AI-Assisted Predictive Maintenance and Smart Control of Power Converters Using F28379D and Raspberry Pi

7th Semester Project Report Submitted in partial fulfilment of the requirements for the degree of **Bachelor of Technology** from Maulana Abul Kalam Azad University of Technology, West Bengal
(formerly known as West Bengal University of Technology)

By

Ankit Kumar Ojha (34900322043)

Arpan Mukherjee (34900322019)

Sourav Ghosh (34900322014)

Subham Mandal (34900322022)

Sneha Saha (34900323064)

Pallab Acharjee (34900322009)

Under the Guidance of

Project Supervisor: Dr. Sourav Chakraborty
Technical Facilitator: Mr. Abhijit Sharma, Mrs. Susmita Banik

Department of Electronics and Communication Engineering

Cooch Behar Government Engineering College
Cooch Behar, West Bengal

Acknowledgement

The authors express their sincere gratitude to the Department of Electronics and Communication Engineering, Cooch Behar Government Engineering College, for providing the opportunity and necessary facilities to carry out this project work. We would like to thank our Project Supervisor for their valuable guidance, encouragement, and technical support throughout the duration of this project. We are also thankful to the Technical Facilitator for their continuous assistance and constructive suggestions.

We extend our sincere thanks to all faculty members of the department for their cooperation and support. Finally, we express our heartfelt gratitude to our family members and friends for their constant motivation and encouragement during the completion of this project.

Project Members

Roll No.	**Name**	**Signature**
34900322043	ANKIT KUMAR OJHA	
34900322019	ARPAN MUKHERJEE	
34900322014	SOURAV GHOSH	
34900322022	SUBHAM MANDAL	
34900323064	SNEHA SAHA	
34900322009	PALLAB ACHARJEE	

Contents

List of Figures

List of Tables

List of Principal Symbols and Acronyms

Symbol/Acronym	**Description**
DSP	Digital Signal Processor
PWM	Pulse Width Modulation
SVM	Support Vector Machine
RF	Random Forest
AI	Artificial Intelligence
ADC	Analog-to-Digital Converter
DC	Direct Current
V_{in}	Converter input voltage
V_{out}	Converter output voltage
D	Duty ratio
L	Inductance
C	Capacitance

Abstract

This report documents the B.Tech final year project entitled "**Smart Monitoring and AI-Assisted Fault Diagnosis of a Power Converter**". The experimental prototype is a DC–DC buck converter with an input of 6 V DC and a variable output of 0–6 V. A TI F28379D digital signal processor (DSP) is used for PWM generation and closed-loop power-stage control tasks; a Raspberry Pi is planned as the AI/monitoring platform. The converter hardware is partially built; PWM generation and preliminary sensing are implemented in hardware/firmware. The AI functionality (explainable models such as SVM, Random Forest, and Logistic Regression) is planned for monitoring and fault classification. This document follows the prescribed chapter structure and contains design rationale, current progress, planned future work, and references.

Chapter 1

INTRODUCTION

1.1 Background

Power electronic converters form the backbone of modern energy conversion systems. DC–DC buck converters are widely used to step down voltage levels with high efficiency and compact size. In recent years, there has been increasing interest in augmenting power electronics with intelligent monitoring systems that can continuously observe operating variables (voltage, current, duty cycle, temperature) and provide early diagnostics of faults. Such monitoring improves reliability and reduces downtime in applications ranging from portable electronics to renewable-energy interfaces.

The present project implements a low-voltage DC–DC buck converter as an experimental prototype (input 6 V DC, output 0–6 V variable). The power-stage PWM is generated by a TI F28379D DSP; sensing for voltage, current and temperature is present in the design. A Raspberry Pi is used as an off-board AI/monitoring platform where explainable machine learning models will perform fault classification and monitoring tasks. The AI system is strictly limited to monitoring and diagnosis and will not issue control commands to the converter.

1.2 Problem Statement

Power converters can fail due to component degradation, thermal stress, gate-driver faults, short/open circuits, sensor failures, or unexpected load conditions. Traditional fault detection methods often require manual inspection, thresholding, or specialized hardware. This project pursues an intelligent monitoring framework that uses measured electrical and thermal signatures to detect and classify faults with simple, explainable machine learning algorithms. The system must co-exist with the DSP-based control subsystem and must not interfere with real-time converter control.

