Semester Long Internship Spring 2026

On

Digital IP Creation in eSim

Submitted by

Sreekrishna K Sasidharan
VIT Bhopal University

Under the guidance of

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

May 11, 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 real-world 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

1 Introduction **1**
- 1.1 Background . . . 1
- 1.2 Overview of eSim . . . 1
  - 1.2.1 KiCad . . . 1
  - 1.2.2 Ngspice . . . 1
  - 1.2.3 KiCad to Ngspice Converter . . . 2
  - 1.2.4 Subcircuit Builder . . . 2
  - 1.2.5 NGHDL . . . 2
  - 1.2.6 NgVeri . . . 2
  - 1.2.7 Makerchip . . . 2
- 1.3 Objectives of the Project . . . 2
- 1.4 Methodology . . . 3

2 Literature Survey **4**
- 2.1 Motor Control and Actuation . . . 4
- 2.2 Feedback and Control Systems . . . 4
- 2.3 Mixed-Signal Processing . . . 5
- 2.4 System Safety and Power Management . . . 5
- 2.5 Digital Communication Protocols . . . 5
- 2.6 Summary . . . 5

3 Problem Statement and Approach **6**
- 3.1 Problem Statement . . . 6
- 3.2 Methodology and Top-Down Design Approach . . . 6

4 Implementation **7**
- 4.1 Soft-Start Inrush Limiter . . . 7
  - 4.1.1 Subcircuit Schematic . . . 7
  - 4.1.2 Test Bench Schematic . . . 8
  - 4.1.3 Plots . . . 8
- 4.2 Sigma-Delta Modulator . . . 9
  - 4.2.1 Subcircuit Schematic . . . 9
  - 4.2.2 Test Bench Schematic . . . 9
  - 4.2.3 Plots . . . 10
- 4.3 Space Vector PWM (SVPWM) . . . 10
  - 4.3.1 Block Diagram . . . 11
  - 4.3.2 Subcircuit Schematic . . . 11
  - 4.3.3 Test Bench Schematic . . . 12
  - 4.3.4 Plots . . . 12
- 4.4 Stepper Indexer . . . 12
  - 4.4.1 Block Diagram . . . 13

4.4.2 Subcircuit Schematic . . . 13
4.4.3 Test Bench Schematic . . . 14
4.4.4 Plots . . . 14
4.5 BLDC Commutator . . . 14
4.5.1 Block Diagram . . . 15
4.5.2 Subcircuit Schematic . . . 15
4.5.3 Test Bench Schematic . . . 16
4.5.4 Plots . . . 16
4.6 Proportional-Integral-Derivative (PID) Controller . . . 17
4.6.1 Subcircuit Schematic . . . 18
4.6.2 Test Bench Schematic . . . 18
4.6.3 Plots . . . 19
4.7 Quadrature Encoder Interface (QEI) . . . 19
4.7.1 Subcircuit Schematic . . . 20
4.7.2 Test Bench Schematic . . . 20
4.7.3 Plots . . . 21
4.8 Windowed Watchdog Timer (WWDT) . . . 21
4.8.1 Test Bench Schematic . . . 22
4.8.2 Plots . . . 22
4.9 Inter-Integrated Circuit (I2C) Protocol . . . 23
4.9.1 Test Bench Schematic . . . 24
4.9.2 Plots . . . 24
4.10 Pulse Width Modulation (PWM) Generator . . . 25
4.10.1 Test Bench Schematic . . . 26
4.10.2 Plots . . . 26

5 Conclusion and Future Scope **27**
5.1 Conclusion . . . 27
5.2 Future Scope . . . 27

6 Bibliography **28**

Chapter 1

Introduction

1.1 Background

The semiconductor industry and electronic system design have traditionally been dominated by expensive, proprietary Electronic Design Automation (EDA) software, creating a significant barrier to entry for students, researchers, and independent developers. To address this, the FOSSEE (Free/Libre and Open Source Software for Education) project was established. Supported by the National Mission on Education through Information and Communication Technology (ICT), under the Ministry of Education, Government of India, FOSSEE promotes the use of open-source software tools to improve the quality of technical education. The primary mission of FOSSEE is to reduce the dependency on proprietary software in academic institutions. It encourages the adoption of FLOSS (Free/Libre and Open Source Software) tools by developing new open-source alternatives, upgrading existing platforms to meet advanced research requirements, and fostering a community of developers. By participating in the development of open-source IP cores, this project directly contributes to FOSSEE's vision of democratizing access to professional-grade hardware design environments.

1.2 Overview of eSim

eSim (previously known as Oscad / FreeEDA) is a highly capable, open-source CAD tool that allows electronic system designers to create, test, and rigorously analyze complex circuits. A major advantage of eSim is its extensible and modular architecture, empowering users to modify the source code, add custom components, and tailor the workflow to specific project needs. eSim serves as an integrated platform that seamlessly connects schematic capture, mixed-signal simulation, and PCB layout. It achieves this by bundling several powerful open-source software components together into a single, cohesive ecosystem.

1.2.1 KiCad

KiCad is the core schematic capture and PCB design software integrated within eSim. It provides a highly interactive graphical interface where users can draw electronic circuits, manage extensive component libraries, and assign physical footprints. In the context of eSim, KiCad is primarily utilized to construct the testbenches and bridge the digital-analog components before generating the netlist required for simulation.

1.2.2 Ngspice

Ngspice is the primary simulation engine powering eSim. Based on the widely adopted Berkeley SPICE framework, Ngspice is a general-purpose circuit simulation program optimized for nonlinear DC, nonlinear transient, and linear AC analysis. It handles the heavy mathematical lifting

required to solve complex differential equations inherent in mixed-signal and analog node analysis, providing accurate waveform outputs for system verification.

1.2.3 KiCad to Ngspice Converter

Because KiCad is fundamentally a schematic tool and Ngspice is a simulation engine, a bridge is required to translate the visual schematic into a readable simulation matrix. The KiCad to Ngspice converter module within eSim processes the generated netlist, allowing users to inject specific analysis parameters (such as transient step times and stop times) and configure source details directly from the graphical interface without manually editing text-based SPICE netlists.

1.2.4 Subcircuit Builder

The Subcircuit Builder is a crucial module that promotes modular design and IP reuse. It allows a user to encapsulate a complex schematic or a compiled block of digital logic into a single, reusable subcircuit block (a "black box" with defined input and output pins). Once generated, these subcircuits can be saved to the eSim library and instantiated multiple times in larger, system-level designs.

1.2.5 NGHDL

NGHDL is an integrated module designed to facilitate mixed-signal circuit simulation by integrating VHDL (VHSIC Hardware Description Language) code with Ngspice. It allows designers to simulate digital logic described in VHDL alongside traditional analog SPICE components within the same testbench environment.

1.2.6 NgVeri

Similar to NGHDL, NgVeri is an advanced module integrated into eSim that brings Verilog, SystemVerilog, and Transaction-Level Verilog (TL-Verilog) support to the platform. By leveraging tools like Verilator behind the scenes, NgVeri converts digital HDL code into C++ models that Ngspice can natively interact with using analog-to-digital (ADC) and digital-to-analog (DAC) bridges. This module was heavily utilized in this project to simulate complex digital IP cores (such as motor controllers) alongside analog signal sources.

1.2.7 Makerchip

Makerchip is an innovative, cloud-based browser application developed by Redwood EDA. It provides a highly interactive IDE for digital circuit design and verification. Makerchip supports modern TL-Verilog, which significantly reduces the boilerplate code required for pipelined logic. While external to the local eSim installation, Makerchip code can be exported and integrated into eSim via the NgVeri module for full-system mixed-signal verification.

1.3 Objectives of the Project

The primary objective of this semester-long internship was to conceptualize, design, and validate a comprehensive suite of advanced Intellectual Property (IP) cores using the eSim environment. Specifically, the goals included:

- **IP Core Development:** To design 10 distinct, industry-standard digital and mixed-signal IP blocks, focusing heavily on motor control, feedback regulation, and embedded communication.

- **Library Expansion:** To successfully model these IPs using the Subcircuit Builder and NgVeri modules, officially expanding the FOSSEE eSim component library for future academic and industrial use.
- **Mixed-Signal Verification:** To develop robust schematic testbenches for each IP, utilizing standard analog stimuli to rigorously verify the digital logic via Ngspice transient analysis.
- **Open-Source Contribution:** To meticulously document the design approach, pin configurations, and simulation results to ensure these tools are highly accessible to the broader open-source hardware community.

1.4 Methodology

To achieve the project objectives, a rigorous, top-down engineering methodology was employed:

- **Literature and Specification Review:** For each target IP core, standard industry datasheets and control algorithms were analyzed to clearly define the required inputs, outputs, and functional state machines.
- **Logic Implementation:** The underlying digital logic for the modules was drafted using Verilog HDL, focusing on synthesizability and optimal gate logic.
- **Model Translation:** The Verilog designs were imported into eSim using the NgVeri tool, which generated the necessary Ngspice-compatible digital models.
- **Testbench Architecture:** In the KiCad schematic editor, robust test circuits were created. ADC and DAC bridge components were strategically placed to interface the generated digital IP blocks with continuous analog signal generators and voltage probes.
- **Simulation and Validation:** Transient simulations were executed. The resulting output waveforms were analyzed to confirm that the IP cores operated perfectly under continuous dynamic conditions, meeting all predefined specifications.

Chapter 2

Literature Survey

The development of robust Intellectual Property (IP) cores requires a foundational understanding of standard industry protocols, control algorithms, and mixed-signal processing techniques. Prior to the design and implementation phases in eSim, a comprehensive review of academic literature, standard datasheets, and open-source HDL implementations was conducted. This survey categorizes the reviewed concepts into five primary domains: Motor Control and Actuation, Feedback Systems, Mixed-Signal Processing, System Safety, and Digital Communication.

