THERMAL RATING AND PREDICTION OF ROADWAY-EMBEDDED POWER ELECTRONICS FOR DYNAMIC WIRELESS POWER TRANSFER

by

Forrest D. Nichols

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Regan A. Zane, Ph.D. Major Professor	Hongjie Wang, Ph.D. Committee Member
Don Cripps, Ph.D. Committee Member	David F. Feldon, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2026

ABSTRACT

Thermal Rating and Prediction of Roadway-Embedded Power Electronics for Dynamic Wireless Power Transfer

by

Forrest D. Nichols, Master of Science

Utah State University, 2026

Major Professor: Regan A. Zane, Ph.D.
Department: Electrical and Computer Engineering

Dynamic wireless power transfer enables in-motion charging of electric vehicles through roadway-embedded transmitter coils, reducing onboard battery requirements and extending operational range. Embedding the power electronics alongside the transmitter coils within the roadway eliminates costly high-frequency cabling between roadside cabinets and ground assemblies, but introduces thermal management challenges where passive cooling must maintain component temperatures below rated limits without active cooling infrastructure. Analytical loss models, lumped-parameter thermal circuits, and finite element methods provide accurate component-level thermal insight but require substantial engineering effort per design, creating a bottleneck when multiple printed circuit board layouts undergo iterative revision.

This thesis developed an integrated framework combining system-level loss simulation, automated thermal measurement, and data-driven prediction to address this gap. A variable-speed circuit simulation characterized component-level loss profiles under pulsed operating conditions, revealing that systems designed for continuous full-load operation are substantially overrated for actual roadway utilization. An automated pipeline translated

printed circuit board design data into selective regions of interest for infrared thermal imaging, enabling reproducible component-level temperature extraction across multiple board designs. Three prototype boards were tested in both air and sand environments, with sand serving as a cost-effective surrogate for roadway thermal boundary conditions. A Feature-wise Linear Modulation-conditioned U-Net trained on this multi-board measurement data learned infrared-to-thermocouple temperature residuals, predicting component-level temperatures directly from thermal images and a compact thermal state vector without per-design engineering effort.

Cross-design evaluation on two training boards and a held-out unseen board demonstrated that the trained model captures transferable thermal coupling patterns rather than design-specific spatial memorization. The resulting staged workflow—from simulation-based loss characterization through automated measurement to machine-learning-assisted thermal screening—replaced weeks of per-design finite element simulation setup with evaluations on the order of minutes, providing a practical path toward rapid thermal assessment of embedded power electronics prior to roadway deployment.

(146 pages)

PUBLIC ABSTRACT

Thermal Rating and Prediction of Roadway-Embedded Power Electronics for Dynamic Wireless Power Transfer

Forrest D. Nichols

Electric vehicles can be charged while driving through wireless charging coils buried in the road surface. To avoid running expensive cables from roadside equipment to each charging pad, the electronics that control the power transfer can also be buried in the road. However, burying these circuit boards removes the ability to cool them with fans or flowing air, so the heat generated during operation must escape naturally into the surrounding road material. If the boards get too hot, components fail—a problem that has already been observed in prototype systems.

Predicting how hot each component on a circuit board will get when buried in a road is difficult. Existing methods such as computer simulations and hand calculations are accurate but take days to weeks of engineering work for each new board design. When multiple boards are being developed at the same time and going through frequent design changes, this process cannot keep pace. Current designs are also oversized because engineers assume worst-case heat loads that rarely occur during actual roadway operation, adding unnecessary cost and bulk.

This thesis develops a faster approach. First, a computer simulation estimates how much heat each part of the system produces at different vehicle speeds, showing that actual heat loads under pulsed roadway use are substantially lower than continuous worst-case assumptions. An automated imaging pipeline then uses a thermal camera and contact sensors to measure actual component temperatures on prototype boards tested in both open air and sand—the sand simulating conditions inside a road. Finally, a machine learning

model trained on these thermal measurements learns to predict component temperatures for new boards without repeating the full simulation or measurement process.

Testing shows that the trained model recognizes general heat-flow patterns rather than memorizing the layout of any single board, producing predictions within a few degrees of measured temperatures even on boards it has not seen before. This framework gives engineers a practical tool for checking whether a buried circuit board will overheat before it is installed in a road, reducing thermal evaluation time from weeks of engineering effort to minutes per design revision.

To my loving wife and family, for their unwavering support and encouragement throughout this journey.

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, Dr. Regan Zane, for his guidance and support throughout this research and my progression as a student. I would also like to thank my committee members, Dr. Hongjie Wang and Dr. Don Cripps, for their valuable feedback and insights. Additionally, I am grateful to Mr. Mahmoud Mansour for his continued support and collaboration on this, and all other parts of graduate research.

Forrest D. Nichols

CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

ACRONYMS

EV	Electric Vehicle
ICE	Internal Combustion Engine
ICEV	Internal Combustion Engine Vehicle
BEV	Battery Electric Vehicle
PHEV	Plug-In Hybrid Electric Vehicle
FCEV	Fuel Cell Electric Vehicle
CNG	Compressed Natural Gas
EVSE	Electric Vehicle Supply Equipment
WPT	Wireless Power Transfer
DWPT	Dynamic Wireless Power Transfer
IPT	Inductive Power Transfer
AC	Alternating Current
DC	Direct Current
HF	High Frequency
LCCL	Inductor–Capacitor–Capacitor–Inductor (resonant network)
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor
SiC	Silicon Carbide
PCB	Printed Circuit Board
IC	Integrated Circuit
FLIR	Forward-Looking Infrared
IR	Infrared
DUT	Device Under Test
PLECS	Piecewise Linear Electrical Circuit Simulation
FEM	Finite Element Method
FMEA	Failure Mode and Effects Analysis
SE	Steinmetz Equation
IGSE	Improved Generalized Steinmetz Equation
ML	Machine Learning
CNN	Convolutional Neural Network
MLP	Multi-Layer Perceptron
LSTM	Long Short-Term Memory
FiLM	Feature-wise Linear Modulation
ReLU	Rectified Linear Unit
PDE	Partial Differential Equation
MSE	Mean Squared Error
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
ROI	Region of Interest
SROI	Selective Region of Interest
NUC	Non-Uniformity Correction
DDE	Digital Detail Enhancement
MAD	Median Absolute Deviation
HDF5	Hierarchical Data Format (version 5)
CSV	Comma-Separated Values
JSON	JavaScript Object Notation
ODB++	Open Database (PCB fabrication format)
ASPIRE	Advancing Sustainability through Powered Infrastructure for Roadway Electrification
EVR	Electric Vehicle and Roadway test facility at USU
ANL	Argonne National Laboratory
INDOT	Indiana Department of Transportation
USU	Utah State University
DOE	U.S. Department of Energy

CHAPTER 1

INTRODUCTION

The focus of this thesis is the thermal prediction of embedded DWPT power electronics. While prior work has demonstrated that transmitter coils can be reliably embedded [4–6], embedding the remaining power electronics—inverter, compensation network, and protection circuitry—introduces thermal design constraints that current methods do not efficiently address. This thesis develops a thermally informed design framework for embedded DWPT modules, progressing from system-level loss characterization and analytical estimation through automated experimental validation to machine-learning-based rapid thermal screening. The resulting framework enables engineers to evaluate thermal feasibility at each stage of the design process, supporting the transition from conservatively overrated designs toward properly rated, cost-optimized embedded DWPT systems.

1.1 Background

Dynamic wireless power transfer (DWPT) enables electric vehicles to charge continuously while in motion, removing the constraint of long stationary charging stops and allowing meaningful reduction in onboard battery capacity. These advantages—lower vehicle weight, reduced battery cost, and improved utilization of charging infrastructure—position DWPT as an enabling technology for large-scale EV deployment. However, current DWPT pilot systems face deployment barriers that limit scalability: high hardware cost, large physical footprint, complex installation logistics, and low system utilization relative to rated capacity [7].

Many of these barriers share a common origin: existing systems are conservatively overrated—designed for continuous full-load operation when actual roadway use is pulsed and intermittent. Vehicles cross individual transmitter pads in fractions of a second, producing duty cycles with average component losses well below continuous-load assumptions.

Without thermal models that characterize component behavior under embedded, pulsed conditions, designers have no reliable basis to tighten component ratings. Conservative derating cascades into larger compensation networks, larger enclosures, longer high-frequency cabling, and more disruptive roadway installation—each compounding system cost and deployment complexity.

This overrating challenge exists against a backdrop of accelerating demand for EV charging infrastructure. Global EV sales continue to grow, as shown in Fig. 1.1 [8], and lifecycle analyses confirm that electrified powertrains produce lower cradle-to-grave emissions than internal combustion engine vehicles across multiple vehicle classes [9].

Current plug-in Electric Vehicle Supply Equipment (EVSE) ranges from Level 1 AC overnight charging through DC fast charging, as illustrated in Fig. 1.2. DC fast chargers can provide a suitable charge within 20–30 minutes, but all plug-in systems require the vehicle to stop and physically connect [10].

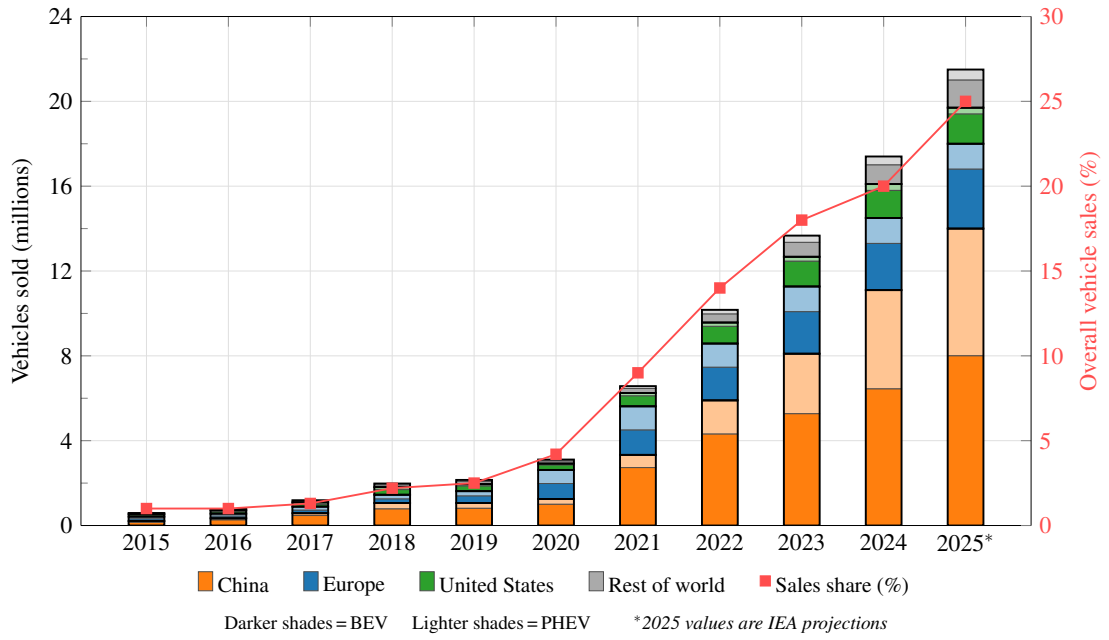


Fig. 1.1: International EV sales by the million.

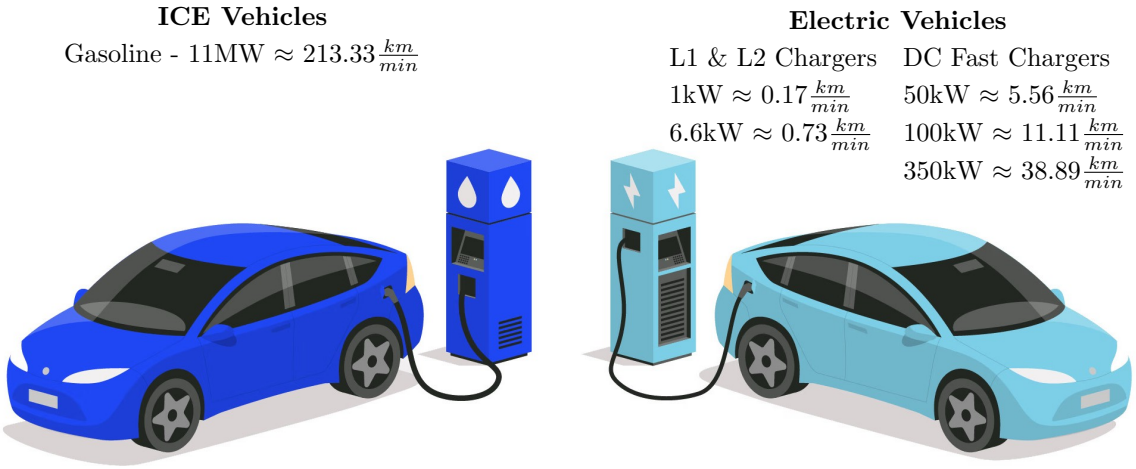


Fig. 1.2: Comparison between EV and ICE vehicles. [1] [2] Assuming 150wh/km energy efficiency for EVs.

Wireless power transfer (WPT) removes the stationary constraint by eliminating physical connectors entirely, reducing maintenance from connector wear and resolving some compatibility issues across vehicle manufacturers. WPT architectures support both stationary and in-motion charging configurations, as seen in Fig. 1.3. Static WPT systems provide hands-free charging but do not reduce onboard battery requirements, as the vehicle must remain parked for the duration of the charge. Dynamic WPT systems charge vehicles in motion, enabling practical battery size reduction and the associated benefits in vehicle weight, cost, and raw-material demand. By reducing the energy drawn from the battery during each trip, dynamic charging also decreases the time required for stationary top-up charging between trips.

Current DWPT implementations house their power electronics in roadside cabinets or underground vaults, with only the transmitter coils embedded in the pavement [7, 11, 12]. High-frequency AC cabling between these external enclosures and the embedded coils increases system inductance, constrains installation geometry, and adds substantial cost. Embedding the power electronics directly in the roadway alongside the coils would eliminate this cabling and enable a modular, scalable architecture—but introduces thermal design constraints that the field has not adequately resolved.

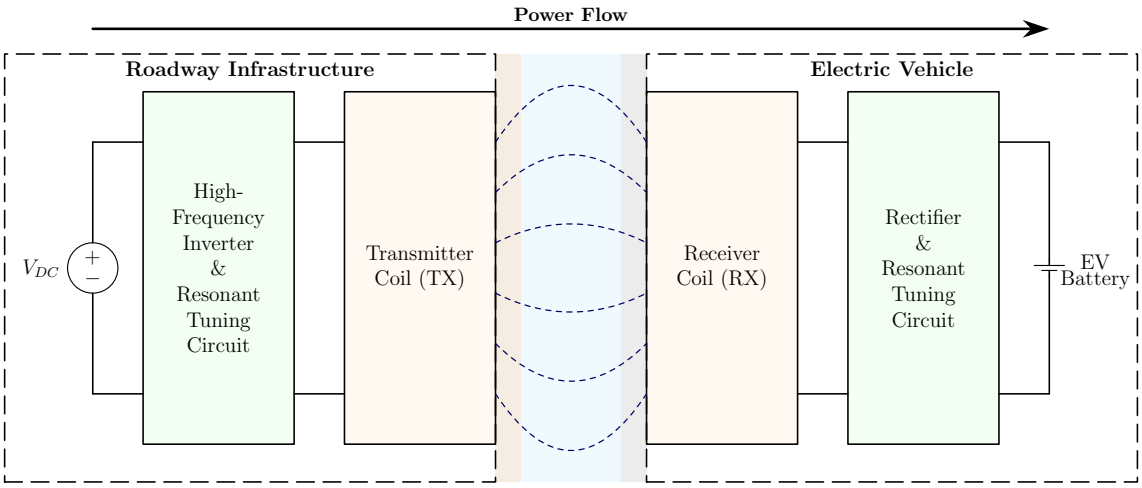


Fig. 1.3: Typical implementation of a DWPT system.

The severity of this thermal barrier was demonstrated during prior research at Utah State University, when an embedded DWPT gate driver board experienced thermal runaway resulting in power loss and component damage [13]. The failure originated in the gate driver ICs and passive components with small thermal mass and narrowly rated temperature margins on the board. Post-failure analysis attributed the failure to thermal design margins rather than a manufacturing defect. The board operated within its manufacturing specification but exceeded component thermal ratings once embedded in the epoxy and concrete encapsulant. Chapter 2 examines this failure in detail and traces the research progression through the thermal prediction methods motivated by it.

1.2 Literature Review

Wireless power transfer encompasses several architectures, of which capacitive and inductive methods are the most applicable for the high power densities required by larger vehicle classes. WPT was first demonstrated in the early 20th century by Nikola Tesla (US Patent No. 593138 and US Patent No. 645576). Since that time, multiple WPT methods have been investigated and adapted for EV charging applications. Those methods outlined in red within Fig. 1.4 represent the most commonly utilized methods for electric vehicle charging.

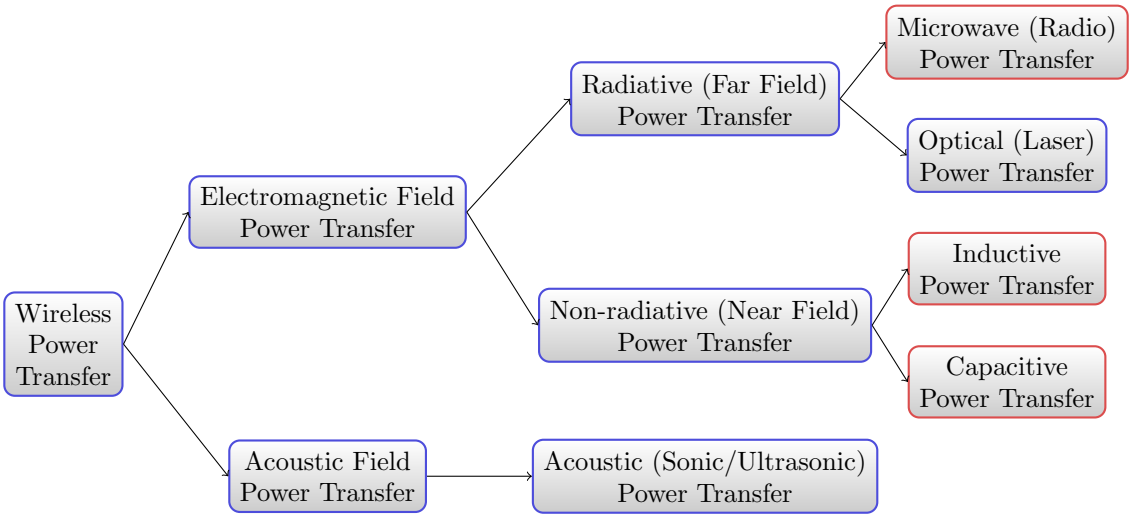


Fig. 1.4: Various implementations of WPT.

1.2.1 Inductive Power Transfer

The most widely used form of WPT for electric vehicles is inductive power transfer (IPT), implemented within wireless charging systems. This method allows for power to be transferred via electromagnetic induction from a primary charging coil within the ground to a secondary charging coil mounted on the underside of the vehicle. For this to occur, the primary charging coil receives a high-frequency (HF) AC signal which generates a magnetic field within the coil. When a secondary charging coil is brought within the proximity of the primary, this magnetic field induces a voltage on the secondary coil allowing for power transfer to occur. This power transfer can be described through the use of Maxwell's equations, and the coils can be modeled as loosely coupled inductors.

Standards outlining the power rating, frequency of transmission, and sizing for WPT systems have recently been published [14]. These standards push commercial systems and researchers to design with cross compatibility in mind between various vehicle manufacturers and vehicle classes. Multiple tuning topologies have been researched and proven for WPT systems, spanning both 3-phase and single-phase configurations [15, 16]. While they share a resonant transfer principle, topologies differ in element placement and design trade-offs. These include series-series, series-parallel, LCCL-series, LCCL-parallel, LCCL-LCCL, and

other series hybrids as seen in Fig. 1.5 [17].

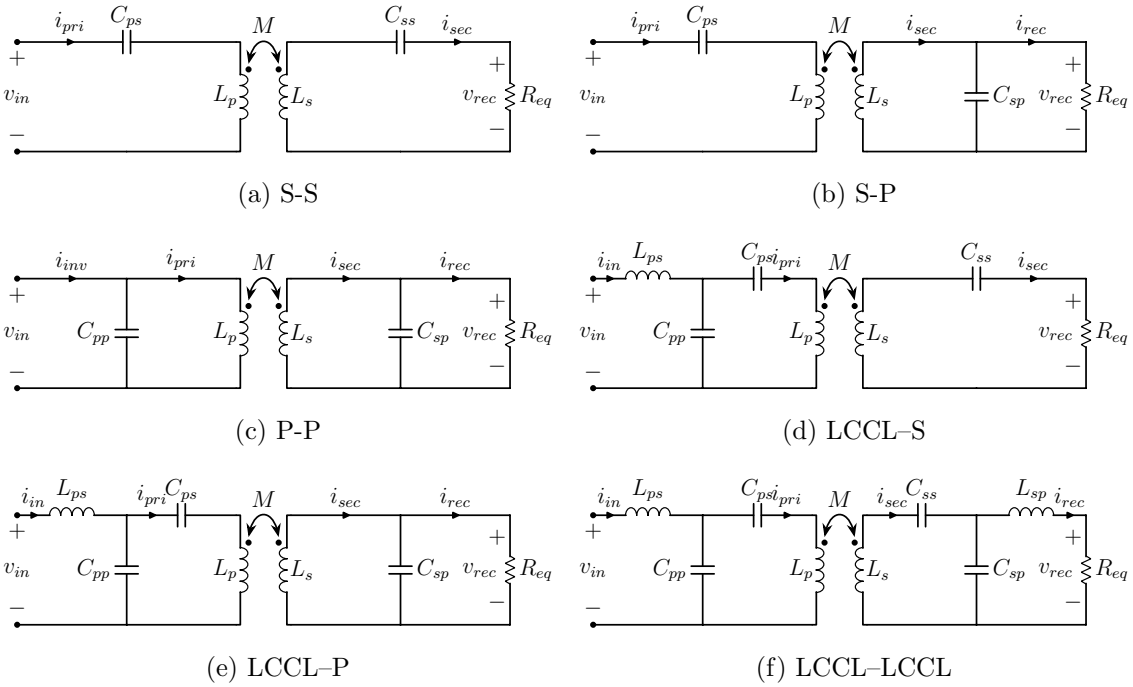


Fig. 1.5: Different compensation networks relevant in inductive DWPT.

Each tuning topology introduces different benefits and drawbacks, however LCCL-LCCL has proven to be an attractive approach due to benefits in the system's operability under DWPT applications within roadways. LCCL tuning allows independent load power regulation and offers flexibility under varying load and coupling conditions [16].

1.2.2 WPT System Design

The architecture of DWPT systems implementing IPT with a resonant tuning network are similar in design and installation. The typical implementation involves a method of sourcing power to the system, which can be of either low or high voltage range. This voltage is then converted to a DC voltage through rectification. An inverter converts this DC voltage to a high frequency (HF) AC voltage which is then connected to the primary charging coil(s), or transmitter(s), through a tuning network to allow for power flow. Fig. 2.2 in Chapter 2 shows a typical implementation of a WPT system for both static and dynamic

applications, representing a LCCL-LCCL tuning topology, which is the topology considered in this research.

1.2.3 Embedded DWPT

Embedding dynamic WPT into roadways allows vehicles to draw power continuously, which in turn allows for size reduction of the onboard battery module. A smaller battery module in turn cuts vehicle weight and cost, boosts efficiency, and reduces raw-material demand; all of which support scaling EV adoption.

When implementing WPT architectures of this nature in a static charging scenario the placement of the transmitter pad and system utilized can be raised from the ground or allow for minimal spacing between the road and the vehicle. This is convenient because it minimizes the distance between the transmitter and receiver coils, allowing for maximized coupling and ideal alignment.

For dynamic system implementation, a transmitter pad on top of a roadway, while feasible, is unsafe for highway driving speeds. This constraint necessitates embedding the primary side, or a portion of the primary side within the pavement of highways, or other installations of DWPT systems. Prior research [4–6] has demonstrated embedded transmitter pads placed within various roadway compositions. In these cases the embedded ground assembly consists of the transmitter coil (composed of litz wire), ferrite to shape the magnetic fields within the roadway, and structural material to hold the coils per the design criteria. These studies confirm that embedding the transmitter pad within the roadway is thermally feasible and achieves expected system behavior during vehicle passes.

1.2.4 Previous Roadway-Embedded Implementations

While the transmitter pads of the ground assembly for embedded WPT systems are included within the roadway, the remaining system hardware—resonant tuning network, primary-side power electronics, and grid connection equipment—is typically housed externally. Several companies and universities have implemented partially embedded DWPT systems following this architecture. ENRX [11], Electreon [12], and Purdue University in

collaboration with INDOT [7] have each partnered with the ASPIRE Engineering Research Center to install DWPT systems at the Utah State University Electrified Vehicle Research (EVR) facility. In each case the power electronics remain in external cabinets or vaults while only the transmitter coils are embedded in the pavement.