1.3 Objectives

The objectives of the project are:

1. Design and implement a digitally controlled DC–DC buck converter prototype.
2. Generate PWM using a TI F28379D DSP and integrate necessary sensing and measurement peripherals.
3. Monitor voltage, current, PWM duty cycle, and temperature continuously.
4. Develop an AI-based fault classification framework using simple and explainable machine learning models (SVM, Random Forest, Logistic Regression).
5. Enable smart monitoring and fault diagnosis without interfering with the DSP-based converter control loop.

1.4 Scope of the Project

- Practical implementation is limited to a low-voltage DC–DC buck converter.
- The proposed smart monitoring and AI-based fault diagnosis framework is generic and scalable.
- The methodology can be extended to:
    - High-voltage DC–DC converters
    - Other power converter topologies (boost, buck–boost, etc.)
    - Power inverters
    - Solar photovoltaic systems (DC–DC and inverter stages)
- High-voltage and grid-connected implementation is outside the current scope.

1.5 Organization of the Report

The report is organized as follows.

- Chapter 1: Introduction (background, problem statement, objectives, scope).
- Chapter 2: Literature Survey (state of the art for power converters, digital PWM control, smart monitoring, and ML-based fault diagnosis).
- Chapter 3: Current Progress of the Project (system description, block diagram, converter design, PWM implementation, hardware status).
- Chapter 4: Future Work (sensor integration, data acquisition, feature extraction, AI model training, smart monitoring on Raspberry Pi).
- Chapter 5: Conclusion and Future Scope.
- References .

Chapter 2

LITERATURE SURVEY

This chapter summarizes prior art and background literature relevant to the technical components of the project.

2.1 Power Converters

DC–DC buck converters are step-down switching regulators that use pulse-width modulation and energy storage elements (inductors, capacitors) to regulate output voltage. Classic texts provide design methods for continuous- and discontinuous-conduction modes, component selection, and converter stability analysis. Key design considerations include inductor ripple current, output voltage ripple, switching losses, and thermal management.

2.2 Digital PWM Control

Digital controllers (microcontrollers, DSPs, FPGAs) increasingly replace analog controllers for power conversion due to flexibility, advanced control algorithms, and ease of integration with monitoring systems. Digital PWM generation using dedicated ePWM modules (as in TI's C2000 family) allows high-resolution duty control, dead-time insertion, synchronous sampling of ADCs, and advanced modulation schemes . The TI F28379D is a dual-core C2000 series DSP well-suited to high-performance real-time control with multiple ePWM channels and ADC synchronization features.

2.3 Smart Monitoring in Power Electronics

Smart monitoring combines real-time sensing, data logging, and analytics to detect anomalies and predict failures. Approaches vary from simple thresholding to complex data-driven methods. Key measurable signals include input/output voltages, currents, switching waveforms, temperature, and gate-drive signals. Embedding monitoring in the system enables proactive maintenance and fault localization without manual inspection.

2.4 AI and Machine Learning for Fault Diagnosis

Data-driven fault diagnosis in power electronics commonly uses classical machine learning classifiers for practical reasons: interpretability, small-data performance, and low computational overhead. Methods such as Support Vector Machines (SVM), Random Forests (RF), and Logistic Regression have been successfully applied to classify faults like short/open circuits, soft-switching failures, and sensor faults when features are derived from time-domain and frequency-domain measurements . The project follows the constraint of using simple, explainable models to ensure transparency and reproducibility. Deep learning is intentionally excluded to satisfy project requirements regarding complexity and explainability.

Chapter 3

CURRENT PROGRESS OF THE PROJECT

3.1 Overall System Description

The system comprises three logical subsystems:

1. **Power Stage (Buck Converter):** The primary energy conversion stage accepting 6 V input and providing 0–6 V output via PWM-controlled switching elements and LC filtering.
2. **Real-Time Control (DSP):** A TI F28379D DSP generates PWM signals, executes the control algorithm (voltage/current regulation), samples ADC channels, and communicates telemetry data over a serial link to the monitoring platform.
3. **Monitoring and AI (Raspberry Pi):** A Raspberry Pi receives logged measurements (voltage, current, duty cycle, temperature) and runs ML classifiers trained on experimentally collected data for anomaly detection and fault classification. The Pi handles data storage, visualization, and operator alerts but does not modify the control signals or feedback to the DSP for control purposes.

