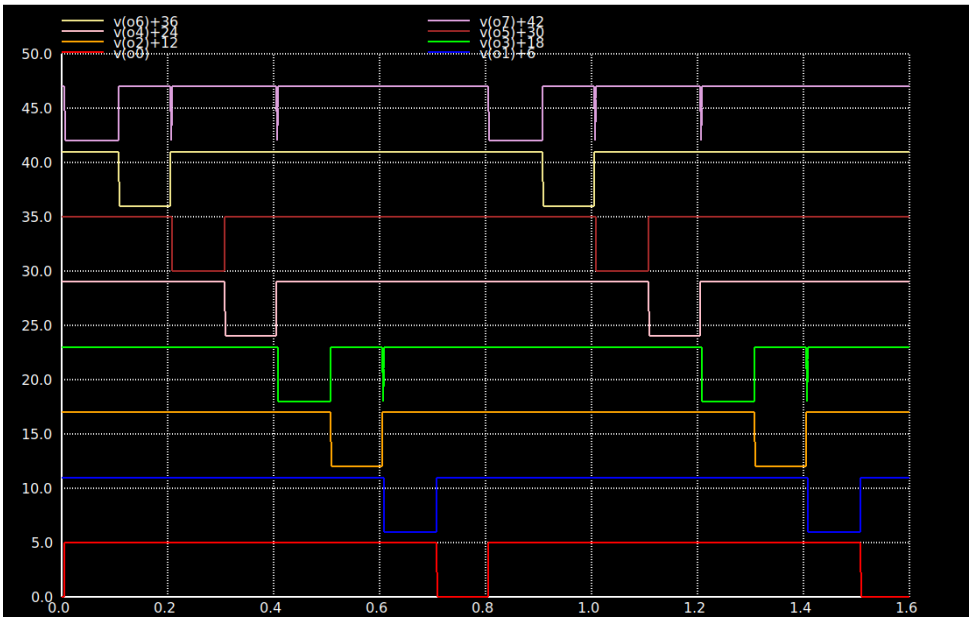

Figure 4.40: Output Waveform of 74S137

4.9 74164 – 8-bit Serial-In Parallel-Out Shift Register

4.9.1 Description

The 74164 is an 8-bit serial-in, parallel-out shift register IC belonging to the standard TTL logic family. Data is entered serially through two AND-gated inputs (Dsa, Dsb), so that a logic high on both inputs is required to shift a high into the register on each rising clock edge (CP). The device provides eight parallel outputs (Q0-Q7) reflecting the current contents of the register, and an asynchronous active-low master reset (MR') that clears all outputs to logic low independent of the clock. It is commonly used wherever serial-to-parallel data conversion is required.

4.9.2 Pin Diagram

The 74164 is available in a 14-pin dual in-line package, with two serial data input pins (Dsa, Dsb), a clock input (CP), an active-low master reset input (MR'), and eight parallel output pins (Q0-Q7).

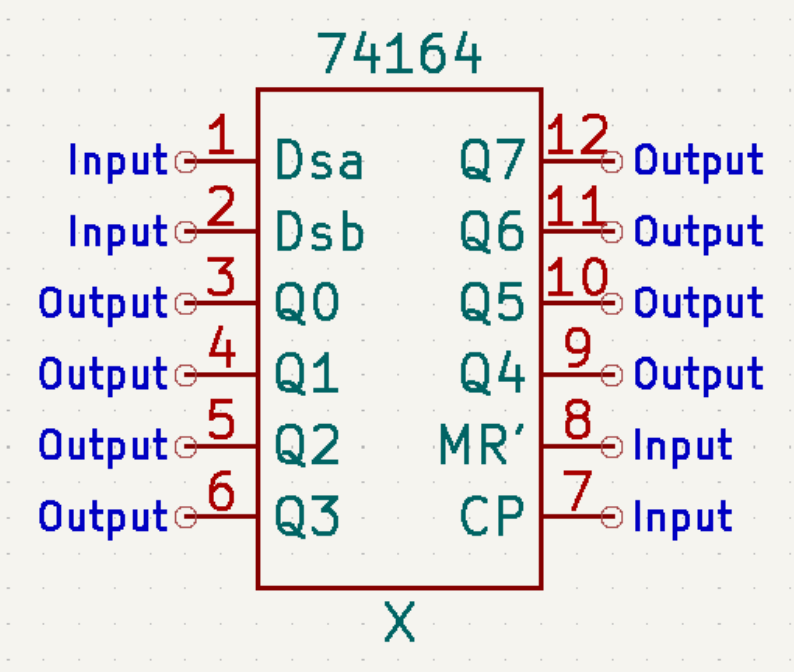

Figure 4.41: Pin Diagram of 74164

4.9.3 Features

- 8-bit serial-in, parallel-out shift register.
- Two AND-gated serial data inputs (Dsa, Dsb) for gating data entry.
- Asynchronous active-low master reset (MR') independent of the clock.
- Data shifts on the rising edge of the clock (CP).
- Standard TTL logic family, compatible with other TTL devices.

4.9.4 Applications

- Serial-to-parallel data conversion.
- Time-delay and sequencing circuits.
- Pattern generation and Johnson/ring counter construction.
- Data buffering between serial communication links and parallel buses.

4.9.5 Subcircuit Design

The internal architecture of the 74164 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The two serial data inputs (Dsa, Dsb) were combined using a `d_and` gate to form the effective serial input to the register, matching the AND-gated data entry of the original device. This combined signal was fed into the first of eight cascaded `d_ff` stages, with the Q output of each flip-flop connected to the D input of the next, forming the 8-bit shift chain. The clock (CP) and master reset (MR') lines were each conditioned through `d_inverter` stages before being distributed to the clock and reset inputs of all eight flip-flops, ensuring consistent active-edge and active-level behavior across the chain. The Q output of each flip-flop was mapped to its corresponding output port (Q0-Q7), and the complete arrangement was encapsulated as a single reusable subcircuit representing the complete 74164 package.

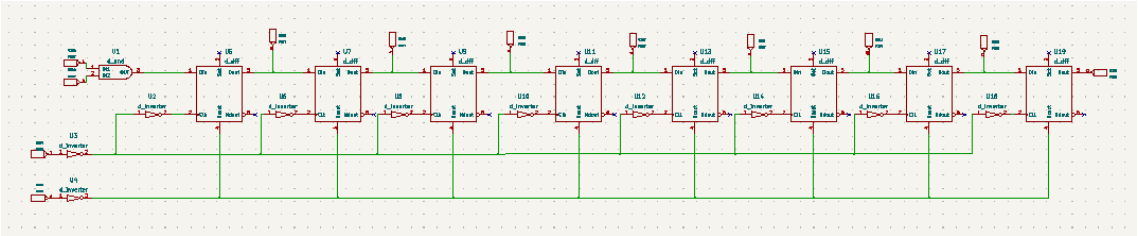

Figure 4.42: Subcircuit Schematic of 74164

4.9.6 Test Circuit

Since the 74164 subcircuit was built using digital gate primitives, an `adc_bridge_4` was used at the input stage to convert the analog voltage signals from four sources (v1-v4) into digital logic levels for the Dsa, Dsb, MR', and CP inputs, and a `dac_bridge_8` was used at the output stage to convert all eight digital outputs (Q0-Q7) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

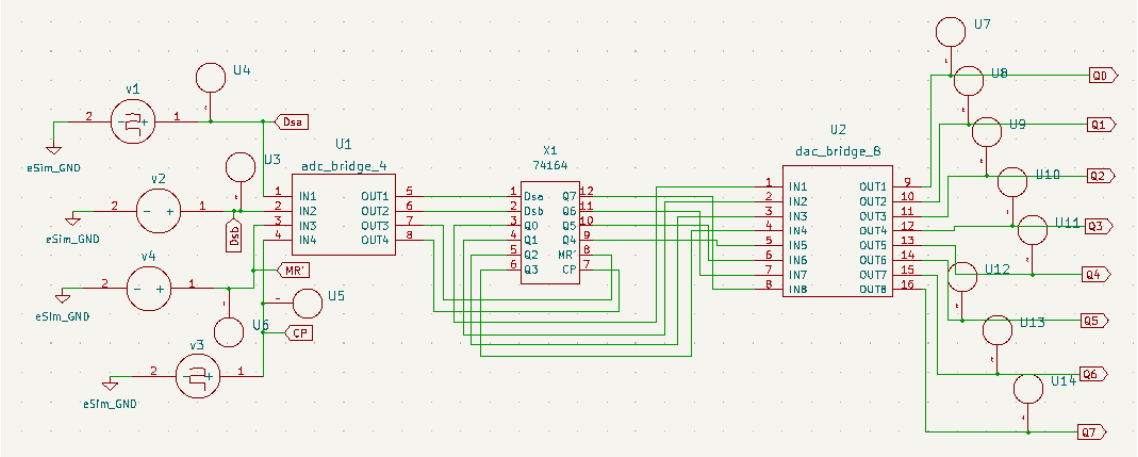

Figure 4.43: Test Circuit of 74164

4.9.7 Output

The input waveform shows the signals applied through the ADC bridge: v("Dsb") and v("Dsa")+6 toggling between logic low and logic high at regular intervals to present a serial data pattern, v("CP")+12 providing the repeating clock pulse train, and v("MRn")+18 held high throughout to keep the master reset inactive.

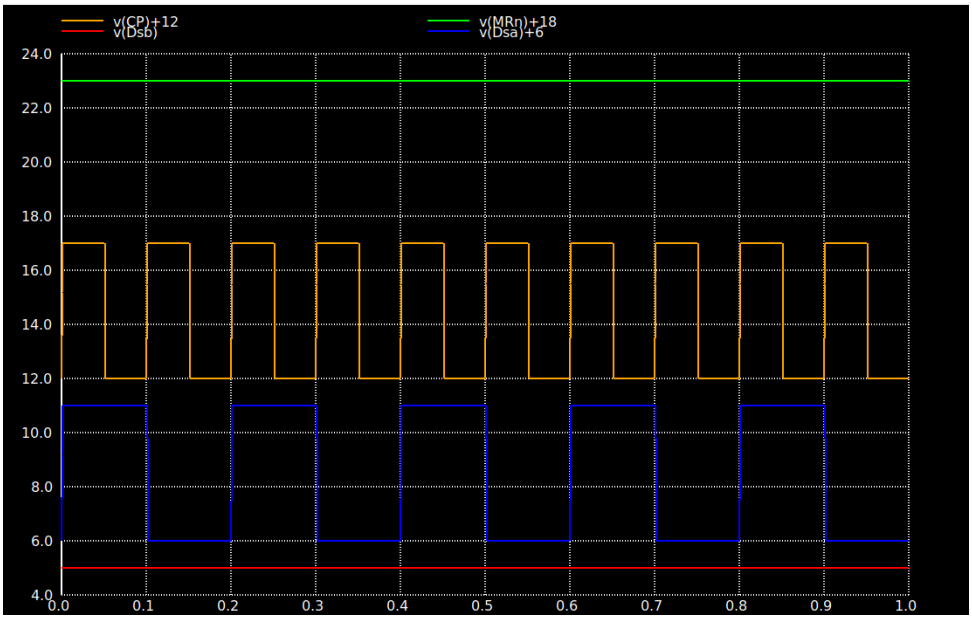

Figure 4.44: Input Waveform of 74164

The resulting output waveforms, v("Q0") through v("Q7")+28 (each offset for clarity of display), show the serial input pattern progressively shifting through the register one stage at a time with each clock pulse, with each output transitioning to reflect the value held by the previous stage on the prior clock edge. This confirms that the developed subcircuit model correctly replicates the serial-in, parallel-out shifting functionality of the 74164.

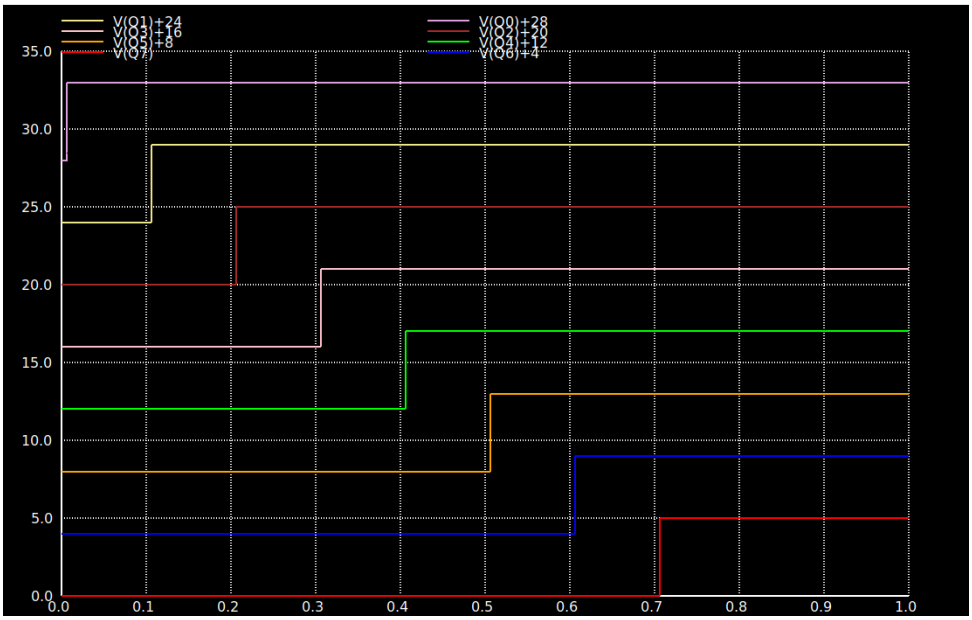

Figure 4.45: Output Waveform of 74164

4.10 74LS393 – Dual 4-Bit Binary Counter

4.10.1 Description

The 74LS393 is a dual 4-bit binary ripple counter IC belonging to the low-power Schottky TTL (LS) logic family. It contains two independent 4-bit counters, each with its own clock input (CP_a', CP_b') and asynchronous reset (MR_a, MR_b), so that a single package can provide two separate binary counting channels. Each counter's four outputs (Q0-Q3) increment in a standard binary ripple sequence on each clock edge, with Q0 toggling fastest and each subsequent bit toggling at half the rate of the one before it. Being part of the LS family, the device offers reduced power consumption compared to standard TTL while retaining familiar TTL-level compatibility.

4.10.2 Pin Diagram

The 74LS393 is available in a 14-pin dual in-line package, with each half of the device providing a clock input, a reset input, and four binary counter outputs (Q0-Q3), for a total of two independent 4-bit counters.

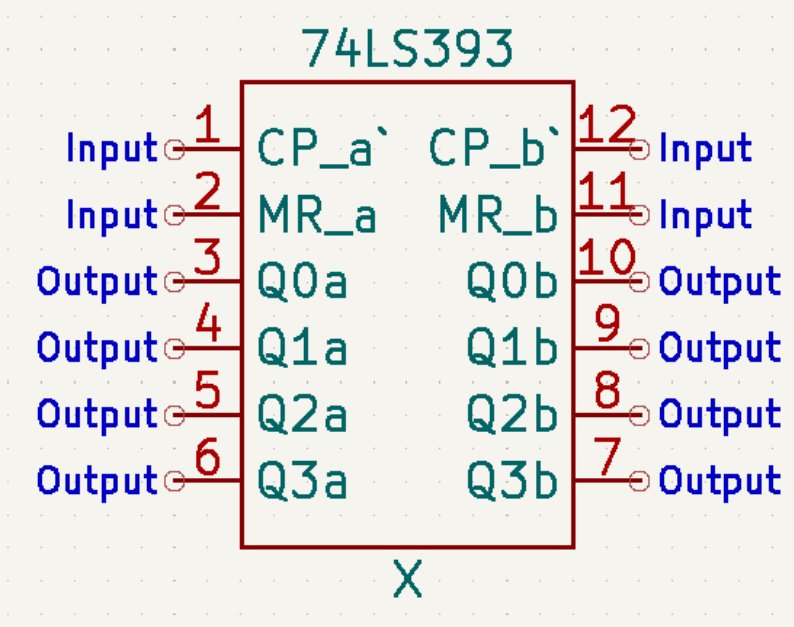

Figure 4.46: Pin Diagram of 74LS393

4.10.3 Features

- Dual independent 4-bit binary ripple counters in a single package.
- Separate clock and reset inputs for each counter section.
- Standard binary ripple-carry counting sequence on each output stage.
- Low-power Schottky TTL (LS) logic family.
- Compatible with other standard TTL/LS family devices.

4.10.4 Applications

- Frequency division and clock scaling.
- Event counting and timing circuits.
- Address generation in memory and sequencing systems.
- Dual independent counting channels in a single package for compact designs.

4.10.5 Subcircuit Design

The internal architecture of the 74LS393 was implemented in eSim using digital gate primitives rather than transistor-level modeling. Each 4-bit counter section was realized using four cascaded `d_jkff` stages configured in toggle mode, with the output of each flip-flop driving the clock input of the next stage to form the ripple-carry

chain. The clock input of each section was conditioned through a `d_inverter` stage before reaching the first flip-flop, and the reset line of each section was distributed to the reset input of all four flip-flops in that section, providing the asynchronous clear behavior. The Q output of each flip-flop was mapped to its corresponding output port (Q0-Q3), and this arrangement was duplicated for both the a-side and b-side counters. The two independent 4-bit sections were encapsulated together as a single reusable subcircuit representing the complete 74LS393 package.

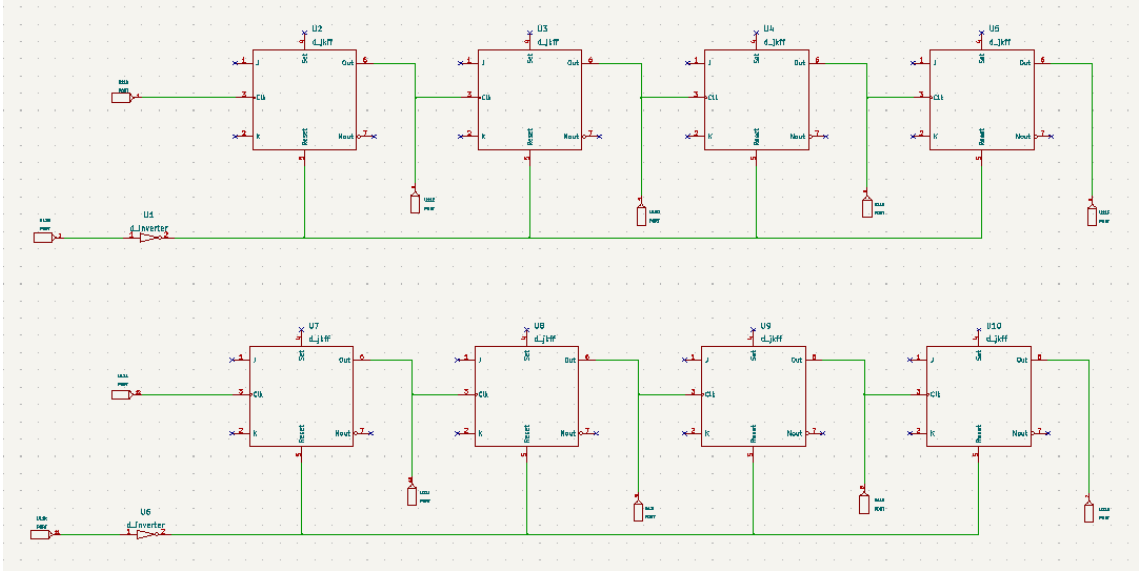

Figure 4.47: Subcircuit Schematic of 74LS393

4.10.6 Test Circuit

Since the 74LS393 subcircuit was built using digital gate primitives, an `adc_bridge_2` was used at the input stage to convert the analog voltage signals from two sources (v1, v2) into digital logic levels for the clock (CP_bar) and reset (MR) lines, with both signals distributed to the corresponding a-side and b-side clock and reset pins so that both counter sections are driven identically. A `dac_bridge_8` was used at the output stage to convert all eight digital outputs (Q0a-Q3a and Q0b-Q3b) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

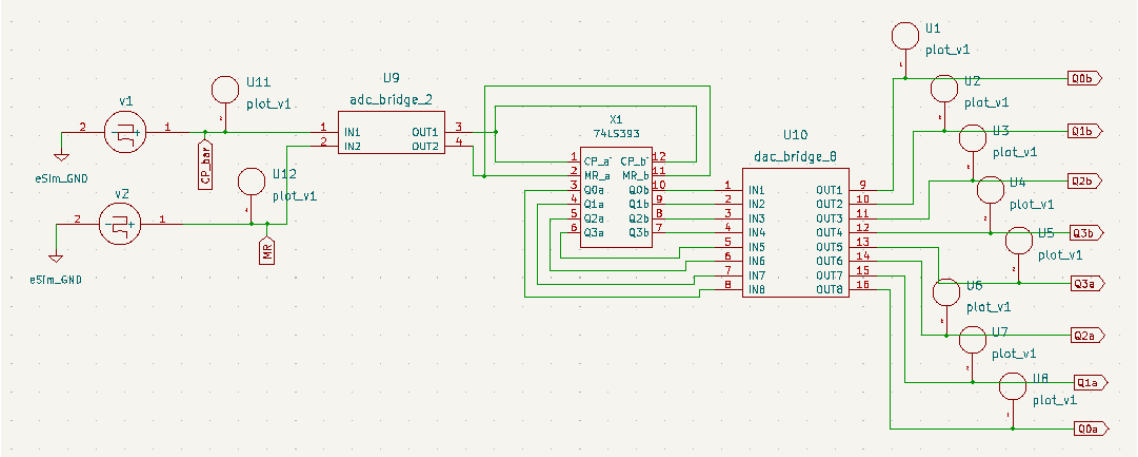

Figure 4.48: Test Circuit of 74LS393

4.10.7 Output

The input waveform shows the signals applied through the ADC bridge: v("cp_in") providing a continuous clock pulse train, and v("mr_in")+6 held low throughout, keeping the reset inactive for both counter sections over the simulation window.