The Purdue system is particularly instructive: the team designed underground vaults to house the power electronics, eliminating the roadside cabinet hazard but introducing the thermal and moisture management challenges discussed in the following subsection [18]. This common architecture—embedded coils with external electronics—reflects a fundamental limitation: the thermal behavior of power electronics in embedded, pulsed-operation conditions is not well enough characterized to confidently embed these components alongside the coils.

1.2.5 Embedded DWPT System Design Challenges

Deploying DWPT at scale introduces challenges in physical footprint, cost, installation logistics, and system utilization. Many of these challenges share a common root cause. Without thermal models that characterize embedded component behavior under pulsed operation, designers apply conservative ratings sized for continuous full-load conditions that the system rarely encounters. This conservative derating cascades through the design. Oversized components require larger compensation networks, which require larger enclosures, which require longer runs of expensive high-frequency litz-wire cabling, which increase installation disruption and cost. Understanding the thermal margins of embedded power electronics is therefore prerequisite to breaking this cascade and enabling properly rated, cost-effective DWPT systems.

The typical design for current DWPT systems is given in Fig. 1.6. For static charging, a cabinet-based architecture allows short connections between the power electronics and the charging pad. For dynamic systems, however, cabinets must be placed at regular intervals along the track or roadway, changing the design constraints substantially.

Housing the main system components within cabinets, while reliable and easily serviceable, presents a hazard to those driving along the road and potentially limits DWPT

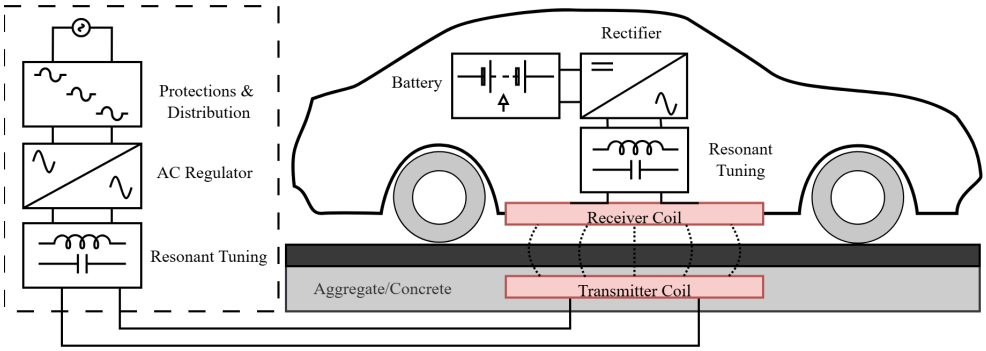


Fig. 1.6: Cabinet-based DWPT architecture with primary-side power electronics housed in an external enclosure and only the transmitter coil embedded in the roadway.

installations to the outside lane of highways. Purdue University addressed this constraint by designing underground vaults to enclose their power electronics hardware [18]. While vaults remove the roadside hazard, they introduce their own challenges. A buried enclosure housing high-power electronics cycles between ambient and elevated temperatures during operation. In climates with wide temperature swings, thermal cycling between ambient and operating temperatures may promote condensation on interior surfaces and PCBs. Accumulated moisture can increase the risk of corrosion and leakage currents on densely populated circuit boards. Managing humidity in a sealed underground environment adds complexity to what was intended as a simpler alternative to roadside cabinets.

In addition to the implications of their installation, connections within DWPT systems require a special type of wire called litz wire. This wire is made of hundreds of bundles of small copper wire, reducing the skin and proximity effects and allowing higher amperage capability while minimizing thermal losses. Prior work has characterized the thermal and electrical behavior of litz wire and other embedded cabling in roadway environments, demonstrating that the resistance and thermal performance remain manageable at typical WPT operating frequencies [6, 19, 20]. However, for current DWPT system designs, high-frequency high-power AC cables must run between the roadside cabinets and the ground assembly.

These cables increase inductance for the tuning network and introduce eddy currents within the system tuning, both of which can vary depending on the system installation and

are undesirable for system installation and calibration. While litz wire is the best solution for connecting the system's modules together, it is expensive due to its manufacturing process leading to a push towards a system design which minimizes wire lengths. The cost of fabrication for these cables is high, leading to increased overall system cost. Due to these constraints, system modularity is especially important when considering the connections between transmitter pads and power electronics.

By embedding hardware which deals with HF AC waveforms directly in the roadway, the proposed design minimizes external hookups and maximizes modularity, streamlining both installation and module-level maintenance. This shift would allow for low frequency cable to be used from the off-road grid connection to the system's modules. Fig. 1.7 illustrates how primary coils and their accompanying power electronics are grouped into roadway-embedded modules.

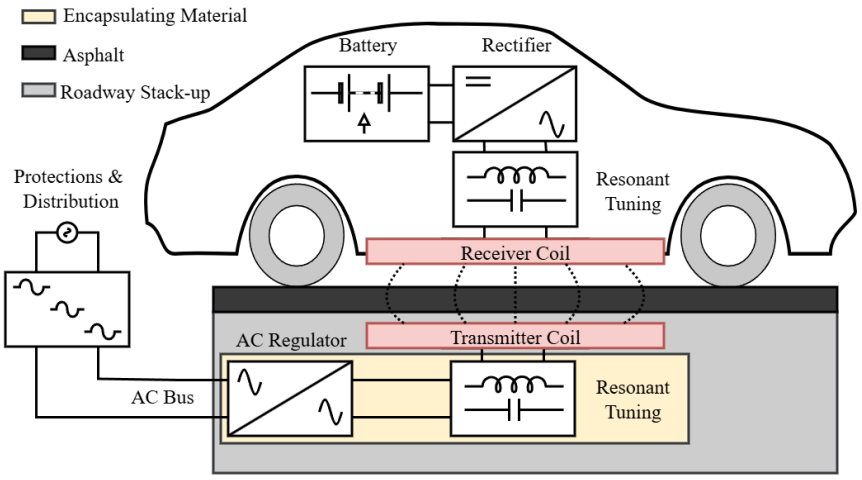


Fig. 1.7: Proposed embedded DWPT architecture with the AC regulator, resonant tuning network, and transmitter coil co-located within the roadway, reducing high-frequency cabling to the grid connection point.

Designing an Embedded DWPT system raises several feasibility questions at both pilot and large-scale deployment levels. These include system thermal performance and loss under variable utilization, reliability and longevity of embedded modules, power electronics designed for repairability, cost-effectiveness of grid and multi-module connections, structural

concerns such as size, weight, and load ratings, and the logistics of roadway conversion and system installation. These considerations highlight the practical challenges of implementing an Embedded DWPT system. The ASPIRE Engineering Research Center at Utah State University operates an Electrified Vehicle Research (EVR) facility that provides a dedicated test track for evaluating DWPT systems under realistic conditions. Multiple industry partners—including ENRX [11], Electreon [12], and Purdue University in collaboration with INDOT [7]—have installed partially embedded DWPT systems at this facility, each employing cabinet-based power electronics with roadway-embedded transmitter coils. The EVR facility enables direct comparison of DWPT architectures under controlled conditions and provides the hardware testing infrastructure used in this thesis for validating embedded thermal models. The thermal challenges identified through prior embedded power electronics research—particularly the thermal failure modes observed during high-power testing—motivate the predictive framework developed in the following chapters.

1.2.6 Thermal Prediction Approaches

Traditional thermal analysis for power electronics relies on either analytical modeling or finite element methods (FEM). Analytical models, such as the Steinmetz equation and RC thermal networks discussed in Chapter 2, provide fast estimates but make simplifying assumptions about geometry and boundary conditions. FEM tools such as ANSYS Icepak offer higher geometric fidelity at the cost of extended model setup time and computational resources [13]. A single FEM simulation requires detailed geometry definitions, material property assignments, and mesh convergence studies before producing results. Neither approach provides fast component-level temperature estimates from measured data without substantial per-design engineering effort.

Dense thermocouple instrumentation represents a measurement-based alternative but incurs hardware costs, requires physical access to component surfaces, and provides limited spatial coverage—typically monitoring fewer than 25 components per board. For embedded DWPT systems where multiple PCB designs undergo concurrent development, the engineering time required for per-design FEM setup or per-board thermocouple measurements

creates a bottleneck in the design iteration cycle. Chapter 2 traces this progression of methods and identifies the specific limitations that motivate a data-driven approach.

1.2.7 Machine Learning for Thermal Prediction

The preceding sections identified three limitations of existing thermal prediction methods: analytical models discard geometric detail, FEM requires prohibitive setup time, and thermocouples provide only sparse spatial coverage. An alternative method must therefore account for the 2D layout of components and their local heat-spreading interactions, transfer across PCB designs without per-board re-engineering, and produce component-level temperature estimates in seconds rather than the hours required for FEM setup. Machine learning methods meet these requirements by learning input-output mappings directly from measurement data.

Fully Connected Networks

The simplest ML architecture for thermal prediction is the fully connected network (FCN), which maps a fixed-length input vector to scalar or vector outputs through a sequence of dense layers. FCNs have been applied to thermal resistance estimation and junction temperature prediction from tabular operating parameters [21, 22]. These models satisfy the speed requirement and can generalize across operating conditions when trained on sufficient data. However, FCNs discard spatial structure entirely. The input is a flat vector with no positional information. For embedded PCB thermal prediction—where component position, spacing, and local heat-spreading patterns carry physical information—this limitation is fundamental.

Convolutional Neural Networks

Convolutional neural networks (CNNs) address the spatial limitation by extracting features through learned convolutional filters that preserve the 2D structure of image inputs. Standard classification CNNs (e.g., ResNet, VGG) have been adapted for regression tasks by replacing the final softmax layer with linear outputs [23]. For thermal imaging,

CNNs can learn to associate local spatial patterns—such as heat-spread gradients around discrete components—with underlying temperatures [24]. The spatial convolution operations inherent to CNN architectures align naturally with the physics of thermal conduction, where local temperature gradients influence neighboring regions through spatial coupling. CNNs satisfy the spatial awareness requirement, but plain architectures produce a single global output and do not generate the spatially resolved, pixel-level predictions needed for component-level thermal screening.

Encoder-Decoder Architectures

Encoder-decoder architectures produce pixel-level outputs by first compressing the input to a low-dimensional bottleneck representation and then upsampling back to the original spatial resolution. The U-Net architecture [25], originally developed for biomedical image segmentation, has become a standard encoder-decoder backbone for dense prediction tasks. Skip connections between corresponding encoder and decoder layers preserve fine-grained spatial detail that would otherwise be lost during downsampling. The encoder captures progressively larger spatial contexts, from individual component neighborhoods at early layers to board-wide thermal patterns at deeper layers. The decoder then reconstructs the full-resolution prediction using both decoded features and preserved spatial detail from the encoder through skip connections.

For embedded PCB thermal prediction, this architecture satisfies both the spatial awareness and resolution requirements. The output is a pixel-level thermal map from which individual component temperatures can be extracted. Other dense-prediction architectures have been explored for physics-field problems. Operator-learning frameworks such as DeepONet [26] and Fourier Neural Operator [27] learn mappings between function spaces and offer theoretical generalization across domain geometries. However, these frameworks target continuous PDE solution fields and require reformulation for the discrete component layout and mixed boundary conditions characteristic of PCB thermal prediction.

Temporal Modeling Approaches

While the spatial architectures above can predict steady-state thermal fields, embedded DWPT power electronics operate under pulsed duty cycles with transient thermal behavior. Recurrent neural networks (RNNs) and their gated variants—Long Short-Term Memory (LSTM) [28] and Gated Recurrent Units (GRU)—model sequential dependencies in time-series data and have been applied to thermal time-series forecasting from historical sensor readings [29]. RNNs satisfy the temporal modeling requirement but operate on 1D sequences and do not natively process 2D spatial inputs such as thermal images. Hybrid CNN-LSTM architectures combine spatial feature extraction with temporal modeling but increase model complexity and training data requirements.

A practical challenge specific to thermal prediction from images is that spatially identical thermal patterns can correspond to different component temperatures depending on the operating phase. During warm-up, a uniform 30°C thermal image has different implications than the same image during cool-down from a higher operating point. This temporal ambiguity cannot be resolved by spatial architectures alone.

Conditioning Mechanisms

Conditioning mechanisms address the temporal ambiguity without requiring sequential input processing, by injecting external context directly into the spatial feature pipeline. Feature-wise Linear Modulation (FiLM) [30] applies learned affine transformations—scale and shift—to convolutional feature maps based on auxiliary inputs such as elapsed time or operating-point statistics. Originally developed for visual reasoning tasks [30], FiLM has been adapted to physics-informed tasks where a small number of scalar parameters must modulate spatial predictions [31]. The mechanism is lightweight, adding only two parameter vectors per conditioned layer with minimal computational overhead. By embedding operating-phase information as channel-wise transformations throughout the encoder-decoder backbone, FiLM enables a single spatial model to produce temporally contextualized predictions. This satisfies the speed requirement by avoiding the overhead of recurrent processing while retaining temporal awareness.

The combination of an encoder-decoder spatial backbone with FiLM conditioning addresses all three requirements: spatial awareness from the convolutional architecture, generalization potential from training across multiple PCB designs, and rapid inference from a single forward pass. The specific model architecture and training procedure developed for this work are described in Chapter 4.

1.3 Thesis Organization

This thesis contributes a thermally informed design framework for embedded DWPT power electronics. The framework addresses the thermal prediction gap through three complementary approaches: system-level loss characterization under pulsed utilization, automated thermal measurement and validation for embedded PCBs, and ML-based cross-design thermal prediction. Together, these tools enable engineers to evaluate thermal performance at each stage of the design process, replacing conservative worst-case ratings with component selections informed by actual embedded thermal behavior.

The thesis is organized into five chapters. Chapter 2 traces the research progression through multiple thermal analysis methods, beginning with a variable-speed PLECS simulation to characterize system-level losses under pulsed utilization, then exploring component-level analytical methods and reviewing the FEM thermal modeling framework established by prior work. The chapter concludes by identifying the PCB-level prediction gap that motivates the remaining chapters. Chapter 3 describes the automated validation framework for embedded PCBs, covering the SROI file generation pipeline for FLIR thermal imaging and the thermal testing methodology that produces training data for the machine learning model. Chapter 4 presents a FiLM-conditioned U-Net thermal prediction model trained on FLIR thermal images and thermocouple ground truth data from multiple PCB designs, evaluating cross-design generalization for component-level temperature prediction. Chapter 5 summarizes the contributions, provides practical design recommendations, and outlines future research directions.

CHAPTER 2

DESIGN CONSIDERATIONS FOR EMBEDDED POWER ELECTRONICS

Thermal stress is the dominant failure mechanism for roadway-embedded DWPT power electronics. Failure Mode and Effects Analysis (FMEA) consistently ranks compensation capacitors and switching devices as the highest-risk components in these systems [32, 33]. The severity of this risk was demonstrated during prior research at Utah State University, when a DWPT gate driver board embedded in an epoxy and concrete encapsulant experienced thermal runaway that resulted in component damage and unintended power flow through the board [13]. Post-failure analysis attributed the failure to thermal design margins rather than a manufacturing defect. The board operated within its manufacturing specification but exceeded component thermal ratings once embedded in the epoxy and concrete encapsulant. This failure underscored the need for thermal prediction methods that can identify at-risk components before a board is committed to roadway deployment, so that design modifications can be validated prior to installation.

Prior work addressed this need through finite element method (FEM) thermal modeling, developing an ANSYS Icepak framework for embedded DWPT components and validating it through a 25% scale capacitor bank prototype tested in both air and sand environments [13]. That work demonstrated passive cooling feasibility for a distributed low-power-density architecture and confirmed that FEM predictions agree with measured temperatures when boundary conditions are well characterized. However, FEM requires substantial per-design engineering effort—geometry construction, meshing, material assignment, and mesh convergence studies—creating a bottleneck when multiple PCB designs undergo concurrent development. The present thesis extends this prior work by investigating a broader suite of thermal prediction methods—from analytical estimation through circuit simulation to machine learning—that trade off fidelity, computational cost, and adaptability to new designs.

Design engineers developing embedded DWPT power electronics face a practical disconnect between design-stage assumptions and deployed thermal behavior. Prior work on this system assumed continuous full-load operation, yet roadway DWPT systems experience pulsed utilization—the system activates only as vehicles cross transmitter pads. Duty cycles vary with traffic density and vehicle speed, and components designed and validated in air may behave differently when embedded in sand, epoxy, or concrete. This mismatch can require costly redesign after deployment. Engineers need methods that provide timely thermal insight across operating conditions and embedding environments, without months of FEM setup or full-system prototype testing for each design revision.

The embedded architecture migrates the transmitter coil and compensation network into the roadway stack-up, as illustrated in Fig. 1.7. Active cooling is not feasible for long-term roadway embedding due to maintenance access and infrastructure constraints, requiring passive thermal management strategies. Prior work [13] employed a distributed low-power-density architecture using polypropylene film capacitors spread over larger surface areas, exploiting roadway thermal mass for conductive heat transfer.

This chapter traces the research progression through multiple approaches to this thermal prediction challenge. Section 2.1 develops a variable-speed PLECS simulation framework that characterizes system-level losses under pulsed utilization as a function of vehicle speed and pad configuration. Section 2.2 explores component-level analytical methods—inductor core loss using the improved generalized Steinmetz equation (IGSE) and capacitor dielectric loss for the compensation network—followed by lumped-parameter RC thermal models for projecting transient and steady-state temperatures from those losses. Section 2.3 reviews the FEM thermal modeling methodology and experimental validation established in [13], framing it as the highest-fidelity benchmark. Section 2.4 confronts the full-PCB prediction challenge that none of the preceding methods solve in isolation. Section 2.4 compares all methods and identifies the gap that motivates the experimental validation framework (Chapter 3) and machine-learning approach (Chapter 4) in subsequent chapters.

2.1 Simulation Development

Prior research on the USU DWPT system assumed steady-state full-load operation, treating the system as if it were continuously energized. However, roadway DWPT systems experience variable utilization depending on traffic density and vehicle speed. The system pulses on and off as vehicles cross transmitter pads, and the resulting duty cycles directly influence component losses and thermal behavior. Understanding how losses change across utilization ratios is important for thermal design of embedded systems, where the embedding medium retains heat and transient dynamics differ from air-cooled operation. To capture these effects, a variable-speed PLECS simulation was developed in MATLAB/Simulink. The simulation models variable coupling between primary and secondary pads across different vehicle speeds, pad lengths, and multi-second transient scenarios representative of roadway operation. Simulation targets include passive component loss magnitudes and overall system loss as functions of speed and utilization. The per-component loss breakdown also motivates the thermal design goal illustrated in Fig. 2.1: identifying the maximum utilization ratio at which each component remains within its thermal rating.

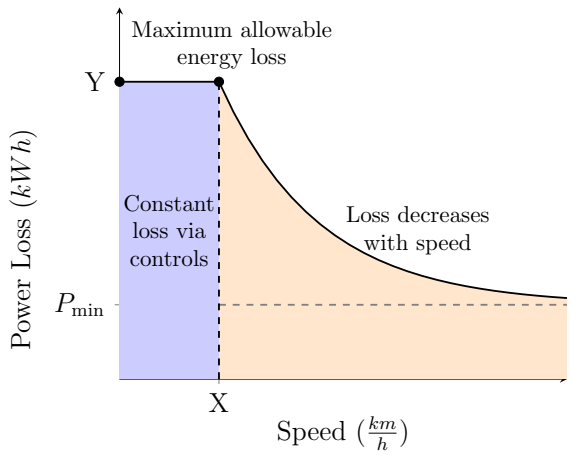


Fig. 2.1: Expected loss behavior of system components under variable utilization.

To begin this process, a simulation capable of variable coupling between the primary and secondary coils was required. Previous work on the simulation of a DWPT system

with a split LCCL-LCCL tuning topology was leveraged as a starting point [6]. Fig. 2.2 shows the split LCCL-LCCL tuned system used in this research, consisting of a full-bridge inverter, primary and secondary LCCL compensation networks, and a diode rectifier feeding the load.

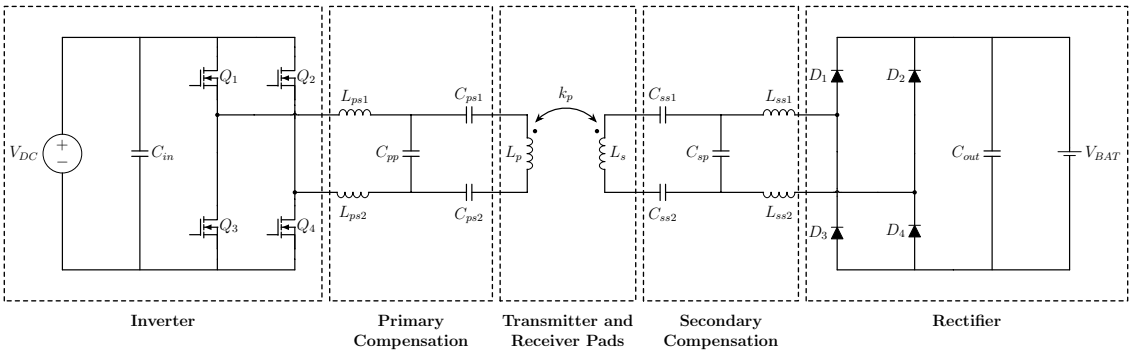


Fig. 2.2: WPT system implementing a split LCCL-LCCL tuning topology.

2.1.1 Coupled Inductor Model

Prior to explaining the variable coupling implementation, the standard coupled inductor model is reviewed. Inductive wireless power transfer systems are typically modeled as a loosely coupled transformer as seen in Fig. 2.3. This example shows a single transmitter and single receiver case for simplicity. The two-port network and current relationships for

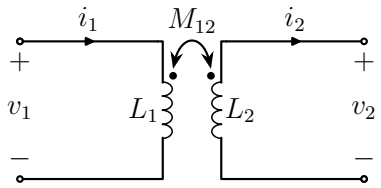


Fig. 2.3: Circuit diagram showing two loosely coupled coils.

this system can be represented in matrix form as given in (2.1).

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} L_1 & M_{12} \\ M_{12} & L_2 \end{bmatrix} \begin{bmatrix} \frac{di_1}{dt} \\ \frac{di_2}{dt} \end{bmatrix} \tag{2.1}$$

where v_1 and v_2 are the terminal voltages of the transmitter and receiver coils, L_1 and L_2 are the self-inductances of each coil, and i_1 and i_2 are the coil currents. M_{12} is the mutual inductance between the two coils and is related to the coupling coefficient k through (2.2).

$$M_{12} = k\sqrt{L_1 L_2} \tag{2.2}$$

For a generalized system with m transmitter coils and n receiver coils, the inductance matrix is an $(m+n) \times (m+n)$ symmetric matrix whose diagonal entries are the self-inductances and whose off-diagonal entries are the pairwise mutual inductances ($M_{ij} = M_{ji}$), as seen in (2.3). In the case of stationary WPT systems with a fixed air gap and alignment, these values are constant.

$$L_{mat} = \begin{bmatrix} L_1 & M_{12} & \cdots & M_{1,m+n} \\ M_{21} & L_2 & \cdots & M_{2,m+n} \\ \vdots & \vdots & \ddots & \vdots \\ M_{m+n,1} & M_{m+n,2} & \cdots & L_{m+n} \end{bmatrix} \tag{2.3}$$

However, for a DWPT system the self and mutual inductances vary as vehicles move over the pads, necessitating a time-varying inductance matrix. The diagonal and off-diagonal entries become functions of time—$L_i(t)$ and $M_{ij}(t)$—as seen in (2.4).

$$L_{mat}(t) = \begin{bmatrix} L_1(t) & M_{12}(t) & \cdots & M_{1,m+n}(t) \\ M_{21}(t) & L_2(t) & \cdots & M_{2,m+n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ M_{m+n,1}(t) & M_{m+n,2}(t) & \cdots & L_{m+n}(t) \end{bmatrix} \tag{2.4}$$

To create this time-varying inductance matrix, a method of calculating the self and

mutual inductances at various positions is needed. These values are dependent on the misalignment between the transmitter and receiver. Because vehicle speed maps position to time, $L_{mat}(t)$ can equivalently be expressed as $L_{mat}(x)$, where x is the longitudinal misalignment between the transmitter and receiver. While lateral misalignment could also be considered, this research focuses on longitudinal misalignment only. The position-dependent inductance matrix is given in (2.5).

$$L_{mat}(x) = \begin{bmatrix} L_1(x) & M_{12}(x) & \cdots & M_{1,m+n}(x) \\ M_{21}(x) & L_2(x) & \cdots & M_{2,m+n}(x) \\ \vdots & \vdots & \ddots & \vdots \\ M_{m+n,1}(x) & M_{m+n,2}(x) & \cdots & L_{m+n}(x) \end{bmatrix} \tag{2.5}$$

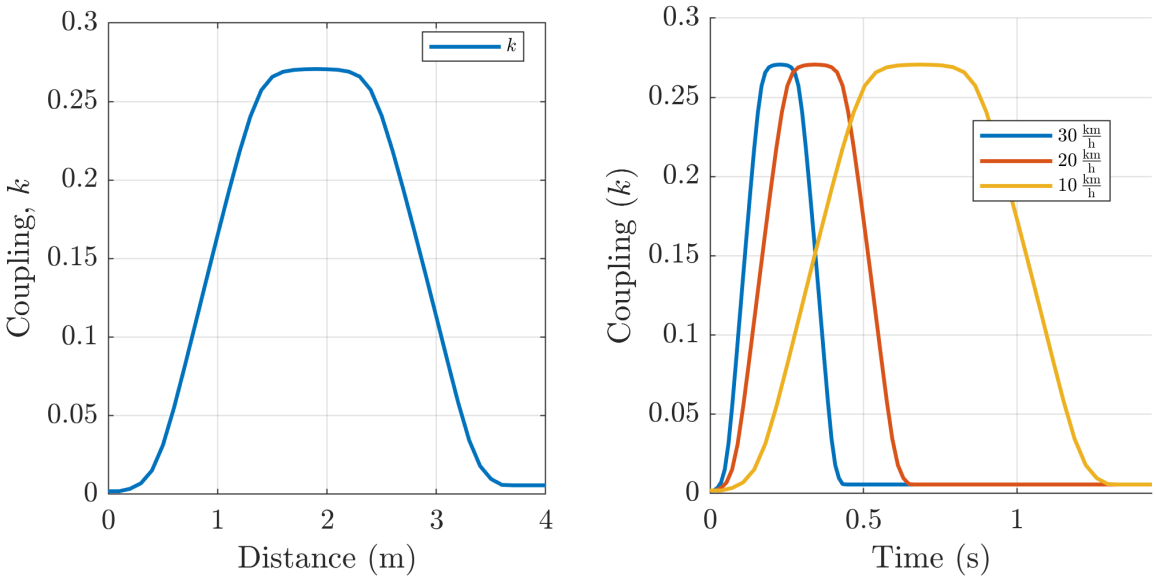
To illustrate this process, Fig. 2.4a shows a simple example of a single transmitter–receiver coil pair with respect to distance across the transmitter coil, while Fig. 2.4b shows the corresponding coupling variation over time as speed changes. For this work, the FEM software Ansys Maxwell was used to calculate the self and mutual inductances at various distances, and prior coil designs were leveraged to create the time-varying $L_{mat}(x)$ matrix within a PLECS simulation.