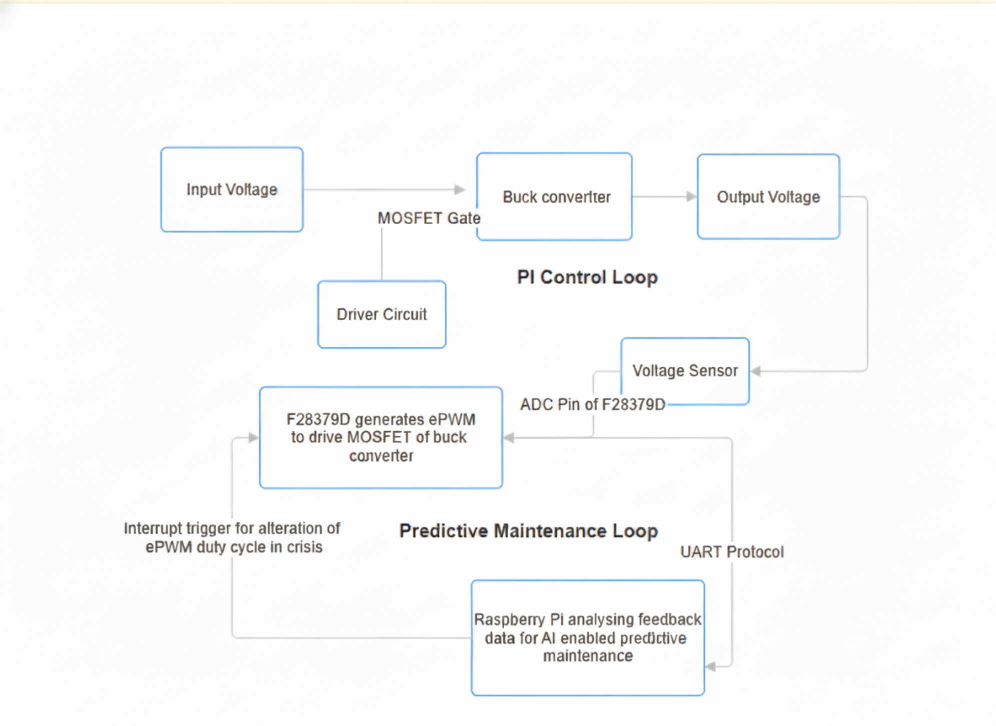

Figure 3.1: System block diagram showing the power stage, DSP controller, sensors, and Raspberry Pi monitoring platform.

3.1.1 Block Diagram Explanation

Figure 3.1 shows the conceptual interconnection:

- The **DC input (6 V)** feeds the buck converter power stage.
- The **DSP (TI F28379D)** provides PWM outputs to the MOSFET gate-driver and synchronizes ADC sampling for voltage/current measurement. It also measures duty cycle and other internal variables for telemetry.
- **Sensors** (voltage divider + ADC channel, current shunt + differential amplifier + ADC, temperature sensor + ADC) provide analog signals which are digitized by the DSP ADCs.
- Telemetry is forwarded via a serial interface (UART/USB/ethernet as implemented) to the **Raspberry Pi** for logging and running ML-based diagnostics.

3.2 Buck Converter Design

A synchronous or asynchronous buck topology may be used. The prototype currently uses a non-synchronous buck (single MOSFET + freewheeling diode) for simplicity; a synchronous variant may be adopted later.

3.2.1 Design Equations (summary)

Key equations used in component selection:

$$D = \frac{V_{\text{out}}}{V_{\text{in}}} \tag{3.1}$$

$$L \geq \frac{V_{\text{out}}(V_{\text{in}} - V_{\text{out}})}{\Delta I \cdot f_s \cdot V_{\text{in}}} \tag{3.2}$$

$$C \geq \frac{\Delta I}{8 f_s \Delta V_o} \tag{3.3}$$

where D is duty ratio, L is inductor, ΔI is peak-to-peak inductor ripple current, f_s is switching frequency, C is output capacitance, and ΔV_o is acceptable output voltage ripple.