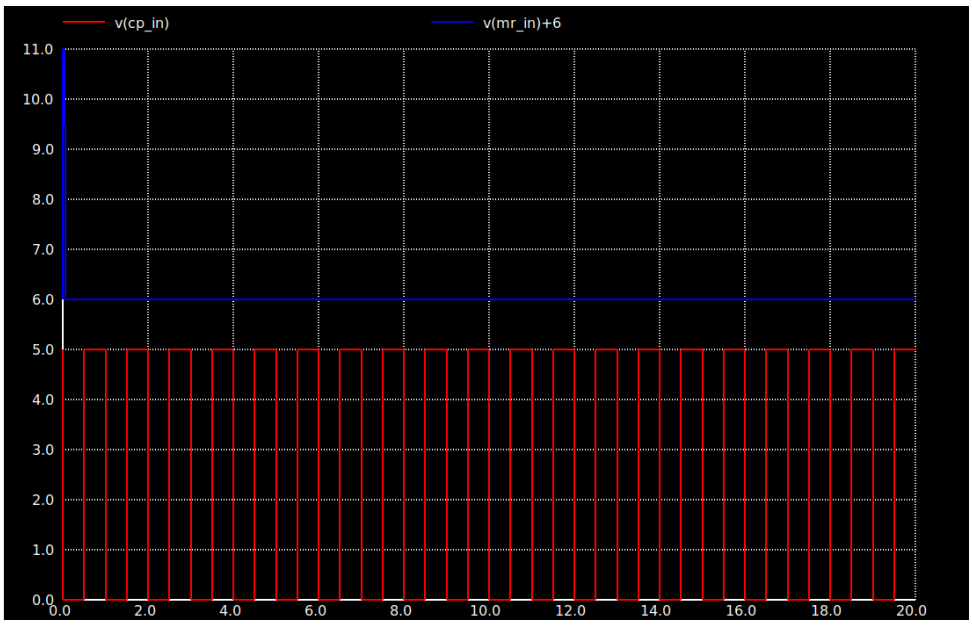

Figure 4.49: Input Waveform of 74LS393

Since both counter sections share the same clock and reset signals in this test, the a-side and b-side outputs follow identical patterns. The a-side output waveforms, v("q0a_o")+18, v("q1a_o")+12, v("q2a_o")+6, and v("q3a_o"), show the expected binary ripple sequence: Q0a toggles on every clock pulse, Q1a toggles at half that rate, Q2a at half the rate of Q1a, and Q3a at half the rate of Q2a, confirming correct 4-bit binary counting behavior.

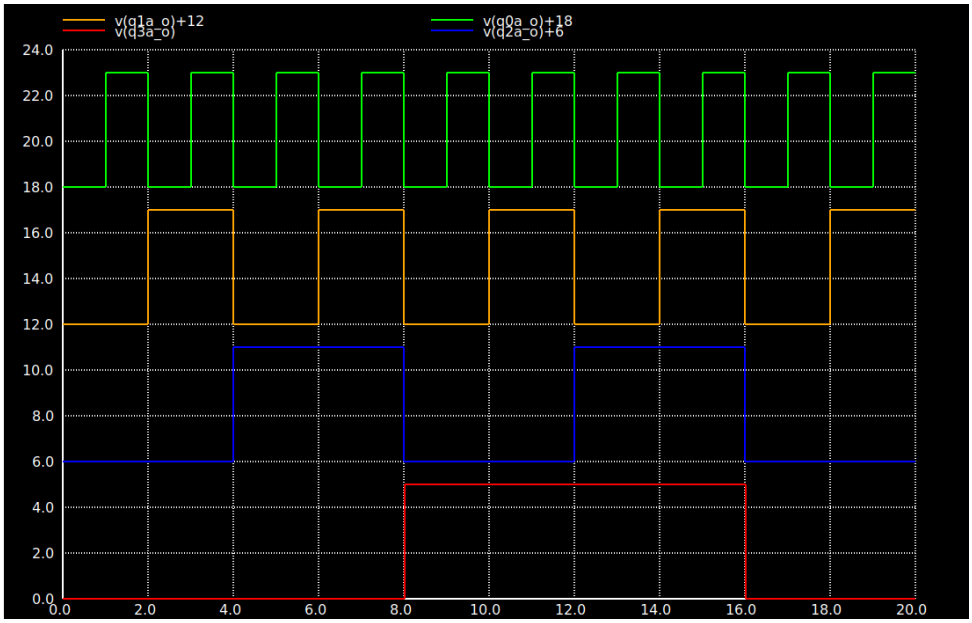

Figure 4.50: Output Waveform of 74LS393 (a-side counter)

The b-side output waveforms, v("q0b_o")+18, v("q1b_o")+12, v("q2b_o")+6, and v("q3b_o"), reproduce the same binary ripple sequence as the a-side counter, confirming that the second independent counter section within the subcircuit operates identically and correctly.

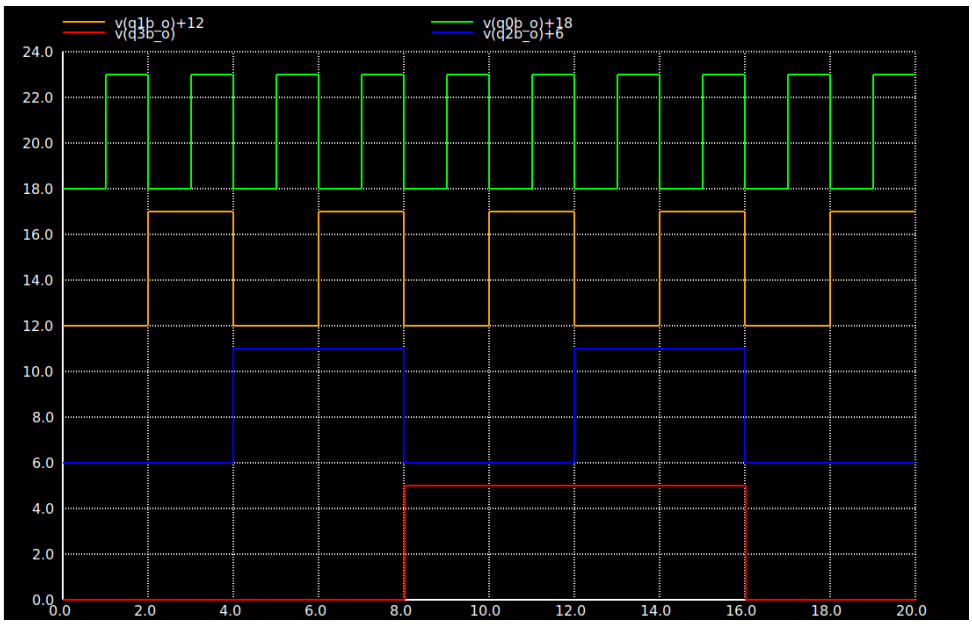

Figure 4.51: Output Waveform of 74LS393 (b-side counter)

4.11 74LS197 – 4-Bit Binary Ripple Counter with Parallel Load

4.11.1 Description

The 74LS197 is a 4-bit binary ripple counter IC belonging to the low-power Schottky TTL (LS) logic family. It provides four outputs (Q0-Q3) that count in a standard binary ripple sequence, driven by two clock inputs (CP0_b, CP1_b), and supports parallel loading of external data (D0-D3) into the counter via an active-low parallel load control (PL_b). An asynchronous active-low master reset (MR_b) clears all outputs independent of the clock. It is commonly used wherever binary counting with the ability to preset a starting count value is required.

4.11.2 Pin Diagram

The 74LS197 is available in a 14-pin dual in-line package, with an active-low parallel load input (PL_b), an active-low master reset input (MR_b), four parallel data inputs (D0-D3), two clock inputs (CP0_b, CP1_b), and four counter outputs (Q0-Q3).

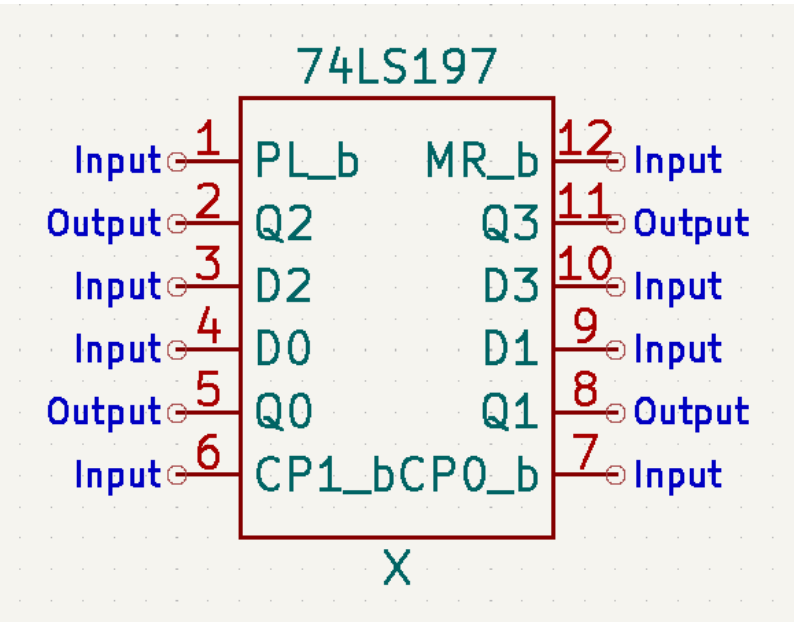

Figure 4.52: Pin Diagram of 74LS197

4.11.3 Features

- 4-bit binary ripple counter with parallel load capability.
- Active-low parallel load (PL_b) for presetting the count value.
- Asynchronous active-low master reset (MR_b) independent of the clock.
- Dual clock inputs (CP0_b, CP1_b) for flexible clocking of the counter stages.

- Low-power Schottky TTL (LS) logic family.

4.11.4 Applications

- Programmable binary counting with a presettable start value.
- Frequency division and clock scaling.
- Address generation and sequencing in digital systems.
- Timing and event-counting circuits requiring a resettable, loadable counter.

4.11.5 Subcircuit Design

The internal architecture of the 74LS197 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The clock input was first conditioned through a `d_buffer` stage, while the parallel load control (PL_b) was processed through a combination of `d_inverter` and `d_or` gates to generate the internal load-enable signal. Each of the four counter bits was implemented using a `d_jkff` stage preceded by `d_nand` gates, which combined the corresponding data input (D0-D3) with the load-enable signal and the toggle feedback from the flip-flop's own output, allowing each stage to either load its parallel data input or toggle based on the incoming clock, depending on the state of PL_b. The output of each flip-flop was cascaded to the clock input of the next stage to realize the ripple-carry counting behavior, and the master reset (MR_b) line was distributed to the reset input of all four flip-flops. The Q output of each flip-flop was mapped to its corresponding output port (Q0-Q3), and the complete arrangement was encapsulated as a single reusable subcircuit representing the complete 74LS197 package.

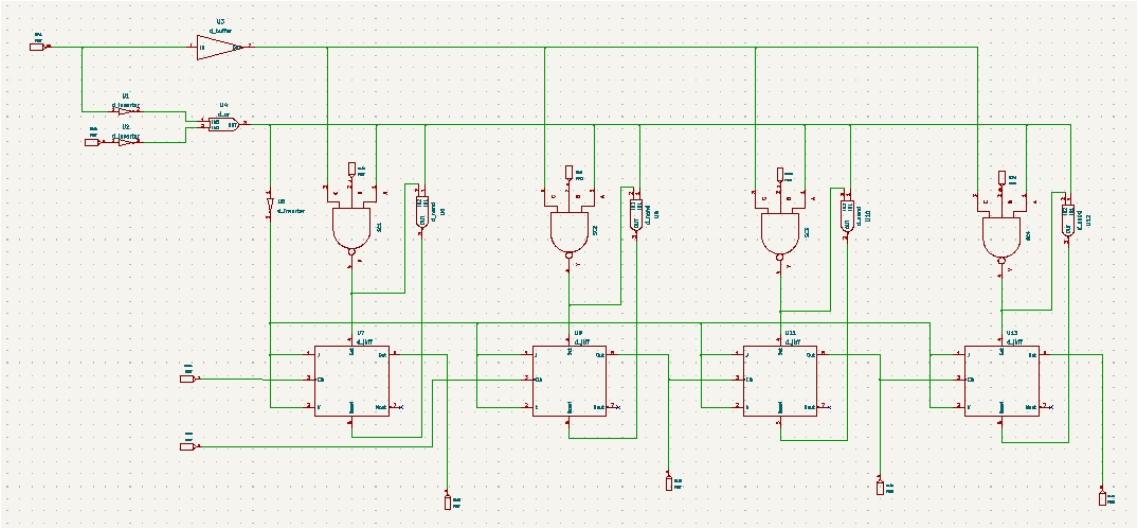

Figure 4.53: Subcircuit Schematic of 74LS197

4.11.6 Test Circuit

Since the 74LS197 subcircuit was built using digital gate primitives, an `adc_bridge_2` was used at the input stage to convert the analog voltage signals from two pulse sources (v2, v3) into digital logic levels for the CP0_b and MR_b lines, with the parallel load input (PL_b) held at a fixed logic level through voltage source v1 to keep the counter in free-running count mode rather than load mode. A `dac_bridge_4` was used at the output stage to convert the four digital outputs (Q0-Q3) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

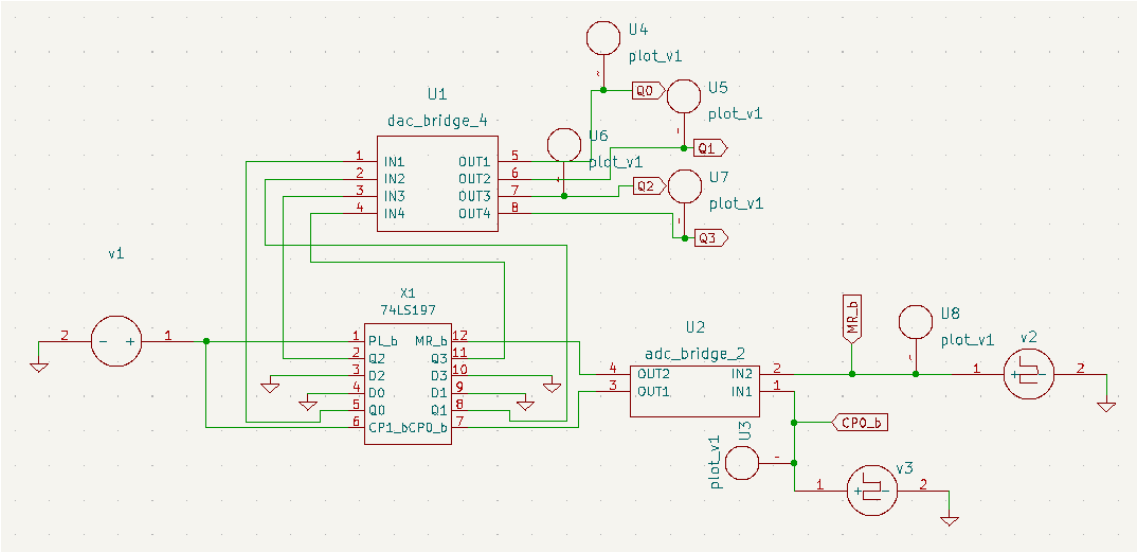

Figure 4.54: Test Circuit of 74LS197

4.11.7 Output

The input waveform shows the signals applied through the ADC bridge: v("cp_in") providing a continuous clock pulse train, and v("mr_in")+6 held low throughout, keeping the master reset inactive over the simulation window.

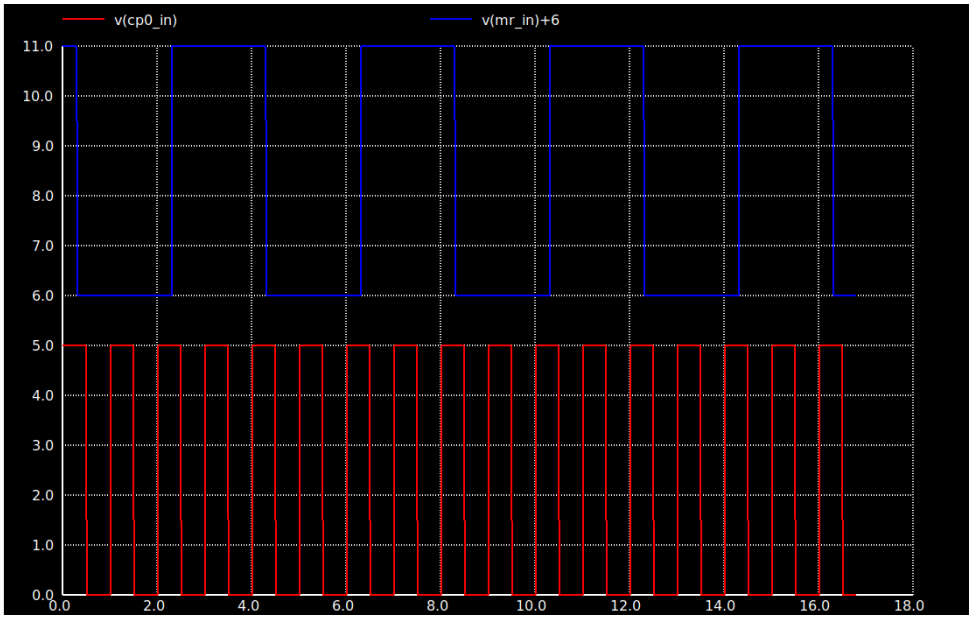

Figure 4.55: Input Waveform of 74LS197

The resulting output waveforms, v("q0")+18, v("q1")+12, v("q2")+6, and v("q3"), show the expected binary ripple sequence: Q0 toggles on every clock pulse, Q1 toggles at half that rate, Q2 at half the rate of Q1, and Q3 at half the rate of Q2. This confirms that the developed subcircuit model correctly replicates the 4-bit binary counting functionality of the 74LS197 with the parallel load held inactive.

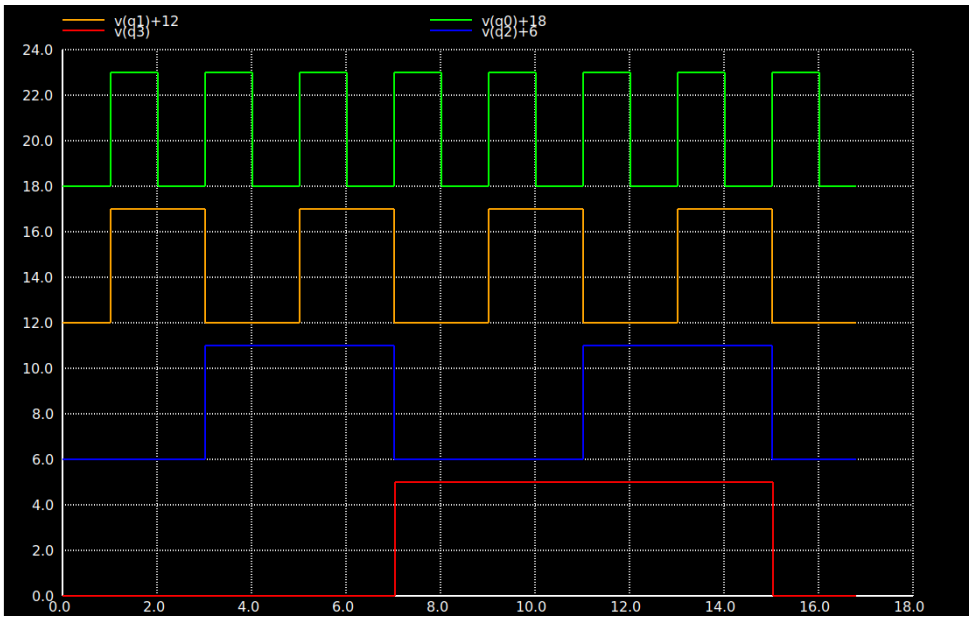

Figure 4.56: Output Waveform of 74LS197

4.12 74LS283 – 4-Bit Binary Full Adder

4.12.1 Description

The 74LS283 is a 4-bit binary full adder IC belonging to the low-power Schottky TTL (LS) logic family. It adds two 4-bit binary numbers (A1-A4 and B1-B4) along with an incoming carry (C0), producing a 4-bit sum output (S1-S4) and an outgoing carry (C4). Internally, the device uses a look-ahead carry structure to minimize propagation delay across the four bit positions, making it well suited for building wider adder chains by cascading the carry output of one stage into the carry input of the next.