2.1 Motor Control and Actuation

Electric motor drives demand precise timing and power management to ensure efficient operation. A review of conventional Pulse Width Modulation (PWM) techniques highlighted its utility in driving simple DC motors and LEDs by varying the duty cycle of a fixed-frequency signal to control average power delivery. However, for 3-phase AC machines, standard PWM is suboptimal. Literature on Space Vector Pulse Width Modulation (SVPWM) demonstrates its superiority in modern motor control. SVPWM algorithms calculate the optimal switching states of an inverter by approximating a rotating reference voltage vector. Studies show that SVPWM provides a 15.5% higher utilization of the DC bus voltage and significantly reduces total harmonic distortion (THD) compared to sinusoidal PWM. For brushless architectures, the Brushless DC (BLDC) Commutator relies on electronic commutation rather than mechanical brushes. Literature on BLDC drives details the necessity of decoding signals from three discrete Hall-effect sensors to determine rotor position. This decoding facilitates a 6-step commutation sequence that sequentially energizes the stator coils to maintain continuous torque. Conversely, for precise open-loop positioning, Stepper Motor Indexers translate high-level step and direction signals into specific phase excitation sequences. Advanced indexer designs incorporate microstepping logic, which uses sine/cosine current profiles to divide standard motor steps into smaller increments, reducing resonance and mechanical noise.

2.2 Feedback and Control Systems

Achieving stability in dynamic systems requires robust closed-loop control mechanisms. The Proportional-Integral-Derivative (PID) controller remains the industry standard for continuous control systems. Literature emphasizes the discretization of the PID algorithm for digital implementation, where the proportional term addresses current error, the integral term eliminates steady-state error, and the derivative term dampens oscillations by predicting future error trends. To supply accurate positional and velocity data to these controllers, a Quadrature Encoder Interface (QEI) is essential. A review of digital motion control systems details how QEIs decode Phase A, Phase B, and Index pulses from optical or magnetic encoders. By analyzing the phase relationship (typically 90 degrees out of phase) between signals A and B, the QEI accurately

determines both the direction of rotation and the absolute/relative position count, which is a prerequisite for advanced servo applications.

2.3 Mixed-Signal Processing

High-resolution analog-to-digital conversion in modern embedded systems frequently utilizes Sigma-Delta Modulators. A review of mixed-signal IC design principles reveals that Sigma-Delta modulation leverages two key concepts: oversampling and noise shaping. By sampling the analog input signal at a frequency significantly higher than the Nyquist rate and utilizing an integrator and a 1-bit quantizer in a feedback loop, the modulator pushes quantization noise into higher frequency bands. This noise is subsequently filtered out by digital decimation filters, yielding a high-resolution digital output from a low-resolution quantizer.

2.4 System Safety and Power Management

Ensuring operational reliability in embedded hardware dictates the integration of autonomous safety mechanisms. Standard watchdog timers often fail to detect scenarios where a runaway software loop continuously clears the timer. The Windowed Watchdog Timer (WWDT) addresses this vulnerability.

Literature on safety-critical automotive and aerospace systems shows that WWDTs require the "feed" or "clear" signal to occur strictly within a predefined temporal window. If the timer is cleared too early or too late, the WWDT interprets it as a software fault and triggers a system reset. Furthermore, electrical safety during system initialization relies on Soft-Start Inrush Limiters. When powering capacitive loads or large motor drives, the initial current surge can exceed steady-state parameters, potentially damaging the power supply or triggering circuit breakers. Literature on power electronics details soft-start mechanisms that gradually ramp up the applied voltage or PWM duty cycle, thereby limiting the inrush current and minimizing thermal and electrical stress on the components.

2.5 Digital Communication Protocols

For intra-board communication between the main controller and peripheral IP cores, the Inter-Integrated Circuit (I2C) protocol is a widely adopted standard. Developed originally by Philips, the literature describes I2C as a robust, two-wire (Serial Data and Serial Clock), multi-master, multi-slave bus protocol. Studying its state machine specifications is crucial for implementing IP cores that require collision detection, clock stretching, and specific start/stop/acknowledge conditions for seamless data transfer.

2.6 Summary

The extensive review of these established architectures and algorithms formed the technical foundation of this internship. Understanding the mathematical and logical operations behind these systems was critical for effectively modeling them as reusable digital blocks and mixed-signal subcircuits within the open-source eSim and Makerchip environments.

Chapter 3

Problem Statement and Approach

3.1 Problem Statement

The design of modern embedded systems and avionics architectures relies heavily on the availability of robust, pre-verified Intellectual Property (IP) cores to accelerate the development of System-on-Chips (SoCs) and complex microcontrollers. Currently, there is a distinct need to populate the open-source eSim ecosystem with advanced, reliable IP blocks related to motor control, closed-loop feedback, system safety, and serial communication. The primary challenge addressed in this project is the conceptualization, design, and simulation of 10 complex mixed-signal and digital IP cores. These range from fundamental timing modules like PWM generators to sophisticated motor control algorithms such as Space Vector PWM (SVPWM) and Brushless DC (BLDC) Commutators. The objective is to ensure these IPs synthesize correctly, interface properly with analog components, and validate their functionality through rigorous mixed-signal transient simulation within the eSim environment.

3.2 Methodology and Top-Down Design Approach

The project follows a systematic, top-down design methodology to ensure each IP core meets standard industry specifications:

- **Specification Definition:** For each IP, the input/output ports, timing constraints, and logical state machines were rigorously defined based on standard datasheets.
- **Hardware Description and Logic Design:** Digital logic for the IP cores was developed using Verilog HDL. State machines, counters, and registers were coded to handle specific tasks such as I2C clock stretching, QEI pulse decoding, and SVPWM sector determination.
- **Mixed-Signal Integration via NgVeri:** A critical step involved using the NgVeri module within eSim. This tool bridges the digital and analog domains, converting the digital Verilog models into Ngspice-compatible mixed-signal blocks that can interact with analog voltage and current levels.
- **Schematic Capture:** Comprehensive test circuits were developed using the KiCad schematic editor integrated into eSim. ADC (Analog-to-Digital) and DAC (Digital-to-Analog) bridge components were meticulously configured to interface standard analog pulse sources with the digital inputs of the IP cores.
- **Simulation and Verification:** Transient analyses were configured and executed in Ngspice. The resulting waveforms were analyzed to verify duty cycles, state transitions, communication protocols, and fault-handling capabilities against the expected theoretical behavior.

Chapter 4

Implementation

4.1 Soft-Start Inrush Limiter

The Soft-Start Inrush Limiter is designed to restrict the initial surge of current when powering up capacitive loads or high-torque electric motors. Uncontrolled inrush currents can severely degrade power supply components and induce mechanical stress on drivetrains. The implemented IP core utilizes a digital ramp generator that gradually increases the duty cycle of the output drive signal over a configurable startup period. In the eSim environment, the Verilog block was integrated with analog threshold triggers to simulate the safe spin-up of a motor load, verifying that the peak transient current remained within safe operational limits.

4.1.1 Subcircuit Schematic

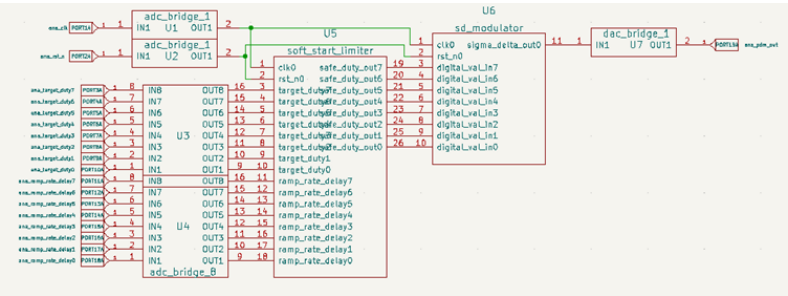

Figure 4.1: Subcircuit schematic showing the cascaded ADC bridges, soft-start core, sd_modulator core, and DAC bridge

4.1.2 Test Bench Schematic

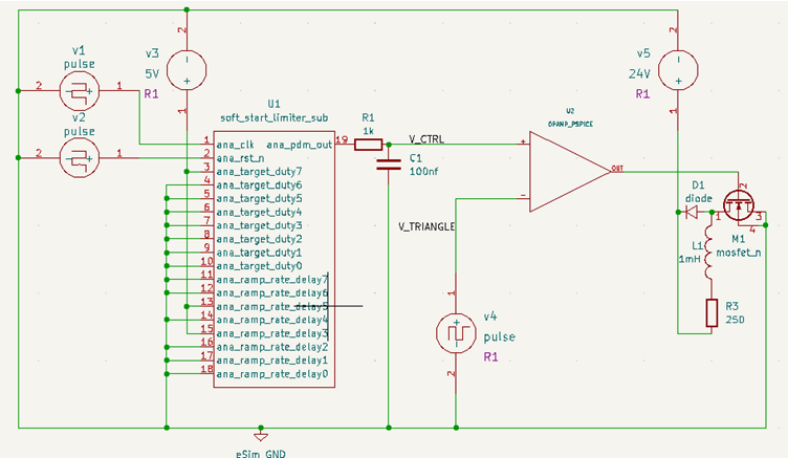

Figure 4.2: Top-level Testbench schematic showing pulse sources, RC filter, OPAMP_PSPICE comparator, and 2N7002 power stage