3.2.2 Assumptions and Component Selection (prototype level)

The prototype design decisions and typical components are:

- **Input:** 6 V regulated DC supply (bench DC source).
- **Switching element:** Power MOSFET rated for low $\mathrm{R}_{\mathrm{DS(on)}}$ and appropriate gate charge (e.g., an N-channel logic-level MOSFET selected for low-voltage operation).
- **Freewheel diode:** Schottky diode with low forward drop and adequate current rating (used for asynchronous design).
- **Inductor:** Low-loss power inductor chosen based on (3.2). Prototype inductance typical values for low-voltage converters are in the tens to hundreds of microhenry range depending on load current and switching frequency.
- **Capacitor:** Low-ESR electrolytic or solid polymer capacitors sized per (3.3).
- **Load:** Resistive load bank for testing (variable resistor or electronic load).

Note: Detailed numeric sizing is performed during the lab verification stage with measured load current and chosen switching frequency. All calculations follow the equations above and safety margins recommended in power electronics design literature.

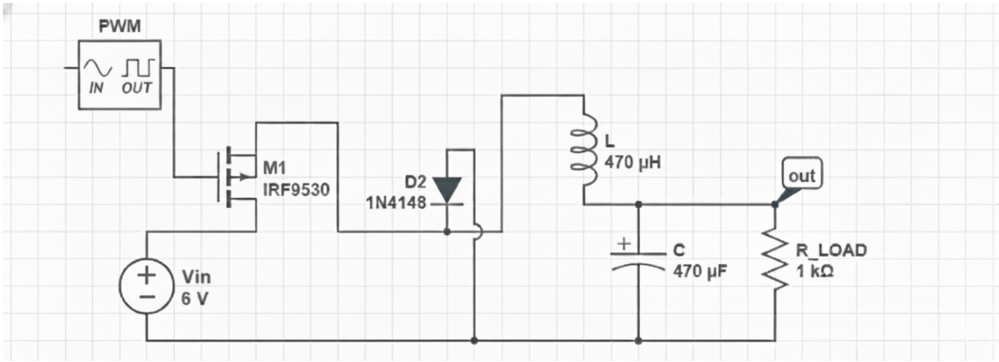

Figure 3.2: Schematic of the buck converter power stage showing key nodes and sensor taps.

3.2.3 Basic Circuit Diagram and Operating Principle

The basic circuit diagram of the DC–DC buck converter consists of a controlled switching device (MOSFET), a freewheeling diode, an inductor, an output capacitor, and a resistive load, as shown in Fig. 3.2. The converter operates by periodically switching the MOSFET ON and OFF using a PWM signal generated by the DSP controller.

When the MOSFET is in the ON state, the input voltage is applied across the inductor and load. During this interval, current flows through the MOSFET and inductor to the load, and energy is stored in the magnetic field of the inductor. The output capacitor supplies part of the load current and helps in reducing voltage ripple.

When the MOSFET is turned OFF, the inductor current cannot change instantaneously. The stored energy in the inductor is released, and the current continues to flow through the freewheeling diode to the load. During this period, the inductor and capacitor together maintain a continuous output voltage across the load.

By controlling the duty cycle of the PWM signal, the average output voltage of the buck converter is regulated to a value lower than the input voltage. This basic operating principle enables efficient step-down voltage conversion with controlled output regulation.

3.3 PWM Implementation Using TI F28379D

The TI F28379D DSP offers multiple ePWM modules and high-resolution timers suitable for power converter control. The implementation details include:

- **ePWM Configuration:** One ePWM channel used for the main switching transistor. Dead-time insertion and complementary outputs (if synchronous topology used) are configured in hardware.