4.12.2 Pin Diagram

The 74LS283 is available in a 16-pin dual in-line package, with eight data input pins (A1-A4, B1-B4), an incoming carry input (C0), four sum output pins (S1-S4), and an outgoing carry output (C4).

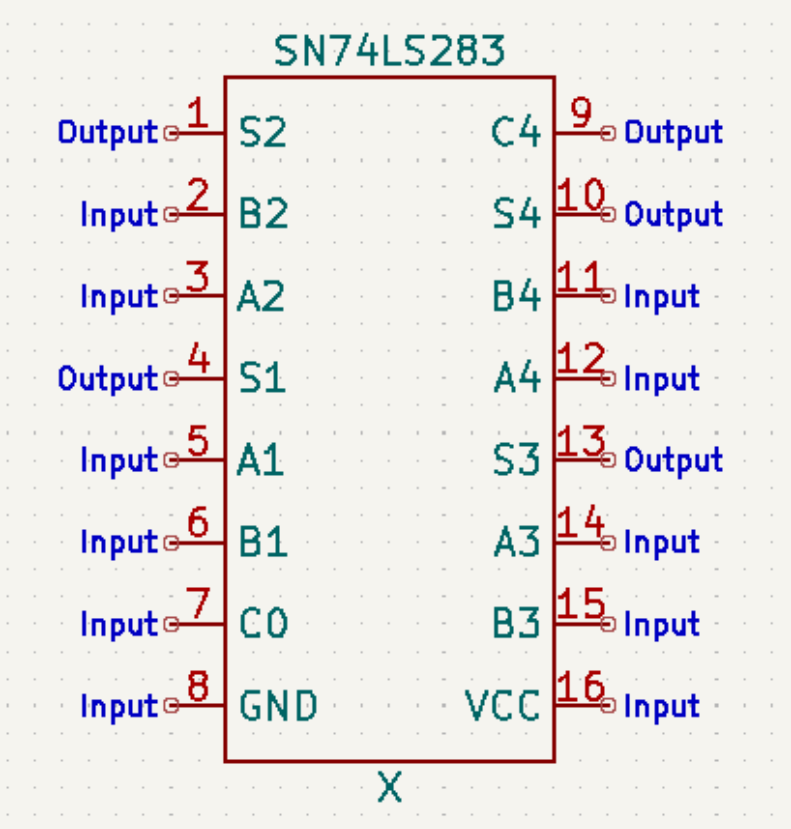

Figure 4.57: Pin Diagram of 74LS283

4.12.3 Features

- 4-bit binary full addition with internal carry look-ahead.

- Cascadable carry input (C0) and output (C4) for building wider adders.
- Low-power Schottky TTL (LS) logic family.
- Fast propagation delay suited to arithmetic-intensive digital systems.
- Compatible with other standard TTL/LS family devices.

4.12.4 Applications

- Binary arithmetic units in ALUs and processors.
- Cascaded multi-bit adder chains for wider word-length addition.
- Address computation and offset calculation in digital systems.
- Building block for subtractors and comparators using additional gating logic.

4.12.5 Subcircuit Design

The internal architecture of the 74LS283 was implemented in eSim using digital gate primitives rather than transistor-level modeling. Each bit position combines its A and B inputs along with the carry propagating from the previous stage using a combination of XOR gates (via inverter/AND/OR gate arrangements) to generate the sum output, and separate AND-OR gate networks to generate the carry term for that stage. Rather than rippling the carry sequentially through each stage, the carry into each of the four bit positions was derived directly from the A, B, and C0 inputs using AND-OR combinations mapped out in parallel, replicating the look-ahead carry structure of the original device. The four sum outputs (S1-S4) and the final carry output (C4) were each mapped to their respective output ports, and the complete arrangement was encapsulated as a single reusable subcircuit representing the complete 74LS283 package.

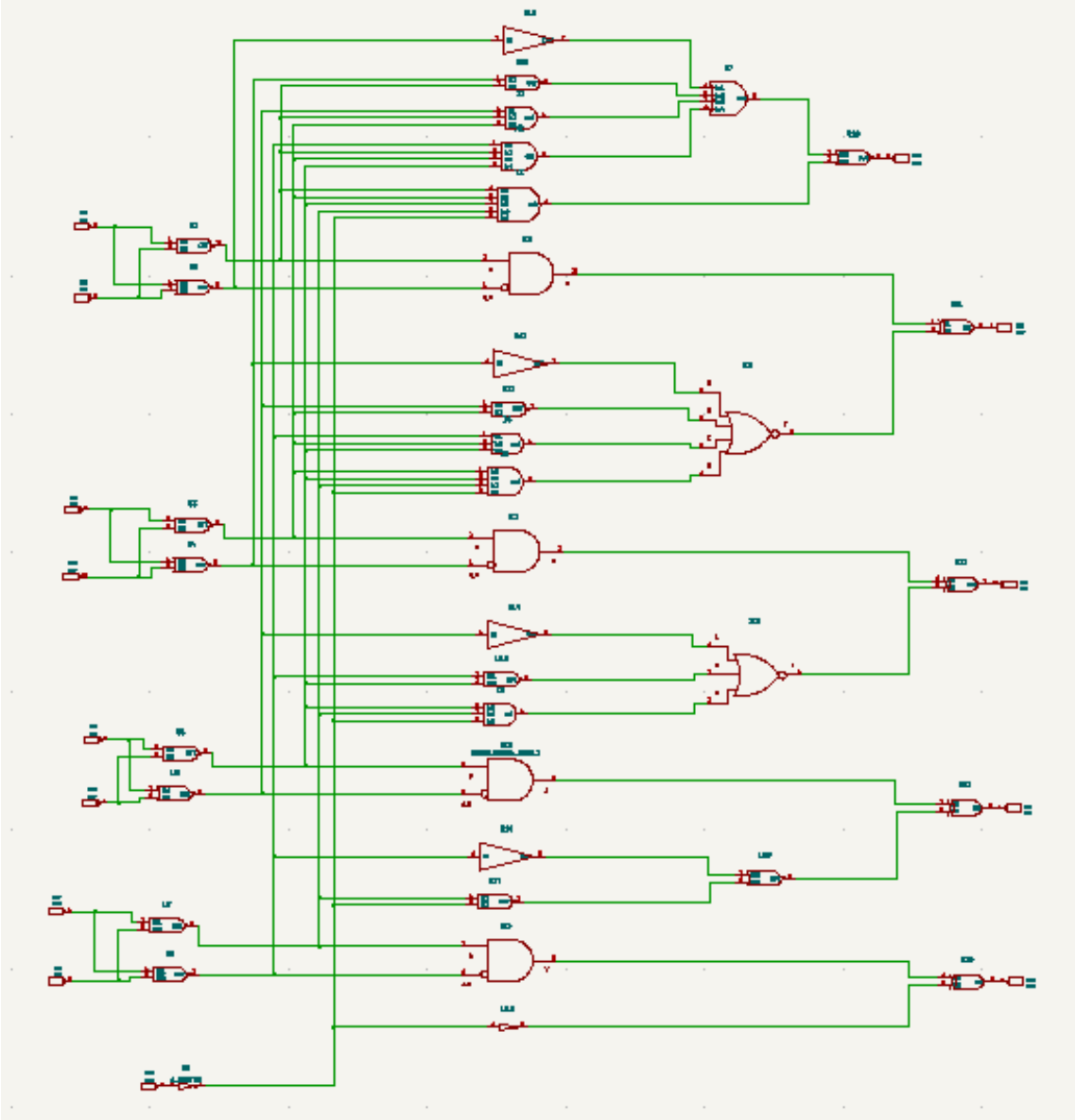

Figure 4.58: Subcircuit Schematic of 74LS283

4.12.6 Test Circuit

Since the 74LS283 subcircuit was built using digital gate primitives, an `adc_bridge_8` was used at the input stage to convert the analog voltage signals from eight pulse sources (v2-v9) into digital logic levels for the eight data inputs (A1-A4, B1-B4), and a separate single-channel bridge (U16) was used to convert the incoming carry signal (C0) from voltage source v1. A `dac_bridge_5` was used at the output stage to convert the five digital outputs (S1-S4, C4) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

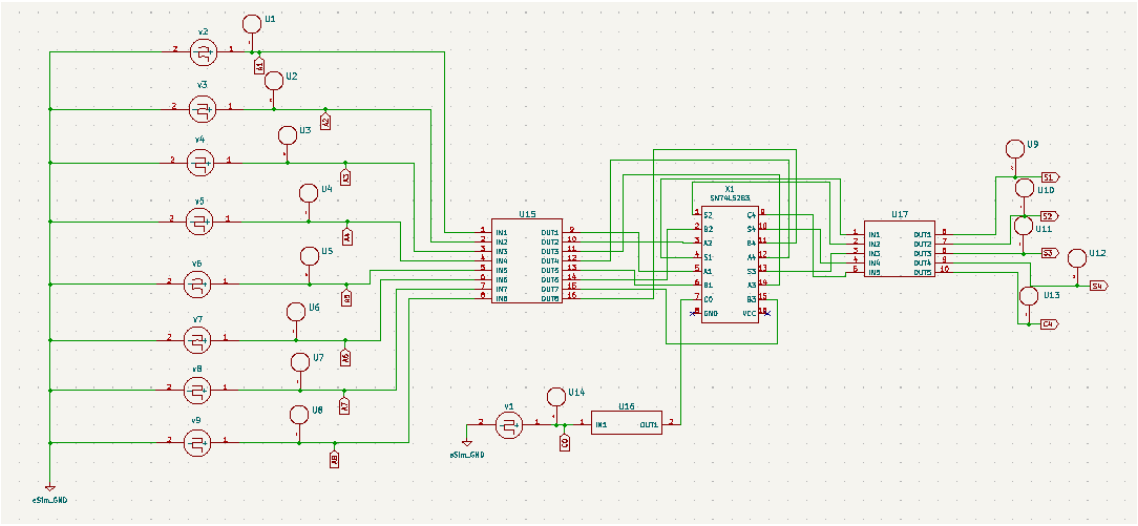

Figure 4.59: Test Circuit of 74LS283

4.12.7 Output

The input waveform shows the nine signals applied through the ADC bridges: v("c0")+40, v("b4")+35, v("a4")+30, v("b3")+25, v("a3")+20, v("b2")+15, v("a2")+10, v("b1")+5, and v("a1") (each offset for clarity of display), toggling at different rates so that the two 4-bit operands and the incoming carry together exercise a wide range of addition combinations over the simulation window.

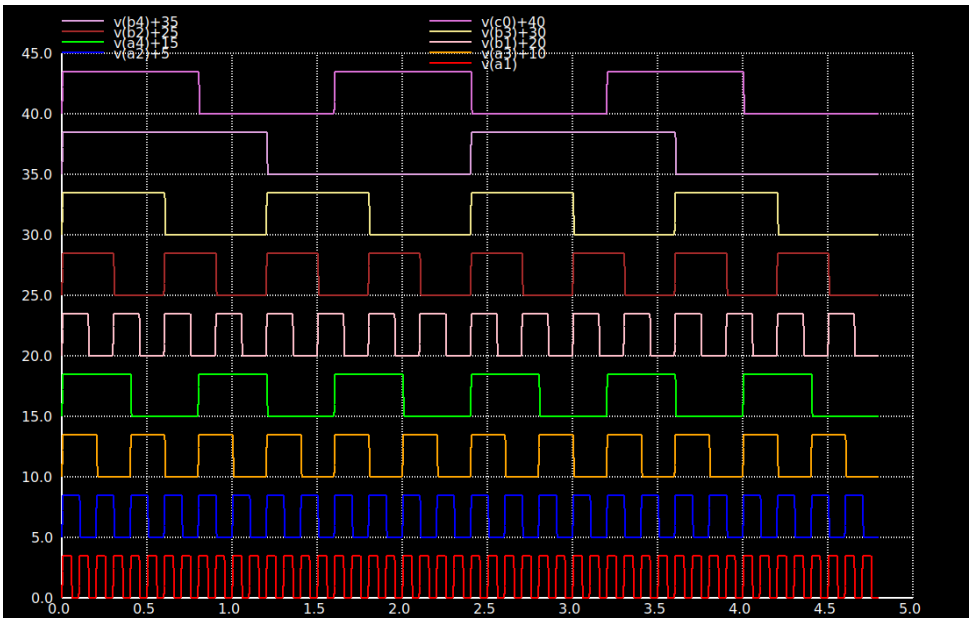

Figure 4.60: Input Waveform of 74LS283

The resulting output waveforms, v("c4")+20, v("s4")+15, v("s3")+10, v("s2")+5, and v("s1") (each offset for clarity of display), show the sum and carry outputs responding to the changing A, B, and C0 inputs, with each sum bit and the carry-out toggling in a pattern consistent with binary addition of the applied operands. This

confirms that the developed subcircuit model correctly replicates the 4-bit full-adder functionality of the 74LS283.

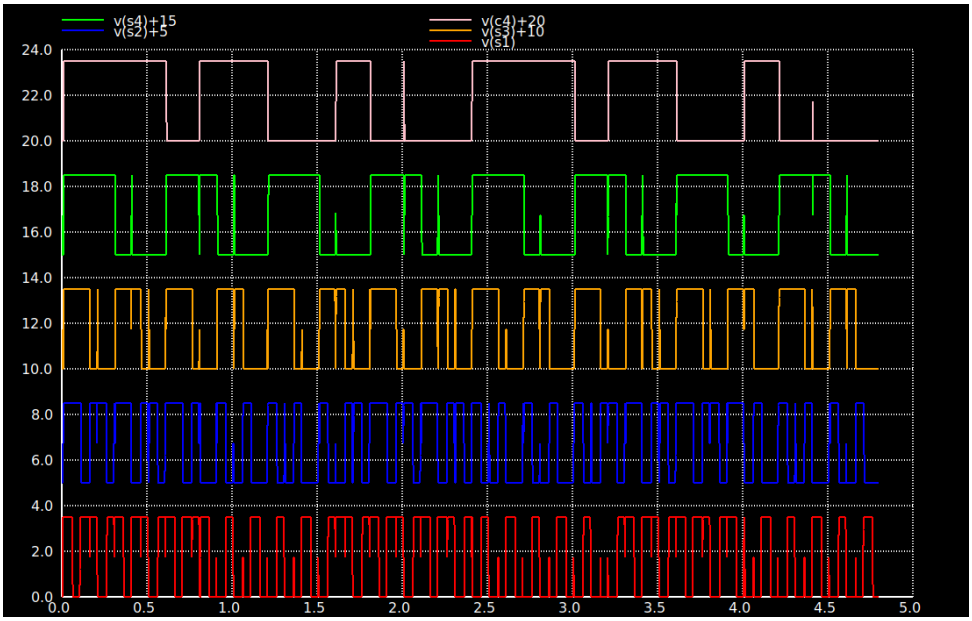

Figure 4.61: Output Waveform of 74LS283

4.13 74HC280 – 9-Bit Parity Generator/Checker

4.13.1 Description

The 74HC280 is a 9-bit parity generator/checker IC belonging to the high-speed CMOS (HC) logic family. It accepts nine data inputs (I0-I8) and produces two complementary outputs: an even-parity output (PE), which is high when the number of high inputs is even, and an odd-parity output (PO), which is high when the number of high inputs is odd. Being part of the HC family, the device offers CMOS-level power efficiency while maintaining speed performance comparable to standard TTL logic, and is commonly used wherever error detection or parity generation across a wide data word is required.

4.13.2 Pin Diagram

The 74HC280 is available in a 14-pin dual in-line package, with nine data input pins (I0-I8), one unused/no-connect pin (NC), and two parity output pins (PE, PO).

4.13.3 Features

- Nine-input parity generation with both even (PE) and odd (PO) outputs.
- PE and PO are always mutually exclusive/complementary.
- High-speed CMOS (HC) logic family with low static power consumption.

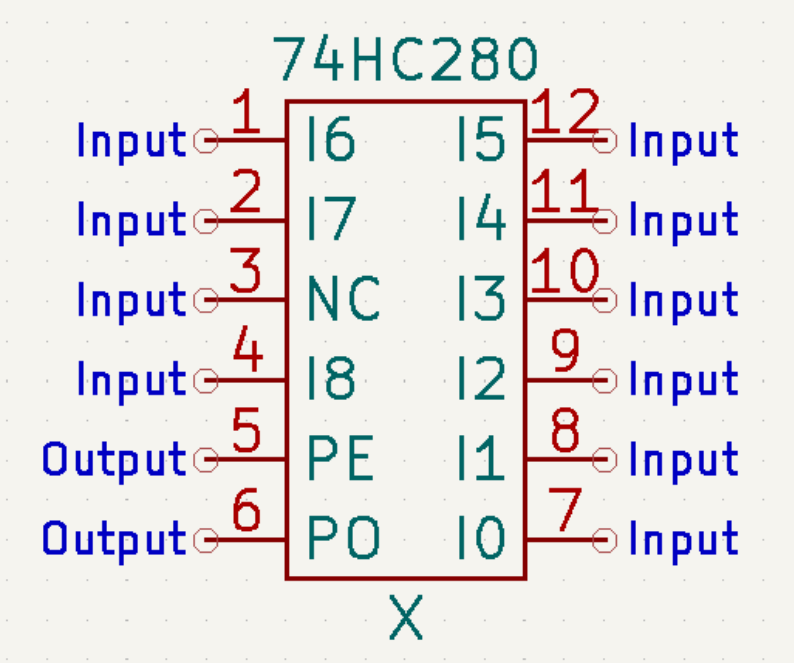

Figure 4.62: Pin Diagram of 74HC280

- Cascadable for parity generation across word widths wider than 9 bits.
- Compatible with other standard CMOS/HC family devices.

4.13.4 Applications

- Error detection in data transmission and storage systems.
- Parity generation for memory and bus interfaces.
- Building block for larger parity trees in wide data-word systems.
- Data integrity checking in communication protocols.

4.13.5 Subcircuit Design

The internal architecture of the 74HC280 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The nine inputs were first split into three groups of three, with each group's inputs conditioned through `d_inverter` stages and then combined using a set of `d_and` gates followed by a `d_nor` gate, producing an intermediate parity term for each group of three inputs. The three group-level parity terms were then combined together using another layer of `d_and` and `d_nor` gates to merge them into overall even and odd parity signals across all nine inputs. This hierarchical combining structure was mapped to the PE and PO output ports, and the complete arrangement was encapsulated as a single reusable subcircuit representing the complete 74HC280 package.