4.1.3 Plots

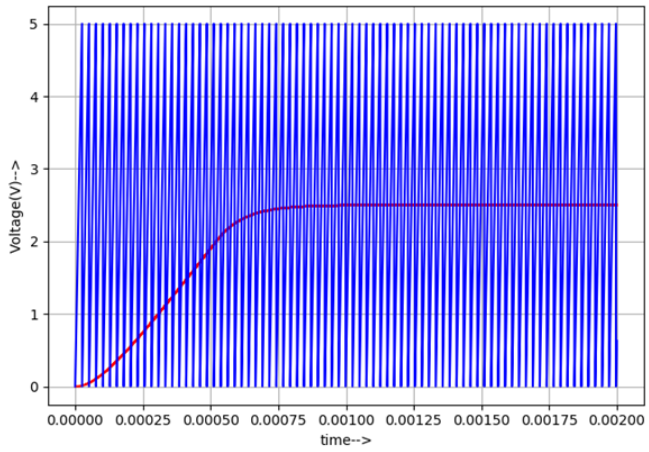

Figure 4.3: Plot of v_ctrl (red) and v_triangle (blue) demonstrating the 1ms analog ramp

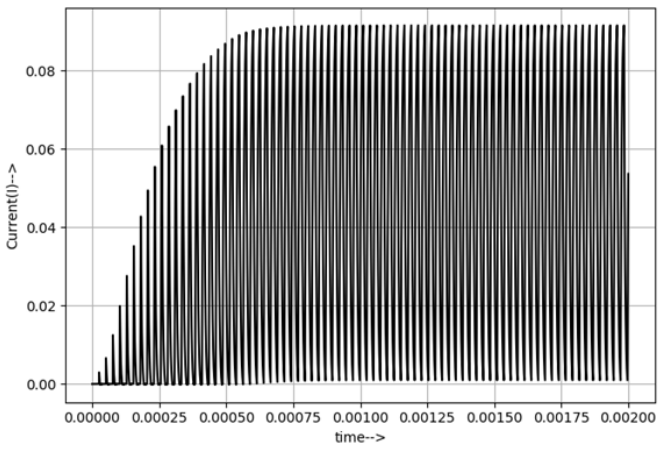

Figure 4.4: Plot of Inductor Current showing the smooth 0 to 90mA ramp

4.2 Sigma-Delta Modulator

The Sigma-Delta Modulator forms the backbone of high-resolution, low-frequency Analog-to-Digital Converters, making it highly suitable for precision sensor telemetry. The architecture employs a difference amplifier, an integrator, and a 1-bit comparator in a feedback loop. This configuration oversamples the input signal and shapes the quantization noise, pushing it into higher frequency bands. During implementation, NgVeri was utilized to model the digital filtering aspects, while Ngspice handled the continuous-time analog integration. The simulation verified the characteristic high-frequency 1-bit bitstream output corresponding to varying DC input levels.

4.2.1 Subcircuit Schematic

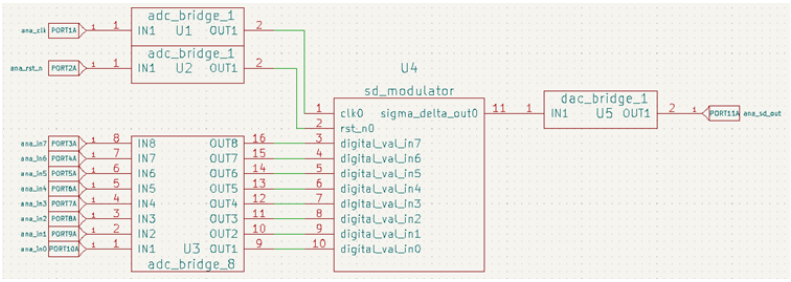

Figure 4.5: Internal Control Subcircuit Schematic utilizing ADC/DAC bridging and the NgVeri instantiated logic cores.

4.2.2 Test Bench Schematic

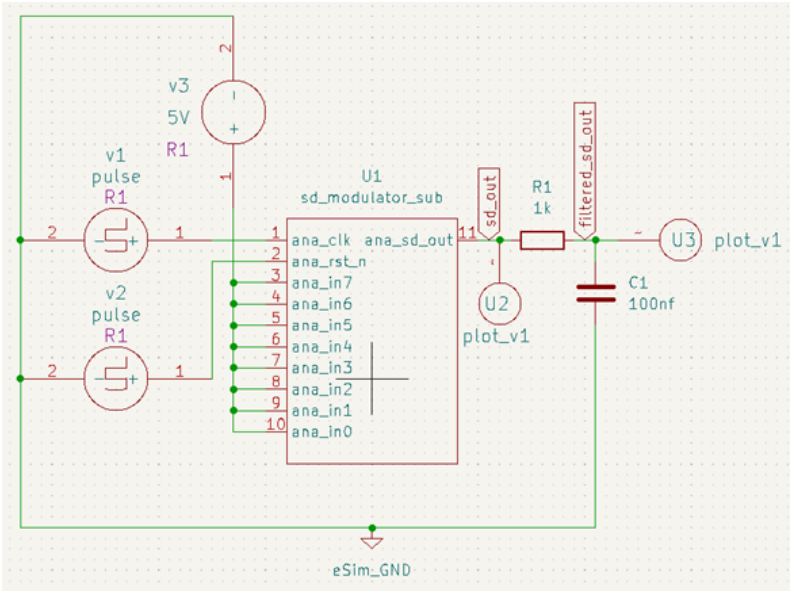

Figure 4.6: Top-Level Mixed-Signal Testbench Schematic

4.2.3 Plots

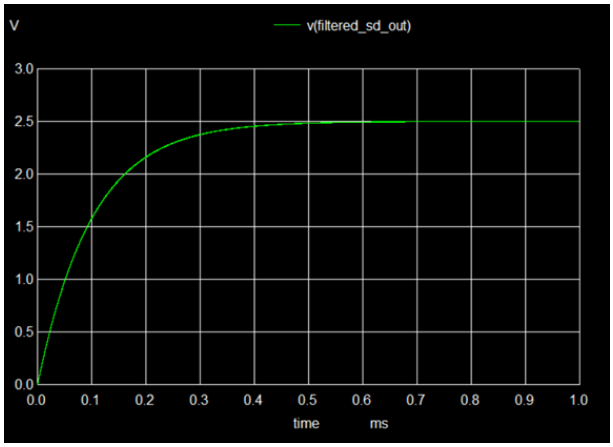

Figure 4.7: Filtered Analog Output Waveform (v(filtered_sd_out) at Mid-Scale)

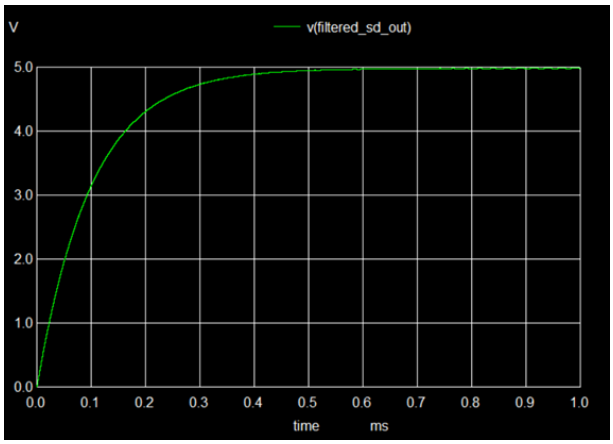

Figure 4.8: Filtered Analog Output Waveform (v(filtered_sd_out) at Full-Scale)

4.3 Space Vector PWM (SVPWM)

Space Vector Pulse Width Modulation (SVPWM) is an advanced algorithm essential for the efficient control of 3-phase AC machines, frequently utilized in electric propulsion and high-performance actuation systems. The IP core was designed to accept a voltage vector magnitude and angle, determine the active sector in the complex hexagonal plane, and calculate the optimal switching times (T1, T2, T0) for the inverter transistors. The Verilog implementation featured a complex finite state machine to manage the symmetric switching sequence. Ngspice simulations confirmed that the generated gate drive signals produced the correct center-aligned PWM waveforms required to approximate a rotating stator flux vector.

4.3.1 Block Diagram

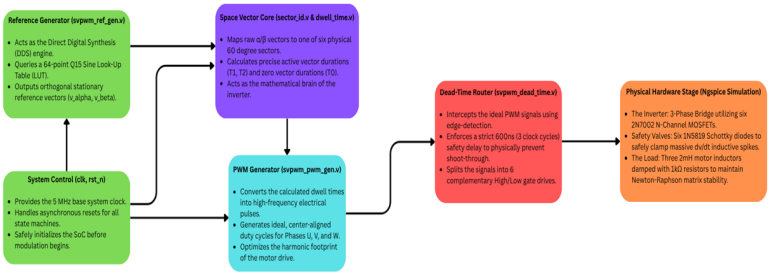

Figure 4.9: SVPWM Block Diagram

4.3.2 Subcircuit Schematic

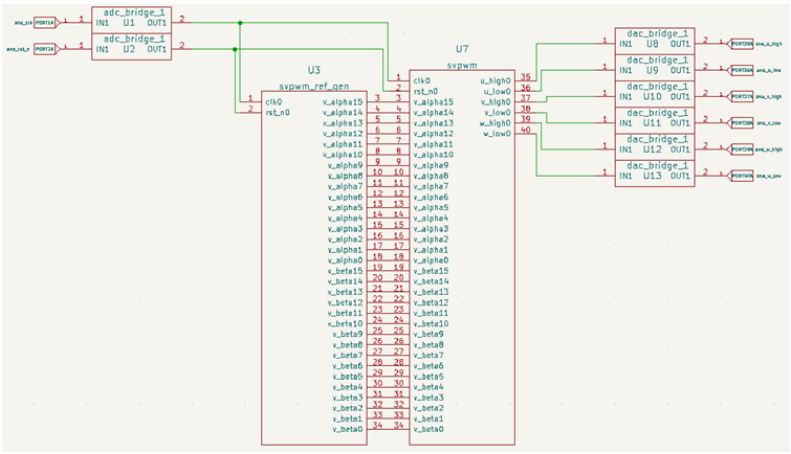

Figure 4.10: Internal Control Subcircuit Schematic utilizing ADC/DAC bridging and the NgVeri instantiated logic cores.