2.1.2 System Parameters and Loss Models

The circuit model matches the USU DWPT hardware operating at a resonant frequency of $f = 85\,\text{kHz}$ with a DC bus voltage of $V_{\text{DC}} = 600\,\text{V}$ and a rated current of $I_{\text{DC}} = 300\,\text{A}$ per turn. Primary and secondary LCCL component values—bridge inductors, parallel capacitors, series capacitors, and filter elements—are derived from the system's nominal coupling point and the resonant design constraints described in [6].

To capture real component behavior, parasitic resistances are assigned to each reactive element through quality-factor-based models following the methodology used in [34, 35]. Each inductor's equivalent series resistance is computed as

$$R_L = \frac{\omega L}{Q_L} \tag{2.6}$$



(a) Distance vs Coupling

(b) Coupling vs Time with Speed Variation

Fig. 2.4: Single transmitter and single receiver coil longitudinal misalignment and speed variation effects.

where $\omega = 2\pi f$ is the angular switching frequency. The inductor quality factor is set to $Q_L = 400$, and each capacitor's equivalent series resistance is

$$R_C = \frac{1}{\omega C \cdot Q_C} \tag{2.7}$$

with $Q_C = 500$.

These quality factors are representative of high-frequency Litz-wire inductors and celem capacitors included within the system. Including these parasitics ensures that the simulated loss distribution across reactive components reflects measured hardware behavior rather than ideal lossless assumptions.

Switching device losses are modeled using the PLECS thermal description of the Wolfspeed CAB450M12XM3 silicon carbide (SiC) half-bridge module. PLECS computes conduction and switching losses at each simulation time step from manufacturer-provided lookup tables of on-state voltage drop versus current and switching energy versus current and voltage. A junction-to-case thermal resistance of $R_{\theta,\text{jc}} = 0.3\,\text{K/W}$ and an ambient temperature

of $T_{\text{amb}} = 25\,^{\circ}\text{C}$ define the thermal boundary for the semiconductor model, allowing the simulation to track estimated junction temperatures alongside electrical waveforms.

2.1.3 Variable-Speed Sweep Framework

Because a DWPT system activates only as a vehicle crosses each transmitter pad, the speed of the vehicle determines both the duration of power transfer and the resulting duty cycle. Converting the position-dependent coupling profiles from Section 2.1.1 into time-domain waveforms requires a distance-to-time mapping governed by vehicle speed. For a vehicle of length l_{veh} and a nominal inter-vehicle period T_{car} (the time between successive vehicle arrivals at a given transmitter pad), the time that a single receiver pad overlaps a transmitter at speed v is

$$t_{\text{ext}} = \frac{l_{\text{veh}}}{v} \cdot N_{\text{sec}} \tag{2.8}$$

where N_{sec} is the number of secondary (receiver) pads on the vehicle. The simulation sweeps vehicle speeds from 10 to 100 $\frac{km}{h}$ in 10 kph increments and evaluates one-, two-, and three-pad receiver configurations at each speed. This produces a matrix of operating conditions that spans the utilization range from near-continuous to lightly loaded operation.

The simulation workflow is shown in Fig. 2.5. The setup begins with user-defined parameters including initial speed, speed increment, vehicle period, and the number of secondary pads. For each transmitter–receiver pad combination, Ansys-Maxwell-derived coupling profiles versus distance are imported. These inductive coupling values, along with primary and secondary resonant component parameters and parasitic resistances, define the electrical model. The process loops over multiple vehicle speeds, as shown in Fig. 2.5, where distance–time scaling is adjusted to reflect pad transitions at different velocities.

PLECS Standalone executes each operating point with a variable-step solver. The maximum time step is set to $(1/f)/50 \approx 235\,\text{ns}$, providing at least 50 solver evaluations per switching period for adequate resolution of SiC switching transients. A relative tolerance of 10^{-3} and a simulation time span of 50 ms per operating point balance accuracy against computation time. For each case, mutual inductance matrices are generated to cap-

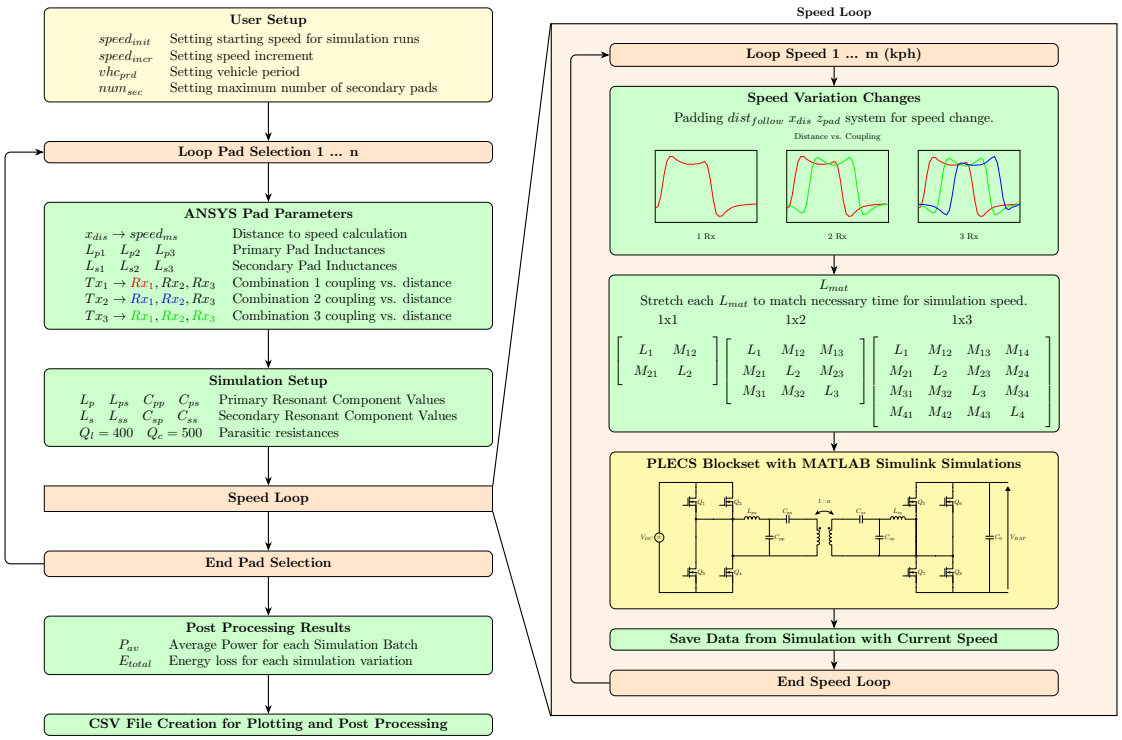


Fig. 2.5: Simulation flowchart for variable speed and coupling DWPT system.

ture multi-receiver interactions, and the corresponding PLECS circuit model is simulated. Each speed and pad combination produces an independent results file, enabling parallelized execution across multiple MATLAB workers when sweep times would otherwise become prohibitive. The simulation setup is seen within the PLECS Standalone environment in Fig. 2.6.

2.1.4 Simulation Results

Post-processing scripts collect the per-case output files, compute average transferred power and total energy loss for each speed and pad configuration, and export the results to CSV files for visualization. This framework enables systematic evaluation across the full matrix of speeds and pad arrangements. Fig. 2.7 shows an example of an application of the simulation to be used for simulating the primary-side power transfer for multiple vehicle speeds and receiver pad configurations, this figure shows the variations of the simulation

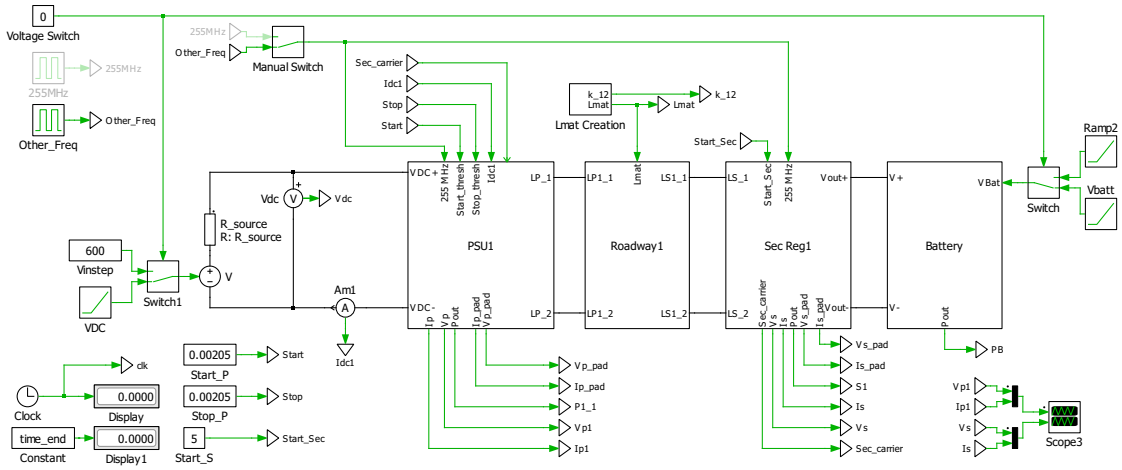


Fig. 2.6: Schematic of the split LCCL-LCCL tuned DWPT system used for simulation development.

with a single transmitter pad and one, two, and three receiver pads for the example split LCCL-LCCL system.

The simulation results validate the power loss distribution within the system and, by extension, the thermal loading on each component. As a first check, the average power transfer was confirmed to decrease with speed, as shown in Fig. 2.8. The parasitic losses within the system were then extracted and the total power loss is plotted in Fig. 2.9a. As speed increases, the duty cycle drops, the average power transfer decreases, and the total loss follows the same downward trend. This can also be seen in the breakdown of losses across system components, shown in Fig. 2.9b, where the loss magnitudes decrease with speed for each component group within the system itself.

However, PLECS produces electrical losses, not temperatures, and translating those losses into thermal predictions requires a separate thermal model. PLECS does support built-in thermal simulation through manufacturer-provided thermal descriptions—for example, the Wolfspeed CAB450M12XM3 SiC module used in Section 2.1 includes conduction- and switching-loss lookup tables and a junction-to-case thermal network. In principle, a designer could construct a complete electrothermal model within PLECS by chaining component thermal descriptions with board- and enclosure-level thermal networks. In practice, however, manufacturer thermal data covers only a limited set of power semiconductor mod-

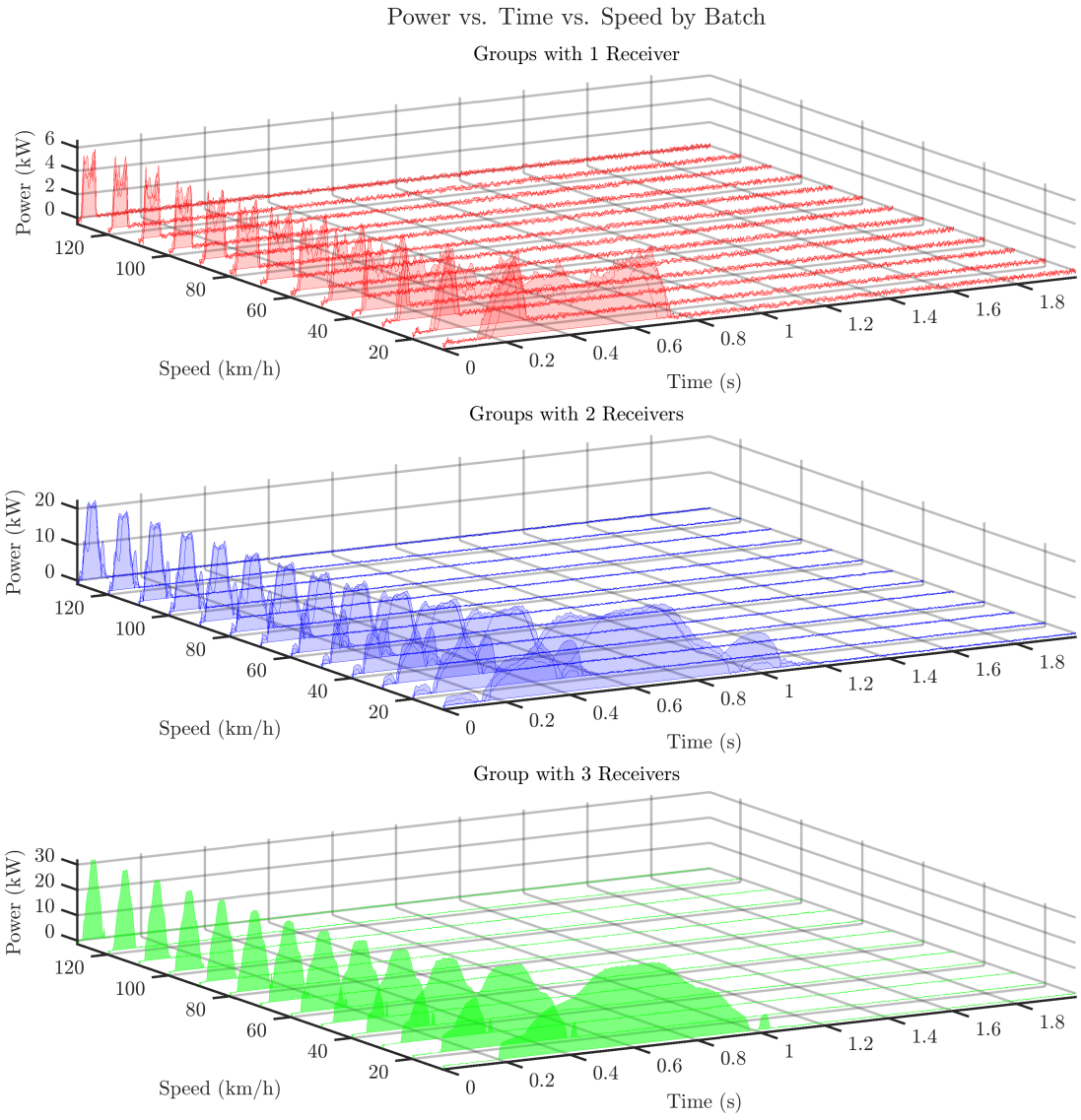


Fig. 2.7: Primary side power transfer for multiple vehicle speeds and multiple receiver pads for example split LCCL-LCCL system.

ules, and extending thermal models to passive components, PCB substrates, and embedding media would require the designer to build custom thermal descriptions for each element—an effort comparable to FEM setup.

These simulation results carry a direct design implication. The component loss data in Fig. 2.9 confirm that systems designed under continuous full-load assumptions are overrated for actual roadway utilization: pulsed operation produces substantially lower average

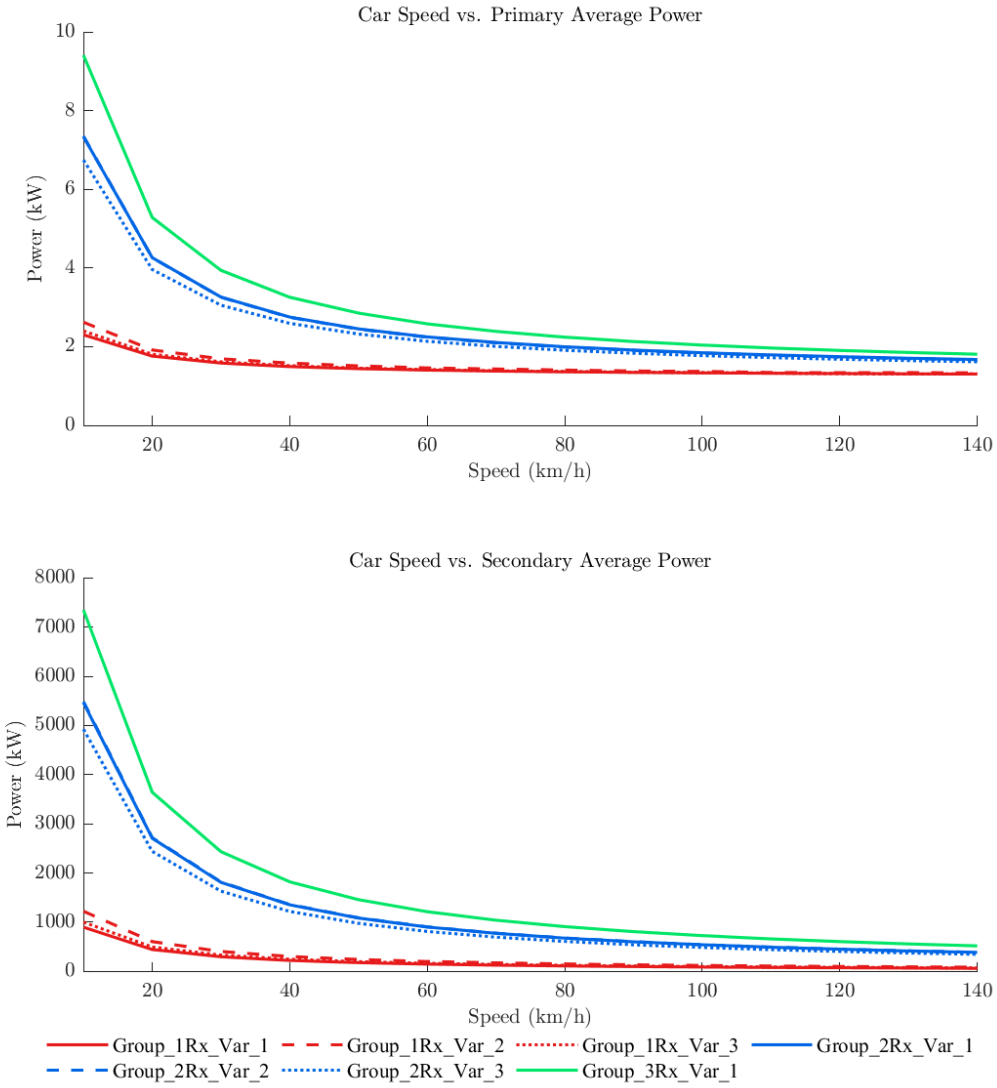
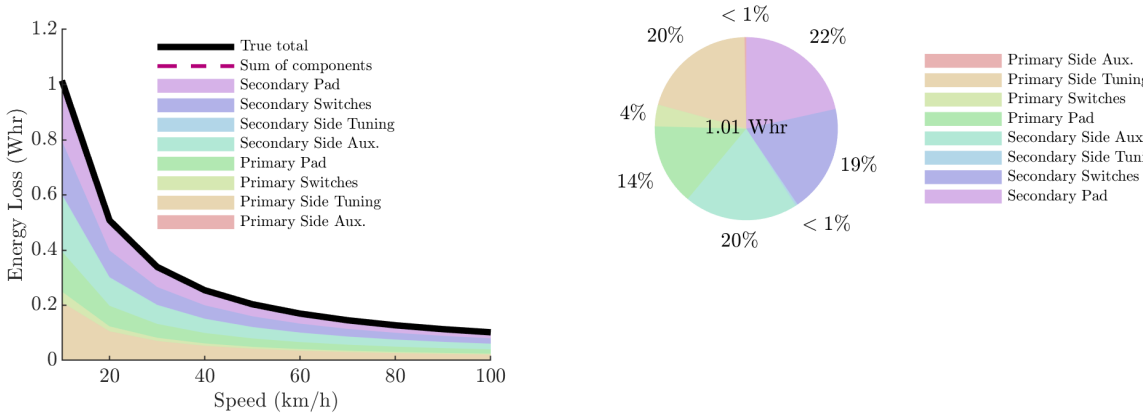


Fig. 2.8: Average power pull of system under variable utilization scenarios for the primary and secondary sides of the example split LCCL-LCCL tuned system.

losses in every component group—compensation inductors, capacitors, and switching devices alike. In principle, a designer could exploit this margin by selecting components rated closer to their actual pulsed thermal envelope rather than their continuous full-load envelope, reducing component size, enclosure volume, and overall system cost. Realizing this design freedom, however, requires thermal models that accurately predict component temperatures under embedded, pulsed conditions. Without such models, a designer has



(a) (b)

Fig. 2.9: (a) Power loss breakdown across system components for multiple vehicle speeds from 10 $\frac{km}{h}$ to 100 $\frac{km}{h}$ to capture the effect of pulsed utilization on component losses. (b) Power loss breakdown across system components for 10 $\frac{km}{h}$ to illustrate the relative contribution of each component to total system loss.

no way to determine how much derating margin can be safely recovered, and conservative continuous-load ratings remain the only defensible choice. The simulation therefore quantifies the *opportunity* for properly rated design but does not, by itself, provide the thermal prediction tools needed to capture it. The remaining sections of this chapter trace the progression toward those tools.

The RC thermal circuit model uses nominal material properties assigned from component datasheets and published literature values; no calibration against the measured thermocouple records was performed. The model is therefore best interpreted as providing order-of-magnitude thermal estimates and relative comparisons between component groups rather than a calibrated predictive tool. A parameterized calibration study—varying the thermal interface paste conductivity, the convective heat transfer coefficient at the exposed resistor and heatsink surfaces, and the sand thermal conductivity and specific heat for the buried configuration—against the experimental thermocouple records would identify a best-fit parameter set with quantified uncertainty bounds. This calibration effort is identified as

future work in Section 5.4.

2.2 Component-Level Thermal Exploration

With system-level losses characterized by the PLECS simulation, the next step was to determine whether individual component thermal behavior could be predicted through analytical methods and scripted calculations. These approaches would give a designer fast turnaround without requiring FEM simulation for each component. The system topology follows the LCCL-LCCL design in Section 2.1. The dominant losses within the system include conduction losses, switching losses, and core losses, distributed across the inverter, rectifier, resonant network, and coils. Previous work addressed the loss mechanisms of an approximate load, capacitor bank, and inverter switching module for a 100 kW split LCCL-LCCL DWPT system [13].

Those calculations assumed steady-state operation and did not account for the pulsed nature of DWPT. This section extends those calculations to include pulsed operation and examines whether the resulting loss estimates, combined with lumped-parameter thermal resistance networks, can produce temperature predictions accurate enough to guide embedded design decisions. Two analytical approaches are examined. First, the Improved Generalized Steinmetz Equation (IGSE) [36] estimates frequency-dependent core losses for the series resonant inductors under pulsed DWPT excitation. Second, lumped-parameter RC thermal circuits following the methodology of [37] translate expected losses within the inverter of the split LCCL-LCCL tuned example system into temperature predictions for a power resistor assembly in air-cooled and sand-embedded configurations.