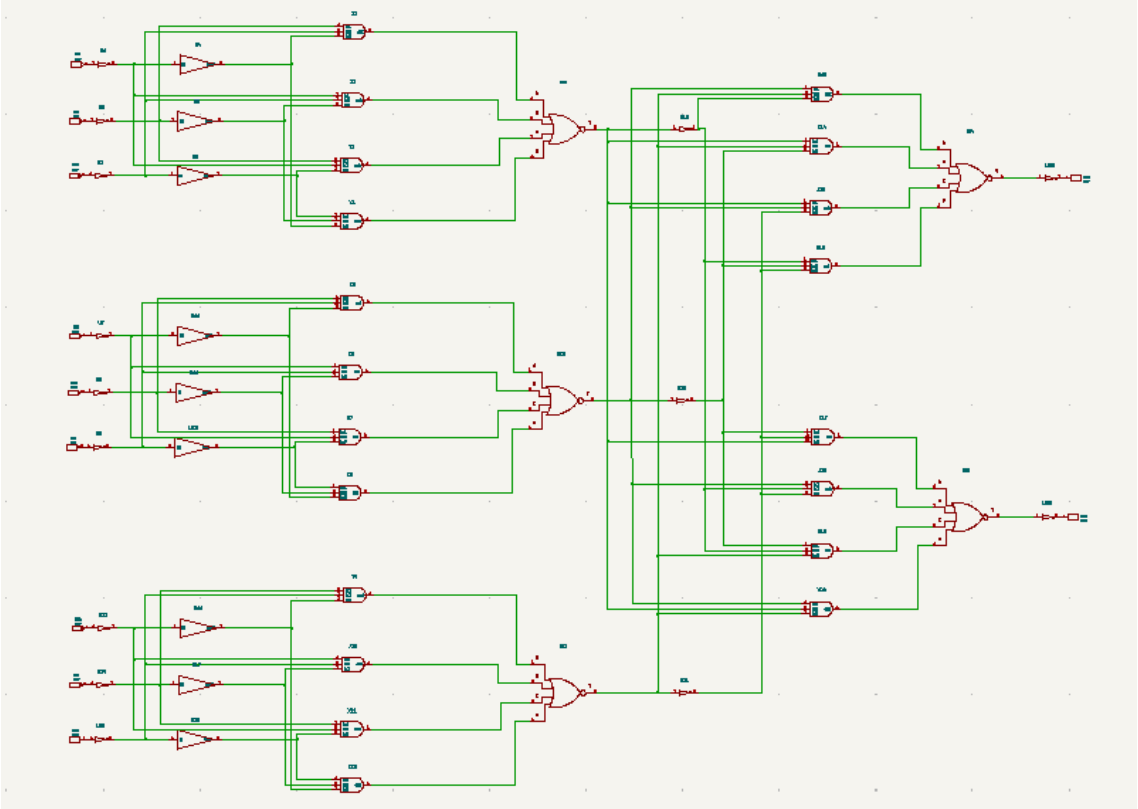

Figure 4.63: Subcircuit Schematic of 74HC280

4.13.6 Test Circuit

Since the 74HC280 subcircuit was built using digital gate primitives, an `adc_bridge_8` was used at the input stage to convert the analog pulse signals from eight voltage sources (v1-v8) into digital logic levels for eight of the nine data inputs, with a separate single-channel `adc_bridge_1` used for the ninth input signal (v9). A `dac_bridge_2` was used at the output stage to convert the two digital outputs (PE, PO) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

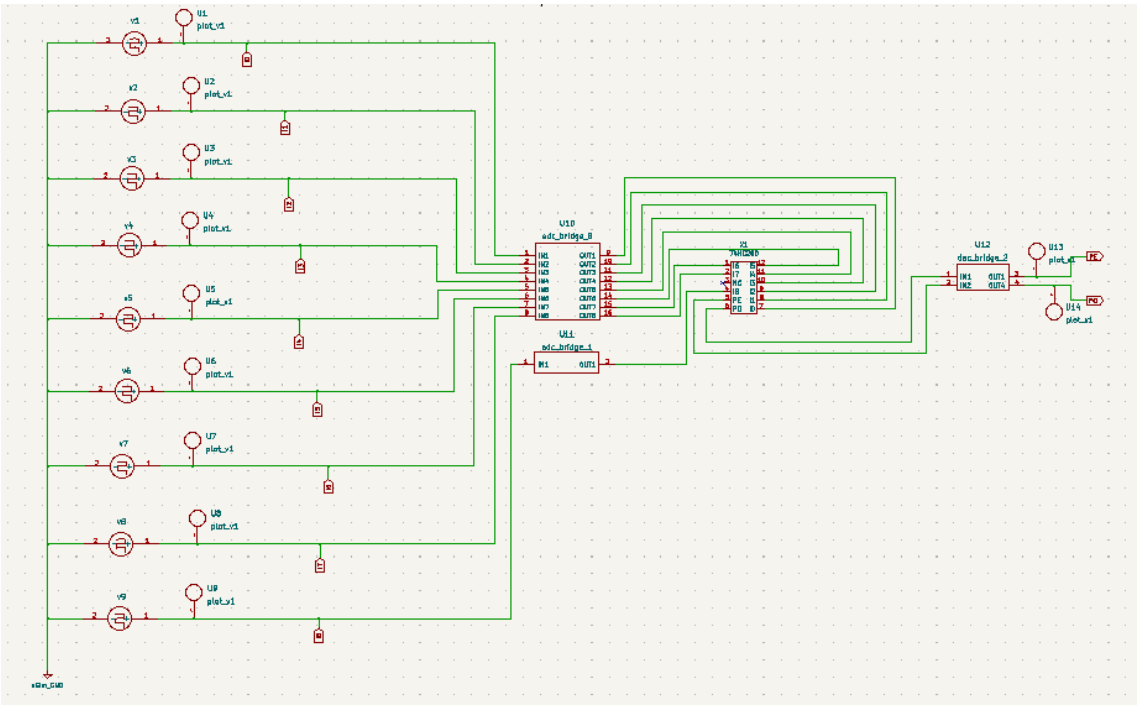

Figure 4.64: Test Circuit of 74HC280

4.13.7 Output

The input waveform shows the nine signals applied through the ADC bridges: v("i8")+48, v("i7")+42, v("i6")+36, v("i5")+30, v("i4")+24, v("i3")+18, v("i2")+12, v("i1")+6, and v("i0") (each offset for clarity of display), each toggling at a different rate so that all nine inputs exercise a wide range of combinations over the simulation window.

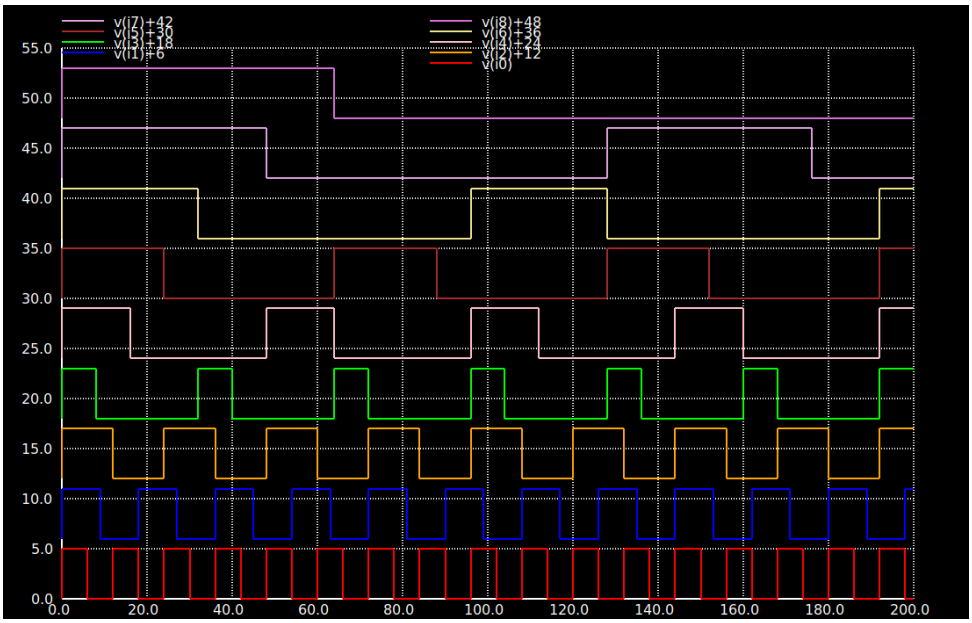

Figure 4.65: Input Waveform of 74HC280

The resulting output waveforms, v("pe") and v("po")+6, remain complementary throughout the simulation, with PE going high whenever the number of high inputs among I0-I8 is even, and PO going high whenever that count is odd. This confirms that the developed subcircuit model correctly replicates the 9-bit parity generation functionality of the 74HC280.

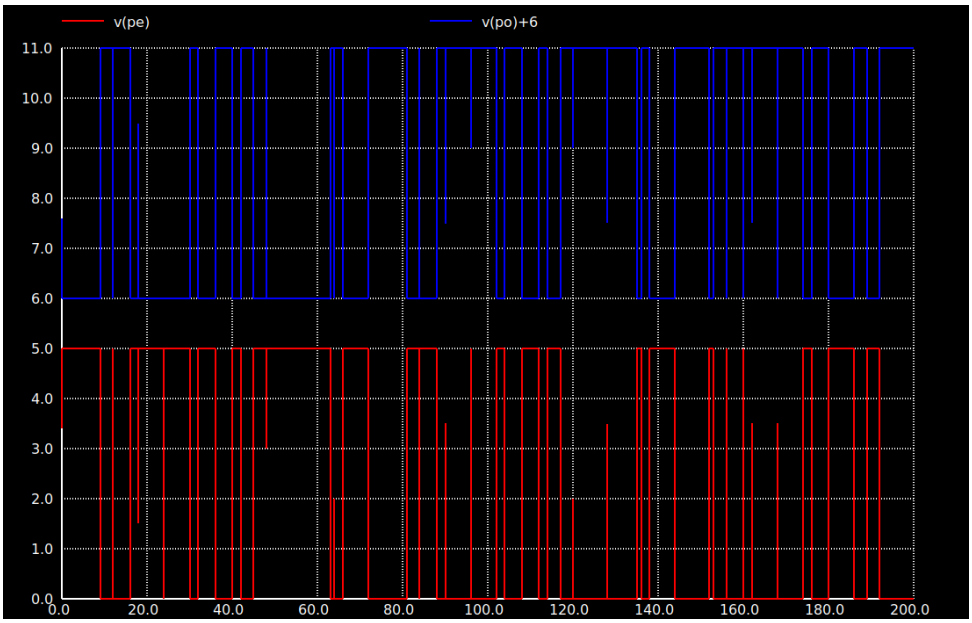

Figure 4.66: Output Waveform of 74HC280

4.14 74LS245 – Octal Bus Transceiver

4.14.1 Description

The 74LS245 is an octal bidirectional bus transceiver IC belonging to the low-power Schottky TTL (LS) logic family. It provides eight I/O lines on each side (A0-A7 and B0-B7) that can pass data in either direction between the two buses, controlled by a direction pin (DIR) that selects whether data flows from A to B or from B to A, and an active-low output enable (OE') that, when high, disables both sides into a high-impedance state. It is commonly used wherever bidirectional data transfer between two bus segments is required.

4.14.2 Pin Diagram

The 74LS245 is available in a 20-pin dual in-line package, with a direction control input (DIR), an active-low output enable input (OE'), and two sets of eight bidirectional I/O pins (A0-A7 and B0-B7).

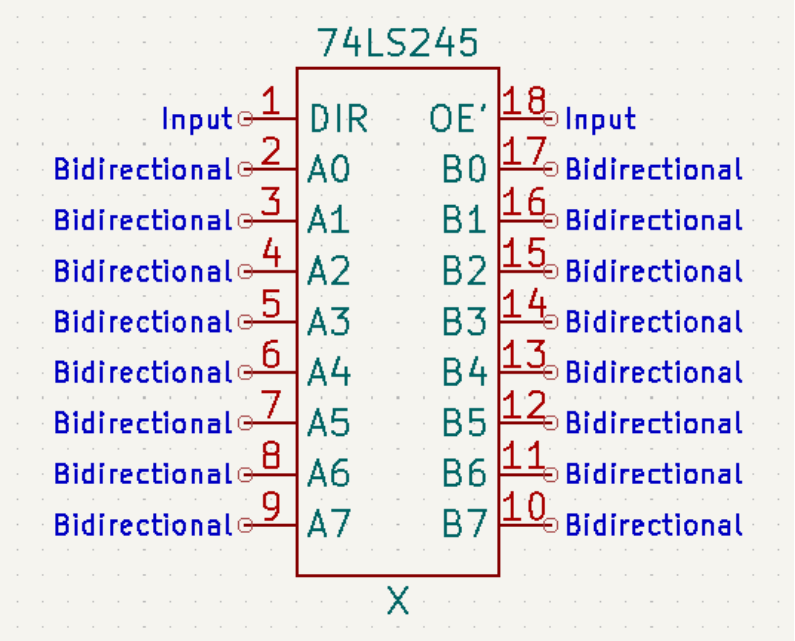

Figure 4.67: Pin Diagram of 74LS245

4.14.3 Features

- Octal bidirectional transceiver with a single direction control (DIR).
- Active-low output enable (OE') for high-impedance bus isolation.
- Non-inverting data transfer in either direction.
- Low-power Schottky TTL (LS) logic family.
- Compatible with other standard TTL/LS family devices.

4.14.4 Applications

- Bidirectional data bus interfacing between processors and peripherals.
- Bus isolation and buffering in multi-device systems.
- Direction-controlled data routing in memory and I/O interfaces.
- Level restoration and drive strength buffering on long bus lines.

4.14.5 Subcircuit Design

The internal architecture of the 74LS245 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The direction (DIR) and output-enable (OE') inputs were first combined using a `d_and` gate to generate the enable condition for the A-to-B direction, and this same combination was passed through

a `d_inverter` stage to derive the complementary enable condition for the B-to-A direction. Each of the eight bus lines was then implemented using a pair of back-to-back `d_tristate` buffers, one driving from A to B and gated by the A-to-B enable term, and the other driving from B to A and gated by the B-to-A enable term, so that only one direction is active at a time depending on the state of DIR, while OE' disables both directions simultaneously when inactive. The eight such bidirectional stages were mapped to their respective A0-A7 and B0-B7 ports and encapsulated together as a single reusable subcircuit representing the complete 74LS245 package.

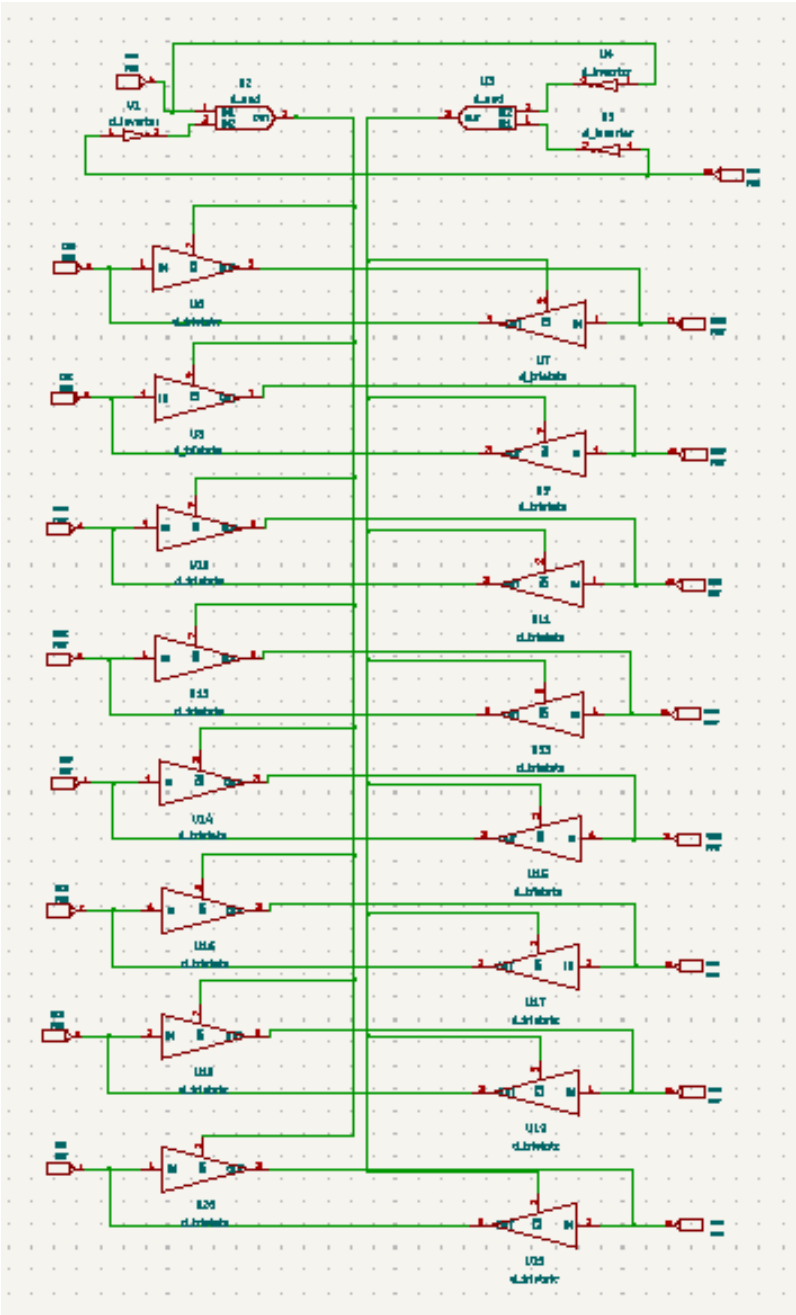

Figure 4.68: Subcircuit Schematic of 74LS245

4.14.6 Test Circuit

Since the 74LS245 subcircuit was built using digital gate primitives, an `adc_bridge_8` was used at the input stage to convert the analog pulse signals from eight voltage sources (v2-v9) into digital logic levels for the driving bus, and a `dac_bridge_8` was used at the output stage to convert the eight digital outputs of the receiving bus back into analog voltage signals for observation. The DIR pin was set to route data from A to B for one test run and from B to A for the other, with OE' held active throughout. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

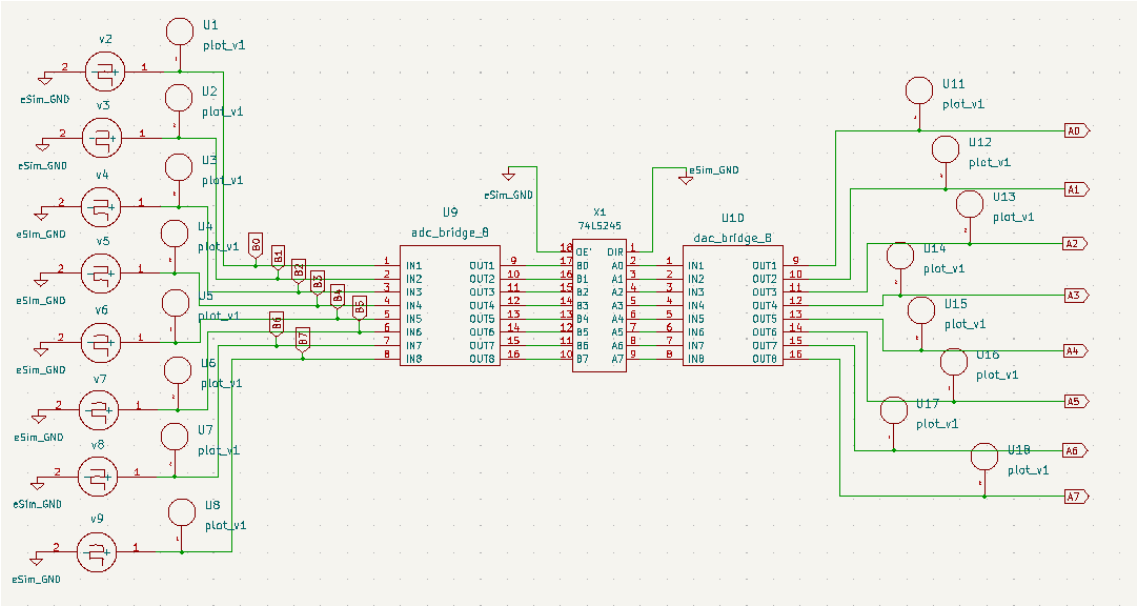

Figure 4.69: Test Circuit of 74LS245

4.14.7 Output

A-to-B Direction

With DIR set to route data from A to B, the input waveform shows the eight signals applied to A0-A7 through the ADC bridge, each toggling at a different rate so that all eight lines exercise a wide range of combinations over the simulation window.

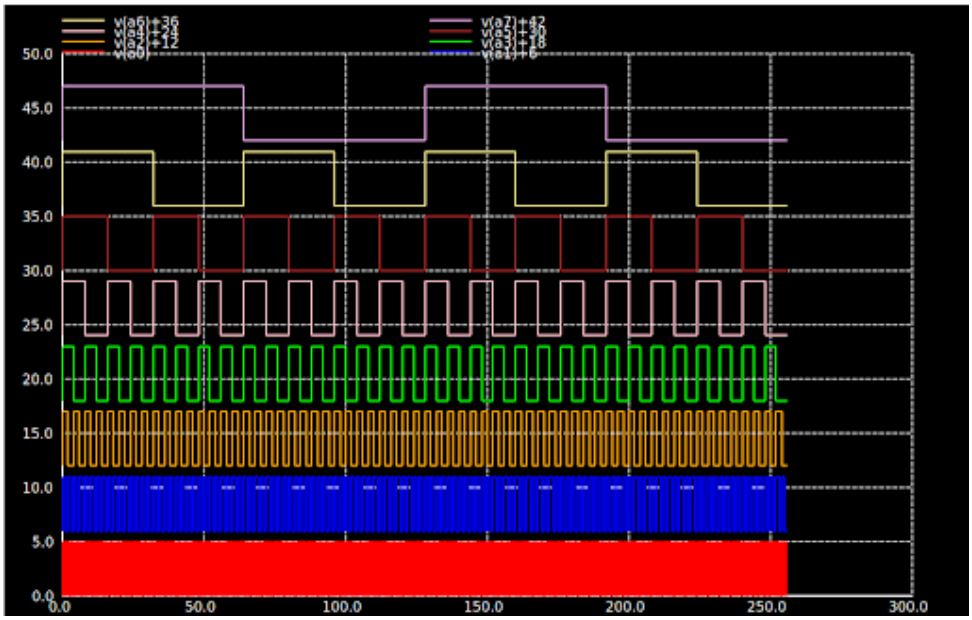

Figure 4.70: Input Waveform of 74LS245 (A-to-B direction)

The resulting output waveforms on B0-B7 reproduce the same pattern applied on A0-A7, confirming that the developed subcircuit model correctly passes data through in the A-to-B direction.