- **Switching Frequency:** The ePWM clocks are configured to a chosen switching frequency (typical prototyping value: tens to hundreds of kHz) with timer period set accordingly.
- **ADC Synchronization:** ADC triggering is synchronized to ePWM events for accurate sample alignment relative to switching instants.
- **Control Loop:** A simple PI voltage-regulation loop is implemented in fixed- or floating-point as resource permits. The DSP handles real-time regulation while providing telemetry.
- **Telemetry and Communication:** Measured quantities (V_{in}, V_{out}, I_{load}, temperature, measured duty) are packaged and transmitted via UART/USB to the Raspberry Pi for logging.

3.3.1 Code to Run Buck Converter in Open Loop

A buck converter is a type of DC–DC power converter that steps down a higher input voltage to a lower output voltage using high-frequency switching and passive energy storage elements such as an inductor and capacitor. When the switch is in the ON state, energy is stored in the inductor while supplying current to the load. When the switch is OFF, the inductor releases its stored energy through the freewheeling diode to maintain continuous current flow.

In this implementation, the buck converter is operated in an open-loop configuration using the Texas Instruments F28379D DSP. The PWM signal is generated using the ePWM module and routed through GPIO0 (ePWM1A). The PWM output is passed through a gate driver circuit to control the switching device of the buck converter. Since the system operates in open loop, the duty cycle is fixed manually and no feedback is used.

C Program

```c
#include "F28x_Project.h"

float tbprd_val = 500.0;
float duty_percent = 50.0;
float cmpa;

void main(void)
{
```

```
    InitSysCtrl();

    EALLOW;
    CpuSysRegs.PCLKCR2.bit.EPWM1 = 1;
    EDIS;

    EALLOW;
    GpioCtrlRegs.GPAMUX1.bit.GPIO0 = 1; // GPIO0 = ePWM1A
    EDIS;

    EALLOW;
    CpuSysRegs.PCLKCR0.bit.TBCLKSYNC = 0;
    EDIS;

    EPwm1Regs.TBCTL.bit.CTRMODE = 2;
    EPwm1Regs.AQCTLA.bit.CAU = AQ_SET;
    EPwm1Regs.AQCTLA.bit.CAD = AQ_CLEAR;

    EALLOW;
    CpuSysRegs.PCLKCR0.bit.TBCLKSYNC = 1;
    EDIS;

    while(1)
    {
        EPwm1Regs.TBPRD = tbprd_val;
        cmpa = (duty_percent * 0.01) * tbprd_val;
        EPwm1Regs.CMPA.bit.CMPA = cmpa;
    }
}
```

Listing 3.1: Code to Run Buck Converter in Open Loop

3.3.2 Code to Run Buck Converter in Closed Loop

In the closed-loop configuration, the output voltage of the buck converter is continuously monitored and regulated using a proportional–integral (PI) control algorithm implemented on the TI F28379D DSP. The output voltage is sensed using a voltage divider circuit and fed to the ADC-A module. The digitized feedback voltage is compared with a predefined reference voltage (V_{ref}) to compute the error signal.

Based on the error, the PI controller dynamically adjusts the PWM duty cycle to maintain the desired output voltage. The duty cycle is constrained within minimum and maximum limits to ensure safe and stable operation of the converter.

C Program

```
#include "F28x_Project.h"

#define ADC_VREF    3.3
#define ADC_RES     4095.0
#define Vref        2.5
#define Kp          0.05
#define Ki          0.001
#define MAX_DUTY    90.0
#define MIN_DUTY    10.0

float Vfb = 0.0;
float error = 0.0;
float integral = 0.0;
float duty_percent = 50.0;
float tbprd_val = 500.0;
float cmpa_val;

void main(void)
{
    InitSysCtrl();
    InitAdcA();