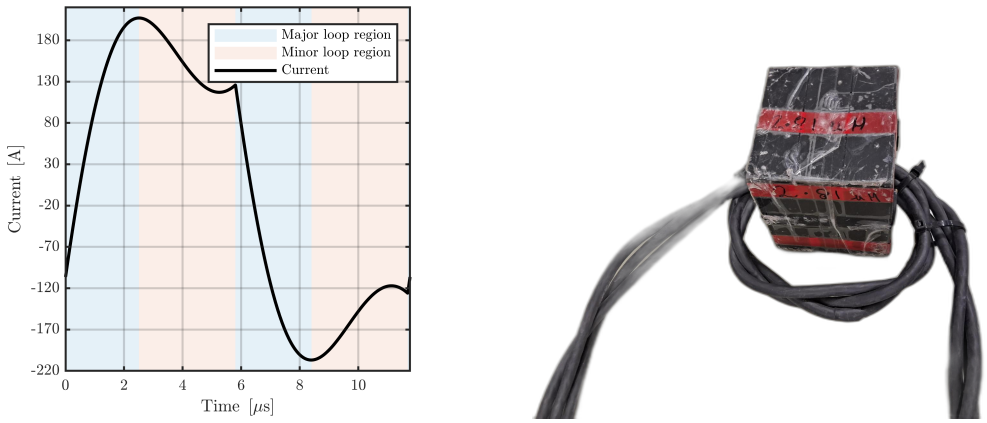
2.2.1 Inductor Core Loss Approximation

For an inductor the losses can be broken down into two main categories, core losses and copper losses. Core losses can be predicted through the use of Steinmetz equations (SE), which approximate core loss by leveraging empirical data provided by core manufacturers. Methods of accurately calculating core loss for an inductor are well known and have been improved upon since the inception of SE [36, 38, 39]. The original Steinmetz equation is

given in (2.9), where P_v is the core loss per unit volume, f is the frequency of operation, B is the peak flux density, and k, α, and β are curve-fitting coefficients determined empirically for the core material.

$$P_v = k\, f^{\alpha}\, B^{\beta} \tag{2.9}$$

While (2.9) allows for accurate approximation of the empirical data provided for various core materials, this equation assumes a sinusoidal current waveform present throughout the inductor. For a DWPT system, a non-sinusoidal current waveform is present due to the behavior of the resonant network, which in turn, presents a challenge for accurately calculating core loss due to the non-ideal waveform as seen in Fig. 2.10a.



(a) (b)

Fig. 2.10: (a) Current waveform for the primary series inductor of a split LCCL-LCCL topology, (b) LCCL tuned inductor used for IGSE core loss approximation and case study of complexity of loss approximation due to non-sinusoidal waveforms.

For a more accurate approximation of the core losses within an inductor, approximating the waveform as a piecewise function is possible, and allows for a more accurate core loss approximation. This process involves identifying the minor and major loops of the waveform, calculating the loss for each segment, and then summing these losses. This process is outlined in [36] and improved upon in [39].

For this research, the MATLAB script developed within [36] is accurate enough when

using one period of a waveform to accurately predict the core loss based on only Steinmetz parameters, i.e. k, α, and β. The equations utilized within this script are given in Equations 2.10 and 2.11, where k_i is a modified Steinmetz coefficient, ΔB is the peak-to-peak flux density, T is the period of the waveform, V_j is the voltage across the inductor during time step j, N is the number of turns of the inductor, A_c is the cross-sectional area of the core, and Δt_j is the time duration of time step j.

$$P_v = \frac{k_i (\Delta B)^{\beta-\alpha}}{T} \sum_j \left| \frac{V_j}{N A_c} \right|^{\alpha} (\Delta t_j) \tag{2.10}$$

$$k_i = \frac{k}{2^{\beta+1} \pi^{\alpha-1} \left(0.2761 + \frac{1.7061}{\alpha+1.354}\right)} \tag{2.11}$$

Leveraging this original script, and the main core loss calculation function within, a new script was built to automate the process of inputting the empirical data from datasheets, specifically from the power per unit volume vs frequency plot. This new script converts this picture into data points from minimal user input and automatically calculates the Steinmetz parameters to fit the data. The ferrite used in the 100 kW split LCCL-LCCL DWPT system is N87 ferrite from TDK [3]. A typical U and I core combo would be using U 93/76/30 with I 93/28/30, the specifics of which are given in [40]. From these datasheets, the empirical data seen in Fig. 2.11 is given.

Using the script mentioned previously, the Steinmetz parameters were calculated for this core material.

Table 2.1: Steinmetz parameters for TDK N87 ferrite core.

Temperature (°C)	α	β	k
25	1.3467	2.5115	8.2381e-03
100	1.7950	3.0640	9.3162e-05

Figs. 2.12a and 2.12b show the curve fit applied to the empirical data given. Due

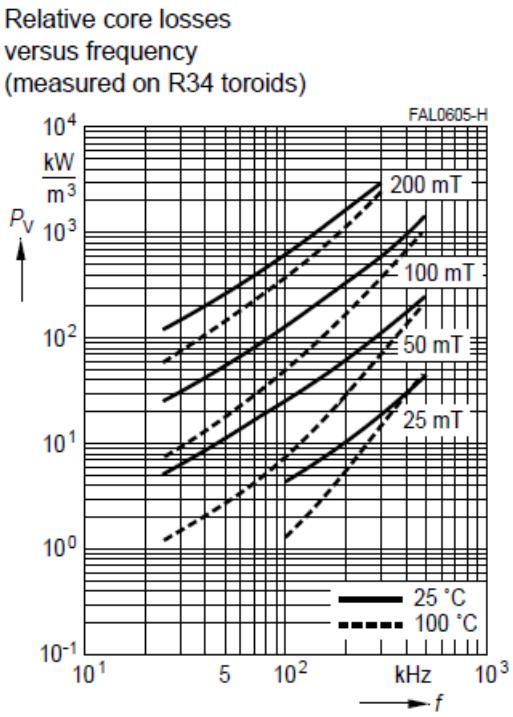


Fig. 2.11: Empirical data for TDK N87 ferrite core loss per unit volume. [3]

to the temperature dependence of the core material, this process was repeated for two temperatures, 25°C and 100°C. These two temperatures were chosen to represent the typical operating temperature of the system, and a high operating temperature which the system may see during operation. The TDK Ferrite Magnetic Design Tool (MDT) was used to verify the accuracy of the Steinmetz parameters calculated at 100°C within 2.5% accuracy and the values calculated at 25°C are expected via extrapolation of this tool's data [41].

This allows flux-density calculation directly from the inductor current waveform, enabling accurate core loss approximation. Fortunately, the inductance, number of turns, and cross-sectional area of the core are known, allowing for conversion from current to flux density via (2.12).

$$B(t) = \frac{i(t)L}{NA_c} \tag{2.12}$$

where $B(t)$ is the flux density, $i(t)$ is the current through the inductor, L is the inductance of the inductor, N is the number of turns of the inductor, and A_c is the cross-sectional area

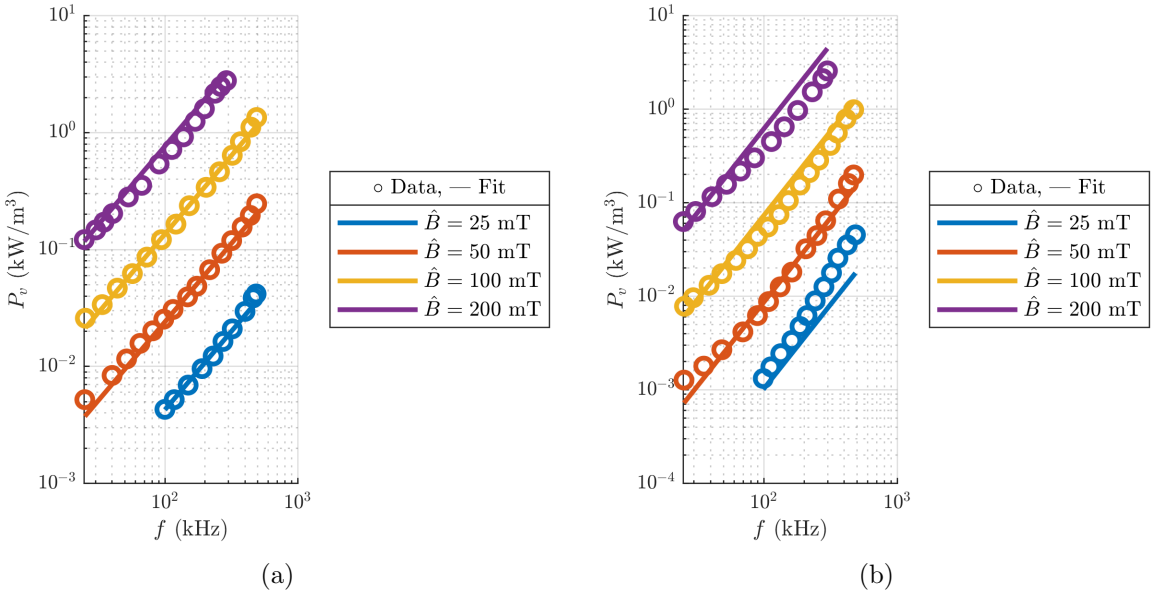


Fig. 2.12: Approximation of empirical data at 25°C and 100°C.

of the core. This waveform mirrors the current waveform, and is shown in Fig. 2.13.

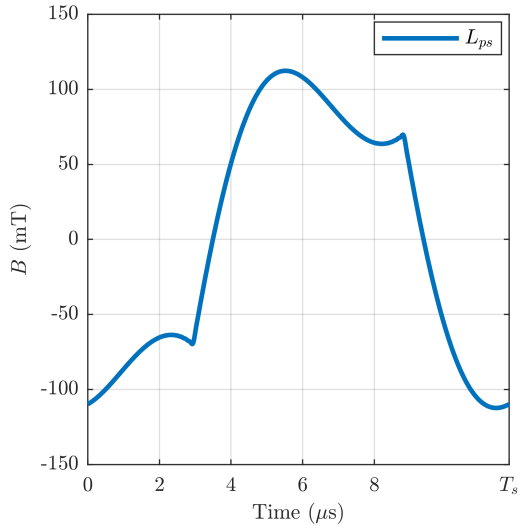


Fig. 2.13: Flux density waveform for the primary series inductor of the split LCCL-LCCL topology.

Utilizing the flux waveform and the Steinmetz parameters seen in Table 2.1, the core loss at the two operating temperatures provided in the datasheet are shown in Table 2.2.

Table 2.2: Core loss for primary series inductor of split LCCL-LCCL topology.

Temperature (°C)	**Core Loss (W)**
25	118.613
100	83.822

As a further double-check, these values match the previous trend seen within the reference data as the core increases in temperature for this material. These data points enable interpolation, as illustrated in Fig. 2.14, to estimate core loss at intermediate temperatures. With the inductor core losses established, the next step is to determine whether

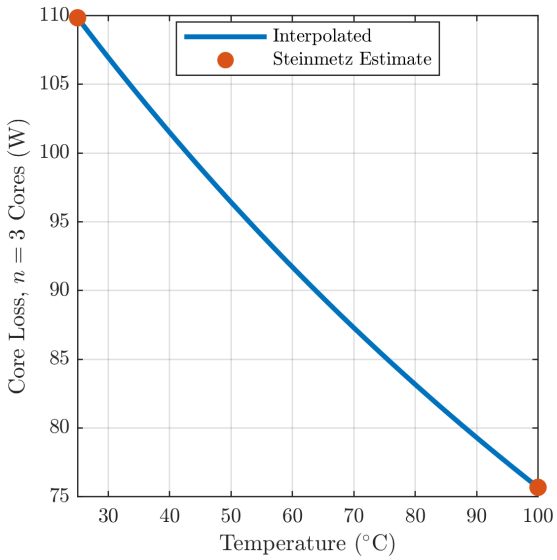


Fig. 2.14: Simple interpolation from 25°C to 100°C for core loss approximation.

these losses—combined with lumped-parameter thermal circuits—can produce useful temperature predictions. These core loss results feed directly into the thermal resistance models developed in Section 2.2.2.

This automated pipeline provides efficient loss prediction for a specific core geometry and material at multiple temperatures. However, inductor core loss represents only one facet of a complete embedded inductor thermal model. Predicting inductor thermal

behavior within a roadway would also require characterizing the magnetic effects of surrounding materials—concrete, sand, and ferrite proximity—on inductance and coupling. The thermal interface between the inductor assembly and the embedding medium adds further complexity. Prior work demonstrated that even modeling the compensation capacitor bank, a single component type, required parameterized FEM simulations to achieve acceptable accuracy [13]. Extending the same methodology to the inductor, with its additional magnetic design complexity, exceeds the scope of this research. The inductor loss analysis presented here provides a validated starting point for core loss estimation, while embedded thermal prediction for this component remains an open challenge.

2.2.2 Lumped-Parameter Thermal Circuit Analysis

With component-level losses established, the next step was to determine whether lumped-parameter RC thermal circuits could translate those losses into usable temperature predictions without FEM. In an RC thermal circuit, heat flow is modeled as current through a resistance–capacitance network: thermal resistance replaces electrical resistance, temperature replaces voltage, and power dissipation replaces current. This approach has been demonstrated for liquid-cooled high-power EV charger modules, constructing multi-node RC networks from manufacturer thermal data and material properties to predict junction temperatures for T-type power modules on cold plates [37]. For this research, an analogous methodology was applied to a simpler target: the power resistor load assembly used in prior proxy inverter load simulation development and testing [13].

The steady-state temperature rise above ambient for any node in such a network is

$$\Delta T = P_{\text{loss}} \cdot R_{\theta,\text{ja}} \tag{2.13}$$

where P_{loss} is the component power dissipation and $R_{\theta,\text{ja}}$ is the total source-to-ambient thermal resistance. For the power resistor assembly, the total resistance is the series sum of individual layer resistances along the dominant heat path: $R_{\theta,\text{ja}} = R_{\theta,\text{core}} + R_{\theta,\text{shell}} + R_{\theta,\text{TIM}} + R_{\theta,\text{hs}} + R_{\theta,\text{amb}}$, where the subscripts denote core-to-shell, shell-to-heatsink (through the

thermal interface material), heatsink spreading, and heatsink-to-ambient paths respectively. Transient behavior is captured by augmenting each node with a thermal capacitance $C_\theta = \rho \cdot c_p \cdot V$, where ρ is the material density, c_p is the specific heat capacity, and V is the element volume. This yields a first-order time constant $\tau = R_\theta \cdot C_\theta$ per layer and a step response of

$$T(t) = T_{\text{amb}} + P_{\text{loss}} \cdot R_{\theta,\text{ja}} \left(1 - e^{-t/\tau_{\text{eq}}}\right) \tag{2.14}$$

where τ_{eq} is the equivalent time constant of the lumped network. For DWPT operation the pulsed nature of vehicle crossings introduces cyclic thermal loading, and when thermal time constants (seconds to minutes) are much longer than the activation pulse (60 ms for a 1 m pad at 60 km/h), the response approaches a time-averaged steady state scaled by the duty cycle $D = t_{\text{ext}}/T_{\text{car}}$.

To test this approach, PLECS thermal circuit models were built for the power resistor assembly in both air-cooled and sand-embedded configurations. Each resistor dissipates 45 W and is mounted on a Wakefield 510-9M heat sink via Laird Tgrease-1500 thermal interface material. Fig. 2.15 shows the physical assembly.

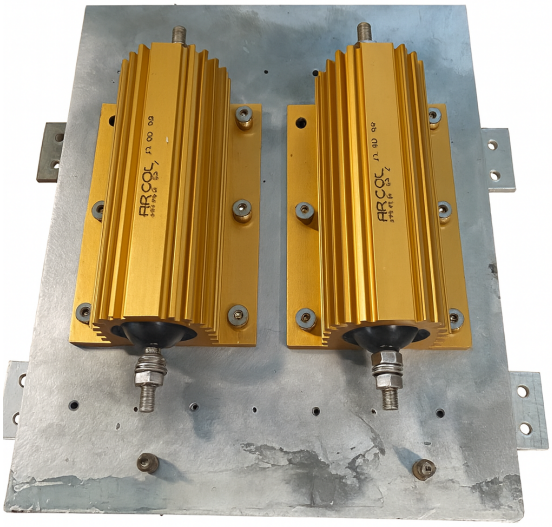


Fig. 2.15: Power resistor load assembly with heatsink.

The air-cooled model traces the heat path from the ARCOL cylindrical resistor body—through the ceramic core, aluminum shell, thermal paste interface, and heat sink fins—to the surrounding air ($T_{\text{amb}} = 20\,^{\circ}\text{C}$). Each layer in the path contributes a thermal resistance computed from its geometry and material conductivity: aluminum ($k = 210$ W/mK), ceramic ($k = 28$ W/mK), and thermal paste ($R''_{pp} = 0.021$ in^2 °C/W per the Laird Tgrease-1500 datasheet, ASTM D5470 test method).

Finally, heat dissipation from the fins to the room air was modeled using published natural convection correlations for vertical fin channels [42, 43]. A blended convective coefficient accounts for air drawn into the fin channels from below and along the open fin tips, and fin efficiency was included to correct for the temperature gradient along each fin. Inter-fin radiation was estimated from fin surface view factors ($\varepsilon_{\text{Al}} = 0.06$), though its contribution is small compared to convection. The enclosure was modeled in a "box" mode with 3 mm clearance beneath the heat sink to approximate the semi-confined test environment. Fig. 2.16 shows the resulting RC circuit representation for the air-cooled configuration.

For the sand-embedded model, the convective fin-to-air boundary was replaced with conductive heat transfer into the embedding medium ($k_{\text{sand}} = 0.22$ W/mK). The sand domain geometry was parameterized from the HDPE test container (0.610 m × 0.406 m × 0.352 m), and sand thermal properties—density ($\rho_{\text{sand}} = 1700\,\text{kg/m}^3$) and specific heat ($c_{p,\text{sand}} = 800\,\text{J/kgK}$)—were estimated from published values for dry play sand. Because sand fills the space around and between the fins, the embedded model requires additional thermal paths not present in the air model. These include conduction from the heatsink envelope into the sand in multiple directions (downward, sideways, upward, and longitudinally), the thermal mass of sand packed between the fins, and the HDPE container walls that form the outer boundary. The outer container wall uses a convective coefficient of $h = 7.5$ W/m^{2}K to room air ($T_{\text{amb}} = 21\,^{\circ}\text{C}$), and the simulation duration was extended to 28,800 s (8 hours) to capture the full experimental transient. Fig. 2.17 shows the RC circuit representation for the sand-embedded configuration, and Fig. 2.18 shows the full PLECS schematic used to implement the embedded thermal model.

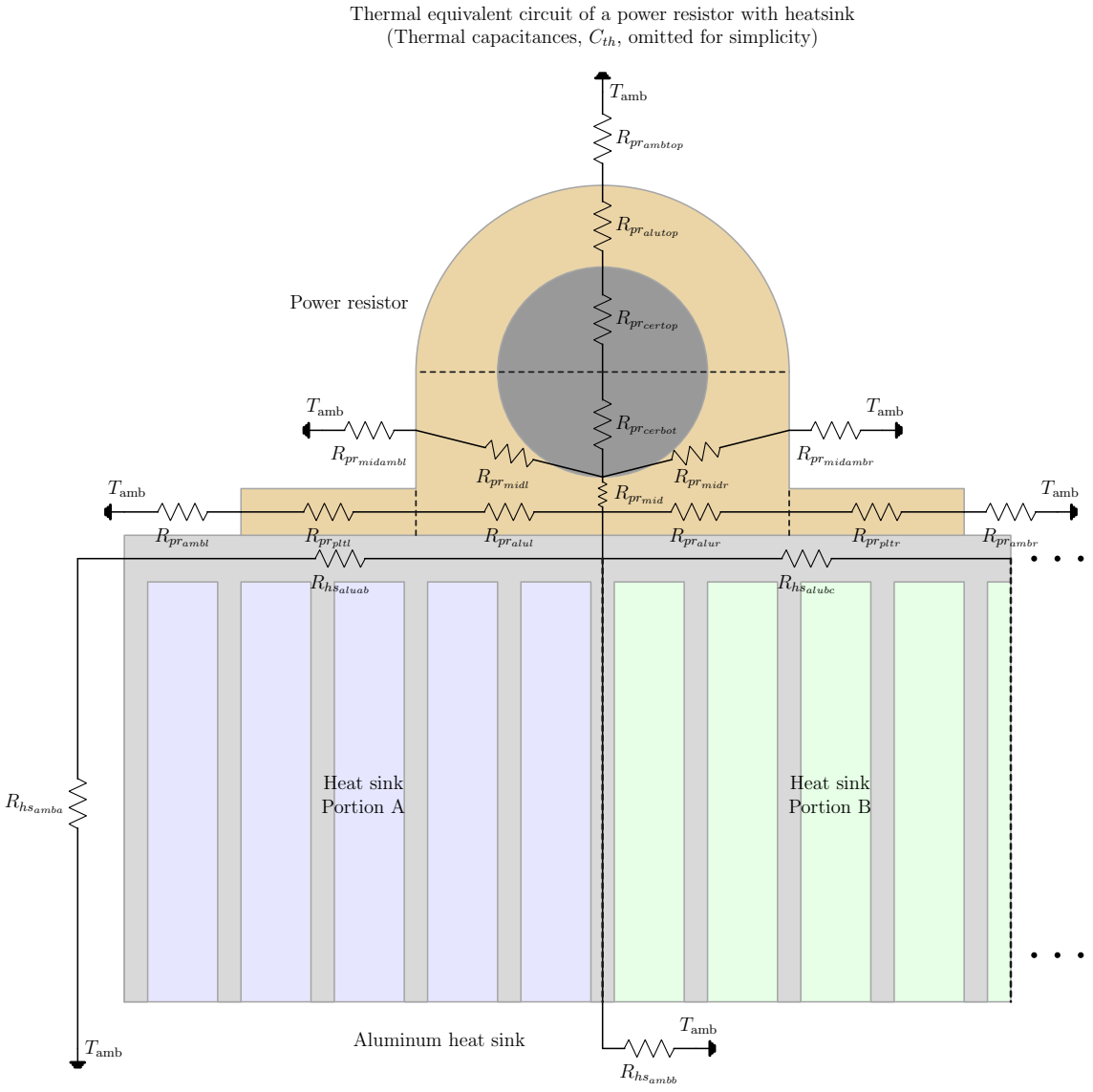


Fig. 2.16: RC thermal circuit model for the air-cooled power resistor proxy inverter load assembly.

MATLAB initialization scripts defined all geometry, material properties, thermal resistances, and capacitances, and the PLECS solver computed the transient temperature response at each circuit node. As shown in Fig. 2.19, the air-cooled analytical model tracks the measured temperature trend well. The steady-state resistor temperature predictions are within approximately 5 °C of the thermocouple measurements, and the heatsink predictions are within approximately 10 °C. The analytical model consistently over-predicts

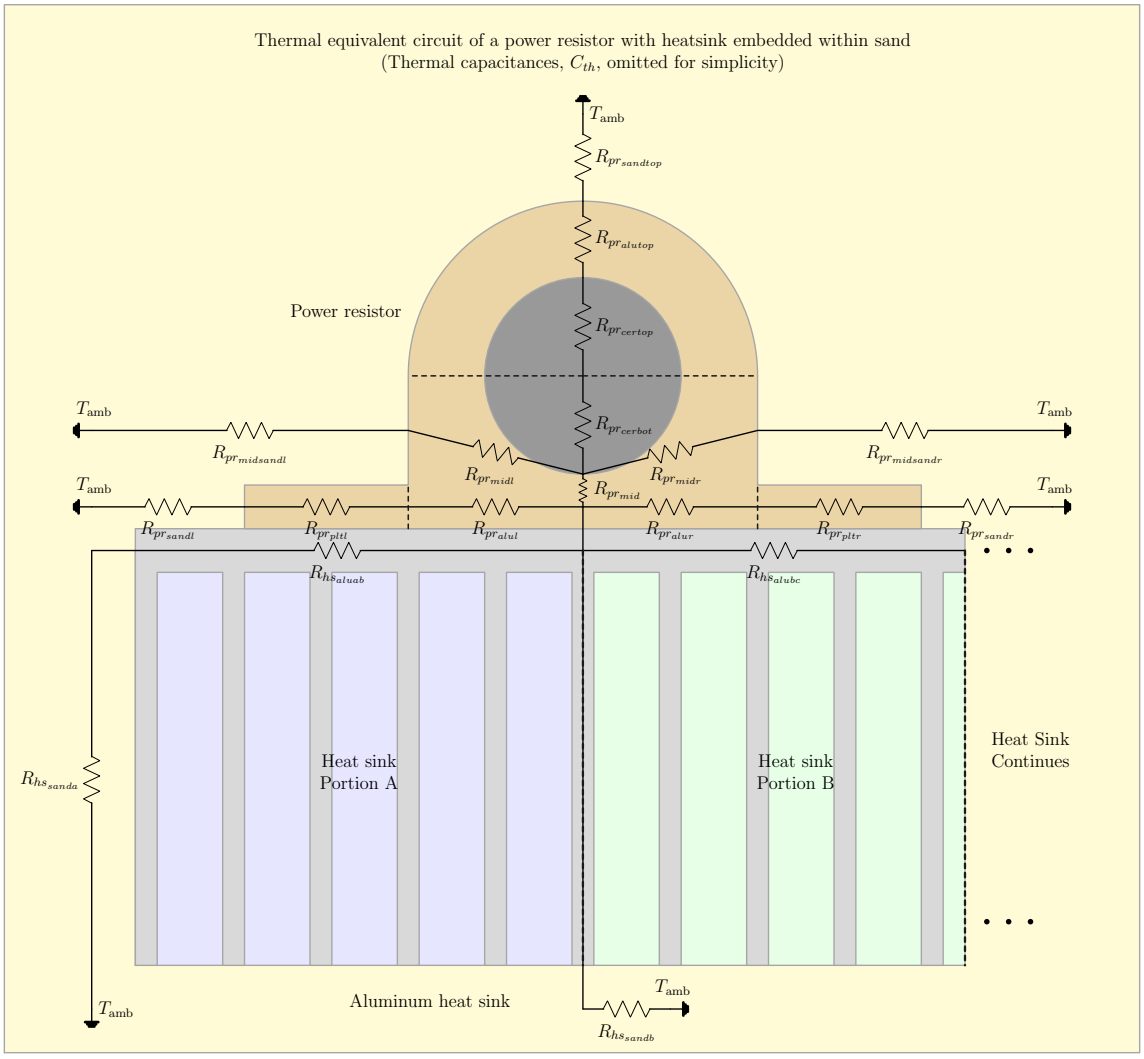


Fig. 2.17: RC thermal circuit model for the sand-embedded power resistor proxy inverter load assembly.

temperature, which is conservative from a design perspective.