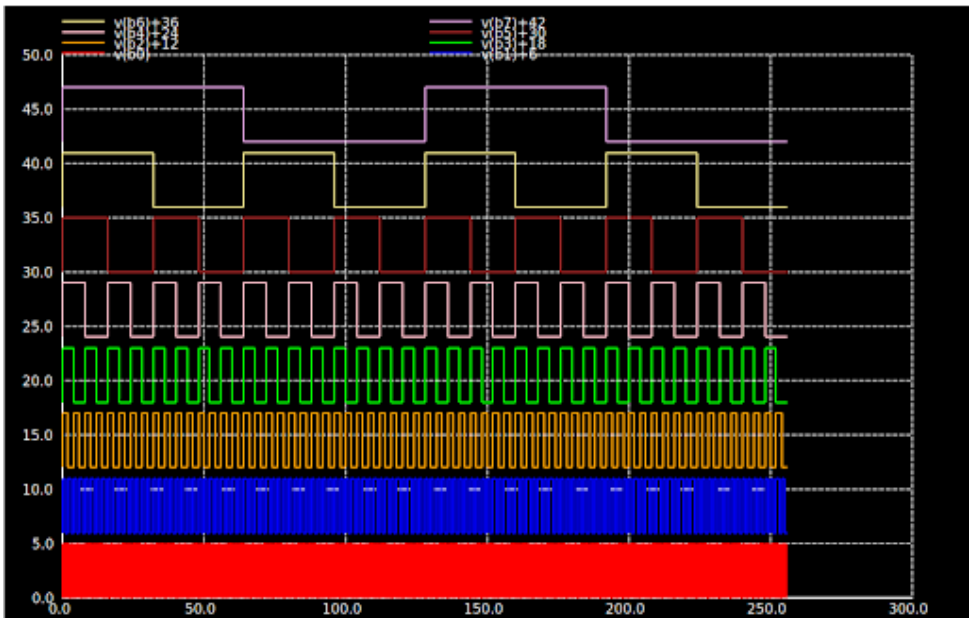

Figure 4.71: Output Waveform of 74LS245 (A-to-B direction)

B-to-A Direction

With DIR set to route data from B to A, the input waveform shows the eight signals applied to B0-B7 through the ADC bridge, each toggling at a different rate so that all eight lines exercise a wide range of combinations over the simulation window.

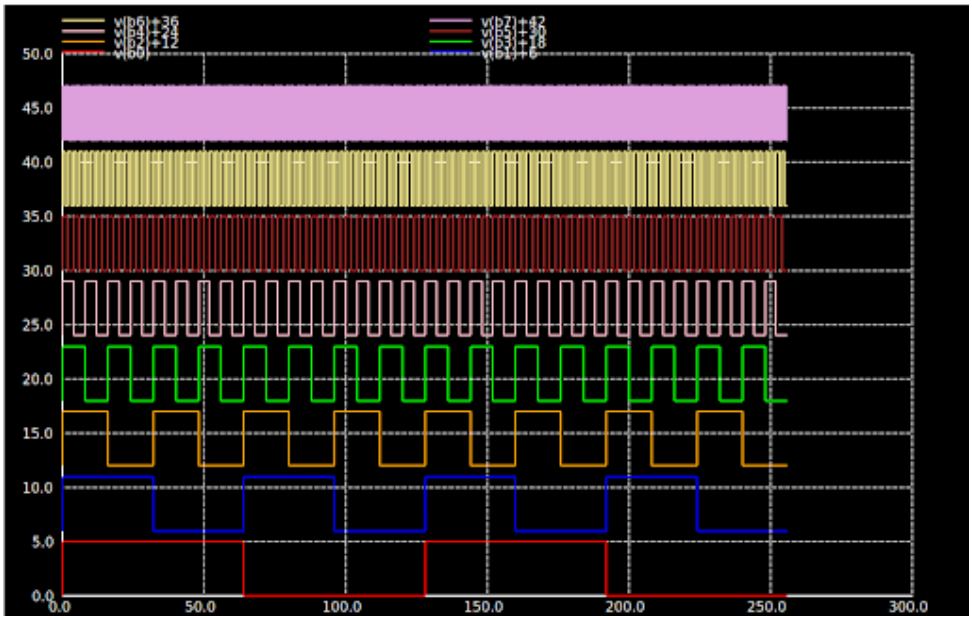

Figure 4.72: Input Waveform of 74LS245 (B-to-A direction)

The resulting output waveforms on A0-A7 reproduce the same pattern applied on B0-B7, confirming that the developed subcircuit model correctly passes data through in the B-to-A direction and that the transceiver operates correctly in both directions.

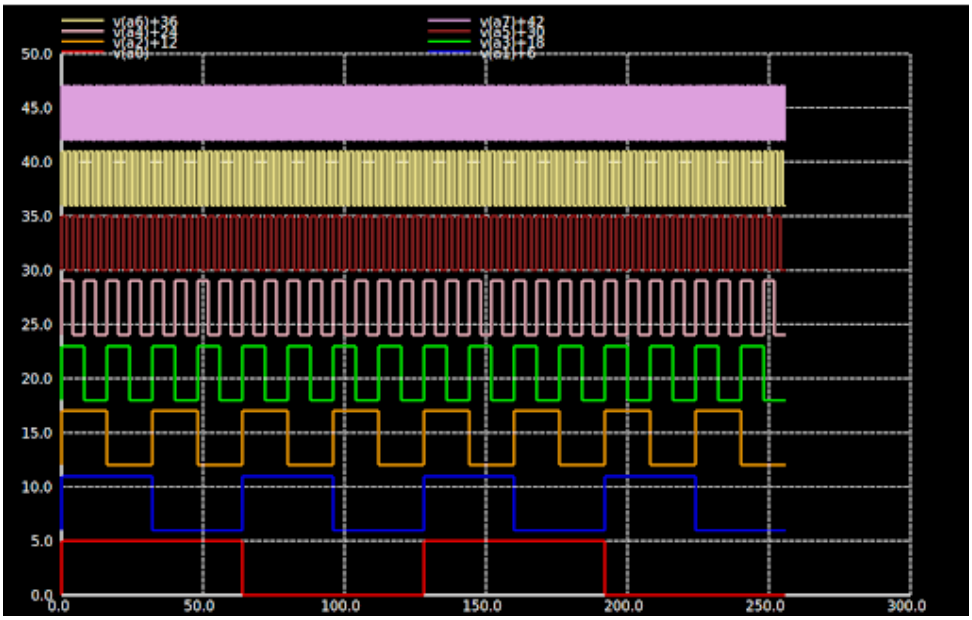

Figure 4.73: Output Waveform of 74LS245 (B-to-A direction)

4.15 74HC4511 – BCD to 7-Segment Latch/Decoder/Driver

4.15.1 Description

The 74HC4511 is a BCD-to-7-segment latch/decoder/driver IC belonging to the high-speed CMOS (HC) logic family. It accepts a 4-bit BCD input (A, B, C, D) and drives seven active-high segment outputs (a-g) to display the corresponding decimal digit on a 7-segment display. The device includes a lamp test input (LT') to force all segments on for display verification, a blanking input (BI') to force all segments off, and a latch enable input (LE') that, when active, freezes the displayed digit regardless of further changes on the BCD inputs. Being part of the HC family, the device offers CMOS-level power efficiency while maintaining speed performance comparable to standard TTL logic, and is commonly used wherever numeric display driving is required.

4.15.2 Pin Diagram

The 74HC4511 is available in a 16-pin dual in-line package, with four BCD data input pins (A, B, C, D), a lamp test input (LT'), a blanking input (BI'), a latch enable input (LE'), and seven segment output pins (a-g).

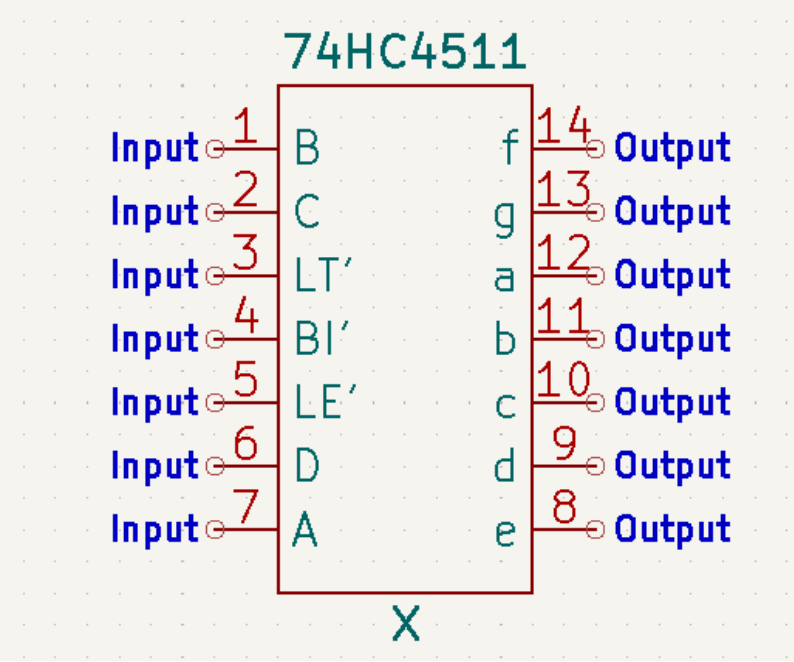

Figure 4.74: Pin Diagram of 74HC4511

4.15.3 Features

- Decodes 4-bit BCD input into 7-segment display drive signals.

- Active-low lamp test (LT') forces all segments on for verification.
- Active-low blanking input (BI') forces all segments off.
- Active-low latch enable (LE') freezes the displayed digit.
- High-speed CMOS (HC) logic family with low static power consumption.

4.15.4 Applications

- Numeric display driving in instrumentation and consumer electronics.
- Digital counters and timers with 7-segment readouts.
- Calculator and measurement device digit displays.
- Latched display outputs for multiplexed or noisy data sources.

4.15.5 Subcircuit Design

The internal architecture of the 74HC4511 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The four BCD inputs (A, B, C, D) were first passed through a latch stage built from `d_latch` primitives gated by the LE' input, so that the inputs are held steady internally whenever the latch is enabled. The latched BCD values and their complements, generated using `d_inverter` stages, were then combined using a network of AND and OR gates to form the individual minterm expressions required to activate each of the seven segments (a-g) for the corresponding decimal digit. The LT' and BI' inputs were combined with this segment logic through additional gating so that LT' overrides the decoded pattern to force all segments on, and BI' overrides it to force all segments off. Each of the seven resulting segment-drive signals was passed through a final inverter/buffer stage and mapped to its respective output port. The complete arrangement was encapsulated as a single reusable subcircuit representing the complete 74HC4511 package.

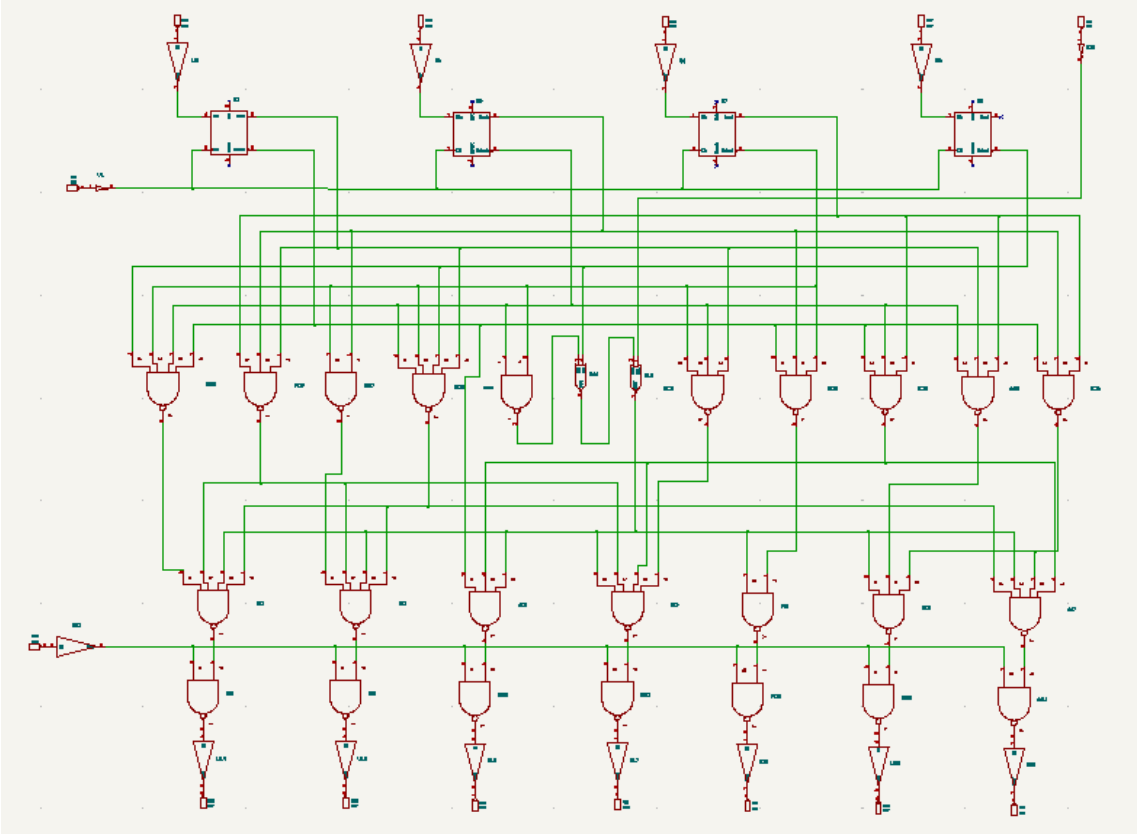

Figure 4.75: Subcircuit Schematic of 74HC4511

4.15.6 Test Circuit

Since the 74HC4511 subcircuit was built using digital gate primitives, an `adc_bridge_7` was used at the input stage to convert the analog voltage signals from six pulse sources (v1-v2, v5-v6) and two fixed-level sources (v3, v4) into digital logic levels for the B, C, LT', BI', LE', D, and A inputs, and a `dac_bridge_7` was used at the output stage to convert the seven digital segment outputs (a-g) back into analog voltage signals for observation. This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment using standard analog voltage sources and plot probes.

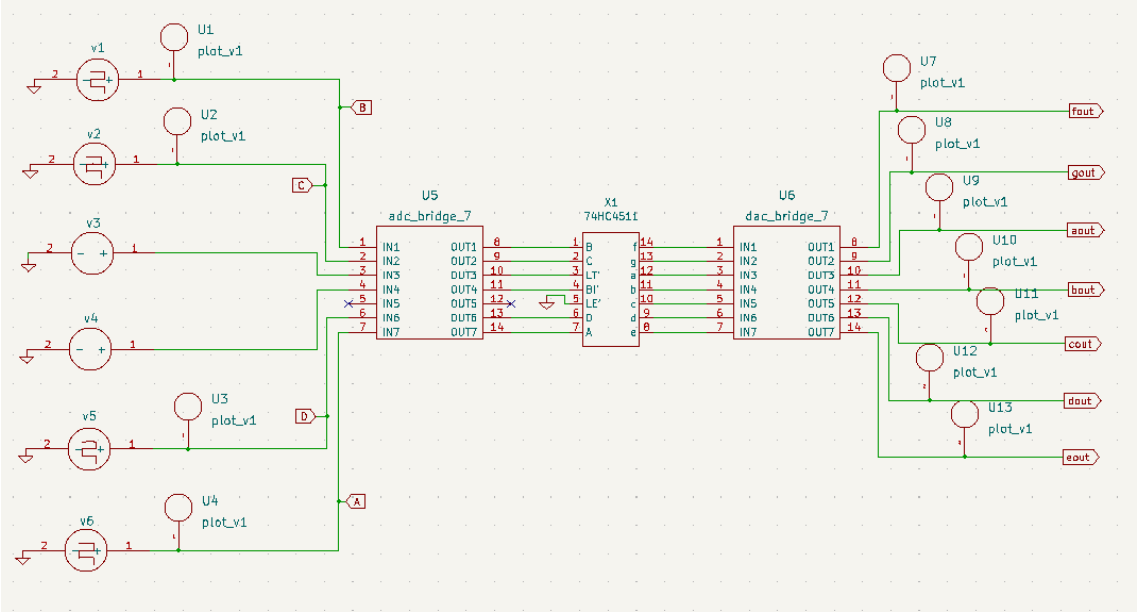

Figure 4.76: Test Circuit of 74HC4511

4.15.7 Output

The input waveform shows the signals applied through the ADC bridge: v("in_a")+21 and v("in_b")+14 toggling at faster rates, v("in_c")+7 toggling at a slower rate, and v("in_d") pulsing periodically (each offset for clarity of display), together stepping the BCD input and the LT'/BI'/LE' control lines through a range of combinations over the simulation window.

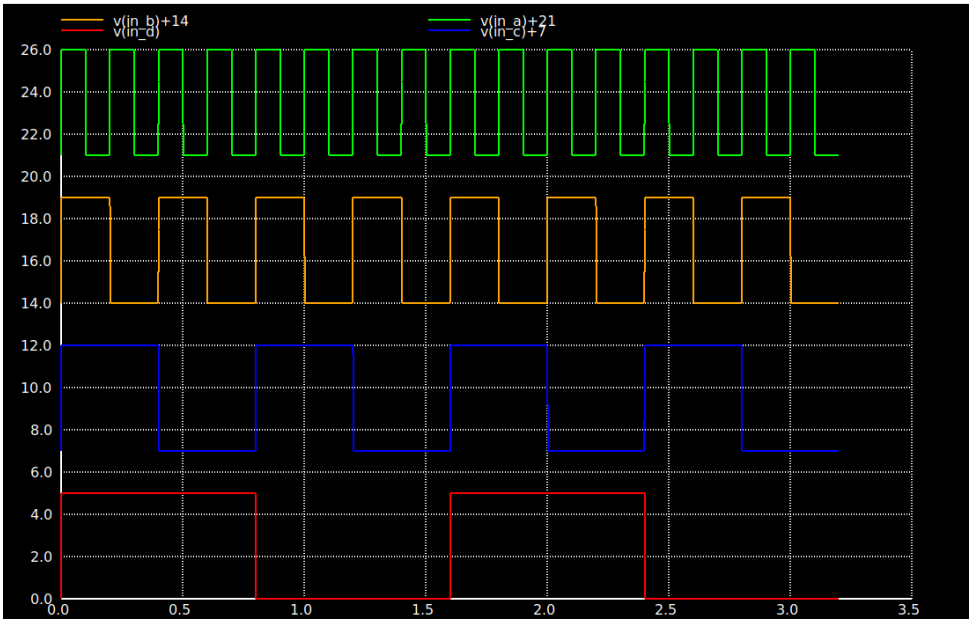

Figure 4.77: Input Waveform of 74HC4511

The resulting output waveforms, v("gout")+42, v("fout")+35, v("eout")+28, v("dout")+21, v("cout")+14, v("bout")+7, and v("aout") (each offset for clarity of

display), show the seven segment-drive signals switching in response to the changing BCD input, with the segment pattern updating to reflect each new decimal digit and holding steady during the periods when the latch enable is active. This confirms that the developed subcircuit model correctly replicates the BCD-to-7-segment decoding and latching functionality of the 74HC4511.

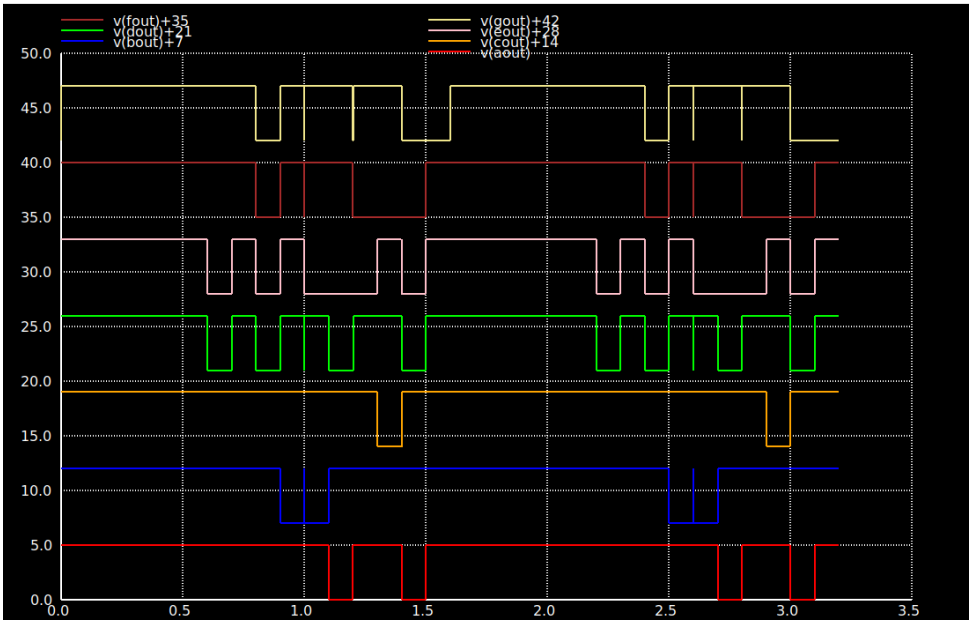

Figure 4.78: Output Waveform of 74HC4511

4.16 8212 – 8-Bit Input/Output Port

4.16.1 Description

The 8212 is an 8-bit input/output port IC used for interfacing microprocessors with peripheral devices. It provides eight data input lines (DI1-DI8) and eight corresponding data output lines (DO1-DO8), with an internal latch that captures the input data on a strobe (STB) pulse. The device includes mode control (MD) and device-select inputs (DS1', DS2) that configure it for either input-port or output-port operation, an active-low clear (CLR') that resets the internal latch, and an active-low interrupt output (INT') that flags when new data has been latched. It is commonly used in microprocessor systems for buffering and latching data between the CPU bus and external devices.