4.3.3 Test Bench Schematic

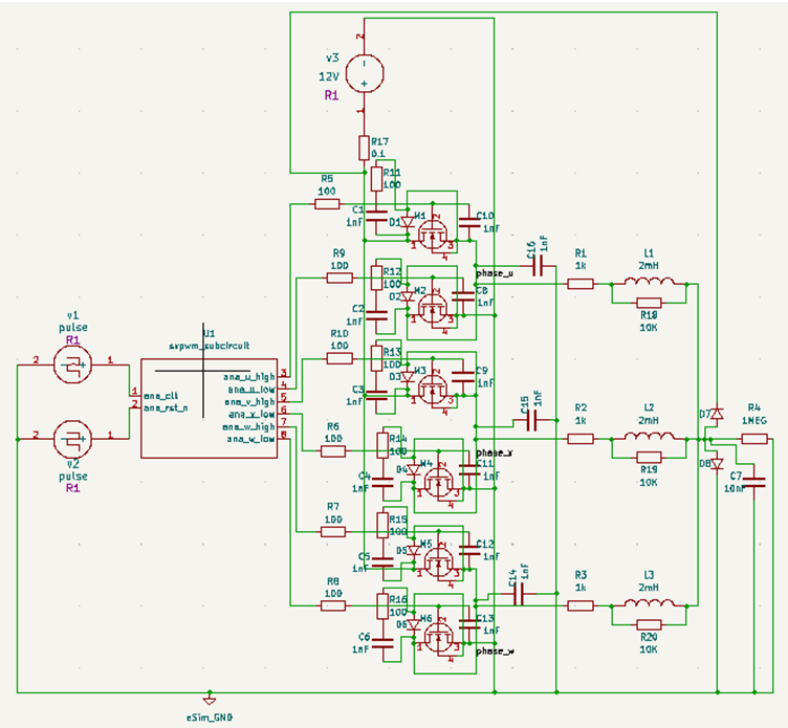

Figure 4.11: Top-Level Mixed-Signal Testbench Schematic demonstrating the 3-Phase Inverter and Motor Inductor Load.

4.3.4 Plots

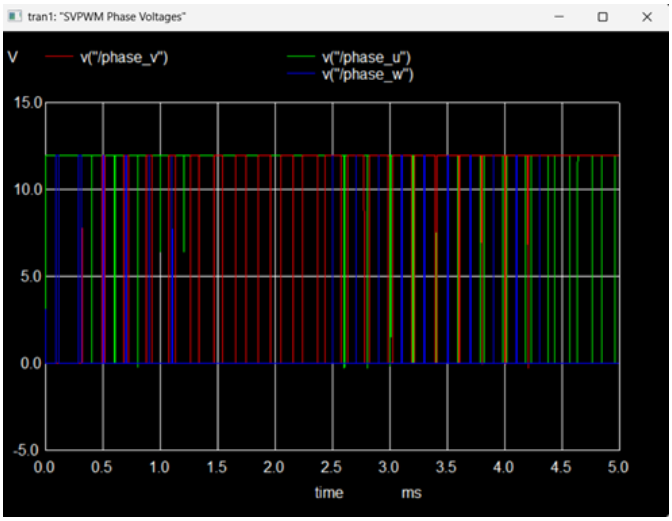

Figure 4.12: Combined 3-Phase SVPWM Voltages.

4.4 Stepper Indexer

The Stepper Indexer IP core abstracts the complex phase sequencing required to drive stepper motors, translating high-level "Step" and "Direction" digital pulses into 4-phase or 2-phase excitation signals. The core was designed to support full-step and half-step configurations via control pins. The internal logic features a bidirectional ring counter synchronized to the step

input clock. The simulation setup in eSim verified that phase outputs (A, B, C, D) energized in the correct sequence depending on the state of the direction pin, ensuring precise open-loop position control.

4.4.1 Block Diagram

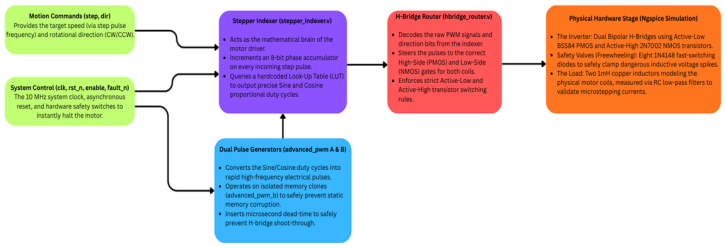

Figure 4.13: Stepper Indexer Block Diagram

4.4.2 Subcircuit Schematic

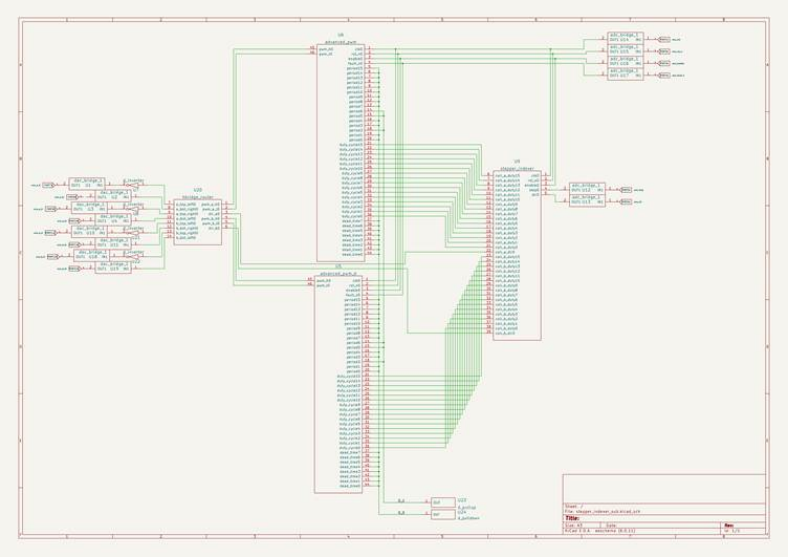

Figure 4.14: Internal Control Subcircuit Schematic utilizing ADC/DAC bridging and the NgVeri instantiated logic cores.

4.4.3 Test Bench Schematic

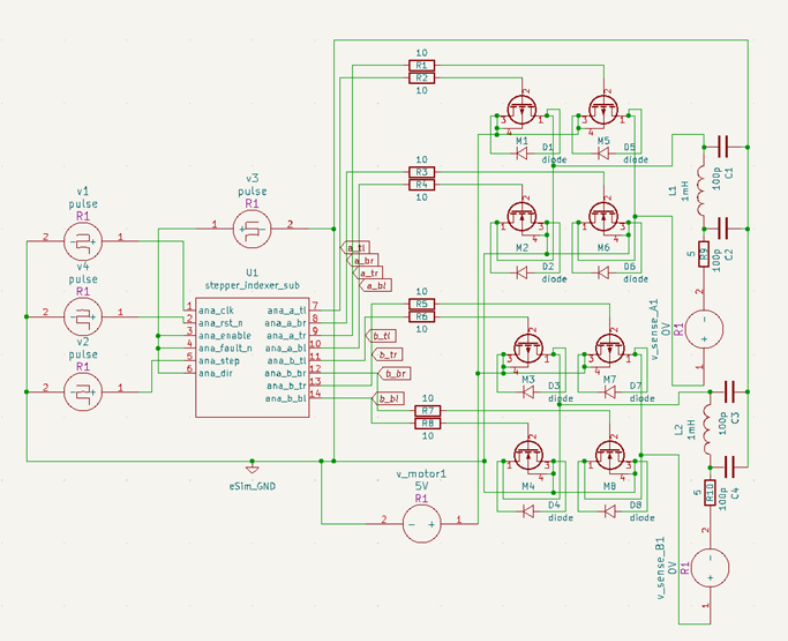

Figure 4.15: Top-Level Mixed-Signal Testbench Schematic demonstrating the Dual Bipolar MOSFET H-Bridges and Motor Inductor Load.

4.4.4 Plots

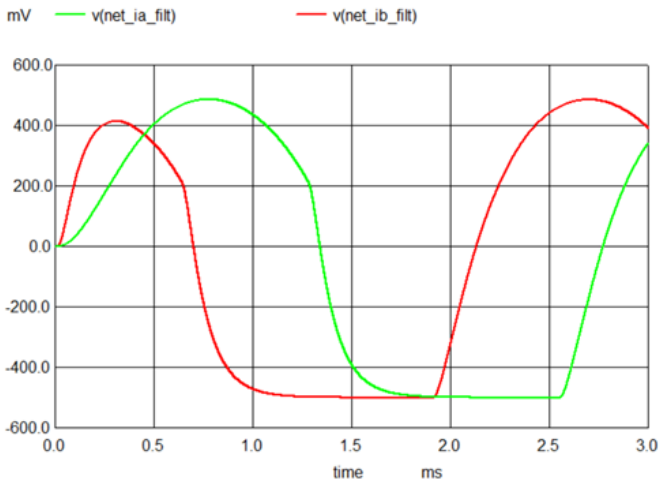

Figure 4.16: Filtered Phase Currents of Coil A (Green) and Coil B (Red) demonstrating perfect 90-degree offset Microstepping.