The sand-embedded configuration (Fig. 2.20) shows similar trend agreement, but the analytical model over-predicts temperatures by approximately 10 °C throughout the entire transient for both the resistor and heatsink locations. This persistent offset reflects the difficulty of accurately modeling sand as a thermal medium: the effective thermal resistance and capacitance of the sand depend on packing density, moisture content, and contact quality between the assembly and the surrounding material—parameters that vary between experimental setups and are not available from any datasheet. As a result, the analytical

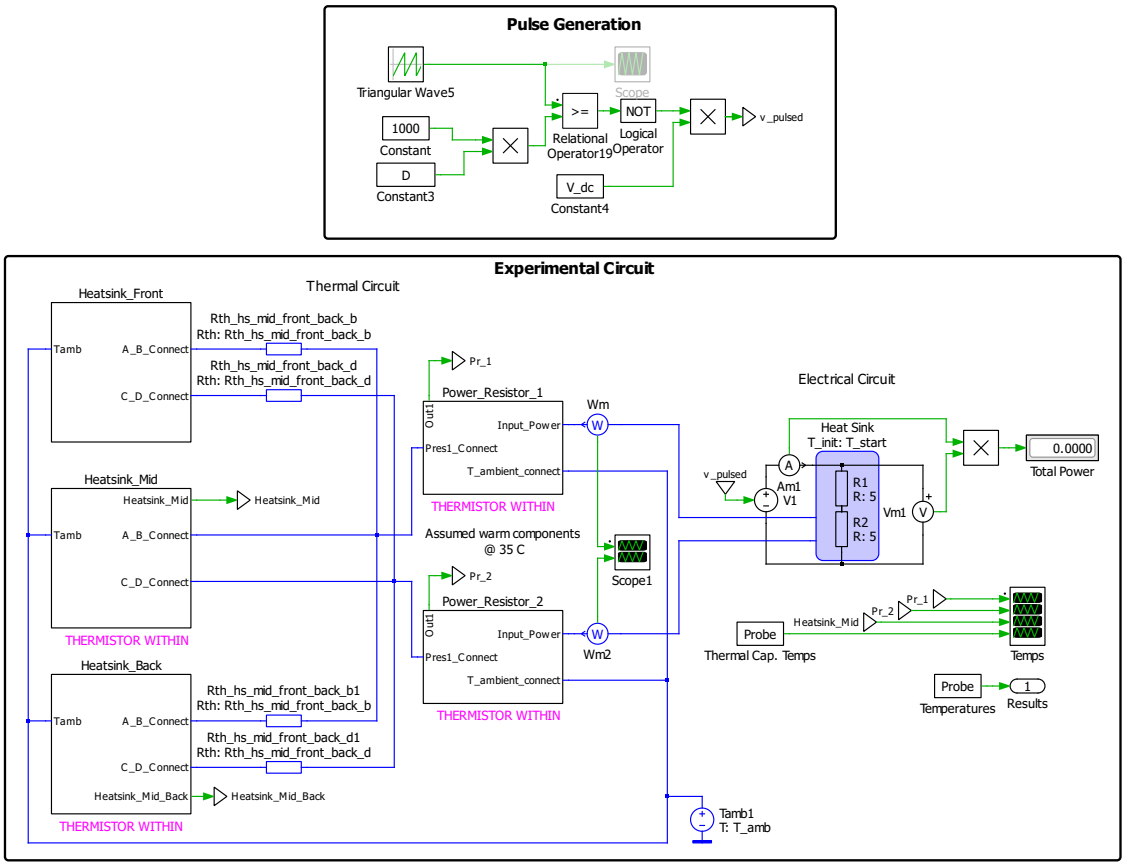


Fig. 2.18: PLECS RC thermal schematic for the sand-embedded power resistor load assembly.

model captures the qualitative thermal behavior but cannot match the ground-truth values without empirical calibration of the sand interface properties.

This exercise revealed a broader limitation. While the RC thermal circuit approach provides fast turnaround—the MATLAB scripts run in seconds on standard hardware—each new component or assembly requires its own custom circuit topology, geometry measurements, and material characterization. The T-type module analysis in [37] succeeded because the cold-plate boundary condition is well defined by the liquid flow rate and inlet temperature.

For embedded geometries, the thermal interface between the component and the surrounding medium introduces uncertainties that analytical models do not resolve without

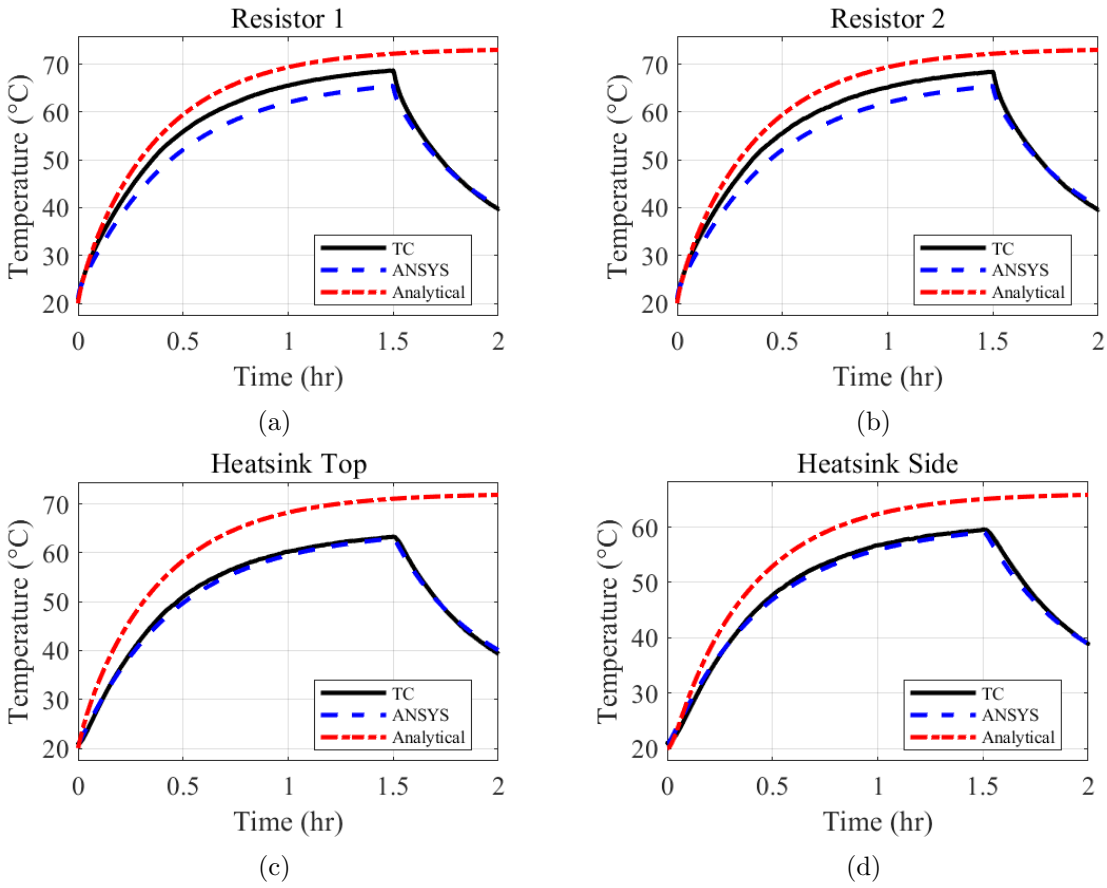


Fig. 2.19: Comparison of measured temperatures, ANSYS Icepak FEM predictions, and lumped-parameter RC thermal circuit predictions for the power resistor proxy inverter thermal load assembly within air: (a) Resistor 1, (b) Resistor 2, (c) Heat sink top, (d) Heat sink side.

empirical calibration for each specific embedding scenario. Building an accurate lumped-parameter model for even the relatively simple power resistor assembly required dozens of geometric parameters, multiple thermal interface assumptions, and iterative tuning—effort that approaches the FEM setup it was intended to replace. Extending this process to a complete PCB with dozens of heterogeneous components would be impractical. This conclusion directed the research toward full-PCB thermal measurement and data-driven prediction methods explored in subsequent chapters.

2.3 FEM Thermal Modeling for Embedded Geometries

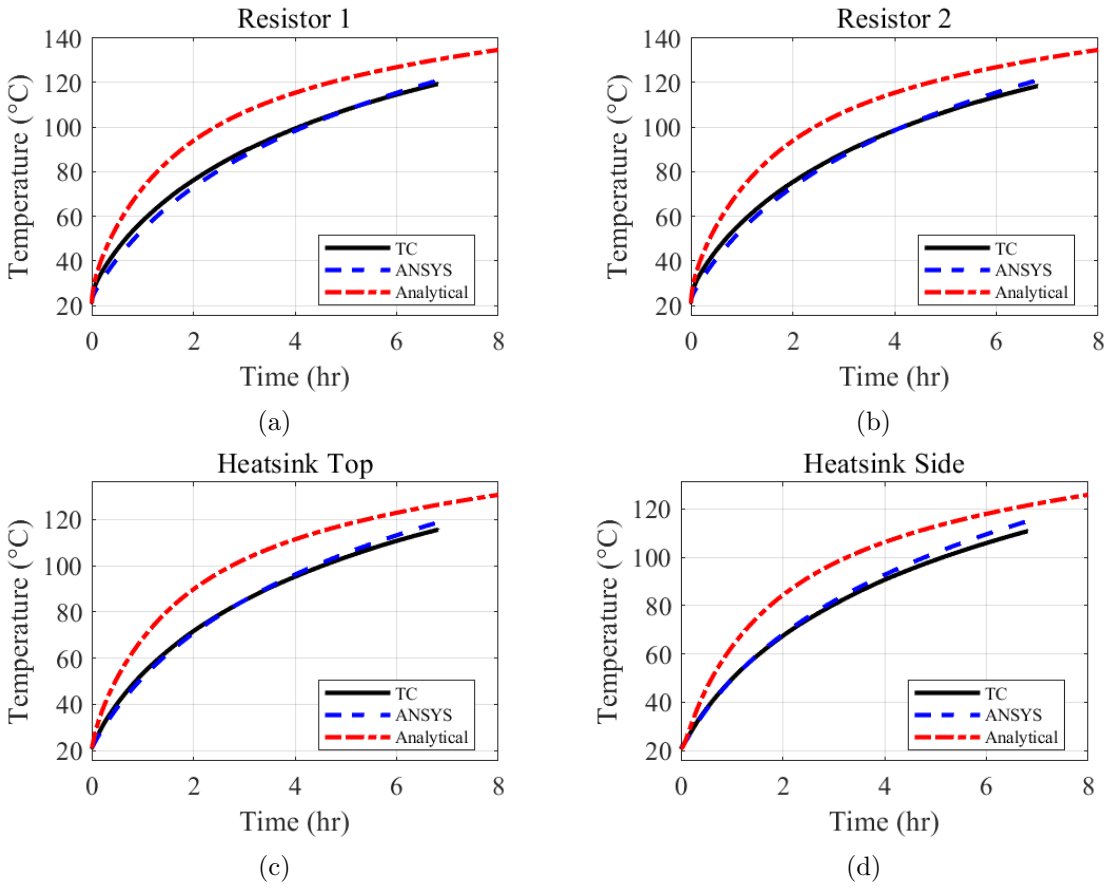


Fig. 2.20: Comparison of measured temperatures, ANSYS Icepak FEM predictions, and lumped-parameter RC thermal circuit predictions for the power resistor proxy inverter thermal load assembly within sand: (a) Resistor 1, (b) Resistor 2, (c) Heat sink top, (d) Heat sink side.

The gate driver failure described in the chapter introduction demonstrated that thermal predictions from electrothermal analytical or simulation methods alone are insufficient for embedded power electronics: the complex thermal interfaces between PCB components and their embedding medium introduce uncertainties that lumped-parameter models do not resolve. Prior work [13] addressed this gap by developing a finite element method (FEM) thermal modeling framework using ANSYS Icepak, validated against hardware prototype measurements in representative embedded environments.

The FEM approach models the complete thermal domain including component internal structure, PCB substrate layers, solder interfaces, and the surrounding embedding material.

Material thermal properties are assigned from manufacturer datasheets and literature values, and boundary conditions are configured for each test environment: natural convection coefficients for air testing, and conductive interface models for sand or roadway-material embedding.

This FEM framework was validated through a 25% scale prototype of the split LCCL-tuned compensation capacitor bank for a 100 kW DWPT system, tested in both air and sand environments [13]. Sand testing served as a practical surrogate for roadway embedding at a fraction of the cost of full roadway prototyping, while still capturing the conductive thermal boundary conditions that distinguish embedded operation from air-cooled operation. The following subsections detail the ANSYS Icepak model development, the prototype fabrication and test setup, and the experimental validation results—all conducted as part of prior research and summarized here to provide the reader with a self-contained understanding of the FEM methodology and its demonstrated capabilities.

2.3.1 ANSYS Icepak Model Development

ANSYS Icepak FEM simulations model thermal behavior using component geometry, material properties, and power dissipation derived from the analytical loss models in Section 2.2. Compensation capacitors in the LCCL resonant network represent the second major source of thermal loss in the DWPT system after inductor core losses. The following design process is illustrated within the generalized thermal modeling and design flow shown in Fig. 2.21. For polypropylene metallized film capacitors operating at 800 V RMS and 85 kHz in the split LCCL topology [44], power dissipation arises from dielectric losses in the polypropylene film. The capacitor power loss for each unit within the bank is calculated using

$$P_{\text{cap}} = V_{\text{rms}}^2 \cdot C \cdot 2\pi f_s \cdot \tan\delta \tag{2.15}$$

where V_{rms} is the RMS voltage across the capacitor, C is the capacitance, f_s is the switching frequency, and $\tan\delta$ is the dissipation factor provided by the manufacturer. For the distributed low-power-density architecture employed in this work, individual capacitors are

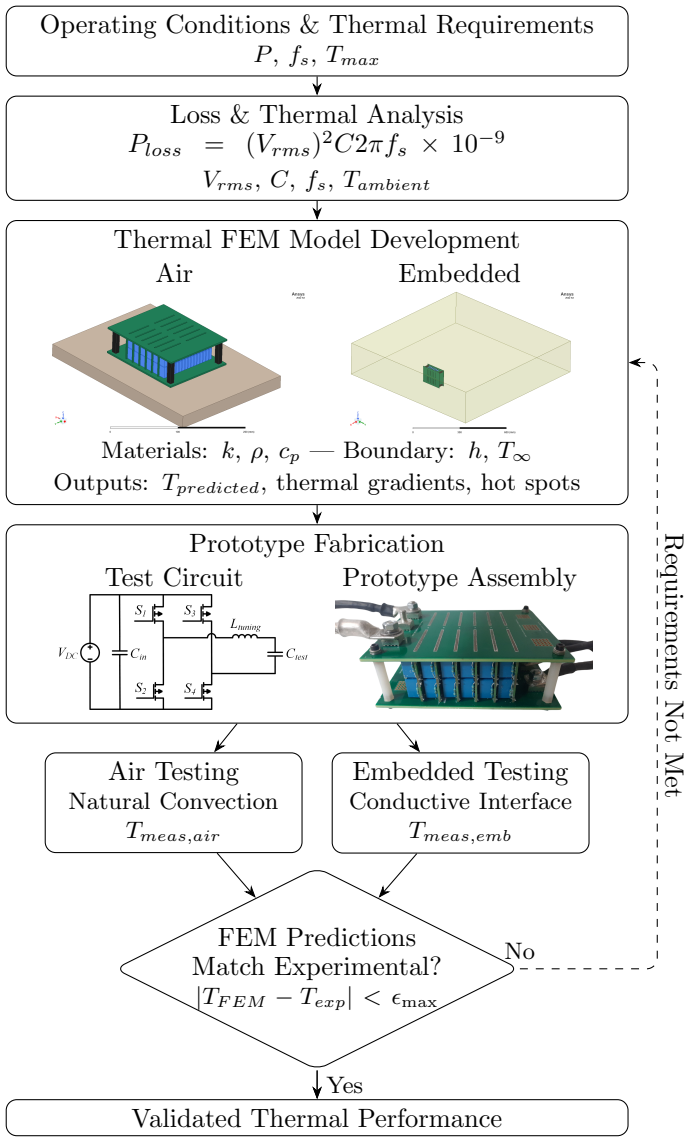


Fig. 2.21: Generalized thermal modeling and design flow for embedded DWPT systems.

derated to 80% of rated voltage and current to maximize longevity under continuous thermal cycling [13]. This derating reduces per-component power dissipation and distributes the total loss over a larger surface area, enabling passive cooling through conductive heat transfer to the surrounding roadway material. The total compensation network loss is the sum of individual capacitor losses across the bank, and this aggregate value serves as the power dissipation input for the FEM thermal model.

The FEM model includes simplified representations of the capacitor internal structure

(metalized film layers, end spray contacts, lead terminations), FR4 PCB substrate with copper traces, solder interfaces, and the surrounding material domain. Material thermal properties are assigned from manufacturer datasheets: polypropylene film ($k = 0.22$ W/m·K), FR4 substrate ($k = 0.3$ W/m·K in-plane, $k = 0.06$ W/m·K through-plane), copper traces ($k = 385$ W/m·K), and solder joints ($k = 50$ W/m·K).

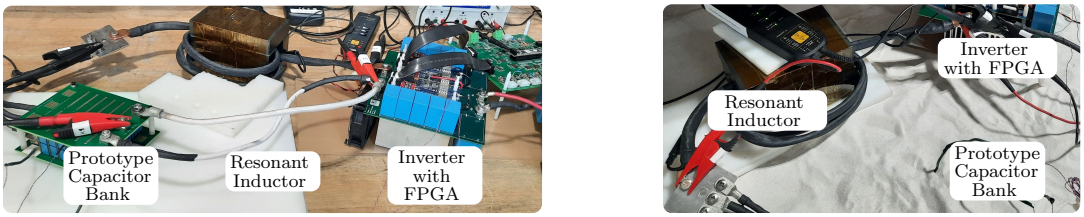
Thermal boundary conditions are configured for each test environment. For air testing, a natural convection coefficient ($h = 5$–10 W/m^2·K) is applied to all exposed surfaces with ambient temperature set to the room temperature (23°C), and radiation exchange is included for surfaces above 60°C where radiative losses become non-negligible. For sand testing, a conductive interface between the capacitor bank and sand medium ($k_{\text{sand}} \approx 0.3$ W/m·K for dry sand) is modeled with a contact resistance parameter at the component-sand interface, and the sand domain extends sufficiently to avoid boundary effects on the thermal solution. Sensitivity analysis is performed on parameters with the highest uncertainty: the thermal interface resistance between capacitor packages and the sand medium, and the effective thermal conductivity of the sand which varies with moisture content and compaction.

2.3.2 Capacitor Bank Prototype Testing

A 25% scale prototype capacitor bank was fabricated and tested using a simpler test circuit with equivalent 100 kW power dissipation (matching full LCCL power losses) in air and sand environments. The prototype uses TDK polypropylene film capacitors selected for the distributed low-power-density architecture. Thermocouples monitor temperatures at key stress points identified through the FEM thermal model, and experimental results are compared against FEM predictions for both environments.

The hardware test setup for air testing, shown in Fig. 2.22a, uses natural convection in a controlled ambient environment (23°C). Sand testing, shown in Fig. 2.22b, embeds the prototype in a sand bed to emulate roadway thermal boundary conditions.

2.3.3 Air and Sand Validation Results

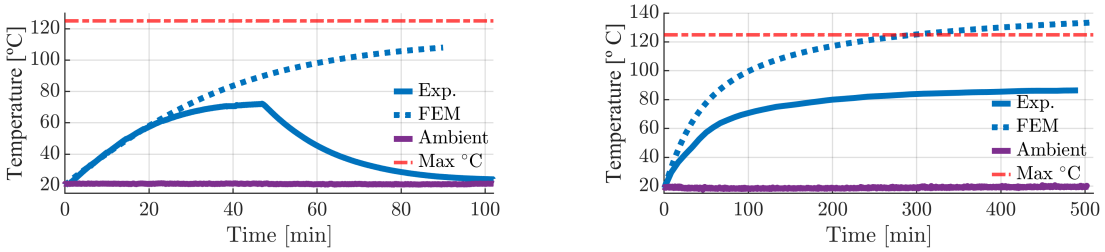


(a) Air test setup

(b) Sand-embedded test setup

Fig. 2.22: Prototype capacitor bank test setup: (a) air test configuration with natural convection in a controlled ambient environment (23°C), and (b) sand-embedded configuration emulating roadway thermal boundary conditions.

Measured temperatures remained below the 125°C thermal limit in both air and sand environments, demonstrating passive cooling feasibility for the distributed low-power-density architecture. FEM predictions showed good agreement with experimental results in air, confirming the natural convection modeling approach. Larger discrepancies in sand testing are attributable to thermal interface uncertainties between the capacitor surfaces and the granular sand medium, an expected challenge for embedded thermal modeling.



(a) Air test: FEM vs. experimental

(b) Sand test: FEM vs. experimental

Fig. 2.23: Prototype capacitor bank experimental validation: (a) air test results comparing FEM predictions with experimental temperatures at 100 kW equivalent power, and (b) sand test results comparing FEM predictions with experimental temperatures.

The comparison between FEM predictions and experimental measurements is shown in Fig. 2.23. These results validate two conclusions: FEM thermal modeling accurately predicts embedded thermal performance when boundary conditions are well characterized, and distributed low-power-density architectures enable passive cooling where compact high-power-density designs require active cooling incompatible with roadway embedding.

The divergence between the FEM prediction and the experimental thermocouple record in Fig. 2.23a reflects a difference in test conditions, not a model error. The ANSYS Icepak simulation assumed continuous full-power operation for the entire recording window. The power resistor and heatsink assembly was powered off at approximately the midpoint of the experimental test; the resulting thermal cool-down is visible in the second half of the air test results. The FEM and experimental temperatures agree during the powered phase; the post-shutdown divergence is an expected consequence of the asymmetric test protocol. Calibrated FEM models can then be adapted to target roadway material properties (concrete, asphalt, or epoxy composites) for predicting deployment performance across different embedding strategies.

2.3.4 FEM Summary and Limitations

The FEM framework represents the highest-fidelity thermal analysis method available for embedded DWPT designs, and the experimental validation summarized above confirms its accuracy when boundary conditions are well characterized. However, each new PCB design requires constructing a new geometric model, generating and converging the mesh, assigning material properties, and configuring boundary conditions—a process that typically requires days to weeks of engineering effort per design. In concurrent multi-PCB development environments—where gate drivers, sensing interfaces, load shedding boards, and compensation networks undergo simultaneous development and revision—repeating this FEM process for each design iteration creates a bottleneck.

Modeling the compensation capacitor bank—a single component type with relatively simple geometry—required detailed geometry construction, mesh convergence studies across multiple mesh densities, and parameterized simulations to explore the thermal design space. Even for this well-defined case, achieving consistent agreement between FEM predictions and experimental measurements demanded careful calibration of thermal interface parameters.

For a complete PCB such as the CREE gate driver board—which contains dozens of components spanning ICs, discrete passives, connectors, and chip-scale packages—the

FEM challenge escalates substantially. The computational mesh required to resolve sub-millimeter chip-scale thermal behavior alongside the full board geometry and surrounding embedding medium becomes impractical at the resolution needed for failure prediction. At coarser mesh resolutions, the thermal behavior of the smallest components is lost—yet these were precisely the components that experienced the original thermal failure. This limitation does not diminish the value of FEM for targeted component-level analysis, but it constrains its applicability as a practical screening tool for full PCB designs. This per-design engineering overhead motivates the search for methods that retain FEM-level accuracy while reducing the setup effort for new designs. The experimental validation framework in Chapter 3 and the machine-learning approach in Chapter 4 address this gap.

2.4 Comparison of Methods and Path Forward

The progression through the methods explored in this chapter—from system-level simulation through component-level analytical modeling to FEM—reveals a recurring trade-off between speed and accuracy for embedded thermal prediction. Each method was pursued as a potential solution to the designer's core challenge: predicting component temperatures in embedded power electronics rapidly enough to inform design decisions during concurrent PCB development.

The PLECS simulation (Section 2.1) addressed the first question in this progression: quantifying actual component-level losses under pulsed DWPT operation. By modeling variable coupling, multi-speed vehicle passes, and transient system behavior, the simulation captured utilization-dependent loss magnitudes that steady-state assumptions miss. These results established that pulsed operation produces loss profiles different from continuous full-load assumptions—a finding that directly affects all downstream thermal analysis. Despite having the capability of producing electrical losses and providing some options to translate those losses into thermal predictions, PLECS requires a separate model to accurately capture the electrothermal dynamics of the devices within larger board- and enclosure-level thermal networks.