4.16.2 Pin Diagram

The 8212 is available in a 24-pin dual in-line package, with eight data input pins (DI1-DI8), eight data output pins (DO1-DO8), mode and device-select control pins (MD, DS1', DS2), a strobe input (STB), an active-low clear input (CLR'), and an active-low interrupt output (INT').

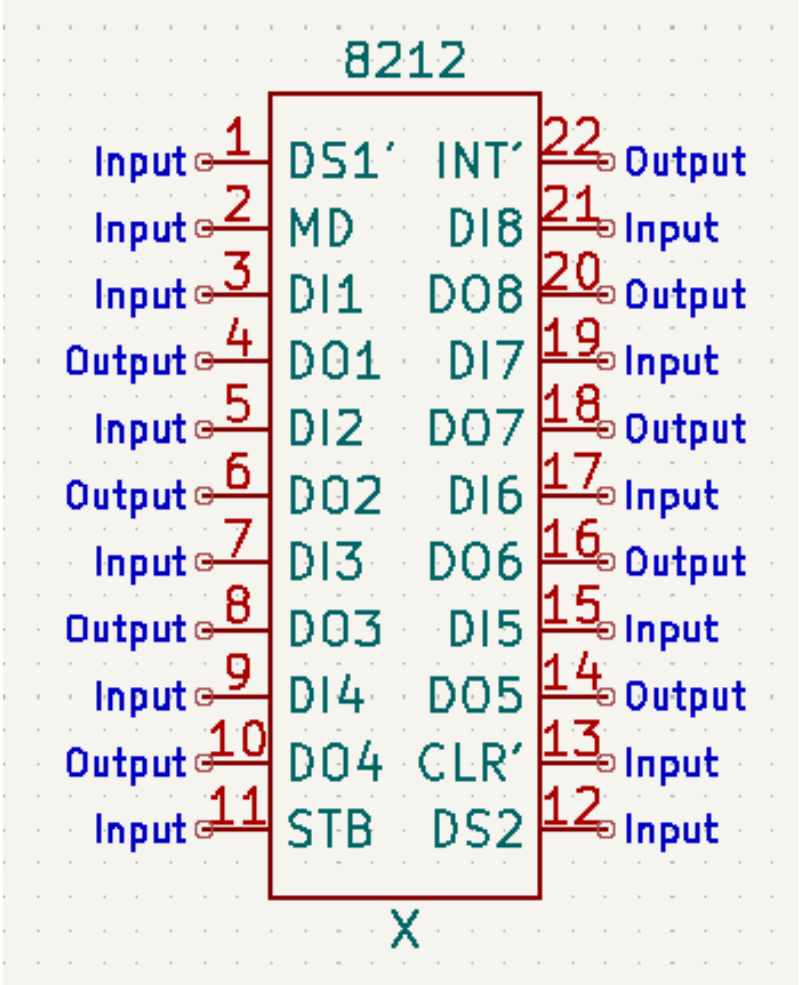

Figure 4.79: Pin Diagram of 8212

4.16.3 Features

- 8-bit latched input/output port with strobe-controlled data capture.
- Mode and device-select inputs (MD, DS1', DS2) for input/output port configuration.
- Active-low clear (CLR') for asynchronously resetting the latch.
- Active-low interrupt output (INT') to flag newly latched data.
- Commonly used for microprocessor bus interfacing and peripheral buffering.

4.16.4 Applications

- Latched input/output ports in microprocessor-based systems.
- Data buffering between a CPU bus and peripheral devices.

- Interrupt-driven data capture from external sources.
- General-purpose bus interfacing and signal buffering in digital systems.

4.16.5 Subcircuit Design

The internal architecture of the 8212 was implemented in eSim using digital gate primitives rather than transistor-level modeling. The mode (MD) and device-select (DS1', DS2) inputs were combined using a network of `d_and`, `d_nor`, and `d_inverter` gates to generate the internal control signals governing latch transparency and output enabling. The strobe (STB) and clear (CLR') inputs were combined through similar gating to produce the clocking and asynchronous reset signals distributed to the internal latch stages. Each of the eight data bits was implemented using a `d_latch` stage, clocked by the derived strobe signal and reset by the derived clear signal, with its output passed through a `d_tristate` buffer gated by the output-enable control before reaching the corresponding DO output pin. A separate small gating network combining the strobe and mode-derived control signals was used to generate the active-low interrupt output (INT'). The eight latch/buffer stages were mapped to their respective DI/DO ports, and the complete arrangement was encapsulated as a single reusable subcircuit representing the complete 8212 package.

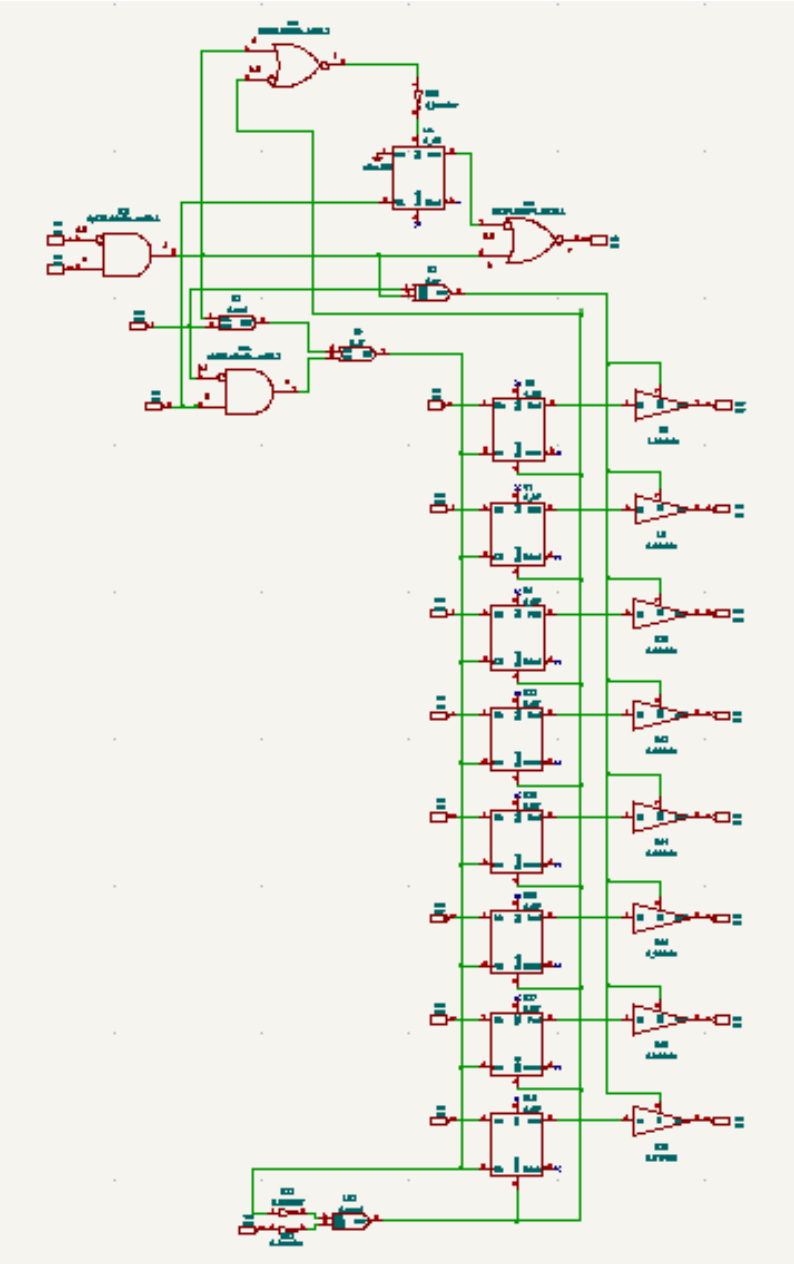

Figure 4.80: Subcircuit Schematic of 8212

4.16.6 Test Circuit

Since the 8212 subcircuit was built using digital gate primitives, an `adc_bridge_8` was used at the input stage to convert the analog voltage signals from eight sources (v2-v9) into digital logic levels for the eight data inputs (DI1-DI8), and a separate `adc_bridge_1` was used to convert the strobe (STB) signal from voltage source v10. A `dac_bridge_8` was used at the output stage to convert the eight digital outputs (DO1-DO8) back into analog voltage signals, with a separate `dac_bridge_1` used for the interrupt output (INT'). This bridging arrangement allowed the digitally modeled IC to be tested and probed within eSim's mixed-signal simulation environment

using standard analog voltage sources and plot probes.

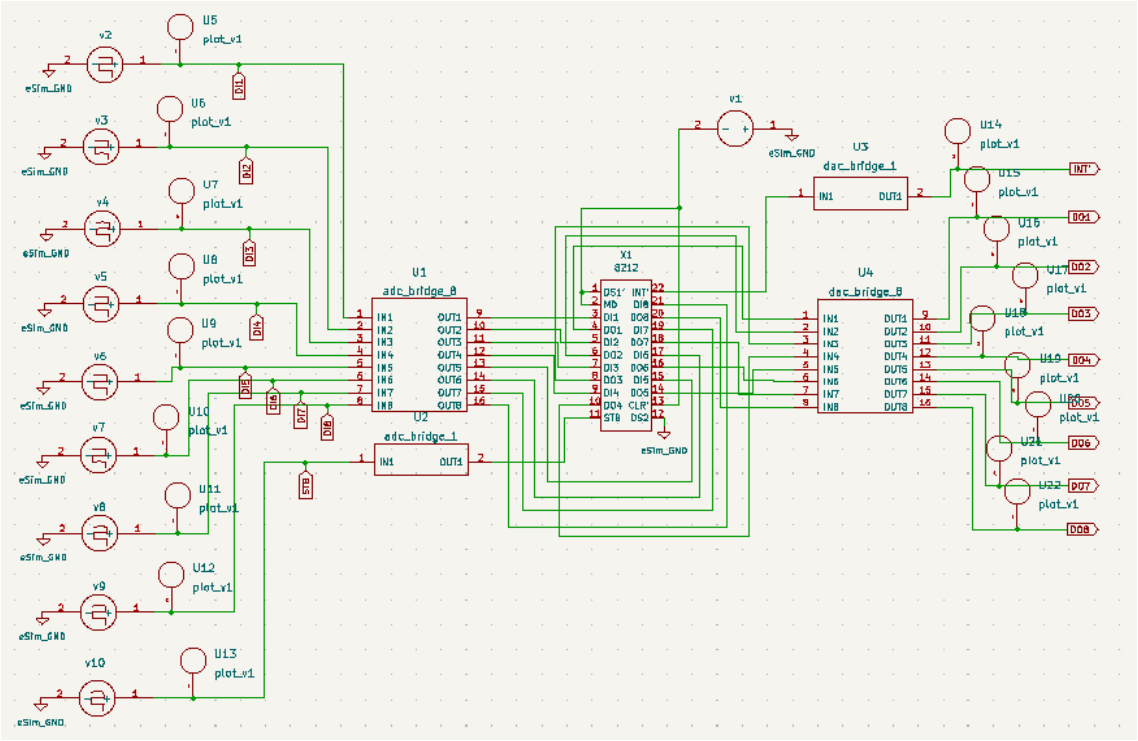

Figure 4.81: Test Circuit of 8212

4.16.7 Output

The input waveform shows the eight data signals applied through the ADC bridge: v("di8")+16, v("di7")+8, v("di6")+32, v("di5")+24, v("di4")+40, v("di3")+40, v("di2")+48, and v("di1")+56 (each offset for clarity of display), with each bit stepping to a new logic level at a different point in the simulation window so that the input data word changes progressively over time.

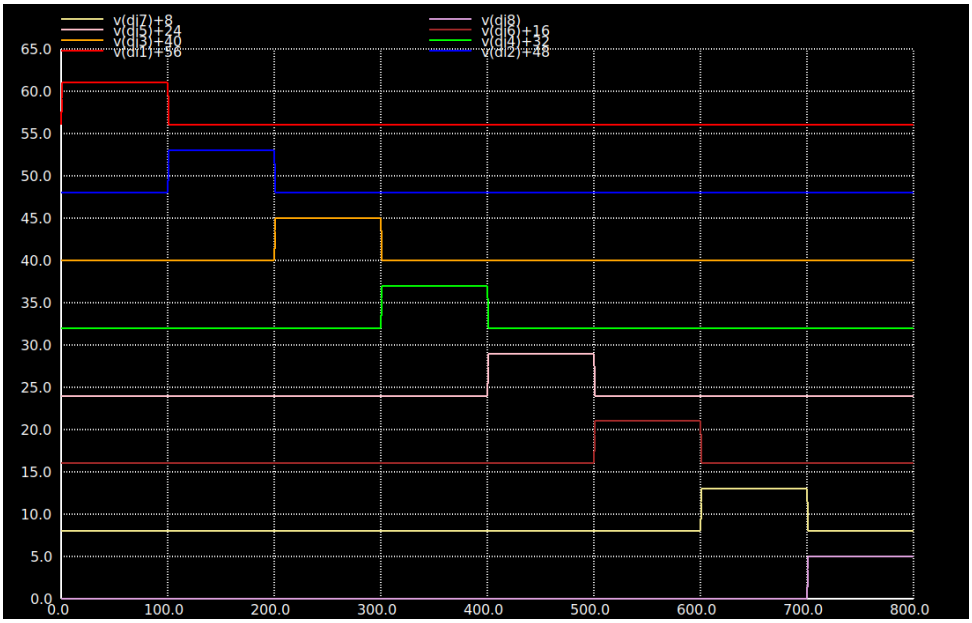

Figure 4.82: Input Waveform of 8212

The resulting output waveforms, v("do8")+16, v("do7")+8, v("do6")+32, v("do5")+24, v("do4")+40, v("do3")+40, v("do2")+48, and v("do1")+56 (each offset for clarity of display), closely track the corresponding input signals, with each output bit taking on the value of its input shortly after the input changes. This confirms that the developed subcircuit model correctly replicates the latched input/output port functionality of the 8212.

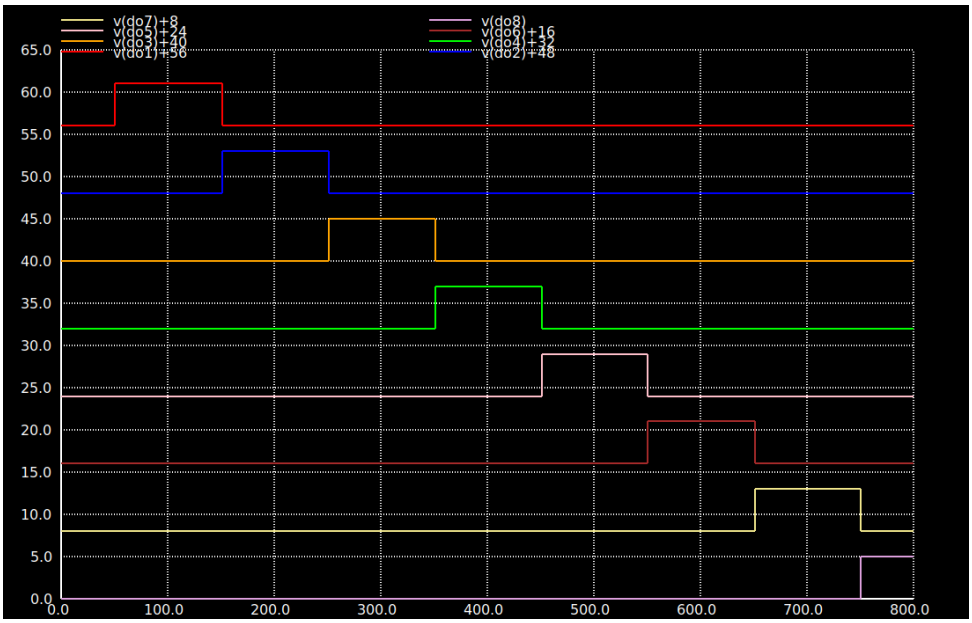

Figure 4.83: Output Waveform of 8212

4.17 TLC274 – Quad Low-Power CMOS Op-Amp

4.17.1 Description

The TLC274 is a quad low-power operational amplifier fabricated using Texas Instruments' LinCMOS process. It integrates four independent, high-input-impedance operational amplifiers on a single chip, each designed for low supply current operation while maintaining wide bandwidth and low input offset voltage. Because of its CMOS input stage, the TLC274 draws significantly less supply current than comparable bipolar-input quad op-amps, and its output stage is capable of swinging to within a few hundred millivolts of either supply rail, making it well suited for low-power signal-conditioning and comparator applications.

4.17.2 Pin Diagram

The TLC274 is available in a 14-pin dual in-line package, with each of the four operational amplifiers sharing common `VDD` (pin 4) and `GND` (pin 11) supply pins, while having independent non-inverting input, inverting input, and output pins.

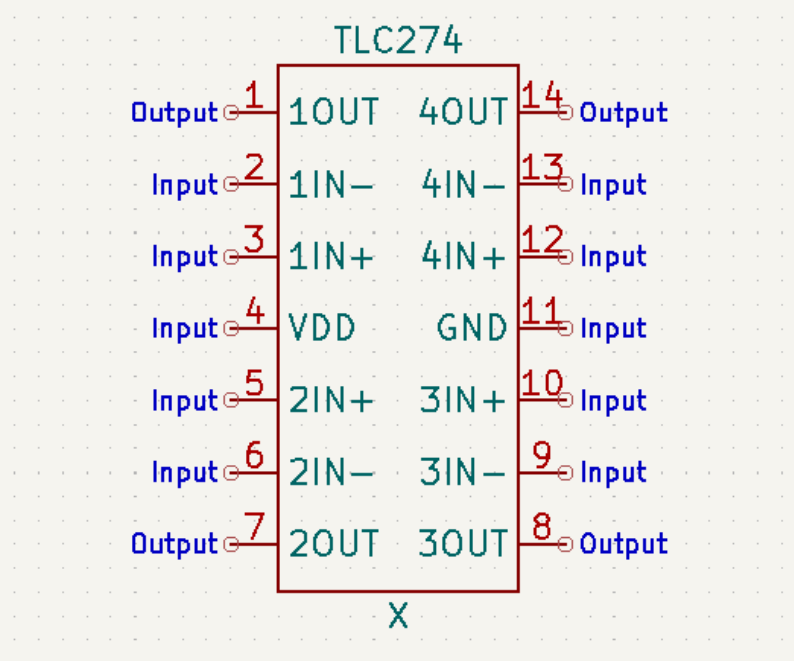

Figure 4.84: Pin Diagram of TLC274

4.17.3 Features

- Four independent operational amplifiers in a single package.
- Low-power CMOS input stage with high input impedance.
- Wide bandwidth with low input offset voltage.
- Output voltage swing to within a few hundred millivolts of either supply rail.

- Single-supply operation suitable for battery-powered systems.

4.17.4 Applications

- Signal conditioning in battery-powered instrumentation.
- Voltage comparators and threshold-detection circuits.
- Active filters and precision amplifier stages.
- Sensor interfacing requiring high input impedance.

4.17.5 Subcircuit Design

The internal architecture of the TLC274 was reconstructed at the transistor level in eSim, built around a CMOS differential input stage feeding an intermediate gain stage and an output driver. The four amplifier channels within the package were each implemented using this transistor-level structure, with their non-inverting input, inverting input, output, `VDD`, and `GND` terminals mapped to the corresponding package pins, and the complete arrangement encapsulated as a single reusable subcircuit representing the TLC274.

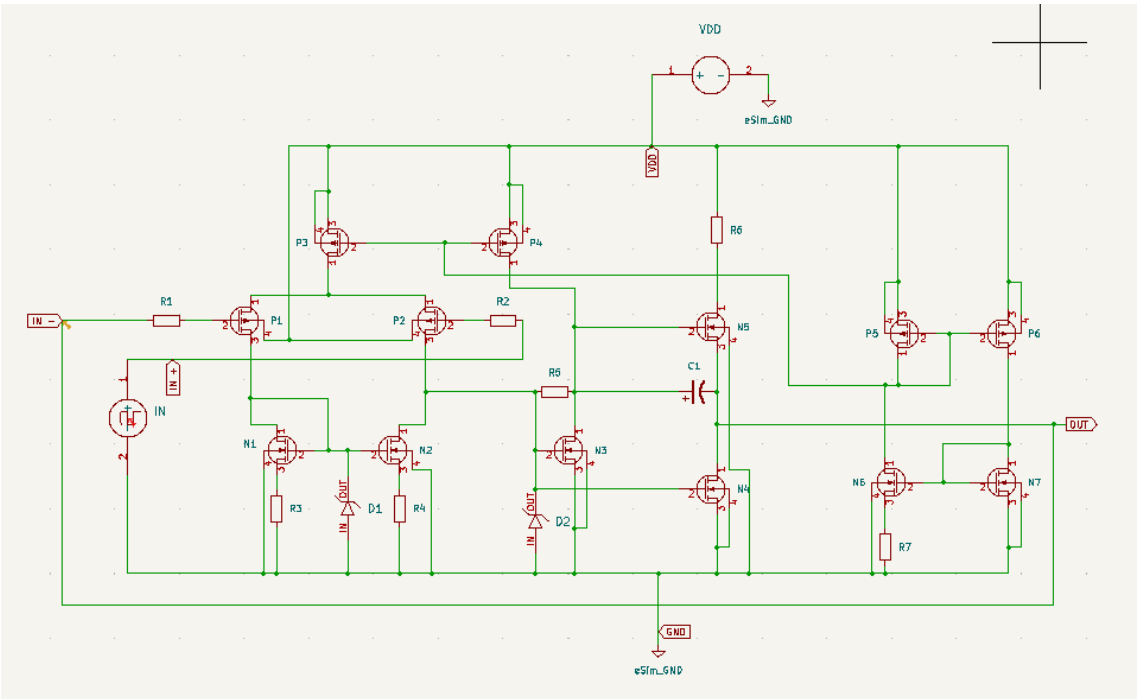

Figure 4.85: Subcircuit Schematic of TLC274

4.17.6 Test Circuit

A test circuit was designed using channel 1 of the TLC274 subcircuit, configured in an open-loop comparator arrangement. The non-inverting input (1IN+) was driven by a pulse voltage source, while the inverting input (1IN−) was held at a fixed reference voltage, with the device biased using `VDD` and `GND` supply sources. The output (1OUT) was probed to observe the comparator response as the non-inverting input voltage crossed the reference level.