4.5 BLDC Commutator

To drive Brushless DC (BLDC) motors, continuous rotor position feedback is required to electronically commutate the stator phases. The BLDC Commutator IP core decodes the digital signals from three 120-degree electrically spaced Hall-effect sensors. The logic utilizes a lookup table to generate the standard 6-step commutation sequence, driving three high-side and three low-side inverter switches. The Verilog design includes dead-time insertion logic to prevent shoot-through faults in the inverter bridge. Simulations validated that the correct phase pairs

were energized sequentially as the simulated Hall states progressed through a full electrical revolution.

4.5.1 Block Diagram

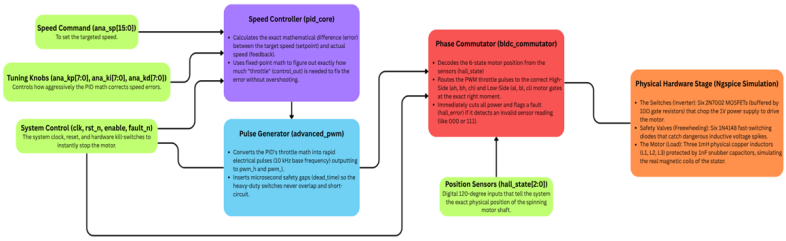

Figure 4.17: BLDC Commutator Block Diagram

4.5.2 Subcircuit Schematic

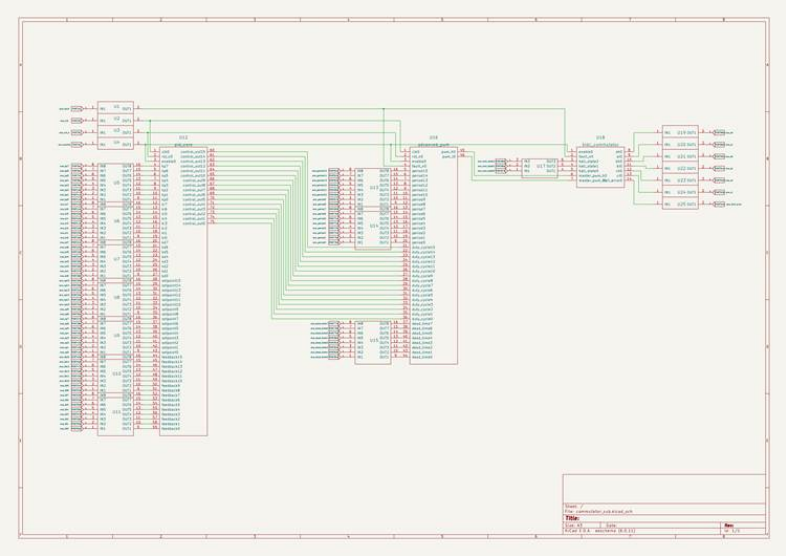

Figure 4.18: Internal Control Subcircuit Schematic utilizing ADC/DAC bridging and the NgVeri instantiated logic core.

4.5.3 Test Bench Schematic

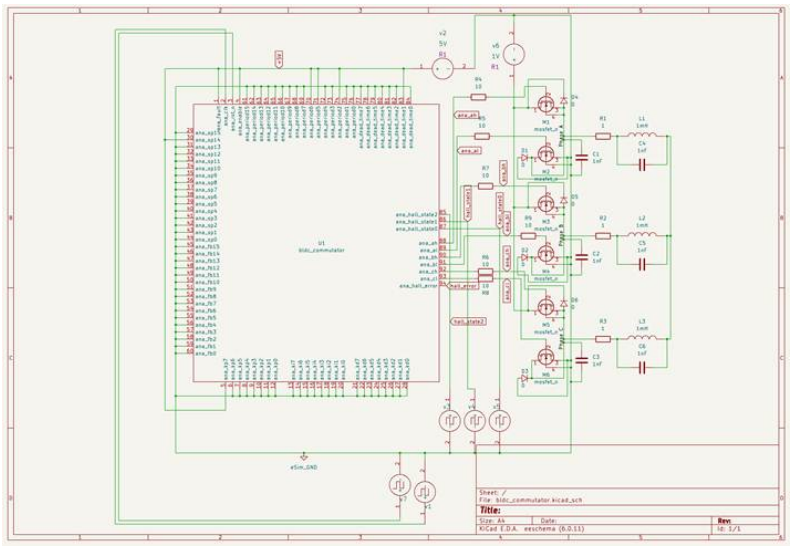

Figure 4.19: Top-Level Mixed-Signal Testbench Schematic demonstrating the 3-Phase MOSFET Inverter and Motor Inductor Load.

4.5.4 Plots

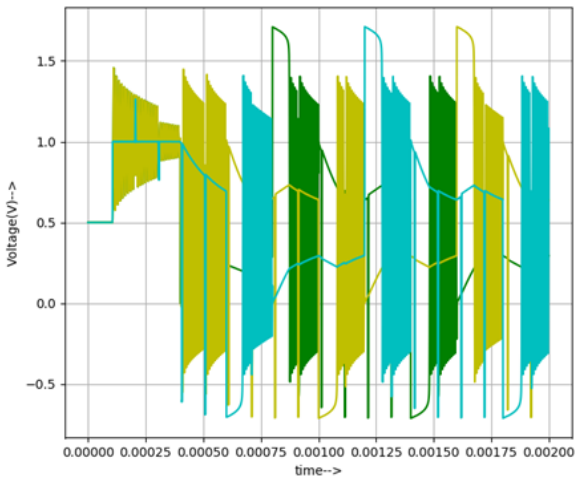

Figure 4.20: Phase A, B, and C Voltages demonstrating active PWM chopping and floating inductive states.

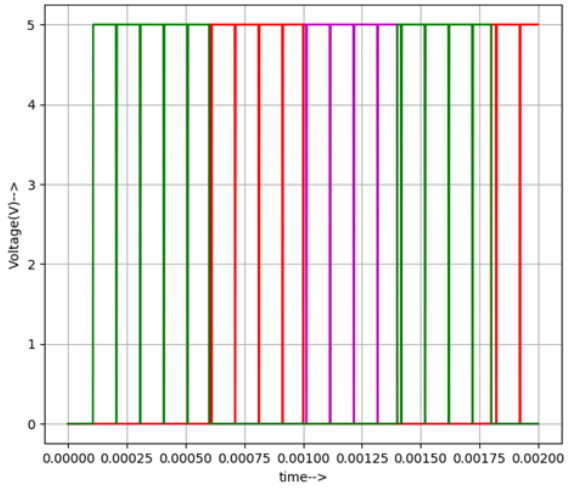

Figure 4.21: Unipolar PWM Scheme - High-Side Drive. The active High-Side gate is actively modulated at a 10 kHz switching frequency to precisely regulate the motor phase current.

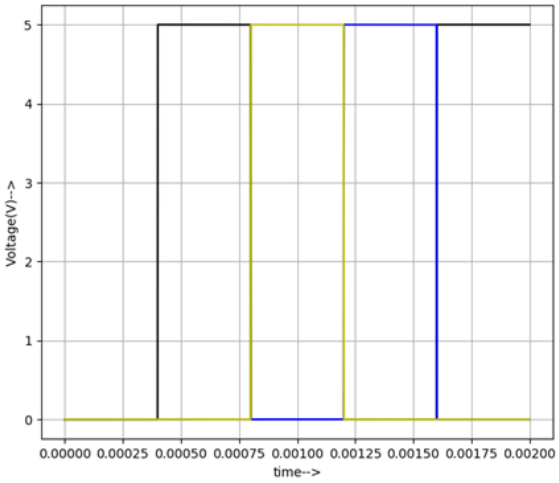

Figure 4.22: Unipolar PWM Scheme - Low-Side Drive. The complementary Low-Side gate is held continuously ON throughout its active sector to provide a stable, highly efficient freewheeling return path.

4.6 Proportional-Integral-Derivative (PID) Controller

The PID controller is the foundational algorithm for closed-loop dynamic systems, critical for tasks ranging from autonomous flap control in UAVs to industrial temperature regulation. A discrete-time digital PID IP core was developed in Verilog. It continuously samples the digital error signal, applying proportional gain to current error, integral gain to accumulated past error, and derivative gain to the rate of error change to compute a corrective output. The simulation in eSim utilized testbench loops to model a plant response, successfully demonstrating the controller's ability to drive the process variable toward the setpoint while minimizing overshoot and steady-state error.

4.6.1 Subcircuit Schematic

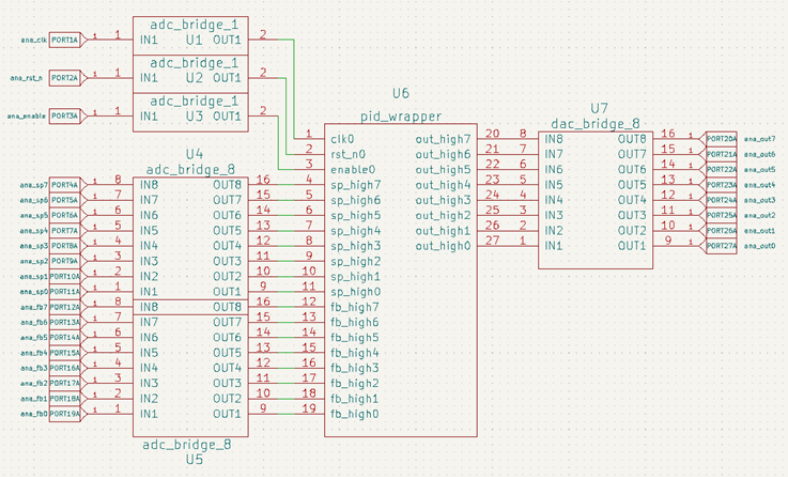

Figure 4.23: Custom 27-pin KiCad symbol and subcircuit routing, bypassing eSim array limitations.