The analytical methods (Section 2.2) attempted to provide that thermal translation

through scripted calculations. Inductor core loss estimation through the IGSE pipeline produced validated loss inputs within minutes. Lumped-parameter RC thermal circuits projected losses from previously approximated thermal losses within the inverter PCB using material properties and manufacturer geometry data, following a methodology similar to the analysis of liquid-cooled T-type modules in [37]. For the air-cooled power resistor assembly, this approach captured the general thermal trajectory. However, the sand-embedded case (Section 2.2.2) demonstrated a fundamental limitation: when the embedding medium replaces air as the thermal boundary, uncertain and variable thermal interface properties introduce errors that analytical models do not resolve without empirical calibration. Moreover, each new component or assembly requires its own custom circuit topology and dozens of geometric parameters, making the approach impractical for full PCBs with heterogeneous components.

FEM thermal analysis (Section 2.3) overcame this limitation by directly solving heat conduction equations with realistic material properties and boundary conditions. The 25% scale prototype testing confirmed that FEM predictions agree with measured temperatures in both air and sand environments when boundary conditions are well characterized. However, the per-design engineering overhead detailed in Section 2.3.4 creates a bottleneck incompatible with concurrent multi-PCB development cycles.

The original gate driver failure brings the PCB-level challenge into focus. The thermal runaway occurred on a multi-component PCB where the failure originated in the smallest, most thermally sensitive components—chip-scale gate driver ICs and small ceramic capacitors. These components operate at the tightest thermal margins and are the hardest to model in FEM due to their sub-millimeter feature sizes. The failure was not in a large, well-characterized component like a compensation capacitor or power inductor; it was in the components that are easiest to overlook and hardest to simulate at adequate resolution. The component density and geometric complexity of full PCBs push FEM beyond practical limits for per-revision screening, while analytical models lack the spatial resolution needed for multi-component boards where thermal coupling between adjacent components through

shared copper planes governs the thermal response.

PCBs occupy a unique position in this design space. They are straightforward to prototype—fabrication services offer turnaround times of days to weeks at costs ranging from tens to hundreds of dollars per board—making physical testing feasible. However, they are exceptionally difficult to simulate at the component level due to the number and diversity of components, the multi-layer substrate structure, and the complex inter-component thermal coupling. In an iterative research program with evolving design requirements, where several PCB designs undergo concurrent revision, the designer needs a method that is rapid enough for design iteration, accurate enough for thermal screening at the component level, and applicable across different PCB designs without requiring per-design FEM setup.

The absence of a method satisfying all three requirements defines the gap addressed by the remaining chapters. Chapter 3 develops an automated measurement framework that provides the ground-truth thermal data needed to bridge this gap—establishing the "measure rapidly" side of the solution. Chapter 4 develops a machine-learning model trained on that measured data to provide rapid cross-design thermal prediction—addressing the "predict rapidly" side by aiming to combine the speed of analytical methods with accuracy approaching that of calibrated FEM.

CHAPTER 3

VALIDATION FRAMEWORK FOR EMBEDDED PCBs

Chapter 2 traced a progression from system-level simulation through analytical modeling to FEM, concluding that existing methods do not provide adequate accuracy and speed for thermal prediction of embedded PCBs within the prototyping and iterative design process. Bridging this gap requires empirical thermal data collected under representative boundary conditions. This chapter develops the experimental framework that provides that data: an automated region-of-interest (ROI) generation pipeline from PCB design files for FLIR thermal imaging, and a systematic dual-environment (air and sand) measurement methodology using thermocouple and FLIR instrumentation. Sand testing serves as a practical surrogate for full roadway embedding, capturing conductive thermal boundary conditions at reduced prototyping effort [13]. The framework is designed for reproducibility across multiple PCB designs and generates the training datasets used by the machine-learning model in Chapter 4. DUT-Test serves two evaluation roles: it is held comple tely out of training for unseen-board cross-design generalization testing, and it is separately included in the training set to confirm that the model performs well when all three boards contribute data.

Three PCB DUTs are used throughout this chapter and the subsequent ML development:

- **DUT-1**: H-Bridge Sensing and Gate Driver Interface (227.381 mm × 164.937 mm)—the primary training board, yielding 211 top-layer ROIs.
- **DUT-2**: Load Shedding AC Switch (167.0 mm × 78.4 mm)—a second training board with 60 top-layer ROIs.
- **DUT-Test**: Thermal Gate Driver (145 mm × 76 mm)—a held-out board used to evaluate whether the trained model generalizes to PCBs not seen during training.

DUT-1 and DUT-2 provide training and validation data, while DUT-Test assesses cross-design generalization.

3.1 SROI File Generation Method

This work contributes an automated pipeline for generating FLIR Selective Region of Interest (SROI) files directly from Altium PCB design data. The SROI file maps each component location from the PCB layout to pixel coordinates in the FLIR thermal image, enabling automated extraction of component-level temperature data from thermal image sequences. Without this pipeline, ROIs must be placed manually within ResearchIR software for each component on each board—a tedious, error-prone process that does not scale to boards with hundreds of components. The automated pipeline eliminates this bottleneck and enables reproducible multi-board testing. The pipeline consists of three stages: PCB data processing, coordinate transformation with board-specific calibration, and binary file generation with ROI pixel map export for the ML pipeline. These stages correspond to Sections 3.1.1, 3.1.2, and 3.1.3, respectively. The overall pipeline architecture is shown in Fig. 3.1.

3.1.1 Altium PCB Data Processing

The pipeline begins with Altium Pick&Place CSV export files containing component designators, X/Y coordinates in millimeters, layer assignment, and footprint information. These files are generated through Altium's standard fabrication output and contain the physical placement coordinates for every component on the board. The enhanced parser processes these files with automatic TopLayer filtering to remove bottom-side components not visible to the overhead FLIR camera. The parser identifies the data header row by searching for fields including *Designator*, *Layer*, and *Center-X*, then extracts component coordinates and layer assignments using standard CSV parsing. TopLayer filtering removes solder-side passives and ICs that cannot be thermally imaged from above; the fraction of bottom-layer components varies by board design.

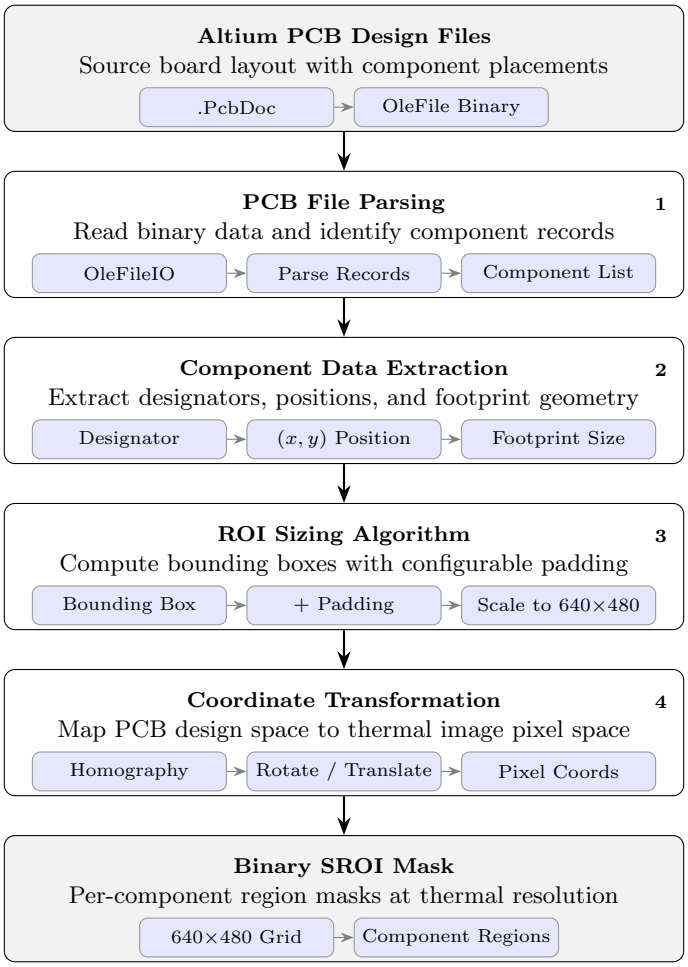


Fig. 3.1: Binary SROI mask generation by including the fabrication output Pick&Place CSV file from Altium PCBs as input, and producing a binary SROI file for FLIR ResearchIR 4.0 as well as other relevant outputs for the ML model training pipeline.

For PCB designs with significant power dissipation on bottom-layer components, restricting FLIR imaging to the top surface is a limitation of the current framework. Practical mitigation strategies include: two-pass imaging with the board physically flipped between tests to capture both surfaces separately, supplemental thermocouple instrumentation on high-power bottom-layer components with those measurements added to the thermal state vector, or encoding known bottom-layer loss estimates as additional model inputs. For the three DUTs evaluated in this work—DUT-1, DUT-2, and DUT-Test—power dissipation is dominated by top-layer components, so top-surface-only imaging captures the dominant thermal behavior; the extent of this assumption should be verified for any new board design

before applying the pipeline.

Components are classified by designator prefix into initial ROI type categories. Small passive components (C*, R*) are assigned single-pixel ROIs, test points (TP*) also receive single-pixel ROIs, and larger components (U*, J*, VR*, DL*, PS*, CR*, L*, F*, H*, SW*) receive 3×3 pixel area ROIs. During SROI generation in the second pipeline stage, these assignments are re-evaluated: the converter examines each component's designator prefix and assigned dimensions to determine the final ROI type—either a 1×1 single-pixel point measurement or a 3×3 pixel area—ensuring consistent sizing regardless of the initial classification. The system maintains a centralized PCB configuration database supporting the three DUTs defined in the chapter introduction: DUT-1, DUT-2, and DUT-Test. Each configuration entry contains board-specific parameters: physical dimensions in millimeters, default corner pixel coordinates for the FLIR field of view, and filename patterns for automatic PCB identification. This multi-PCB support enables seamless switching between DUTs—the pipeline auto-detects the target board from the input filename and loads the appropriate configuration without manual intervention. Additionally, the pipeline loads board-specific transformation parameters from a centralized JSON configuration file shared with the thermal post-processing system. This configuration includes optional axis mirroring flags that account for physical board mounting orientation during thermal imaging—for example, when a PCB is mounted upside-down or mirrored relative to the Altium coordinate system. These transformations are applied to the PCB coordinates before the pixel-space conversion described in Section 3.1.2.

3.1.2 Coordinate Transformation and Corner Calibration

Transforming PCB coordinates (mm) to thermal camera pixel coordinates requires calibration against the physical board position in the FLIR field of view. The coordinate transformation proceeds through four steps: board-specific mirroring, scaling from millimeters to pixels, Y-axis inversion for image coordinate conversion, and translation to the board's position in the image. The corner-based calibration system uses two reference points—the bottom-left and top-right corners of the PCB visible in the thermal image—to establish the

transformation between physical and pixel coordinate spaces. From these reference points, the pixel-space dimensions are computed as $\Delta x_{px} = x_{TR} - x_{BL}$ and $\Delta y_{px} = y_{BL} - y_{TR}$, noting the inverted Y-axis in image coordinates. Independent scaling factors are then calculated as $s_x = \Delta x_{px}/W_{mm}$ and $s_y = \Delta y_{px}/H_{mm}$, where W_{mm} and H_{mm} are the board width and height from the configuration database. Because the FLIR camera may not image the board at its native aspect ratio, different X and Y scaling factors can arise. To maintain geometric fidelity, the pipeline uses uniform scaling: $s = \min(s_x, s_y)$, applying the same scale factor in both axes to preserve the PCB aspect ratio and prevent spatial distortion of ROI positions.

Before scaling, board-specific mirroring is applied when the physical mounting orientation differs from the Altium coordinate system. If the configuration specifies horizontal mirroring, the X coordinate is reflected as $x'_{mm} = W_{mm} - x_{mm}$; similarly, vertical mirroring reflects the Y coordinate as $y'_{mm} = H_{mm} - y_{mm}$. These transformations ensure correct ROI placement regardless of how the board was oriented during thermal imaging. With mirroring and scaling defined, the complete transformation maps each Altium component coordinate (x_{mm}, y_{mm}) to a FLIR pixel location through four sequential steps:

1. Apply board-specific mirroring (if configured)
2. Scale to pixels: $x_{sc} = x'_{mm} \cdot s$, $y_{sc} = y'_{mm} \cdot s$
3. Invert Y-axis: $y_{inv} = \Delta y_{px} - y_{sc}$
4. Translate to board position: $x_{px} = x_{BL} + x_{sc}$, $y_{px} = y_{TR} + y_{inv}$

where (x_{BL}, y_{BL}) and (x_{TR}, y_{TR}) are the bottom-left and top-right corner pixel coordinates respectively.

Table 3.1 summarizes the per-board coordinate transform results, including scaling factors, ROI counts, and SROI binary file sizes. DUT-Test uses axis mirroring to compensate for its physical mounting orientation during imaging. Corner recalibration is performed whenever the camera or board position changes, ensuring sub-pixel accuracy for component temperature extraction. An interactive mode guides the user through corner selection with

visual feedback and calibration guidance, while an automated mode uses pre-configured corner coordinates from the JSON configuration for batch processing of multiple test sessions. The interactive mode is recommended for initial calibration of new board configurations, while the automated mode enables repeated measurements without user interaction.

3.1.3 SROI and ROI Map Generation

The final stage generates two outputs: a binary SROI file for FLIR ResearchIR and an ROI pixel map CSV for the ML training pipeline. The binary SROI format encodes a file header containing metadata (version, ROI count, image dimensions), followed by individual ROI entries. Each ROI entry contains the region's pixel coordinates (top-left corner), dimensions (width and height in pixels), a label string (the component designator from the Pick&Place file), and display attributes (color, line width) for visualization within ResearchIR. All ROIs in the final SROI file use cursor-type definitions: 1×1 single-pixel point measurements for small passives and 3×3 pixel area measurements for all other components including semiconductors, inductors, and connectors. The Cursor ROI type is required by the ResearchIR binary format for component-level point and area measurements that extract per-frame temperature statistics. When the SROI file is loaded into ResearchIR alongside a thermal recording, each ROI automatically extracts per-component temperature statistics (minimum, maximum, mean, standard deviation) from every frame in the thermal sequence. This eliminates manual data extraction and ensures consistent measurement regions across multi-hour test sessions.

In addition to the SROI binary, the pipeline exports an ROI pixel map CSV that lists every pixel coordinate belonging to each ROI. Each row in the pixel map contains the component designator, the pixel X and Y coordinates, and the ROI type. This file provides the spatial mapping between component names and their pixel locations in the 640×480 thermal frame, serving as the spatial index for the ML training pipeline described in Chapter 4. The pixel map enables the FiLM-conditioned U-Net to extract per-component temperature values directly from raw thermal image arrays without requiring the ResearchIR software interface, decoupling the ML data pipeline from proprietary FLIR tools.

The pipeline includes automated validation at multiple stages. Coordinate validation rejects ROIs whose transformed pixel coordinates fall outside the 640×480 thermal image frame, and component count verification confirms alignment with the filtered Pick&Place data. Binary file integrity is validated through hexadecimal analysis to confirm correct encoding of all ROI entries. An optional visual overlay can be generated for human verification of ROI placement against a reference thermal image, superimposing the ROI positions on a captured FLIR frame to confirm spatial accuracy before committing to a full test session. However, this step can also be performed within ResearchIR after loading the SROI file and thermal recording. Fig. 3.2 presents the DUT-2 layout, 3D rendering, and FLIR ROI overlay as a representative example, verifying that pipeline-generated SROI positions align with actual component locations in the thermal image. Output SROI files are auto-named based on PCB type and configuration parameters for reproducible multi-session testing. The corresponding ROI pixel maps contain the expanded per-pixel coordinate data for each ROI, enabling direct array indexing into thermal image frames. In Table 3.1, the SROI size in bytes reflects the total file size of the generated binary SROI file, which scales with the number of ROIs and their dimensions.

Table 3.1: SROI generation summary for each DUT.

	DUT-1	**DUT-2**	**DUT-Test**
PCB Size (mm)	227 × 165	167 × 78	145 × 76
Scale (px/mm)	1.52	2.38	4.07
Valid ROIs	211	60	208
SROI Size (B)	75 842	21 086	73 034

3.2 Thermal Testing Methodology

Reliable training data requires consistent instrument configuration, calibration-artifact removal, and a standardized packaging format across all test sessions. Section 3.2.1 describes the FLIR camera configuration and 15-second recording protocol used across all DUTs. Section 3.2.2 covers thermocouple placement and synchronization with the FLIR

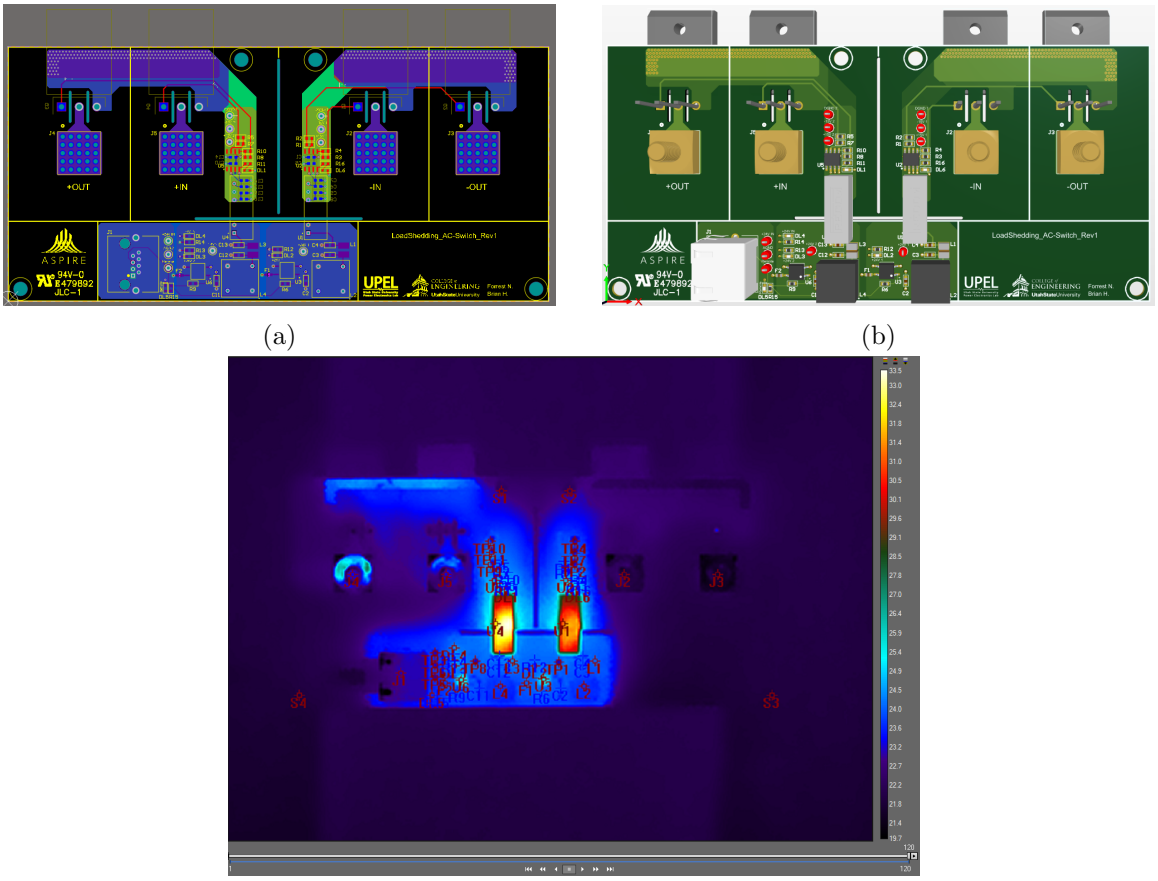


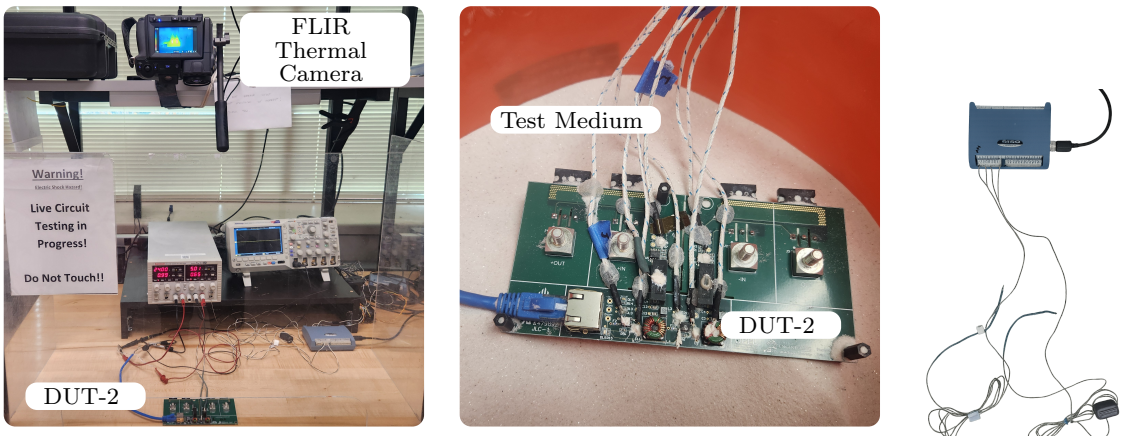
Fig. 3.2: DUT-2 (LoadShedding AC switch): (a) 2D PCB layout showing component placement and designators. (b) 3D rendering illustrating the physical arrangement of components. (c) FLIR thermal image with all 60 ROI positions overlaid, confirming spatial registration between Altium coordinates and FLIR pixel space.

acquisition. Section 3.2.3 addresses non-uniformity correction artifact removal and median filtering applied before dataset assembly. Section 3.2.4 describes how filtered frames, thermocouple logs, and ROI masks are packaged into the HDF5 format consumed by the ML training pipeline.

3.2.1 FLIR Camera Configuration and Measurement Protocol

Thermal measurements are captured using a FLIR T5590 thermal camera (640×480 resolution) controlled through ResearchIR software. The camera is positioned directly above the device under test (DUT) with image orientation configured to match the PCB coordi-

nate system through horizontal and/or vertical flipping as required by the board orientation. Required camera configuration includes object parameters—distance to object, ambient humidity, and atmospheric temperature—for accurate radiometric compensation. The color palette is configured consistently across test sessions for repeatable thermal visualization. The measurement protocol uses 15-second sampling intervals for thermal transient characterization, capturing both rapid warm-up dynamics and gradual approach to steady state. Steady-state criterion is defined as component temperature variation less than 0.5°C over a 1-minute interval. Test durations typically span 60–90 minutes for complete thermal transient characterization, depending on the thermal time constants of the DUT and environment. The SROI file generated in Section 3.1 is loaded into ResearchIR prior to recording, enabling automatic per-component temperature extraction throughout the test. Fig. 3.3a shows the annotated FLIR test setup with the camera positioned overhead.



(a) (b) (c)

Fig. 3.3: Thermal testing setup for DUT-2 (Load Shedding board): (a) air cooling environment with FLIR imaging, (b) sand cooling environment with thermocouple instrumentation, and (c) close-up of thermocouple wiring and data acquisition module.

3.2.2 Thermocouple Instrumentation

Thermocouples attached to critical components provide ground-truth temperature measurements that complement the FLIR surface imaging. Unlike FLIR measurements that

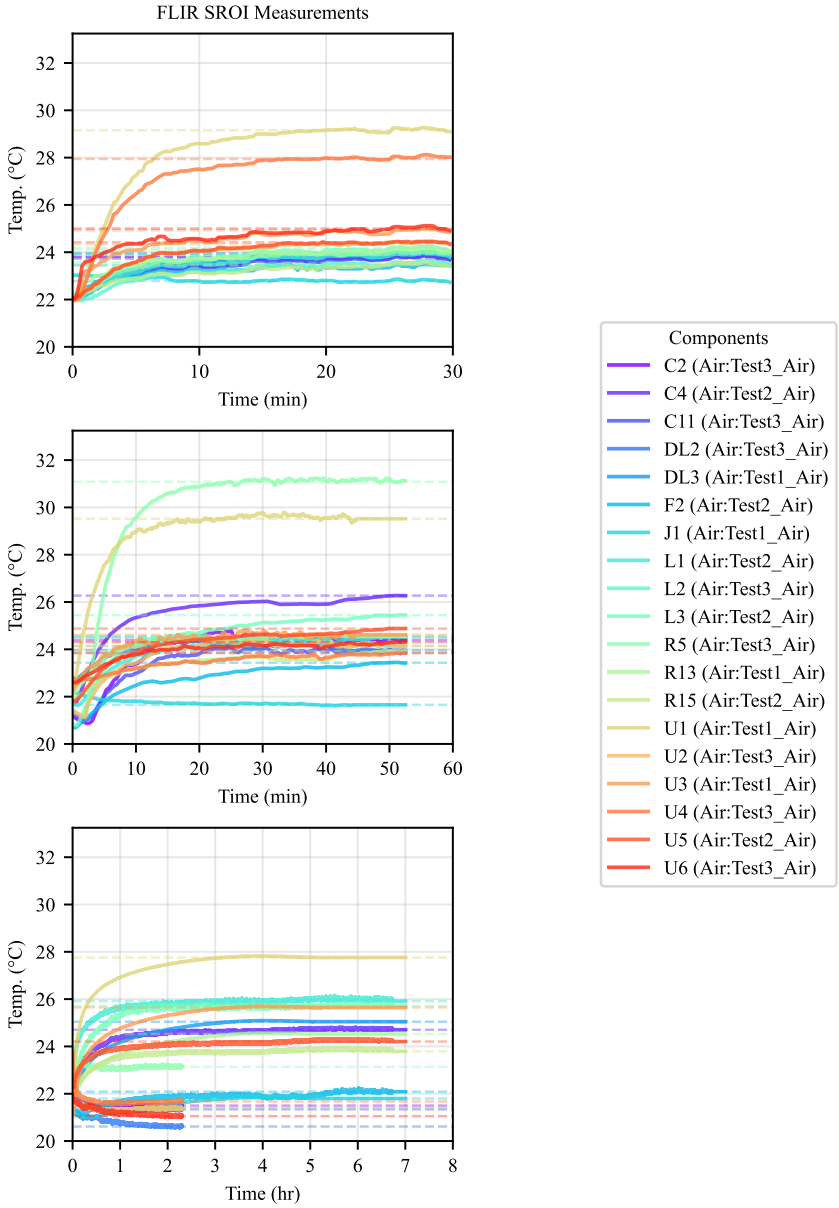
capture surface radiation subject to emissivity variation across different component materials and surface finishes, thermocouples provide direct contact temperature measurement at the component-board interface. This dual-measurement approach enables cross-validation between imaging and contact sensing, and provides the calibration data necessary for the ML model training described in Chapter 4.