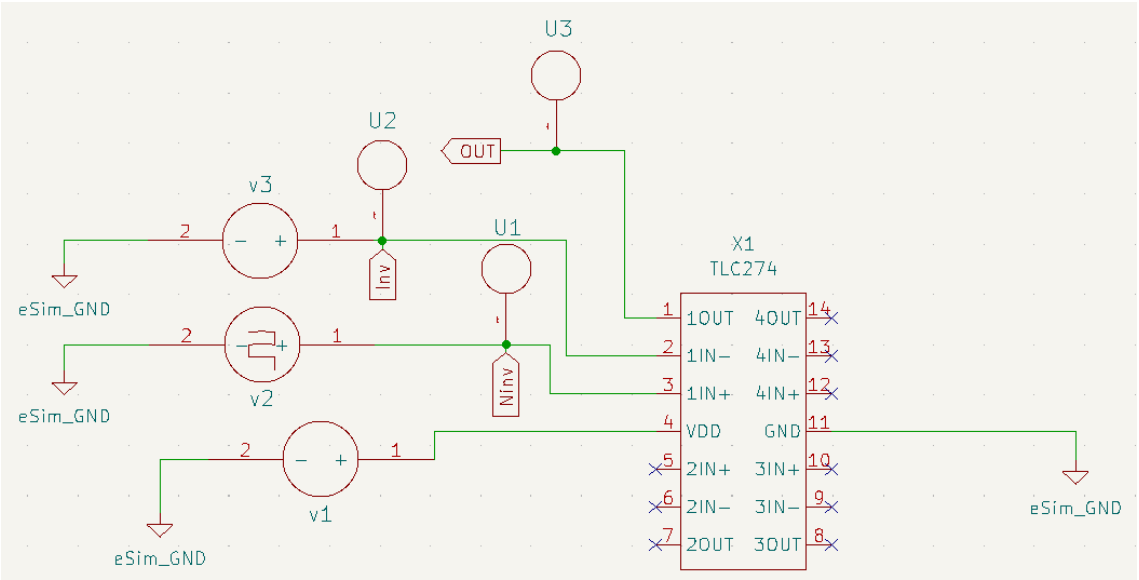

Figure 4.86: Test Circuit of TLC274

4.17.7 Output

The input and output responses were plotted together on a single combined waveform for direct comparison. The trace v("in+")+5 (offset for clarity) shows the pulsed non-inverting input signal, while v("in−") shows the fixed reference voltage applied to the inverting input, and v(out)+10 (offset for clarity) shows the resulting comparator output. The output switched to its high state whenever the non-inverting input exceeded the reference level applied at the inverting input, and returned to its low state whenever the non-inverting input fell below it, confirming that the developed subcircuit model correctly replicated the comparator behavior expected of the TLC274.

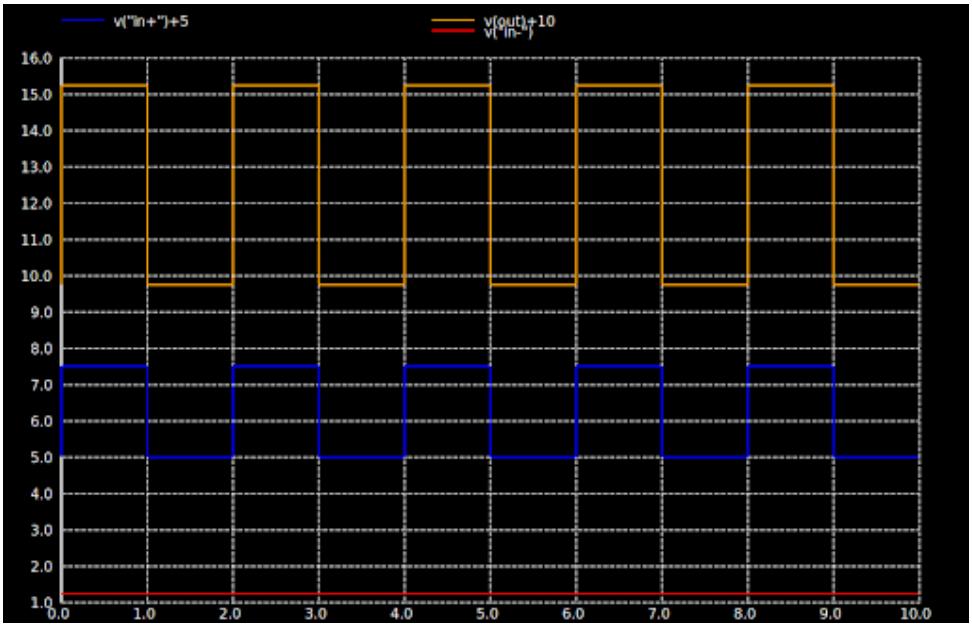

Figure 4.87: Input and Output Waveform of TLC274

4.18 CA3098 – Programmable Schmitt Trigger

4.18.1 Description

The CA3098 is a programmable Schmitt trigger IC that provides hysteresis-based threshold switching, with its upper and lower trigger points independently programmable through externally applied reference voltages. Unlike a conventional Schmitt trigger with fixed internal thresholds, the CA3098 allows the high reference (H_Ref) and low reference (L_Ref) levels, along with a bias current (I_bias), to be set externally, giving the designer control over the exact voltage window at which the output switches state. This flexibility makes it useful in applications where variable or precisely tunable hysteresis is required.

4.18.2 Pin Diagram

The CA3098 is available in an 8-pin package, with pins for the low reference input (L_Ref), bias current input (I_bias), output (Out), negative supply (V−), non-inverting input (+In), high reference input (H_Ref), positive supply (V+), and a compensation pin (CC).

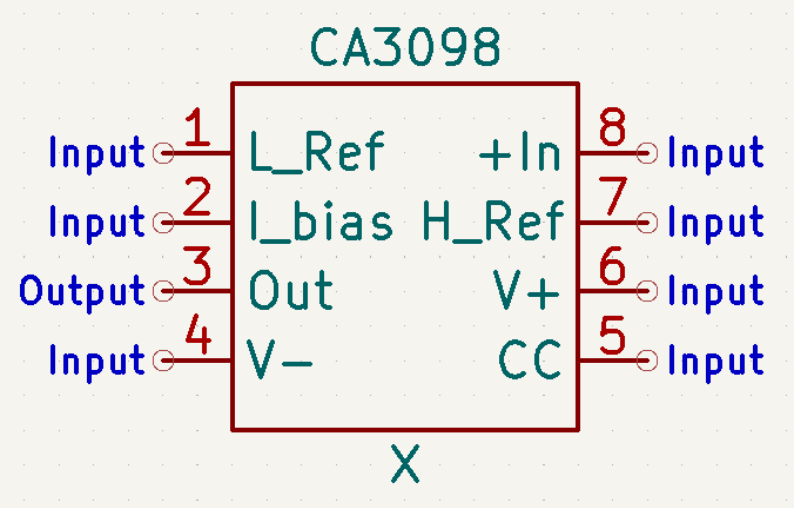

Figure 4.88: Pin Diagram of CA3098

4.18.3 Features

- Independently programmable high and low trigger threshold voltages.
- External bias current control for tunable switching characteristics.
- Hysteresis-based output switching, immune to noise near a single fixed threshold.
- Single output stage with well-defined high and low logic states.

4.18.4 Applications

- Level-detection circuits with adjustable hysteresis bands.
- Waveform shaping and noise-immune threshold detection.
- Voltage-controlled oscillators and pulse-width modulation circuits.
- Signal conditioning where variable switching thresholds are required.

4.18.5 Subcircuit Design

The internal architecture of the CA3098 was reconstructed at the transistor level in eSim, comprising a multi-stage BJT network built around an input differential pair, cross-coupled comparator stages referencing the externally applied L_Ref and H_Ref voltages, and an output driver stage. Two diodes (D1, D2) were included in the bias network to set the reference levels for the internal comparator stages, and current-mirror-like transistor arrangements were used to bias the internal stages consistently. The external terminals (L_Ref, I_bias, Out, V−, +In, H_Ref, V+, CC)

were mapped to the corresponding package pins, and the complete transistor-level circuit was encapsulated as a single reusable subcircuit representing the CA3098.

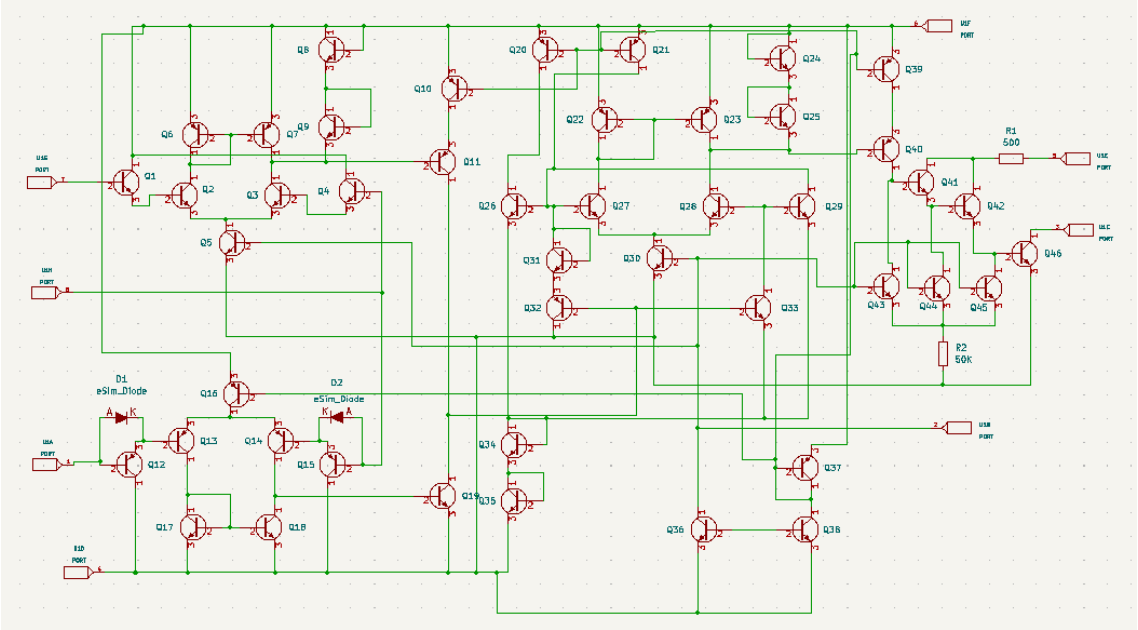

Figure 4.89: Subcircuit Schematic of CA3098

4.18.6 Test Circuit

A test circuit was designed using the CA3098 subcircuit, with the non-inverting input (+In) driven by a triangular voltage source to sweep the input voltage continuously between its minimum and maximum values. Separate fixed voltage sources were connected to the L_Ref and H_Ref pins to set the lower and upper trigger thresholds respectively, while V+ and V− supplied the device's operating rails. A bias resistor (R1 = 120 kΩ) was connected to the I_bias pin to set the internal bias current, and the output (Out) was probed to observe the resulting switching behavior.

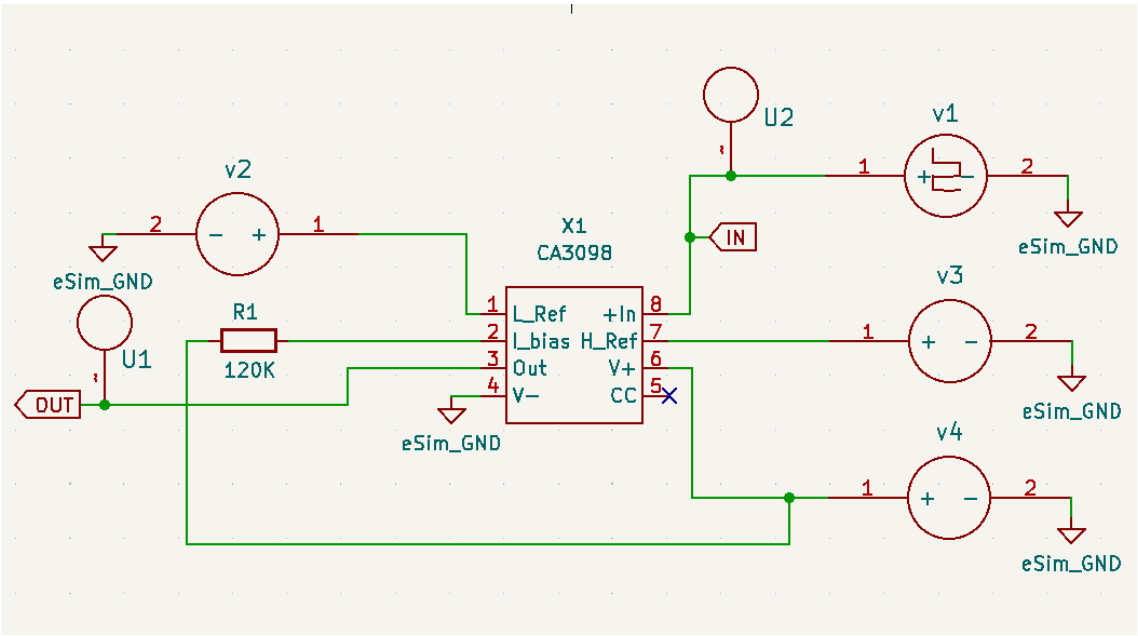

Figure 4.90: Test Circuit of CA3098

4.18.7 Output

The input waveform, v(in), shows the triangular voltage signal applied to the non-inverting input, ramping linearly between 0 V and 12 V and back over each cycle.

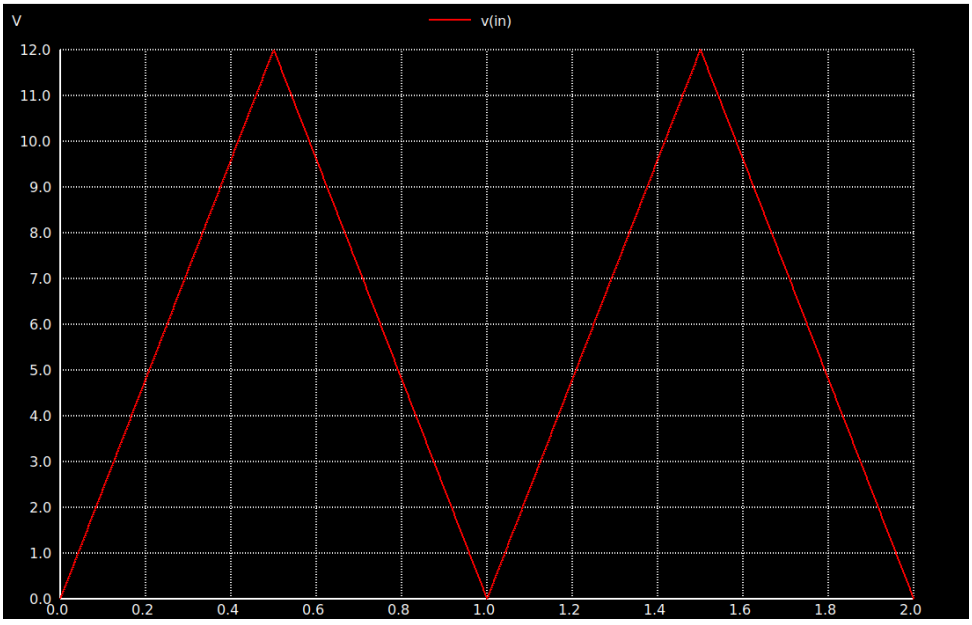

Figure 4.91: Input Waveform of CA3098

The resulting output waveform, v(out), remained low as the input ramped upward until it crossed the upper trigger threshold set by H_Ref, at which point the output switched high and remained latched high until the input, now ramping downward, fell below the lower trigger threshold set by L_Ref, causing the output to switch

back low. This hysteresis-based switching behavior, with distinct upper and lower trigger points, confirmed that the developed subcircuit model correctly replicated the programmable Schmitt trigger functionality of the CA3098.

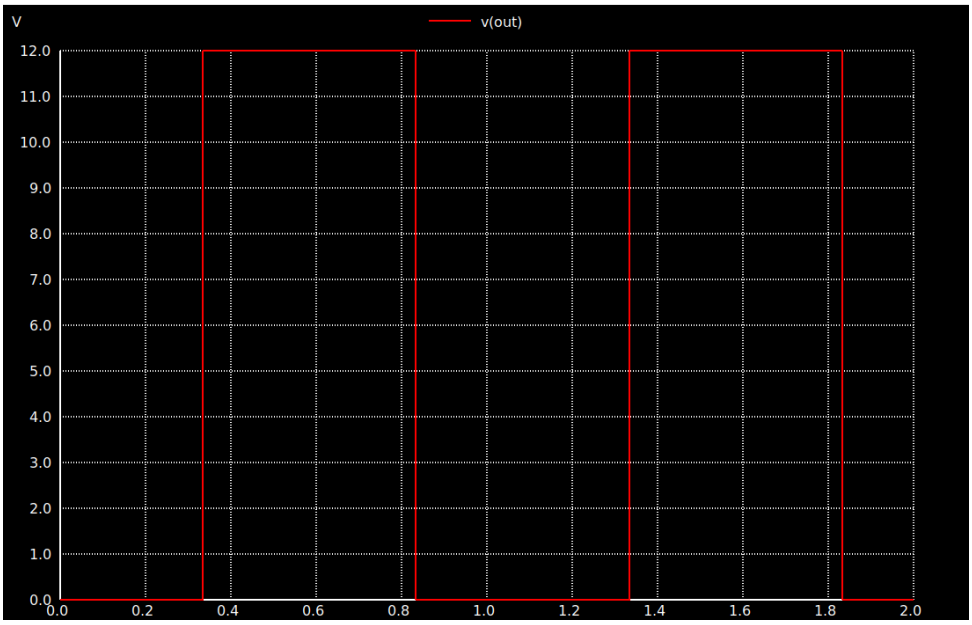

Figure 4.92: Output Waveform of CA3098

4.19 ICL8038

4.19.1 Description

The ICL8038 is a precision waveform generator IC, originally introduced by Intersil (now Renesas), capable of simultaneously producing sine, triangle, and square wave outputs from a single monolithic circuit with a minimum of external components. The operating frequency, adjustable from below 0.001 Hz to above 300 kHz, is set externally using resistors and a timing capacitor, and can additionally be swept or modulated through an external control voltage applied to the FM sweep input. Internally, the device generates a triangular waveform by charging and discharging a timing capacitor with constant currents, which is then used directly as the triangle output, converted through a non-linear shaping network to produce the sine output, and used to drive a comparator stage to produce the square wave output. Note that the ICL8038 was officially discontinued by Intersil in 2002, making it a legacy device that is not available in eSim's default library and therefore required subcircuit-level reconstruction for use in this project.

4.19.2 Pin Diagram

The ICL8038 is available in a 14-pin dual in-line package, with dedicated pins for sine wave adjustment (SIN_ADJ, present on both pins 1 and 12), sine output (SIN_OUT),

triangle output (TRI_OUT), frequency-setting inputs (FREQ_A, FREQ_B), positive and negative supply (V+, V−), FM bias and FM sweep inputs (FM_BIAS, FM_SWP), square wave output (SQ_OUT), and an external timing capacitor terminal (CEXT). Pins 13 and 14 are not internally connected (NC).