4.6.2 Test Bench Schematic

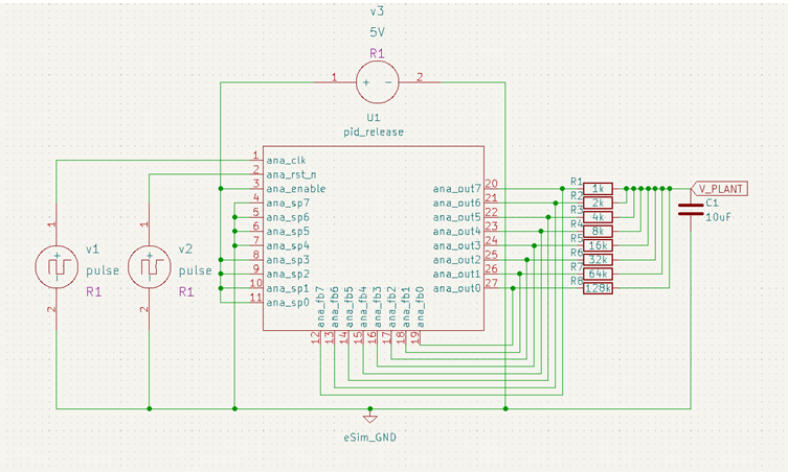

Figure 4.24: Mixed-Signal Testbench Schematic utilizing the encapsulated 27-pin PID subcircuit driving a 10 µF RC plant.

4.6.3 Plots

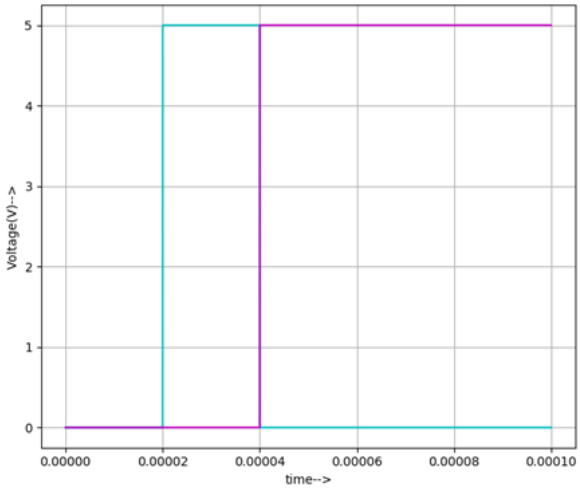

Figure 4.25: Internal logic and timing validation. The transient plot illustrates the core's step-by-step digital response to the input pulse sequencing. Once the reset and enable conditions are satisfied, the arithmetic pipeline actively evaluates the hardwired error. The trace confirms that the internal 32-bit integral accumulator safely accumulates while strictly obeying the anti-windup bounds, and the fractional scaling pipeline reliably right-shifts the raw summation into the saturated output range without bit-rollover or logic inversion.

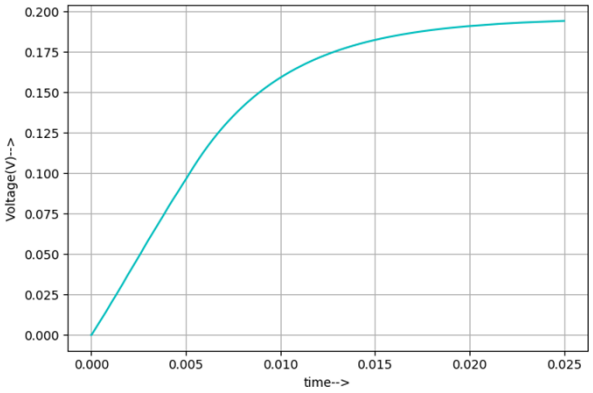

Figure 4.26: V_PLANT closed-loop step response. The digital PI logic successfully drives the analog RC plant to a steady-state asymptote without overshoot.

4.7 Quadrature Encoder Interface (QEI)

The QEI module tracks the rotation and position of motor shafts by interpreting the Phase A, Phase B, and Index pulse signals generated by quadrature encoders. The core's digital logic utilizes an edge-detection circuit to monitor state transitions on the A and B lines. Based on the sequence of these transitions, a direction flag is updated, and an internal position counter is either incremented or decremented (4x resolution tracking). The Index pulse acts as a zero-reference reset. Simulation plots verified that the 16-bit position counter accurately tracked simulated forward and reverse shaft rotations without dropping counts at high frequencies.

4.7.1 Subcircuit Schematic

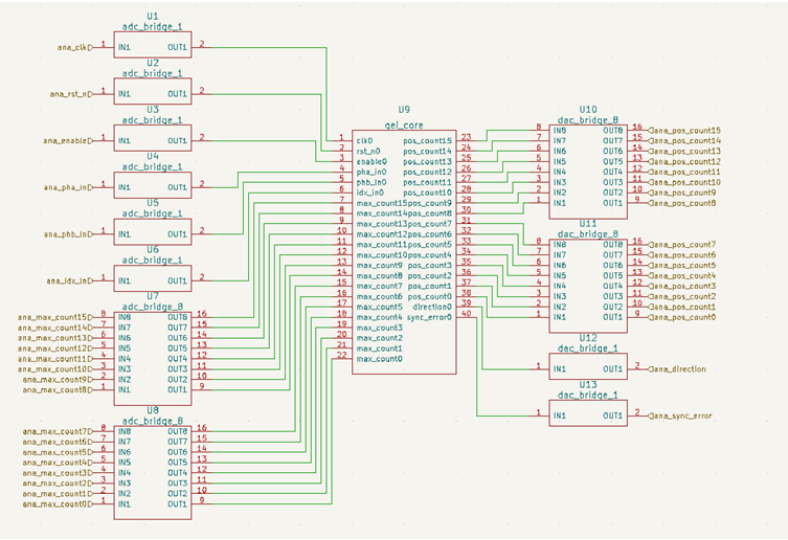

Figure 4.27: Custom 40-pin KiCad symbol encapsulating the Verilog core and analog-to-digital bridges.

4.7.2 Test Bench Schematic

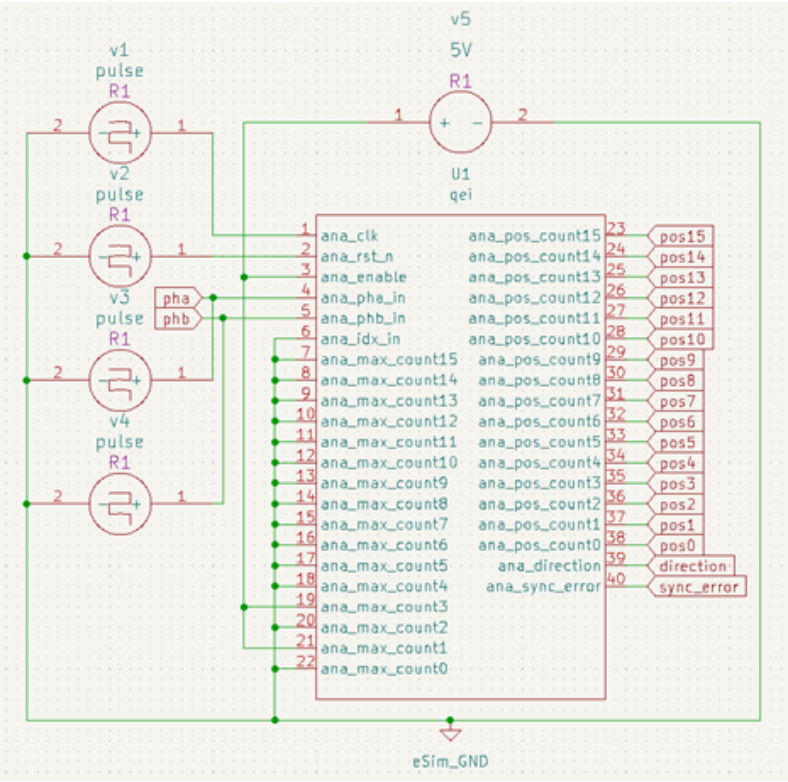

Figure 4.28: Mixed-Signal Testbench Schematic utilizing the encapsulated 40-pin QEI subcircuit, hardwired max_count rails, and simulated motor phase pulse generators.

4.7.3 Plots

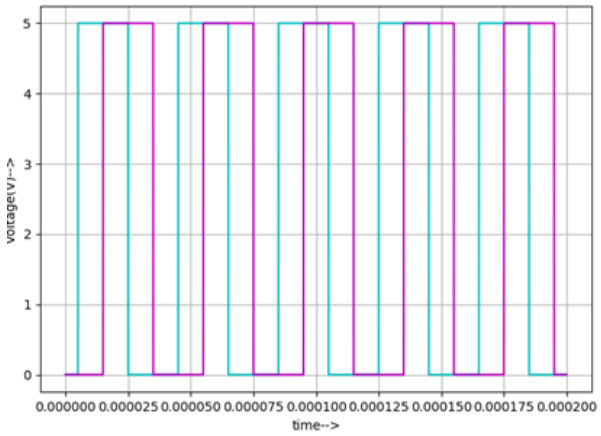

Figure 4.29: Forward Rotation Inputs. Phase A (Cyan) transitions at 5µs, leading Phase B (Magenta) at 15µs.

