System Requirement Specification (SYS2)

Revisions

Version	Author	Description	Date
v1.0.0	*Hairu Mossa*		*28/09/2025*

Document Approvals

Role	Name	Signature	Date

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this system is to design and implement embedded software for a non-isolated Buck Converter. The software shall regulate the output voltage to 24V under varying input voltages and load conditions, limit the output current to a safe maximum, by providing real-time monitoring and telemetry over CAN bus.

1.2 Scope

In Scope

- Closed-loop PI/PID-based voltage regulation.
- Current limiting with fault logging.
- Periodic telemetry transmission over CAN (voltage, current, temperature, state, errors).
- Error detection and reporting (overcurrent, ADC failure, instability).

Out of Scope

- Hardware simulation.

1.3 Definitions, Acronyms, and Abbreviations

- ADC - Analog-to-Digital Converter
- CAN - Controller Area Network
- PWM - Pulse Width Modulation
- PI/PID - Proportional-Integral / Proportional-Integral-Derivative Control
- SyRS - System Requirements Specification

1.4 References

1. Saykal Electronics Embedded Systems Interview Test – Rev v1.0/02.05.2025
2. Internal Coding Standard: STD.COD.01.

2. Specific Requirements

2.1 Functional Requirements

SYS_0001: The system shall measure output voltage and output current via ADC channels.
SYS_0002: The system shall implement a PI/PID controller to regulate the PWM duty cycle.
SYS_0003: The system shall adjust PWM duty cycle within 0%–95% range.

SYS_0004: The system shall limit output current to 10A maximum by reducing duty cycle.
SYS_0005: The system shall periodically (100 ms) transmit telemetry over CAN, including

Input Voltage, Input Current, Output Voltage, Output Current, Temperature, State, Error Flags.
SYS_0006: The system shall log and report error conditions (overcurrent, ADC failure, instability).

2.2 Usability Requirements

SYS_0007: The system shall use consistent naming conventions for easier maintenance (per STD.COD.01).
SYS_0008: The system shall provide readable and modular code to reduce debugging complexity.
SYS_0009: Errors and states shall be human-interpretable through CAN signal definitions (DBC).

2.3 Performance Requirements

SYS_0013: ADC shall support 12-bit resolution.

2.4 System Interfaces

SYS_0015: CAN messages shall comply with DBC specification including scaling, offsets, units, and limits.

2.5 System Operations

SYS_0016: The system shall allow IDLE, RUNNING, and ERROR states.
SYS_0017: The system shall autonomously transition from IDLE → RUNNING when control starts.
SYS_0018: The system shall transition to ERROR state if overtemperature or ADC failure occurs.

2.6 System Modes and States

SYS_0019: Modes include:

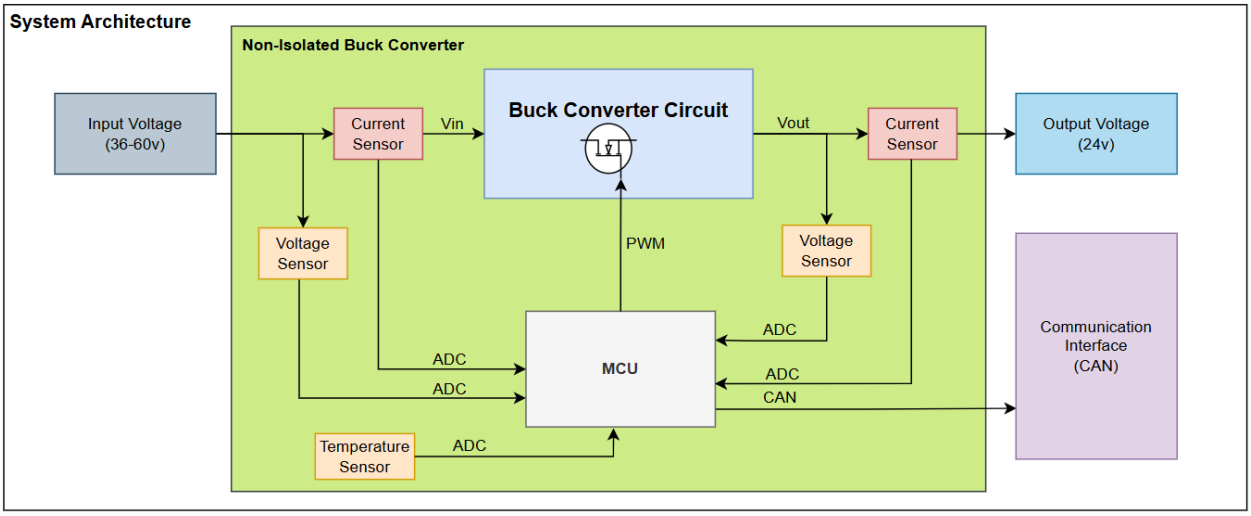
- Idle Mode: No PWM output, awaiting start.
- Running Mode: Actively regulating and transmitting telemetry.
- Error Mode: PWM disabled, errors broadcast periodically.

2.7 Policies and Regulations

SYS_0028: All code shall comply with internal coding standard STD.COD.01.
SYS_0029: The system shall follow functional safety guidelines (ISO 26262 principles) where applicable.

Appendices

Appendix A:



Overall System Architecture