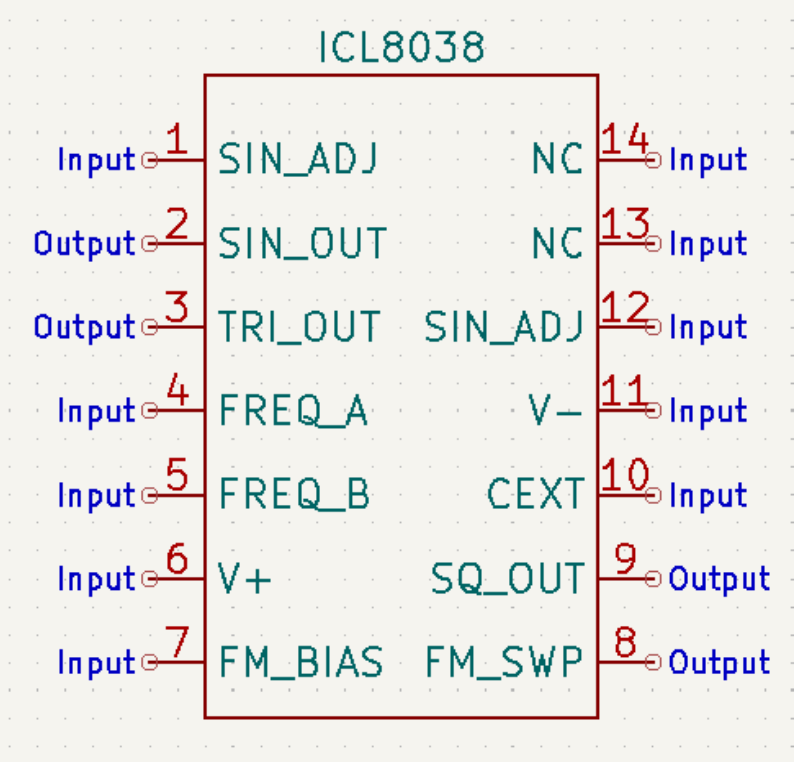

Figure 4.93: Pin Diagram of ICL8038

4.19.3 Features

- Simultaneous sine, triangle, and square wave outputs from a single IC.
- Wide frequency range, adjustable from sub-Hz to several hundred kHz using external R and C.
- Frequency modulation and sweeping capability via an external control voltage.
- Low frequency drift with temperature and supply voltage variation.
- Adjustable duty cycle and sine wave distortion via external trim resistors.

4.19.4 Applications

- Function and signal generators for test and measurement.
- Voltage-controlled oscillators (VCOs) in phase-locked loop systems.
- Frequency modulation and sweep generation circuits.
- Tone burst and audio signal generation.

4.19.5 Subcircuit Design

The internal architecture of the ICL8038 was reconstructed at the transistor level in eSim, comprising a current-source charge/discharge stage driving the timing capacitor to generate the core triangular waveform, a comparator-based threshold detection stage to derive the square wave output, and a non-linear sine-shaping network of cascaded differential BJT stages to convert the triangular waveform into a sine output. Current-mirror transistor arrangements were used extensively throughout the design to bias the internal stages and ensure symmetric charge/discharge currents, which is critical to the linearity of the triangle and sine outputs. The external terminals (SIN_ADJ, SIN_OUT, TRI_OUT, FREQ_A, FREQ_B, V+, FM_BIAS, FM_SWP, SQ_OUT, CEXT, V−) were mapped to the corresponding package pins, and the complete transistor-level circuit was encapsulated as a single reusable subcircuit representing the ICL8038.

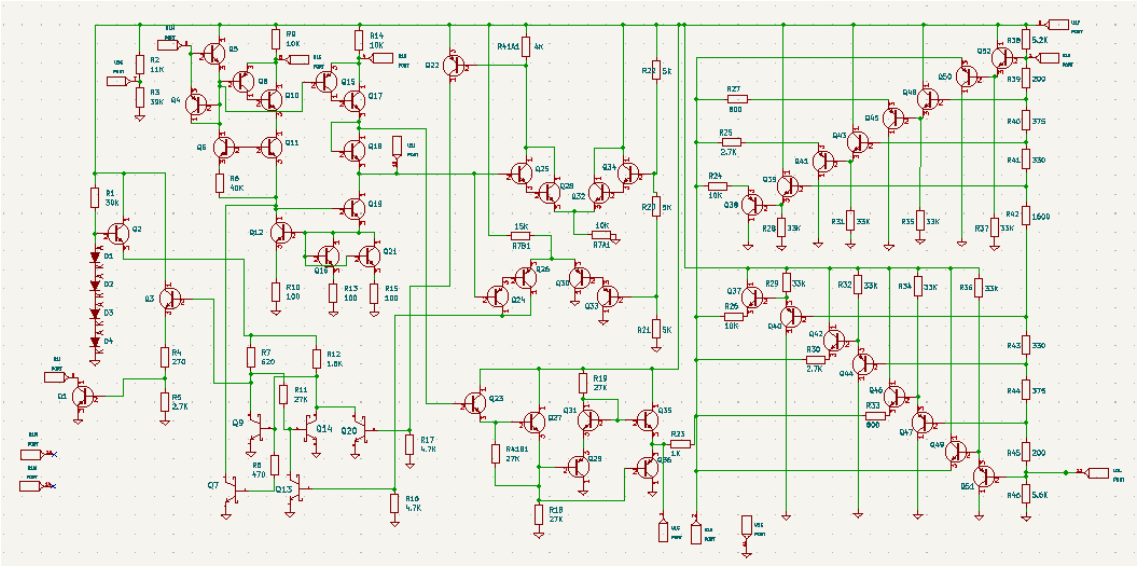

Figure 4.94: Subcircuit Schematic of ICL8038

4.19.6 Test Circuit

A test circuit was designed using the ICL8038 subcircuit, with a single DC voltage source (V1 = 20 V) supplying V+ as well as the frequency-setting inputs FREQ_A and FREQ_B through resistors R1 and R2 (10 kΩ each), which together set the charge and discharge current for the internal timing capacitor. A timing capacitor C1 (3.3 nF) was connected between CEXT and V− to set the oscillation frequency, while resistor R5 (82 kΩ) was connected from V− to the SIN_ADJ pin to condition the sine-shaping network. Since the SQ_OUT pin has an open-collector output stage, a pull-up resistor R6 (10 kΩ) was connected from SQ_OUT to the supply to obtain a valid logic-high level. The SIN_OUT, TRI_OUT, and SQ_OUT pins were each probed to observe the three simultaneously generated waveforms.

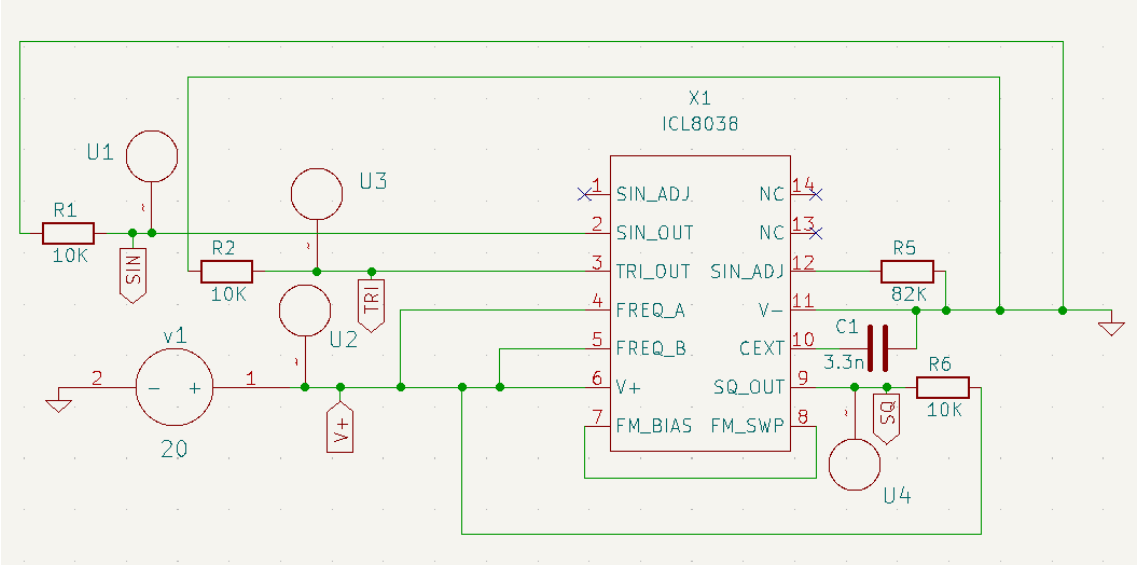

Figure 4.95: Test Circuit of ICL8038

4.19.7 Output

Since the ICL8038 is biased using a constant DC supply rather than a time-varying input, the input waveform simply confirms the fixed 20 V DC level applied at V+, denoted v("v+"), remaining steady throughout the simulation.

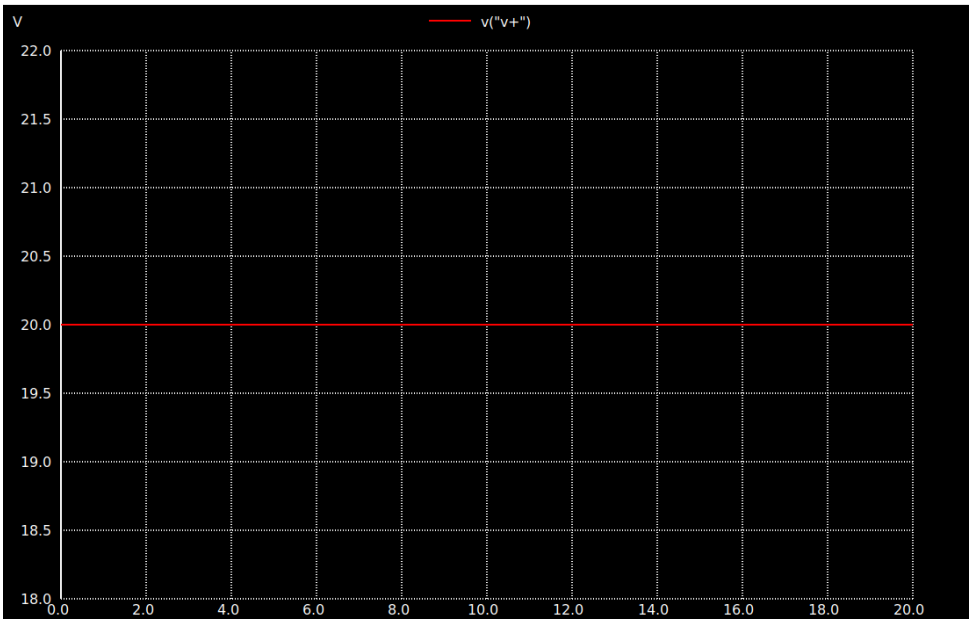

Figure 4.96: Input (DC Supply) Waveform of ICL8038

The resulting output waveforms show all three signals generated simultaneously from the same internal oscillator core: the square wave output (sq) switched cleanly between its low and high logic levels at the set oscillation frequency; the triangle wave output (tri+20, offset for clarity) rose and fell linearly in phase with the

square wave transitions; and the sine wave output (sin+30, offset for clarity) tracked the same frequency with a smoothly shaped sinusoidal profile derived from the triangular waveform. The consistent frequency and phase relationship among all three outputs confirmed that the developed subcircuit model correctly replicated the multi-waveform generation functionality of the ICL8038.

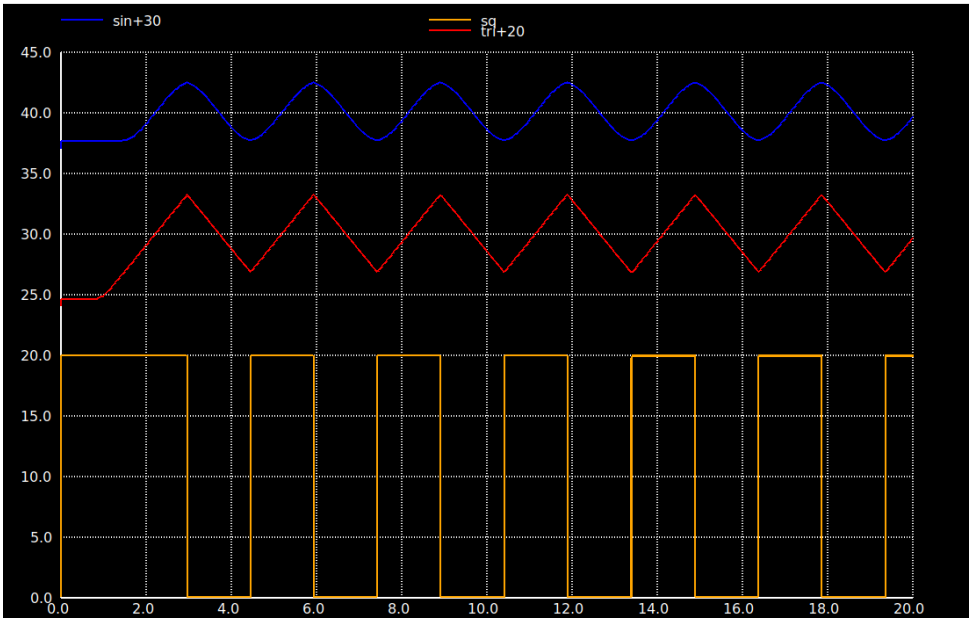

Figure 4.97: Output Waveforms of ICL8038

Chapter 5

Conclusion and Future Scope

5.1 Conclusion

This project set out to address a specific and practical limitation of eSim: the absence of accurate, datasheet-validated subcircuit models for a functionally diverse set of integrated circuits commonly encountered in digital and analog electronics education. Over the course of this work, eighteen ICs, spanning basic logic gates, data-selection devices, sequential storage and counting elements, arithmetic and parity circuits, bus interfacing and display-driving devices, and analog/mixed-signal components, were successfully modeled, tested, and validated within the eSim environment.

For each IC, the internal architecture was reconstructed based on manufacturer datasheets, either at the transistor level using BJT and CMOS device models for devices such as the SN7401, SN7402, TLC274, and CA3098, or using digital gate primitives for devices such as the 74HC386 and 8212, depending on the nature and complexity of the device. Each resulting subcircuit was encapsulated with a pin-accurate schematic symbol, subjected to a dedicated test circuit designed around realistic input conditions, and simulated using eSim's integrated Ngspice engine. In every case, the simulated behavior was verified against the expected truth table, timing diagram, or characteristic response documented in the corresponding datasheet, confirming the functional correctness of the developed models.

This work demonstrates that eSim, when combined with careful datasheet-driven subcircuit modeling, is capable of accurately replicating the behavior of a wide range of real-world integrated circuits, including devices with tri-state outputs, sequential clocked behavior, arithmetic logic, bus interfacing, and even analog comparator characteristics with externally programmable hysteresis. The successful completion of this project directly extends eSim's component library, making it more capable and self-sufficient as a free and open-source alternative to commercial EDA tools for digital and analog circuit simulation. In doing so, this work also contributes meaningfully to FOSSEE's broader mission of reducing the dependency of Indian academic institutions on costly proprietary software, by providing reusable, verified models that future students, educators, and developers can build upon.

5.2 Future Scope

While this project has successfully modeled and validated eighteen ICs, the work carried out here can be extended and built upon in several ways:

1. **Expansion of the IC library** — The eighteen ICs modeled in this project represent only a fraction of the commonly used TTL, CMOS, and analog ICs still missing from eSim's default library. Future work could extend this effort to additional logic families, more complex sequential circuits (such as programmable counters, universal shift registers, or memory elements), and further analog devices such as timers, voltage regulators, and instrumentation amplifiers.

2. **Integration into eSim's public subcircuit library** — The models developed in this project could be formally packaged, documented, and contributed to eSim's official component library, making them directly accessible to all eSim users without requiring them to be rebuilt from scratch.

3. **Automated or semi-automated model verification** — The manual process of comparing simulation output against datasheet truth tables and characteristic curves could be supplemented with automated test scripts that systematically verify each subcircuit's behavior across a broader range of input conditions, improving both coverage and confidence in model accuracy.

4. **PCB-level validation** — Since eSim also supports PCB layout design through its KiCad integration, future work could extend beyond simulation-level validation to designing and testing physical PCB layouts incorporating these subcircuit models, further demonstrating their practical usability in real hardware design workflows.

5. **Higher-level system design using developed subcircuits** — With a growing library of verified building blocks, future projects could use these subcircuits as components in larger, more complex digital and mixed-signal systems, such as small microprocessor-adjacent designs, counters and display systems, or programmable logic demonstrations, showcasing the practical utility of the models developed here in composite designs.

6. **Community feedback and iterative refinement** — As these models are used more widely by students and educators, feedback from real-world usage could be incorporated to refine device accuracy, improve documentation, and address any edge cases not covered during the original validation process.

In summary, this project lays a solid foundation for the continued expansion of eSim's component ecosystem, and the methodology followed here, datasheet-driven reconstruction, subcircuit encapsulation, application-driven test design, and simulation-based validation, provides a repeatable framework that can guide future contributions to eSim well beyond the scope of this particular set of ICs.

Chapter 6

Bibliography

FOSSEE, "Free/Libre and Open Source Software for Education," IIT Bombay. https://fossee.in[Link]

eSim, "Open Source EDA Tool," FOSSEE, IIT Bombay. https://esim.fossee.in[Link]

eSim Resources, Manuals, and Spoken Tutorials, FOSSEE, IIT Bombay. https://esim.fossee.in/resour

eSim Source Code Repository, FOSSEE, IIT Bombay. https://github.com/FOSSEE/eSim[Link]

eSim User Manual (Version 2.5), FOSSEE, IIT Bombay. https://static.fossee.in/esim/installation-files/eSim$_M anual_2$.5.*pdf* [*Link*]

Ngspice, "Open Source Mixed-Level/Mixed-Signal Circuit Simulator." https://ngspice.sourceforge.io

KiCad EDA, "Open Source Electronics Design Automation Suite." https://www.kicad.org[Link]

Texas Instruments, "SN5401, SN7401 Quadruple 2-Input Positive-NAND Gates with Open-Collector Outputs," Datasheet. https://www.tvsat.com.pl/pdf/7/7401$_t i$.*pdf* [*Link*]

Texas Instruments, "SN5402, SN7402 Quadruple 2-Input Positive-NOR Gates," Datasheet. https://www.ti.com/lit/ds/symlink/sn7402.pdf[Link]

Texas Instruments, "SN5409, SN7409 Quadruple 2-Input Positive-AND Gates with Open-Collector Outputs," Datasheet. https://www.ti.com/lit/ds/symlink/sn74ls09.pdf[Link]

Texas Instruments, "SN5433, SN7433 Quadruple 2-Input Positive-NOR Buffers with Open-Collector Outputs," Datasheet. https://www.ti.com/lit/ds/symlink/sn74ls33.pdf[Link]

STMicroelectronics, "74HC386 Quad 2-Input Exclusive-OR Gate," Datasheet. https://datasheet4u.c Microelectronics/74HC386/524055[Link]

Nexperia, "74HC158 Quad 2-Input Multiplexer," Datasheet. https://assets.nexperia.com/documents sheet/74HC158.pdf[Link]

Nexperia, "74HC251 8-Input Multiplexer with Three-State Output," Datasheet. https://assets.nexperia.com/documents/data-sheet/74HC$_H CT$251.*pdf* [*Link*]

Texas Instruments, "SN74S137 3-Line to 8-Line Decoder/Demultiplexer with Address Latches," Datasheet. https://www.futurlec.com/74LS/74LS137.shtml[Link]

STMicroelectronics, "SN54164, SN74164 8-Bit Parallel-Out Serial Shift Registers," Datasheet. https://www.alldatasheet.com/datasheet-pdf/pdf/23048/STMICROELECTRONICS/74

Texas Instruments, “SN54LS393, SN74LS393 Dual 4-Bit Binary Counters,” Datasheet. https://www.futurlec.com/74LS/74LS393.shtml[Link]

Motorola, “SN54LS197, SN74LS197 Presettable Binary Counters,” Datasheet. https://www.alldatash pdf/pdf/5684/MOTOROLA/74LS197.html[Link]

National Semiconductor, “SN54LS283, SN74LS283 4-Bit Binary Full Adders with Fast Carry,” Datasheet. https://www.alldatasheet.com/datasheet-pdf/pdf/8069/NSC/74LS283.html

Nexperia, “74HC280 9-Bit Odd/Even Parity Generator/Checker,” Datasheet. https://www.nexperia. logic-ics/logic/specialty-logic/parity-generators-checkers/serie/74hc280-74hct280[Link]

Texas Instruments, “SN54LS245, SN74LS245 Octal Bus Transceivers with Three-State Outputs,” Datasheet. https://eater.net/datasheets/74ls245.pdf[Link]

Nexperia, “74HC4511 BCD-to-7-Segment Latch/Decoder/Driver,” Datasheet. https://assets.nexperi sheet/74HC$_H CT$4511.*pdf*[*Link*]

Intel Corporation, “8212 8-Bit Input/Output Port,” Datasheet. https://deramp.com/downloads/inte

Texas Instruments, “TLC274/TLC279 LinCMOS Quad Operational Amplifier,” Datasheet. https://www.ti.com/lit/ds/symlink/tlc279.pdf[Link]

Renesas Electronics (formerly Intersil/RCA/Harris), “CA3098 Programmable Schmitt Trigger,” Datasheet. https://www.renesas.com/en/document/dst/ca3098-datasheet[Link]

Renesas Electronics (formerly Intersil), “ICL8038 Precision Waveform Generator/-Voltage Controlled Oscillator,” Datasheet. https://www.renesas.com/en/document/dst/icl8038-datasheet[Link]