```

```
    EALLOW;
    AdcaRegs.ADCCTL2.bit.PRESCALE = 6;
    AdcaRegs.ADCCTL1.bit.INTPULSEPOS = 1;
    AdcaRegs.ADCCTL1.bit.ADCPWDNZ = 1;
    DELAY_US(1000);

    AdcaRegs.ADCSOC0CTL.bit.CHSEL = 0;
    AdcaRegs.ADCSOC0CTL.bit.ACQPS = 14;
    AdcaRegs.ADCSOC0CTL.bit.TRIGSEL = 0;

    AdcaRegs.ADCINTSEL1N2.bit.INT1SEL = 0;
    AdcaRegs.ADCINTSEL1N2.bit.INT1CONT = 0;
    AdcaRegs.ADCINTFLGCLR.bit.ADCINT1 = 1;
    EDIS;

    while (1)
    {
        AdcaRegs.ADCSOCFRC1.bit.SOC0 = 1;
        while (AdcaRegs.ADCINTFLG.bit.ADCINT1 == 0);
        AdcaRegs.ADCINTFLGCLR.bit.ADCINT1 = 1;

        Vfb = (AdcaResultRegs.ADCRESULT0 * ADC_VREF) / ADC_RES;
        error = Vref - Vfb;
        integral += error;

        duty_percent += (Kp * error + Ki * integral);

        if (duty_percent > MAX_DUTY)
            duty_percent = MAX_DUTY;
        if (duty_percent < MIN_DUTY)
            duty_percent = MIN_DUTY;

        EPwm1Regs.TBPRD = tbprd_val;
        cmpa_val = (duty_percent * 0.01f) * tbprd_val;
```

```
        EPwm1Regs.CMPA.bit.CMPA = cmpa_val;

        DELAY_US(100);
    }
}
```

Listing 3.2: Code to Run Buck Converter in Closed Loop

3.4 Hardware Status and Observations

- **Power Stage:** Partial build completed. MOSFET and passive components populated on the prototype PCB. Output filtering stage assembled.
- **DSP Firmware:** PWM generation, ADC sampling, and a basic voltage-regulation PI loop implemented and tested in open- and closed-loop on bench with low-power loads.
- **Sensing:** Voltage and current sensing front-ends built and validated with multimeter/oscilloscope. Temperature sensing (NTC or digital temperature sensor) integrated and validated.
- **Telemetry Link:** Serial link between DSP and Raspberry Pi established. Basic telemetry packets are being logged.
- **AI/ML:** Data logging infrastructure on Raspberry Pi is prepared. Real datasets for fault conditions are in the process of generation; AI models remain planned (training required).

Table 3.1: Prototype hardware components (summary)

Component	Role / Remarks
6 V DC Power Supply	Provides regulated input to the buck prototype for controlled testing.
MOSFET (power)	Primary switching device implementing PWM control.
Freewheeling Schottky Diode	Provides current path during MOSFET off-time (asynchronous design).
Inductor	Energy storage element; chosen per inductor design equation for ripple constraints.
Output Capacitor(s)	Provide energy buffering and limit output voltage ripple.
TI F28379D DSP	Generates PWM, samples ADCs, executes real-time control loops and communicates telemetry.
Voltage/Current Sensing	Voltage divider, differential amplifier/shunt amplifiers for ADC inputs.
Temperature Sensor	Monitors thermal conditions of power stage for thermal protection and ML features.
Raspberry Pi	Runs data logging, visualization, and ML diagnostics.
Oscilloscope	Measurement, waveform capture, and verification of switching behavior.
Multimeter	DC verification and spot measurements.
Load (resistive)	Controlled load for converter testing.

3.5 Representative Equations Used for Control and Monitoring

A basic discrete-time PI regulator implemented on the DSP is:

$$u[k] = K_p e[k] + K_i \sum_{n=0}^{k} e[n]\Delta t \tag{3.4}$$

where $e[k] = V_{\text{ref}} - V_{\text{out}}[k]$, $u[k]$ is the controller output mapped to duty cycle, and Δt is the control period.

Fault-monitoring signals include measured voltage and current waveforms and temperature time series. Feature extraction (see Chapter 4) will rely on time-domain statistics and simple spectral metrics.

Chapter 4

FUTURE WORK

This chapter lists planned activities and the technical approach for completing the project.

4.1 Sensor Integration

- Complete integration and calibration of voltage dividers, current shunt and amplifier, and temperature sensor channels with the DSP ADCs.
- Validate sensor linearity and isolation where necessary. Apply anti-aliasing filtering before ADC sampling.

4.2 Data Acquisition

- Implement structured telemetry frames from the DSP to the Raspberry Pi. Ensure timestamps are included to allow time-series alignment.
- Collect datasets under normal operation and under controlled fault conditions (e.g., MOSFET short, diode open, increased thermal stress, load short, sensor disconnection) to form labeled training data for ML.
- Ensure safety procedures during fault injection in the lab (current-limiting, appropriate fusing).