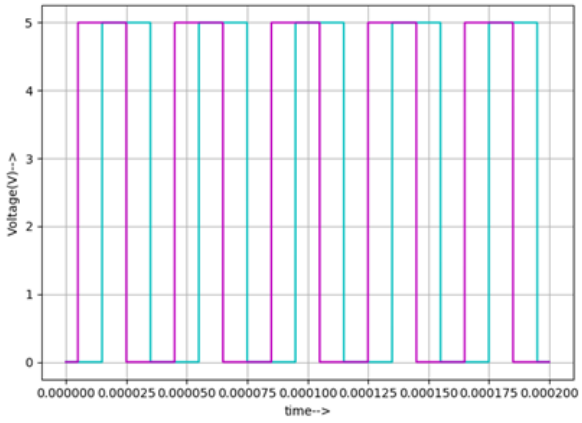

Figure 4.30: Reverse Rotation Inputs. Phase B (Magenta) transitions at 5µs, leading Phase A (Cyan) at 15µs.

4.8 Windowed Watchdog Timer (WWDT)

Designed for high-reliability and mission-critical environments, the Windowed Watchdog Timer (WWDT) offers superior fault detection compared to standard watchdogs. It requires the host microcontroller to clear the timer strictly within a specific time window (e.g., between 25% and 75% of the timeout period). The Verilog IP core tracks two internal counters to define this upper and lower boundary. If the clear signal is received before the window opens or after the window closes, a system reset interrupt is triggered. The Ngspice simulation rigorously tested both the "early clear" fault and the "late clear" fault to validate the reset generation.

4.8.1 Test Bench Schematic

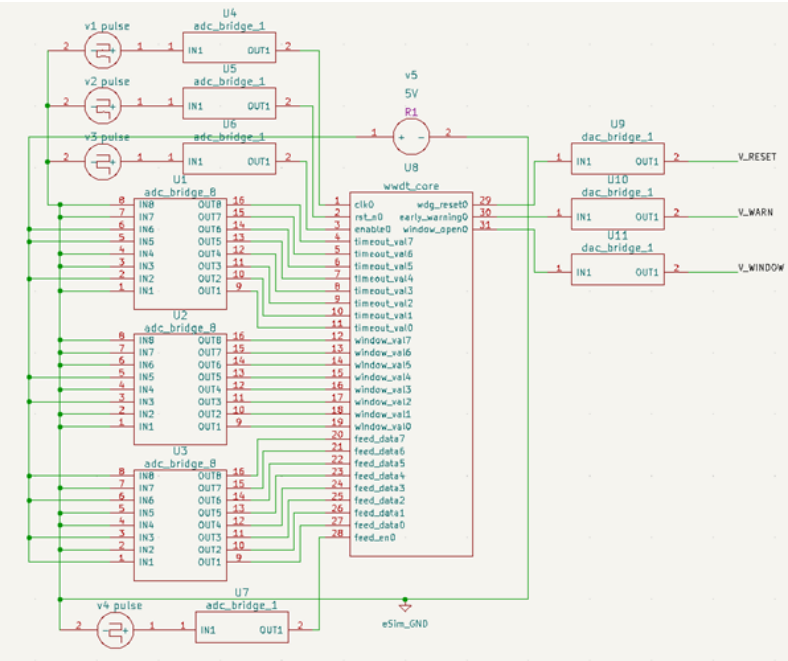

Figure 4.31: Mixed-Signal Testbench Schematic utilizing 8-bit parallel ADC bridges for configuration.

4.8.2 Plots

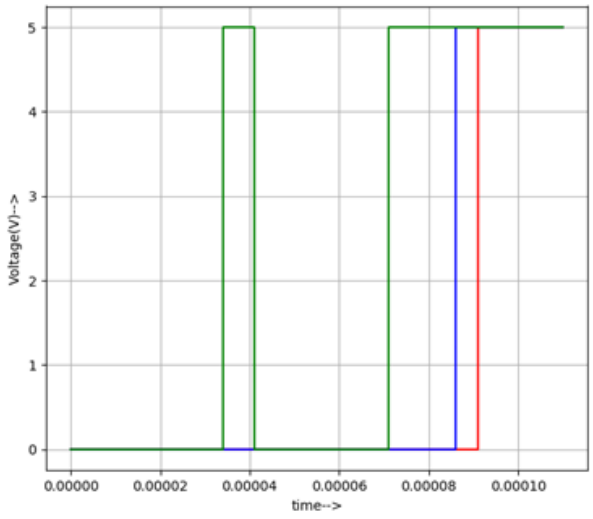

Figure 4.32: Ngspice Output Plot demonstrating a successful feed, followed by an Early Warning Interrupt and fatal starvation timeout.

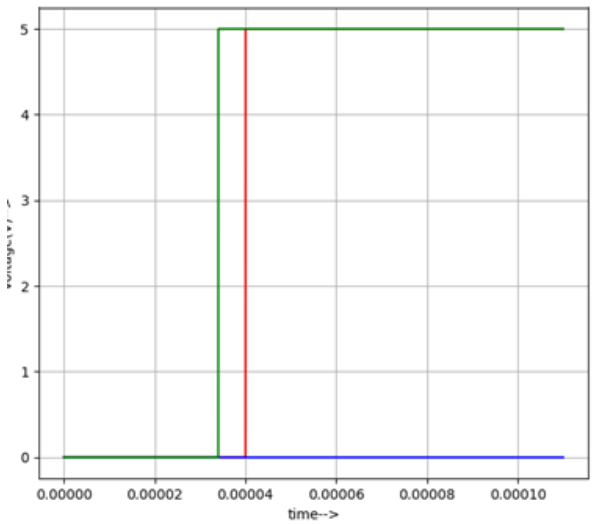

Figure 4.33: Ngspice Output Plot demonstrating instantaneous fatal reset upon receiving an invalid magic key.

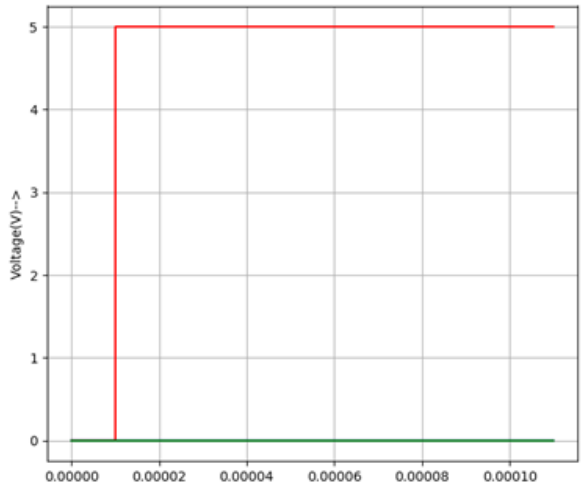

Figure 4.34: Ngspice Output Plot demonstrating closed-window violation detection

4.9 Inter-Integrated Circuit (I2C) Protocol

The I2C IP core implements the physical and data link layers of the ubiquitous two-wire serial communication protocol. Designed to act as a Master interface, the core handles the generation of the Serial Clock (SCL) and the bidirectional management of the Serial Data (SDA) line. The implementation features a complex state machine transitioning through IDLE, START, ADDRESS, ACK_WAIT, DATA_TX/RX, and STOP states. The simulation in eSim modeled a data writing sequence, verifying the correct generation of start conditions, 7-bit addressing, standard 100kHz clock timing, and appropriate handling of acknowledgment bits from a simulated slave device.

4.9.1 Test Bench Schematic

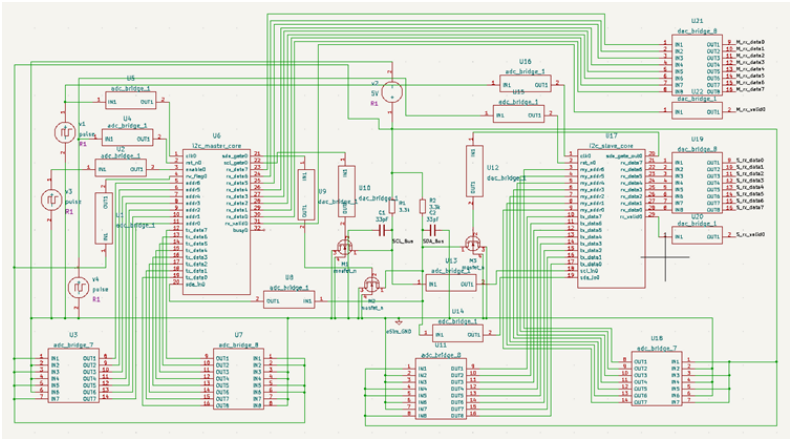

Figure 4.35: Mixed-Signal Testbench Schematic

4.9.2 Plots

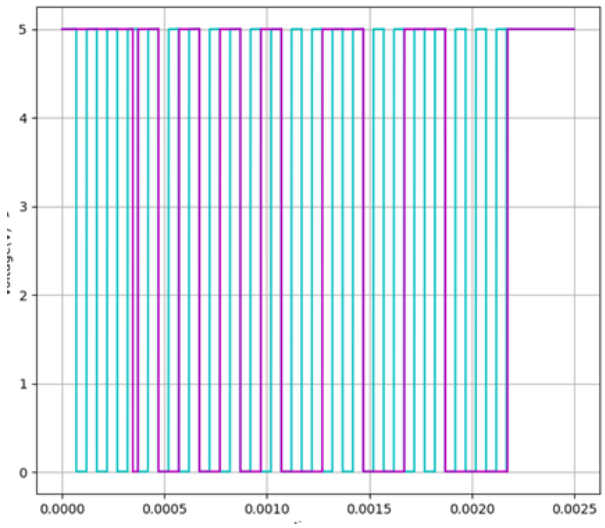

Figure 4.36: I2C Bus Waveforms for SDA(Magenta) & SCL(Cyan) during WRITE Transaction