The placement strategy prioritizes components with the highest thermal risk and largest expected temperature differential from ambient, as well as baseline low power components to ensure the ML model dataset is varied and representative. Each thermocouple is attached to the component surface with thermal paste for good thermal contact and secured in place with GC Electronics Hi-Volt Silicone Rubber Putty (P/N 10-8880), which provides mechanical stability throughout multi-hour test sessions without affecting the thermal measurement. Data acquisition is performed using a Measurement Computing USB-TEMP module, and the thermocouple data is later synchronized with the FLIR thermal video using timestamp alignment to ensure accurate pairing of temperature readings between air and embedded test conditions.

This synchronization is critical for generating the paired training data (FLIR image $\leftrightarrow$ thermocouple ground truth) used by the FiLM-conditioned U-Net. Figs. 3.3b and 3.3c show the sand embedding setup and data acquisition hardware. Fig. 3.4 compares the steady-state temperatures of all monitored components under air, sand, and potted boundary conditions for DUT-2, illustrating the thermal impact of different cooling environments on component temperatures, as well as the relative temperatures of each component under each condition.

3.2.3 NUC Artifact Correction and Median Filtering

During extended recordings, the FLIR T5590 periodically performs a Non-Uniformity Correction (NUC) event that temporarily interrupts the sensor readout. These NUC events introduce brief temperature artifacts—typically manifesting as isolated spikes or jagged discontinuities in the per-component time series—that do not reflect actual component temperature changes. While the magnitude of these artifacts is generally small relative to



(a)

Fig. 3.4: Three-condition thermal comparison for all monitored components on DUT-2. Each subplot compares steady-state temperatures under air, sand, and potted boundary conditions.

the overall thermal transient, leaving them uncorrected can degrade ML model training by introducing non-physical discontinuities that the network may attempt to learn as legitimate thermal behavior. The filtering stage therefore serves as a preprocessing step to ensure

smooth, physically consistent thermal trajectories prior to dataset generation.

Five filtering algorithms were evaluated to address NUC artifacts and general measurement noise. Savitzky-Golay polynomial smoothing [45] (window length 11, third-order polynomial) preserves peak amplitudes while removing high-frequency noise. A hybrid method combines median absolute deviation (MAD) outlier detection with Savitzky-Golay smoothing, achieving the highest overall artifact removal at the cost of minor peak attenuation. A fourth-order Butterworth low-pass filter [46] extracts smooth thermal trends but attenuates transient features. Statistical outlier removal using Z-score thresholding (default threshold 3.0) replaces individual data points exceeding three standard deviations from the local mean. Median filtering (kernel size 5) targets isolated refocus spikes without affecting the underlying thermal trend and is used as the primary correction method for all datasets in this work. The temporal median filter operates per-pixel across the time dimension: for each pixel in the 480×640 frame, a sliding window of 5 consecutive frames replaces the center value with the local median. This suppresses single-frame outliers while preserving the true thermal trajectory.

Table 3.2 summarizes the per-board NUC correction magnitude measured across all monitored components. The correction magnitudes vary across boards due to differences in thermal operating range and component density. DUT-1 shows the smallest corrections (max 0.30°C, median 0.17°C across 22 components), indicating that NUC artifacts on this board are minor perturbations relative to its 22–61°C operating range. DUT-2 shows moderate corrections (max 1.63°C for connector J1, median 0.22°C across 19 components), with connector and IC components exhibiting larger artifacts than passive devices due to their greater thermal mass and surface area within the ROI. DUT-Test shows the largest corrections (max 4.44°C for U5B, median 1.32°C across 8 components), reflecting the higher power dissipation and steeper thermal gradients on this board that amplify the NUC-induced measurement perturbation. Despite these variations, the median correction remains below 1.5°C for all boards, confirming that the filtering stage is a refinement step rather than a correction for large-scale measurement errors. The filtered data are used for all

subsequent ML training to ensure that the FiLM-conditioned U-Net learns from smooth, physically representative thermal trajectories.

Table 3.2: Per-board NUC correction magnitude across all monitored components. Max correction is the largest single-frame ROI-mean temperature difference between raw and median-filtered data for any component on the board.

	DUT-1	**DUT-2**	**DUT-Test**
Monitored components	22	19	8
Max correction (°C)	0.30	1.63	4.44
Mean max correction (°C)	0.18	0.38	1.60
Median max correction (°C)	0.17	0.22	1.32
Correction range (°C)	0.10–0.30	0.13–1.63	0.62–4.44

Fig. 3.5 shows the NUC correction process for DUT-2. The raw FLIR frame containing the NUC artifact is shown alongside the computed correction map and the filtered result. The correction map reveals that the NUC artifacting requires an temporal based created offset per pixel within the frame; the median filter removes this offset while preserving the spatial temperature distribution and inter-component thermal gradients across the PCB.

3.2.4 HDF5 Training Dataset Generation

The final stage of the post-processing pipeline bridges the thermal measurement infrastructure to the machine learning model presented in Chapter 4. The filtered FLIR thermal image sequences, thermocouple ground-truth temperatures, and ROI pixel maps from the preceding sections are packaged into HDF5 training datasets. Each HDF5 file contains the complete temporal record for one board and one test condition: the raw FLIR frames (480×640, float32), aligned thermocouple temperatures for all monitored components, timestamps, frame-to-timestamp index mapping, ROI binary masks, and board metadata (PCB corners, component names, board identifier).

This self-contained format allows the CNN training pipeline to reconstruct all derived input channels—spatial features, temporal derivatives, and FiLM conditioning scalars—directly from the archived data, as described in Chapter 4. Separate HDF5 files are gener-

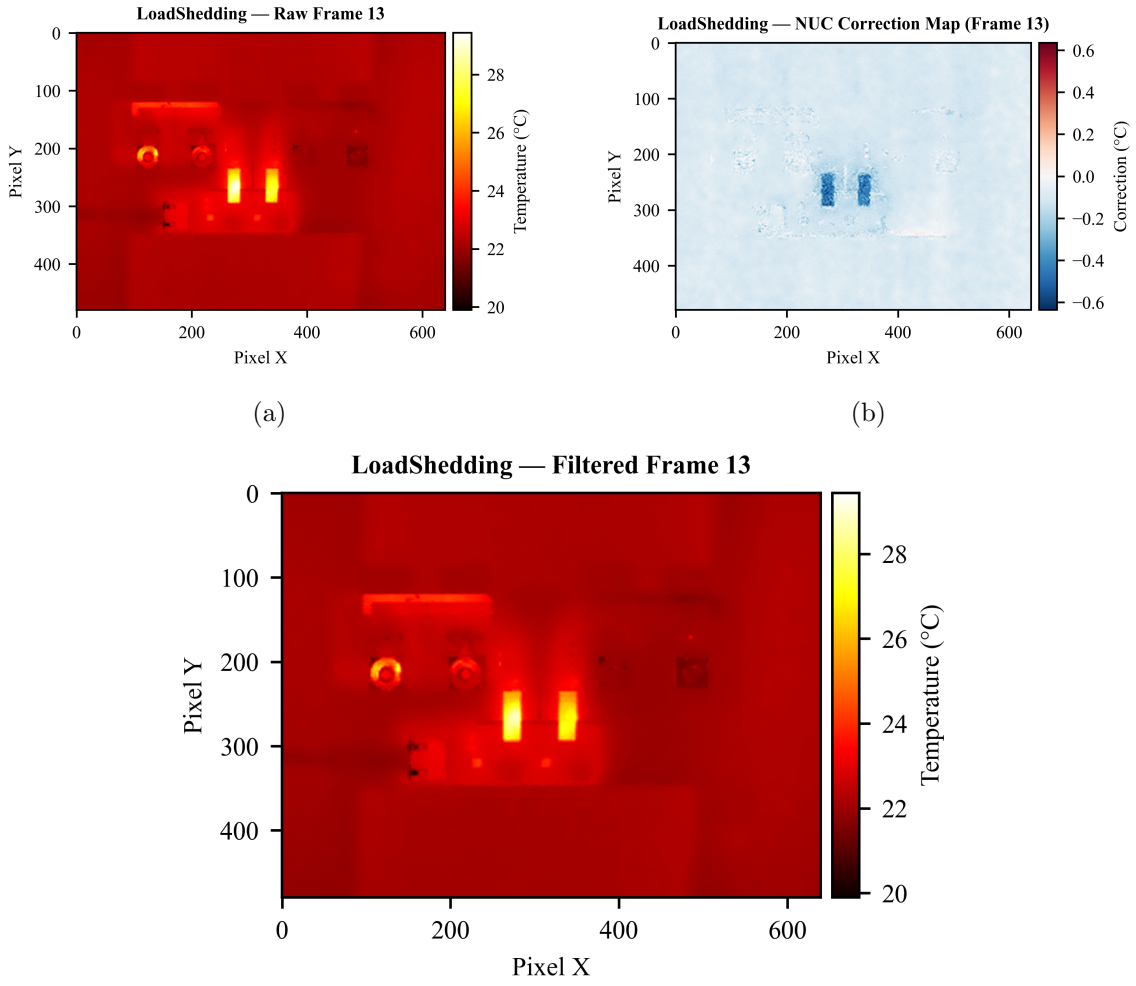


Fig. 3.5: DUT-2 (LoadShedding AC switch) NUC correction: (a) raw FLIR thermal image showing NUC refocusing artifact, (b) computed correction map, and (c) filtered result after median correction.

ated for each board and boundary condition (air, sand), enabling flexible dataset composition during model training. A table showing the summary of the captured data for each board and test condition is shown in Table 3.2.4.

headerbg	**DUT-1**	**DUT-2**	**DUT-Test**	**Total**
PCB size (mm)	227 × 165	167 × 78	145 × 76	—
FLIR ROIs	211	60	208	**479**
TC components	22	19	8	**49**
Frames / session	300	120	300	**720**

3.3 Scope of the Thermal Verification Framework

The framework presented in this chapter converts raw PCB design files and thermal recordings into structured, ML-ready datasets. The SROI pipeline automates ROI generation from Altium exports, the FLIR and thermocouple measurement protocol captures transient thermal behavior in both air and sand environments, and the post-processing stages produce filtered, time-aligned HDF5 files for each board and test condition. Together, these stages provide the paired ground-truth data—FLIR thermal images with per-component thermocouple temperatures—that the FiLM-conditioned U-Net in Chapter 4 requires for training and evaluation.

CHAPTER 4

FILM-CONDITIONED U-NET FOR COMPONENT-LEVEL THERMAL PREDICTION

This chapter presents a machine learning approach to component-level thermal prediction for power electronics PCBs, building on the experimental instrumentation and data pipeline described in Chapter 3. The ML architectures and conditioning techniques reviewed in Section 1.2.7 provide the background for the model design presented here. The thermal measurement framework from Chapter 3 produces FLIR thermal images paired with thermocouple ground-truth temperatures across multiple PCB designs and operating conditions. This dataset enables a data-driven approach. Rather than constructing a per-design FEM model or relying on analytical estimates with uncertain boundary conditions, the model learns the mapping between thermal image features and component-level temperatures directly from measured data. The model is a FiLM-conditioned U-Net that predicts component temperatures from FLIR thermal images, combining a pixel-level encoder-decoder architecture with Feature-wise Linear Modulation (FiLM) conditioning to adapt to different boards and temporal phases without design-specific retraining. This approach provides fast thermal assessment without the computational cost of FEM simulation or the hardware cost of dense thermocouple instrumentation.

Rather than predicting absolute temperatures, the model learns FLIR-to-thermocouple residuals—the offset between surface-measured FLIR values and contact-measured thermocouple readings—which decouples the prediction task from the absolute temperature scale and improves generalization across boards with different thermal operating points. Cross-design validation on two distinct PCB topologies evaluates whether the model learns transferable thermal coupling patterns, a requirement for practical deployment where new board designs must be assessed without design-specific retraining. The data pipeline and model architecture are designed so that adding new DUTs to the training set requires only the FLIR thermal imaging and thermocouple instrumentation described in Chapter 3; no ar-

chitectural changes or retraining from scratch are needed, making the framework extensible as additional PCB designs become available.

Existing ML-based thermal prediction methods for electronics fall into several categories, but none directly address the specific challenge targeted here. Physics-informed neural networks (PINNs) and neural-operator methods [26,27] learn solutions to parametric PDEs and can generalize across geometries, but require access to the governing equations and mesh representations that are unavailable for arbitrary PCB designs encountered in rapid prototyping. Data-driven surrogate models have been applied to data-center thermal management [47] and benchmarked for multiscale IC thermal simulation [31], but these approaches assume fixed facility layouts or chip floorplans and do not transfer to the irregular component placements and variable board sizes.

The approach presented in this chapter differs in three respects. First, the FiLM conditioning mechanism [30] decouples the spatial encoder from the domain identity, allowing a single model to process multiple PCB designs by modulating decoder features based on board-level thermal statistics rather than learning separate spatial representations per board. Second, the residual learning formulation (predicting the FLIR-to-thermocouple offset rather than absolute temperature) enables training across boards with different absolute temperature scales without normalization artifacts. Third, the automated SROI pipeline from Chapter 3 provides a standardized input format—10-channel image tensors and 6-element thermal state vectors—that is constructed identically for any PCB, making the model input-agnostic to board geometry and component count. Together, these design choices enable the cross-design and unseen-board generalization results presented in Section 4.5.

4.1 Model Architecture

The model is a U-Net [25] with four encoder blocks, a bottleneck, and four FiLM-conditioned decoder blocks totaling 7,826,849 trainable parameters. Fig. 4.1 shows the overall architecture. The model accepts two inputs: a 10-channel image tensor $\mathbf{x} \in \mathbb{R}^{480\times640\times10}$ and a thermal state vector $\mathbf{s} \in \mathbb{R}^6$. A more detailed view of the encoder-decoder structure

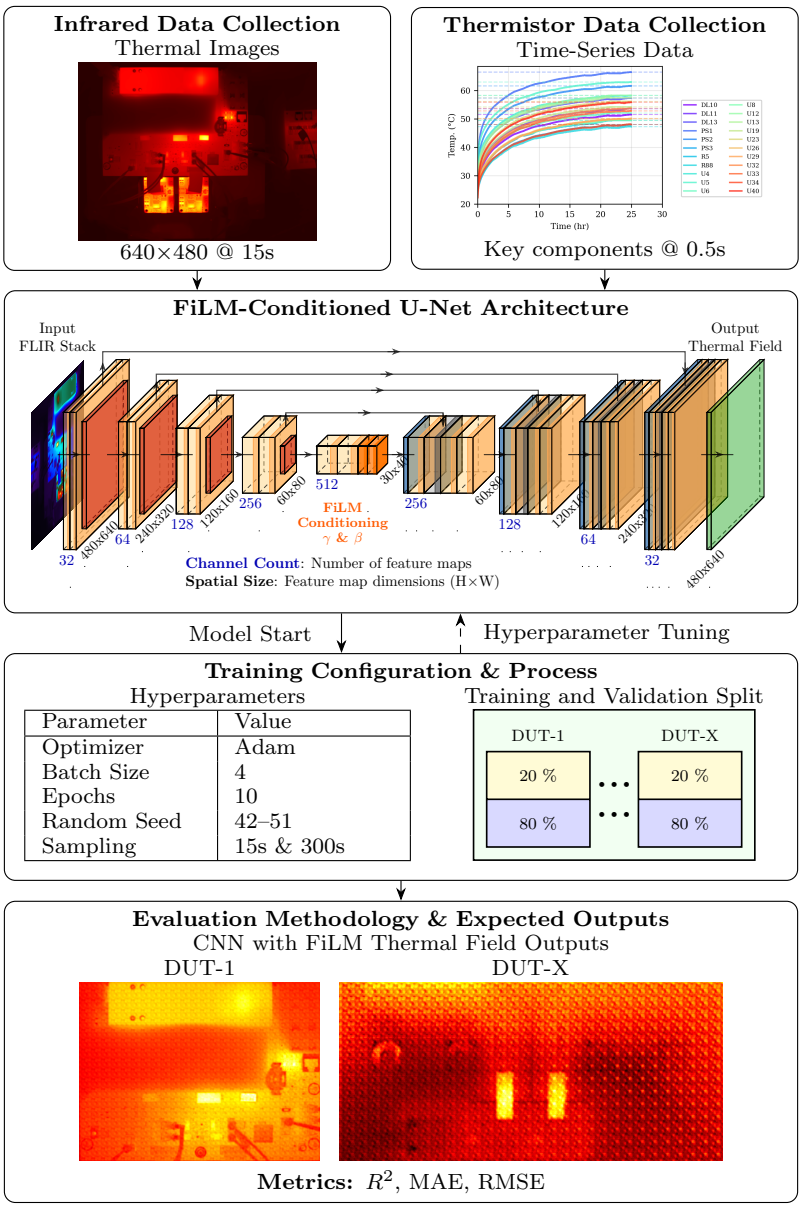


Fig. 4.1: FiLM-conditioned U-Net architecture for thermal prediction. The encoder extracts multi-scale spatial features from the 10-channel input. A shared FiLM embedding maps the 6-element thermal state vector to affine parameters applied at each of the four decoder levels, enabling temporal and domain-aware modulation throughout the reconstruction path.

with FiLM conditioning connections is shown in Fig. 4.2, illustrating the channel dimensions at each encoder and decoder stage, the skip connections that preserve spatial detail, and the FiLM embedding pathway that generates per-level affine parameters from the thermal state vector.

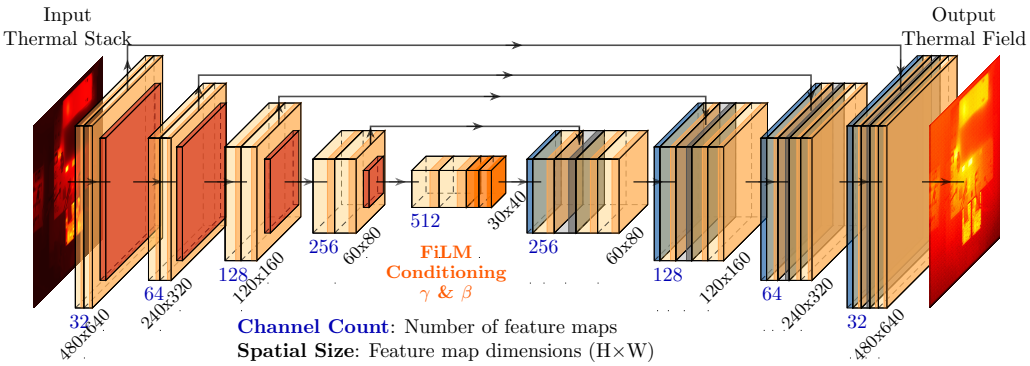


Fig. 4.2: Detailed architecture of the U-Net with FiLM conditioning, showing the flow of data through the encoder, bottleneck, and decoder stages.

4.1.1 Input Construction

Each training sample produces the image tensor $\mathbf{x}$ and thermal state vector $\mathbf{s}$ from a raw FLIR frame $F \in \mathbb{R}^{480\times640}$ (degrees Celsius) and its associated timestamp τ. The 10 channels encode complementary spatial, temporal, and domain information organized into three functional groups. Spatial features (Channels 0–4) characterize the thermal field structure within a single frame, temporal features (Channels 5–7) encode transient dynamics across consecutive frames, and domain descriptors (Channels 8–9) identify the board's thermal operating range. Table 4.1 provides a summary of all channels, their formulations, and their role in cross-board generalization.

The feature design involves two distinct aspects. The 10-channel input representation was specified by the researcher based on domain knowledge: spatial channels encode thermal field structure within a single frame, temporal channels encode transient dynamics, and domain descriptor channels identify the board's operating range. This choice of what to measure is a design decision informed by the physics of the problem. The relative importance of these channels—which ones the model actually learns to rely on—is not prescribed; it is determined entirely by the training process. The permutation feature importance analysis in Section 4.6 quantifies this learned reliance. The temporal conditioning pathway (Channel 7 and the FiLM conditioning network) dominates in the current training configuration; spatial feature channels contribute marginally with the present two-board dataset.

Table 4.1: Summary of the 10-Channel Input Tensor and 6-Element Thermal State Vector.

Ch.	Name	Formulation	Cross-Board Role
Spatial Features (per-pixel, computed from single frame F)			
0	Z-score temperature	$(F_p - \mu_F)/(6\sigma_F) + 0.5$	Board-invariant: removes absolute offset
1	Gradient magnitude	$\|\nabla F\|$ via Sobel, normalized	Board-invariant: edge structure only
2	Local std. dev.	$\sigma_{5\times5}(F_p)$, normalized	Board-invariant: local heterogeneity
3	Laplacian	$\nabla^2 F_p$, normalized to $[0,1]$	Board-invariant: heat source/sink map
4	PCB binary mask	$\{0,1\}$ from SROI corners	Board-specific: defines board region
Temporal Features (per-pixel or broadcast, require $\geq$3 consecutive frames)			
5	$\dot{T}$ (1st derivative)	$(F_p^{(t)} - F_p^{(t-1)})/\Delta\tau$	Board-agnostic: rate of change
6	$\ddot{T}$ (2nd derivative)	$(F_p^{(t)} - 2F_p^{(t-1)} + F_p^{(t-2)})/\Delta\tau^2$	Board-agnostic: transient acceleration
7	Log-normalized time	$\log_{10}(\tau{+}1)/\log_{10}(\tau_{\max}{+}1)$	Board-agnostic: temporal position
Domain Descriptors (spatially broadcast scalars, per-frame statistics)			
8	Thermal amplitude	$\sigma(z\text{-score of } \sigma_F)$	Board-specific: thermal spread identity
9	Thermal level	$\sigma(z\text{-score of } \mu_F)$	Board-specific: thermal baseline identity
Thermal State Vector $\mathbf{s} \in \mathbb{R}^6$ *(FiLM conditioning input)*			
s_1	Log-normalized time	Same as Ch. 7	Temporal context for FiLM
s_2	Thermal level	Same as Ch. 9	Domain context for FiLM
s_3	Thermal amplitude	Same as Ch. 8	Domain context for FiLM
s_4	FLIR range	$\sigma(z\text{-score of } (F_{\max} - F_{\min})_{\text{PCB}})$	Thermal spread within board mask
s_5	Mean $\dot{T}$	$\sigma(z\text{-score of mean } \dot{T}_{\text{PCB}})$	Global heating rate summary
s_6	Gradient energy	$\sigma(z\text{-score of } \overline{\|\nabla F\|^2}_{\text{PCB}})$	Spatial complexity of thermal field

The cross-board column in Table 4.1 reflects a deliberate design choice that enables the model to generalize across PCB designs without retraining. Channels 0–3 are normalized relative to each frame's own statistics (μ_F, σ_F) or use derivative operators that respond to spatial structure rather than absolute temperature, making these features inherently board-invariant: a 10°C hot spot on DUT-1 produces the same z-score pattern as a 10°C hot spot on DUT-2. Channel 4 (the PCB mask) is board-specific by necessity—each board occupies a different region of the FLIR frame—but serves only to mask the background, not to encode board identity. Channels 5–6 capture temporal dynamics that are board-agnostic: the rate at which a component heats depends on its thermal time constant, not on which board it belongs to. Channels 7–9 and the thermal state vector $\mathbf{s} \in \mathbb{R}^6$ are the primary pathways through which board-specific context enters the model, and these are processed exclusively by the FiLM conditioning network rather than the convolutional encoder. This separation ensures that when additional boards are added to the training set, the spatial

encoder requires no modification—only the FiLM pathway sees new domain points, and domain augmentation (Section 4.3.5) pre-conditions it to handle a continuous range of board operating conditions. The channel construction is therefore not limited to the two boards used in this work; any PCB tested with the SROI pipeline from Chapter 3 produces the same 10-channel representation, making the input format inherently extensible. The remainder of this section details the formulation, normalization, and physical interpretation of each channel.