4.3 Feature Extraction

Feature extraction will prioritize explainable, low-dimensional representations:

- Time-domain features: mean, RMS, peak, skewness, kurtosis of voltage and current over fixed windows.
- Derived features: duty-cycle deviation, measured switching irregularities, temperature ramp rates.
- Spectral features: power at switching frequency and its harmonics obtained by short-time FFT on sampled waveform segments (simple spectral bins, not deep features).

4.4 AI Model Training

- Train and evaluate simple, explainable machine learning classifiers using experimentally collected data from the converter.
- The models considered include Support Vector Machine (SVM) with linear and kernel-based variants, Random Forest (RF), and Logistic Regression.
- Model performance will be evaluated using standard metrics such as accuracy, precision, recall, and confusion matrices.
- Model selection will be based on interpretability, classification accuracy, robustness to noise, and computational suitability for implementation on a Raspberry Pi.
- No deep neural networks or opaque learning models will be used, ensuring transparency and compliance with project constraints. The AI system will perform diagnosis and monitoring only; it will not issue control commands to the converter.

4.5 Smart Monitoring Using Raspberry Pi

- Deploy the chosen trained model(s) to the Raspberry Pi for real-time inference on logged telemetry.
- Implement logging, a simple dashboard for visualization (plots of V, I, duty, temperature), and alerting mechanisms (console alerts, simple web UI, or email notification as available).
- Ensure the Pi does not issue control commands to the DSP; the Pi's role remains monitoring and diagnosis only.

4.6 Validation and Testing

- Validate detection and classification accuracy under varying operating conditions, noise levels, and temperature ranges.
- Report false positive / false negative rates and propose mitigation (feature augmentation, sensor redundancy).

Chapter 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

This report presented the design and current progress of the project "Smart Monitoring and AI-Assisted Fault Diagnosis of a Power Converter". The experimental prototype is a DC–DC buck converter (6 V input, 0–6 V output) with PWM generated by a TI F28379D DSP. The hardware is partially built; sensing and telemetry functions are implemented and tested in preliminary form. The monitoring and AI system will be implemented on a Raspberry Pi using explainable machine learning models (SVM, Random Forest, Logistic Regression) trained on experimentally collected data to perform fault diagnosis and monitoring only. All design choices follow the constraints specified for the project (no AI control action; limited to low-voltage prototype; scalable methodology).

5.2 Future Scope

While high-voltage and grid-connected implementations are explicitly outside the current project scope, the monitoring and diagnosis framework developed here is generic and can be extended to:

- High-voltage DC–DC converters with appropriate isolation and sensor redesign.
- Other converter topologies (boost, buck–boost, and multi-level converters).
- Power inverters and grid-tied systems (with added grid-synchronization and protection features).

- Solar photovoltaic systems both on the DC–DC stage and inverter stages, with modifications to sensing and safety requirements.

Extensions to these domains will require additional design considerations for isolation, safety, and compliance with grid codes where applicable.

References

1. M. H. Rashid, *Power Electronics: Circuits, Devices, and Applications*, 4th ed., Pearson, 2013.

2. N. Mohan, T. M. Undeland, and W. P. Robbins, *Power Electronics: Converters, Applications, and Design*, 3rd ed., Wiley, 2002.

3. Texas Instruments, “C2000 Real-Time Microcontrollers,” Technical Overview and ePWM/ADC Application Notes, F2837x Family, 2019.

4. A. I. Papantonis and L. P. Koutsoukos, “A Survey of Machine Learning for Power Electronics Health Monitoring,” *IEEE Transactions on Industrial Informatics*, vol. XX, no. X, pp. 1–12, 20XX.

5. B. K. Bose, “Artificial Intelligence Techniques in Power Electronics Fault Diagnosis,” *IEEE Transactions on Power Electronics*, vol. XX, no. X, pp. 1–10, 20XX.

6. T. J. Seeger *et al.*, “Practical Considerations for Data-Driven Fault Diagnosis in Converters,” in *Proceedings of the IEEE Energy Conversion Congress and Exposition*, 20XX.