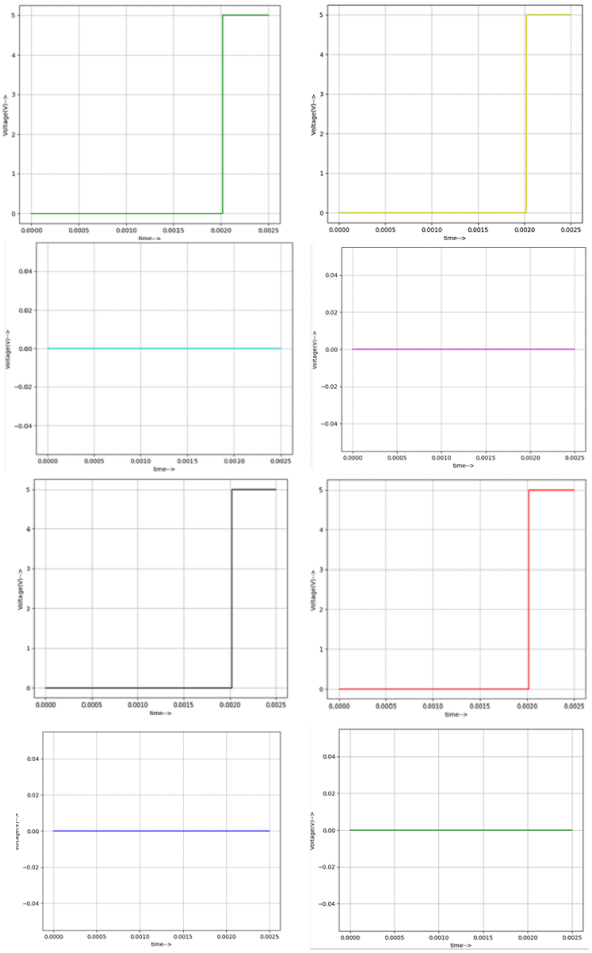

Figure 4.37: Receive Data Register captured during transaction. Note: The output DAC bridge routing was intentionally inverted (0 to 7) to automatically align with the I2C protocol's MSB-first chronological transmission. Reading the stacked physical bus lines from top to bottom at the latching phase correctly yields the binary payload.

4.10 Pulse Width Modulation (PWM) Generator

The generic Pulse Width Modulation (PWM) IP core provides a highly configurable square wave output used to control average power delivery to loads. The design centers around a free-running N-bit counter and a digital comparator. When the counter value is less than the input duty cycle register value, the output is driven HIGH; otherwise, it is driven LOW. The core was enhanced with programmable frequency prescalers. Simulation plots in Ngspice successfully validated the linear relationship between the digital input control vector and the analog duty cycle of the output waveform, confirming its viability for generic drive applications.

4.10.1 Test Bench Schematic

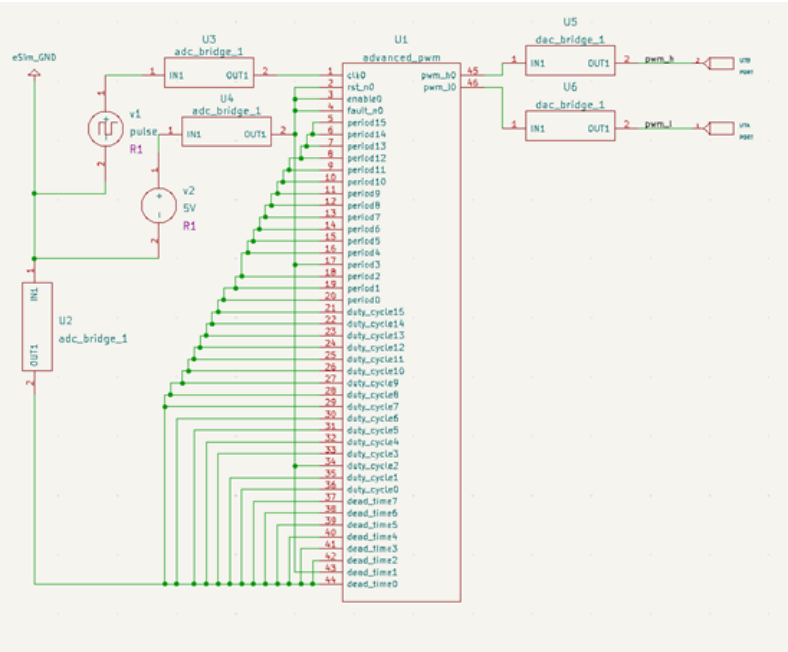

Figure 4.38: Mixed-Signal Testbench Schematic utilizing ADC/DAC bridging.

4.10.2 Plots

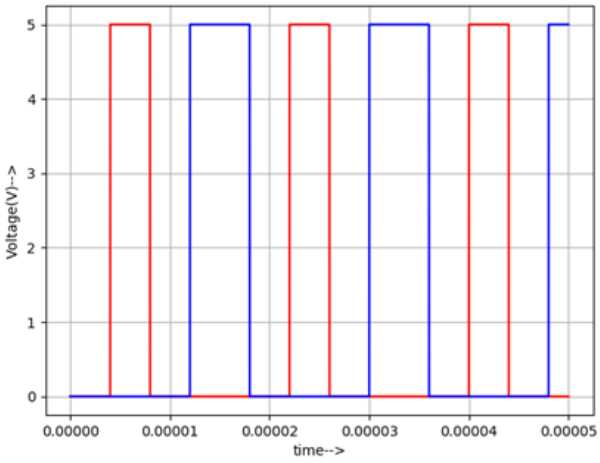

Figure 4.39: Ngspice Output Plot demonstrating Dead-Time insertion and Complementary PWM.

Chapter 5

Conclusion and Future Scope

5.1 Conclusion

This internship project successfully realized the comprehensive design, integration, and validation of 10 complex mixed-signal and digital Intellectual Property cores utilizing the open-source eSim platform. By traversing the complete electronic design flow, from theoretical specification to schematic capture, Verilog logic design, NgVeri integration, and Ngspice transient analysis, a highly robust library of motor control, communication, and safety IP blocks was established. The successful implementation of advanced algorithms such as the Space Vector PWM, BLDC Commutation, and Windowed Watchdog Timer demonstrates the efficacy of eSim as a powerful tool for mixed-signal System-on-Chip (SoC) development. The project reinforced foundational concepts in digital hardware design while providing deep practical exposure to bridging the digital-analog divide using ADC/DAC bridging techniques in open-source EDA environments.

5.2 Future Scope

The IP cores developed during this internship lay the groundwork for advanced embedded system architectures. Future developments can expand upon this work in several directions:

- **FPGA Deployment:** The verified Verilog models can be synthesized, mapped, and deployed onto physical FPGA hardware (such as open-source Lattice toolchains) to validate real-time timing constraints and resource utilization.
- **SoC Integration:** The individual IP blocks can be integrated onto a unified internal bus structure (e.g., Wishbone or AMBA APB) alongside an open-source RISC-V soft core, creating a complete, single-chip motor control solution.
- **Application Specific Modules:** Expanding the library to include aerospace-specific communication protocols (like CAN bus or ARINC 429) or integrating the PID and QEI cores with AI-driven parameter tuning for autonomous flight surface control.
- **Advanced Layout:** Transitioning the verified eSim schematics into PCB layouts using KiCad, moving from pure simulation to functional hardware prototypes for bench-testing with physical motors and sensors.

Bibliography

[1] eSim Official Documentation. FOSSEE, IIT Bombay. https://esim.fossee.in/

[2] "Makerchip Digital Circuit Design." Redwood EDA. https://makerchip.com/

[3] FOSSEE Project Overview. National Mission on Education through ICT, MHRD, Govt. of India. https://fossee.in/

[4] Ngspice Circuit Simulator. http://ngspice.sourceforge.net/

[5] "The I2C-Bus Specification and User Manual," UM10204, NXP Semiconductors, Rev. 6, 2014. https://www.nxp.com/docs/en/user-guide/UM10204.pdf

[6] "Space-Vector PWM With TMS320C24x/F24x Using C and C++," Texas Instruments Application Report, Literature Number: BPRA073. https://www.ti.com/lit/an/bpra073/bpra073.pdf

[7] Yedamale, Padmaraja. "Brushless DC (BLDC) Motor Fundamentals," Microchip Technology Inc., Application Note AN885, 2003. https://ww1.microchip.com/downloads/en/AppNotes/00885a.pdf

[8] Kester, Walt. "ADC Architectures III: Sigma-Delta ADC Basics," Analog Devices, Application Note MT-022, 2009. https://www.analog.com/media/en/training-seminars/tutorials/MT-022.pdf

[9] Sreekrishna K Sasidharan. "Verilog Source Code Implementations for Mixed-Signal and Digital IP Cores." Developed during FOSSEE Semester Long Internship, Spring 2026. https://github.com/sreekrishnaks/Internship-work-FOSSEE-eSim-/tree/main/verilog_codes

[10] Sreekrishna K Sasidharan. "Individual IP Core Specification and Work Reports." Developed during FOSSEE Semester Long Internship, Spring 2026. https://github.com/sreekrishnaks/Internship-work-FOSSEE-eSim-/tree/main/ip_reports