Channel 0: Z-Score Normalized Temperature

Per-frame statistics μ_F and σ_F are computed from all pixels. The z-score normalization [48] maps approximately 99.7% of pixel values into $[0, 1]$—a direct consequence of the 6σ divisor, which spans ± 3 standard deviations of the Gaussian distribution:

$$x_p^{(0)} = \frac{F_p - \mu_F}{6\,\sigma_F} + 0.5 \tag{4.1}$$

Here $x_p^{(0)}$ is the normalized Channel 0 value at pixel p, F_p is the raw FLIR temperature at that pixel in degrees Celsius, μ_F is the mean temperature across all pixels in the frame, and σ_F is the corresponding standard deviation. This removes the board-level absolute temperature offset so the encoder processes only the spatial *pattern* of heating, not which board produced the frame. Board identity is restored through Channels 8–9 and the FiLM conditioning vector.

Channel 1: Gradient Magnitude (Sobel)

Spatial gradients via Sobel operators [49] highlight thermal boundaries between components and the background:

$$x_p^{(1)} = \frac{\sqrt{(F * S_x)_p^2 + (F * S_y)_p^2}}{\max_q \sqrt{(F * S_x)_q^2 + (F * S_y)_q^2}} \tag{4.2}$$

Here $x_p^{(1)}$ is the normalized gradient magnitude at pixel p, F is the full-frame FLIR thermal

image, and S_x, S_y are the standard 3×3 Sobel kernels that approximate horizontal and vertical partial derivatives, with $*$ denoting 2D convolution. The denominator normalizes by the maximum gradient magnitude across all pixels q in the frame, mapping the channel to $[0, 1]$. The gradient magnitude provides a domain-invariant structural signal that marks component edges regardless of absolute temperature.

Channel 2: Local Standard Deviation

A 5×5 sliding window computes the local temperature standard deviation at each pixel, normalized to unit range. High local standard deviation indicates thermal heterogeneity near component boundaries; low values indicate thermally uniform regions [49]. This channel enables the encoder to distinguish component hot spots from uniform background areas without relying on absolute temperature thresholds.

Channel 3: Laplacian (2nd Spatial Derivative)

The discrete Laplacian [49] identifies heat sources and sinks by comparing each pixel's temperature to its four immediate neighbors:

$$\nabla^2 F_p = F_{i-1,j} + F_{i+1,j} + F_{i,j-1} + F_{i,j+1} - 4\,F_{i,j} \tag{4.3}$$

Here $\nabla^2 F_p$ is the discrete Laplacian at pixel p, $F_{i,j}$ is the FLIR temperature at row i and column j, and the four surrounding terms ($F_{i\pm1,j}$, $F_{i,j\pm1}$) are the temperatures of the immediate north, south, east, and west neighbors. Normalized and centered at 0.5, pixels with $\nabla^2 F_p < 0$ (value < 0.5) are warmer than their neighbors and indicate active heat sources, while $\nabla^2 F_p > 0$ (value > 0.5) indicates local heat sinks. When $\nabla^2 F_p = 0$, the steady-state heat equation is satisfied. This channel provides the encoder with a physics-based feature that directly encodes the spatial structure of the thermal field.

Channel 4: PCB Binary Mask

A binary mask derived from known PCB corner coordinates marks the board region

($x_p^{(4)} = 1$) versus background ($x_p^{(4)} = 0$). This channel restricts the model's attention to the relevant board area and prevents background thermal artifacts from influencing component predictions.

Channels 5–6: Temporal Derivatives

Given three consecutive frames $F^{(t)}, F^{(t-1)}, F^{(t-2)}$ at times $\tau_t, \tau_{t-1}, \tau_{t-2}$, standard forward finite-difference approximations [48] capture the rate and acceleration of temperature change:

$$\dot{T}_p = \frac{F_p^{(t)} - F_p^{(t-1)}}{\tau_t - \tau_{t-1}} \tag{4.4}$$

$$\ddot{T}_p = \frac{F_p^{(t)} - 2\,F_p^{(t-1)} + F_p^{(t-2)}}{(\tau_t - \tau_{t-1})^2} \tag{4.5}$$

Here $\dot{T}_p$ is the first temporal derivative (heating rate in °C/s) at pixel p, $\ddot{T}_p$ is the second temporal derivative (heating acceleration), $F_p^{(t)}$ is the FLIR temperature at pixel p in the current frame, $F_p^{(t-1)}$ and $F_p^{(t-2)}$ are the temperatures at the same pixel in the two preceding frames, and τ_t is the timestamp of frame t in seconds. Both channels are normalized and centered at 0.5. Channel 5 ($\dot{T}$) tells the encoder whether each pixel is heating or cooling and at what rate. Channel 6 ($\ddot{T}$) indicates whether the thermal transient is accelerating or decelerating—a positive second derivative during warm-up signals that the system is still far from thermal equilibrium, while a near-zero value indicates approach to steady state. For the first two timesteps, both derivatives default to 0.5 (zero change).

Channel 7: Log-Normalized Time

A spatially broadcast scalar encodes the elapsed time τ (seconds since test start), compressed logarithmically so that the early transient phase and late steady-state phase receive equal representational weight:

$$x_p^{(7)} = t_{\text{norm}} = \frac{\log_{10}(\tau + 1)}{\log_{10}(\tau_{\max} + 1)} \tag{4.6}$$

Here $x_p^{(7)}$ is the Channel 7 value broadcast identically to every pixel p, t_{norm} is the resulting log-compressed time coordinate in $[0, 1]$, τ is the elapsed time in seconds since the start of the test, and $\tau_{\max}$ is the maximum test duration across all training sessions.

Channels 8–9: Board-Level Domain Descriptors

These spatially broadcast channels encode the domain identity through per-frame thermal statistics. Each statistic is first centered by subtracting its training-set mean and dividing by its training-set standard deviation (z-score), then passed through a sigmoid function that sets the result into the bounded range $(0, 1)$. This two-step normalization guarantees that the domain descriptors remain well-behaved even for boards whose temperature statistics fall far outside the training distribution—the sigmoid saturates rather than extrapolating, avoiding the out-of-range values that would occur with simple min-max scaling. Formally:

$$x_p^{(8)} = \text{Ch8} = \sigma\left(\frac{\sigma_F - \bar{\sigma}_F}{\hat{\sigma}_{\sigma_F}}\right) \quad \text{(thermal amplitude)} \tag{4.7}$$

$$x_p^{(9)} = \text{Ch9} = \sigma\left(\frac{\mu_F - \bar{\mu}_F}{\hat{\sigma}_{\mu_F}}\right) \quad \text{(thermal level)} \tag{4.8}$$

Here $x_p^{(8)}$ (thermal amplitude) and $x_p^{(9)}$ (thermal level) are the domain descriptor values broadcast to every pixel p, and σ_F and μ_F are the per-frame standard deviation and mean temperature defined in Section 4.1.1. The notation $\sigma(\cdot)$ denotes the logistic sigmoid [48] $\sigma(z) = 1/(1 + e^{-z})$. The training-set statistics $\bar{\sigma}_F$ and $\hat{\sigma}_{\sigma_F}$ are the mean and standard deviation of per-frame σ_F values computed across all training files; $\bar{\mu}_F$ and $\hat{\sigma}_{\mu_F}$ are the corresponding statistics for per-frame μ_F. The sigmoid guarantees that Ch8 and Ch9 lie strictly in $(0, 1)$ for *any* input temperature—including boards with thermal ranges far outside the training distribution—eliminating the out-of-distribution extrapolation that caused systematic DC offsets with the previous min-max normalization (see Section 4.3.5). The z-score centres the training distribution at $\sigma(0) = 0.5$, preserving dynamic range: HBridge training values span approximately $[0.15, 0.84]$ in Ch9, providing $\sim$0.69 of usable range for

FiLM conditioning.

Thermal State Vector

The 6-element FiLM conditioning input aggregates temporal position, domain identity, and thermal field summary statistics:

$$\mathbf{s} = \begin{pmatrix} t_{\text{norm}} \\ \text{Ch9} \\ \text{Ch8} \\ \sigma(z(F_{\max} - F_{\min})) \\ \sigma(z(\overline{\dot{T}})) \\ \sigma(z(\overline{\|\nabla F\|^2})) \end{pmatrix} \in \mathbb{R}^6 \tag{4.9}$$

This vector enters the model through a separate input and drives FiLM modulation at all four decoder levels. The first three elements duplicate information already present in the spatial channels (Ch. 7–9), providing the FiLM embedding network with a compact summary of *when* the frame was captured and *which thermal operating range* the board occupies. The three additional elements—FLIR temperature range within the PCB mask (s_4), mean first temporal derivative (s_5), and mean squared gradient magnitude (s_6)—provide the FiLM pathway with richer supervision about the board's current thermal state. All six features use the same sigmoid-of-z-score normalization as Channels 8–9, ensuring bounded inputs regardless of the board's absolute temperature range.

Taken together, the 10 spatial channels and 6-element state vector give the model two complementary views of the same thermal scene. Channels 0–6 provide dense per-pixel information—where heat is, how fast it is changing, and what the local spatial structure looks like—while Channels 7–9 and the state vector summarize the frame-level context: when the measurement was taken, how thermally active the board is, and which operating condition produced the frame. Because the per-pixel channels are normalized relative to each frame's own statistics, they transfer across boards without modification. The

frame-level descriptors carry the board-specific context that the FiLM conditioning network (Section 4.1.3) uses to adapt the decoder to each board's thermal characteristics. The following subsections describe how these inputs flow through the encoder, FiLM conditioning pathway, and decoder.

4.1.2 U-Net Encoder and Bottleneck

The encoder compresses the 10-channel input into a compact spatial representation through a series of convolutional blocks. Each of the four encoder blocks applies two 3×3 convolution layers with rectified linear unit (ReLU) activation [48], followed by 2×2 max pooling for spatial downsampling. In the notation used below, Conv_C denotes a 3×3 convolution producing C output feature channels, and $\text{MaxPool}_{2\times2}$ halves each spatial dimension by selecting the maximum value in each 2×2 window. The output feature map of encoder block k after pooling is $\mathbf{h}_k$, and $\mathbf{h}_k^{\text{skip}}$ is the feature map from block k *before* pooling, retained so the decoder can recover fine spatial detail that pooling discards. Each encoder block follows the same pattern:

$$\mathbf{h}_k = \text{MaxPool}_{2\times2}\Big(\text{ReLU}\big(\text{Conv}_{C_k}\big(\text{ReLU}(\text{Conv}_{C_k}(\mathbf{h}_{k-1}))\big)\big)\Big), \quad k = 1, 2, 3, 4 \tag{4.10}$$

where $\mathbf{h}_0 = \mathbf{x}$ is the 10-channel input tensor. Table 4.2 summarizes the spatial resolution and channel count at each encoder stage.

Table 4.2: Encoder block dimensions. Each block doubles the channel count and halves the spatial resolution.

Stage	**Channels (C_k)**	**Output Size**	**Skip Size ($\mathbf{h}_k^{\text{skip}}$)**
Input ($\mathbf{h}_0$)	10	480×640	—
Block 1 ($k = 1$)	32	240×320	$480 \times 640 \times 32$
Block 2 ($k = 2$)	64	120×160	$240 \times 320 \times 64$
Block 3 ($k = 3$)	128	60×80	$120 \times 160 \times 128$
Block 4 ($k = 4$)	256	30×40	$60 \times 80 \times 256$

After the fourth encoder block, a bottleneck applies two additional 3×3 convolutions at 512 channels without further pooling:

$$\mathbf{h}_5 = \mathrm{ReLU}\big(\mathrm{Conv}_{512}\big(\mathrm{ReLU}(\mathrm{Conv}_{512}(\mathbf{h}_4))\big)\big) \in \mathbb{R}^{30\times 40\times 512} \tag{4.11}$$

At this 30×40 resolution, each feature map position has a receptive field covering a large portion of the board, enabling the bottleneck to capture board-wide thermal coupling patterns. The bottleneck output $\mathbf{h}_5$ is then passed to the decoder, where it is progressively upsampled and recombined with the skip connections $\mathbf{h}_1^{\text{skip}}$ through $\mathbf{h}_4^{\text{skip}}$ to recover the original 480×640 spatial resolution.

4.1.3 FiLM Conditioning Mechanism

The encoder processes spatial features from the 10-channel input, but it has no direct access to frame-level context such as elapsed time or the board's thermal operating condition. The Feature-wise Linear Modulation (FiLM) mechanism [30] addresses this by converting the 6-element thermal state vector $\mathbf{s}$ into per-channel scale and shift parameters that modulate the decoder's feature maps at every level.

The first step maps $\mathbf{s} \in \mathbb{R}^6$ to a shared latent embedding $\mathbf{z} \in \mathbb{R}^{64}$ through a two-layer MLP:

$$\mathbf{z}^{(1)} = \mathrm{ReLU}(\mathbf{W}_1\mathbf{s} + \mathbf{b}_1) \tag{4.12}$$

$$\mathbf{z} = \mathrm{ReLU}(\mathbf{W}_2\mathbf{z}^{(1)} + \mathbf{b}_2) \tag{4.13}$$

Here $\mathbf{W}_1 \in \mathbb{R}^{32\times 6}$ and $\mathbf{b}_1 \in \mathbb{R}^{32}$ are the weights and bias of the first fully connected layer, expanding the 6-element input to 32 hidden units, and $\mathbf{W}_2 \in \mathbb{R}^{64\times 32}$ and $\mathbf{b}_2 \in \mathbb{R}^{64}$ are the weights and bias of the second layer, further expanding to 64 units. The rectified linear unit activation $\mathrm{ReLU}(\cdot) = \max(0, \cdot)$ introduces nonlinearity so the embedding can represent complex relationships between the state vector elements. This embedding is computed once per sample and shared across all four decoder levels.

From the shared embedding, separate linear projections produce a scale vector $\boldsymbol{\gamma}_\ell$ and shift vector $\boldsymbol{\beta}_\ell$ for each decoder level $\ell \in \{6, 7, 8, 9\}$:

$$\boldsymbol{\gamma}_\ell = \mathbf{W}_{\gamma_\ell}\,\mathbf{z} + \mathbf{b}_{\gamma_\ell} \in \mathbb{R}^{C_\ell} \tag{4.14}$$

$$\boldsymbol{\beta}_\ell = \mathbf{W}_{\beta_\ell}\,\mathbf{z} + \mathbf{b}_{\beta_\ell} \in \mathbb{R}^{C_\ell} \tag{4.15}$$

where $\mathbf{W}_{\gamma_\ell} \in \mathbb{R}^{C_\ell \times 64}$ projects the 64-dimensional embedding down to C_ℓ channels, and C_ℓ matches the decoder's channel count at level ℓ (256, 128, 64, and 32 for levels 6 through 9, respectively). There is one $(\boldsymbol{\gamma}_\ell, \boldsymbol{\beta}_\ell)$ pair per decoder level, giving the FiLM pathway the flexibility to apply different modulation at different spatial scales.

These parameters are applied to the decoder feature map channel-wise:

$$\hat{\mathbf{h}}_\ell = \boldsymbol{\gamma}_\ell \odot \mathbf{h}_\ell + \boldsymbol{\beta}_\ell \tag{4.16}$$

where $\odot$ denotes element-wise multiplication broadcast over the spatial dimensions (height and width). In plain terms, each channel of the decoder feature map is first scaled by the corresponding entry in $\boldsymbol{\gamma}_\ell$ and then shifted by the corresponding entry in $\boldsymbol{\beta}_\ell$. The scale factors $\boldsymbol{\gamma}_\ell$ amplify or suppress specific learned features based on the thermal state, while the shift terms $\boldsymbol{\beta}_\ell$ add a context-dependent offset. For example, during warm-up the FiLM pathway can amplify features that capture transient heating gradients; near thermal equilibrium it can suppress those same features and emphasize steady-state spatial patterns instead.

For these affine parameters to be effective from the start of training, the FiLM projections use a near-identity initialization [50]. The standard deviation of 0.02 follows the convention established in [51] for initializing conditioning layers in generative networks:

$$\mathbf{W}_{\gamma_\ell} \sim \mathcal{N}(0,\, 0.02^2), \quad \mathbf{b}_{\gamma_\ell} = \mathbf{1}, \quad \mathbf{W}_{\beta_\ell} = \mathbf{0}, \quad \mathbf{b}_{\beta_\ell} = \mathbf{0} \tag{4.17}$$

Setting $\mathbf{b}_{\gamma_\ell} = \mathbf{1}$ and $\mathbf{b}_{\beta_\ell} = \mathbf{0}$ means that at the start of training, $\boldsymbol{\gamma}_\ell \approx \mathbf{1}$ and $\boldsymbol{\beta}_\ell \approx \mathbf{0}$, so the

modulation reduces to $\hat{\mathbf{h}}_\ell \approx \mathbf{h}_\ell$. The decoder therefore begins training as an ordinary U-Net, and the small random perturbation on $\mathbf{W}_{\gamma_\ell}$ breaks this symmetry so the conditioning pathway gradually learns meaningful modulation as training progresses.

The FiLM pathway complements the spatial encoder. The encoder extracts local per-pixel features from the 10-channel input (spatial gradients, temporal derivatives, thermal structure), while FiLM provides a global conditioning signal that tells the decoder *which board* and *which time instant* produced the current frame. The feature importance analysis in Section 4.6 quantifies the relative contribution of each pathway.

4.1.4 Decoder with FiLM Modulation

The decoder reverses the encoder's spatial compression, progressively upsampling the bottleneck representation back to the original 480×640 resolution. Each of the four decoder blocks performs three operations in sequence.

First, the feature map is upsampled and concatenated with the encoder skip connection. A 2×2 transposed convolution ($\text{ConvTranspose}_{C_\ell}$) doubles the spatial dimensions of the preceding feature map. A transposed convolution is the learnable inverse of the pooling/striding used in the encoder: it inserts values between existing pixels and applies a 2×2 learned kernel to produce an output with twice the height and width [52]. The upsampled result is then concatenated channel-wise with the skip connection $\mathbf{h}_k^{\text{skip}}$ from the encoder block at the same spatial resolution:

$$\mathbf{h}'_\ell = \text{Concat}\Big(\text{ConvTranspose}_{C_\ell}(\mathbf{h}_{\ell-1}),\ \mathbf{h}_k^{\text{skip}}\Big) \tag{4.18}$$

where k is the encoder block whose skip connection matches the decoder's current spatial size (e.g., decoder level 6 at 60×80 uses $\mathbf{h}_4^{\text{skip}}$ from encoder block 4, also at 60×80). The concatenation doubles the channel count temporarily, giving the subsequent convolutions access to both the decoder's reconstructed features and the encoder's original fine-grained detail.

Second, two 3×3 convolutions with ReLU activation refine the concatenated features and reduce the channel count back to C_ℓ:

$$\mathbf{h}_\ell = \mathrm{ReLU}\big(\mathrm{Conv}_{C_\ell}\big(\mathrm{ReLU}(\mathrm{Conv}_{C_\ell}(\mathbf{h}'_\ell))\big)\big) \tag{4.19}$$

Third, the FiLM scale and shift parameters $(\boldsymbol{\gamma}_\ell, \boldsymbol{\beta}_\ell)$ from Section 4.1.3 are applied channel-wise via (4.16). Applying FiLM at all four decoder levels—rather than only at the bottleneck—allows the conditioning signal to influence both coarse thermal patterns (early decoder blocks operating at low resolution) and fine component-level predictions (later blocks at high resolution).

Table 4.3 summarizes the decoder dimensions at each level, along with the corresponding encoder skip connection.

Table 4.3: Decoder block dimensions and corresponding encoder skip connections.

Level	**Channels (C_ℓ)**	**Output Size**	**Skip From**
Bottleneck ($\mathbf{h}_5$)	512	30×40	—
Decoder 6 ($\ell = 6$)	256	60×80	$\mathbf{h}_4^{\text{skip}}$ ($60 \times 80 \times 256$)
Decoder 7 ($\ell = 7$)	128	120×160	$\mathbf{h}_3^{\text{skip}}$ ($120 \times 160 \times 128$)
Decoder 8 ($\ell = 8$)	64	240×320	$\mathbf{h}_2^{\text{skip}}$ ($240 \times 320 \times 64$)
Decoder 9 ($\ell = 9$)	32	480×640	$\mathbf{h}_1^{\text{skip}}$ ($480 \times 640 \times 32$)

After the final decoder block, a 1×1 convolution with sigmoid activation [48] reduces the 32-channel feature map to a single-channel output:

$$\hat{y}_p = \sigma\big(\mathrm{Conv}_{1\times1}(\hat{\mathbf{h}}_9)\big)_p \in [0, 1] \tag{4.20}$$

The 1×1 convolution acts as a per-pixel linear combination of the 32 decoder features, and the sigmoid $\sigma(\cdot) = 1/(1 + e^{-z})$ constrains the output to $[0, 1]$, matching the normalized residual target range described next.

4.1.5 Residual Learning Target

With the architectural components defined, the remaining design decision is what the model actually predicts. Rather than predicting absolute temperatures, the model learns temperature residuals—the difference between the thermocouple contact measurement and the mean FLIR surface temperature over each component's ROI. This approach follows the residual learning principle introduced in [23], where predicting offsets rather than absolute values simplifies the learning task.

For each component c at a given timestep, the residual is computed as:

$$\bar{F}_c = \frac{1}{|\mathcal{M}_c|} \sum_{p \in \mathcal{M}_c} F_p, \qquad r_c = T_c^{\text{tc}} - \bar{F}_c \tag{4.21}$$

Here $\mathcal{M}_c$ is the set of pixels belonging to component c's ROI (from the SROI pipeline in Chapter 3) and $|\mathcal{M}_c|$ is the pixel count of that ROI. F_p is the FLIR-measured temperature at pixel p (in °C) and $\bar{F}_c$ is the mean FLIR temperature averaged over all pixels in the component's ROI. T_c^{tc} is the thermocouple contact temperature for component c (in °C), and r_c is the resulting residual—positive when the thermocouple reads hotter than the FLIR surface average, negative when it reads cooler.

Because the decoder output is bounded to [0, 1] by the sigmoid (4.20), the raw residual r_c must be normalized to match this range. The normalization proceeds in two steps: first, the residual is scaled to $[-1, +1]$ by dividing by the largest residual magnitude observed in the training set; then it is shifted to [0, 1]:

$$y_p = \frac{r_c / r_{\text{abs}} + 1}{2} \quad \forall\, p \in \mathcal{M}_c \tag{4.22}$$

where $r_{\text{abs}} = \max(|r_{\min}|, |r_{\max}|)$ is the largest absolute residual across all components and timesteps in the training set. For the datasets used in this work, r_{abs} is on the order of 15–20°C, reflecting the maximum offset between FLIR surface readings and thermocouple contact measurements observed during high-power testing. All pixels outside any component ROI are assigned $y_p = 0$ and are excluded from the primary loss (Section 4.1.6).

Residual learning provides two advantages for cross-board generalization. First, it decouples the prediction from absolute temperature. The encoder does not need to recover the board-level offset removed by z-score normalization, because the residual is measured relative to the local FLIR mean within each component ROI. Second, the residual distribution has lower variance than the absolute temperature distribution, concentrating the model's capacity on learning the correction between surface-measured FLIR values and contact-measured thermocouple values.

At inference, the predicted residual is mapped back to a temperature in degrees Celsius by inverting the normalization:

$$\hat{r}_c = (2\hat{y}_p - 1) \cdot r_{\text{abs}}, \qquad \hat{T}_c = \bar{F}_c + \hat{r}_c \tag{4.23}$$

Here $\hat{r}_c$ is the predicted residual for component c in degrees Celsius, $\hat{y}_p$ is the model's sigmoid-bounded output at pixel p (from (4.20)), r_{abs} is the normalization constant defined above, $\bar{F}_c$ is the mean FLIR temperature over component c's ROI, and $\hat{T}_c$ is the final predicted contact temperature. The term $(2\hat{y}_p - 1)$ maps the sigmoid output from $[0, 1]$ back to $[-1, +1]$, and multiplying by r_{abs} recovers the residual in degrees Celsius. Adding this predicted residual to the FLIR-measured mean $\bar{F}_c$ produces $\hat{T}_c$. If the model predicts $\hat{y}_p = 0.5$, the predicted residual is zero and $\hat{T}_c = \bar{F}_c$ (the FLIR reading is returned unchanged); values above 0.5 indicate the thermocouple is expected to read hotter than the FLIR surface average, and values below 0.5 indicate cooler.

4.1.6 Loss Function: Adaptive Weighted Masked MSE

The loss function determines what the optimizer penalizes during training. It combines two components: a variance-weighted mean squared error (MSE) over ROI pixels, and a physics-based smoothness regularizer over background pixels. Together, these two terms ensure that the model focuses on predicting accurate component temperatures while producing physically plausible temperature fields between components.

Primary Loss: Inverse-Variance Weighted MSE

The primary term operates only over labeled ROI pixels—pixels that fall within a component's SROI mask and have a corresponding thermocouple ground-truth value. Different boards and test conditions produce samples with different temperature spreads. Without correction, a sample with a 50°C temperature range would produce much larger squared errors than one with a 5°C range. This causes the high-range sample to dominate the gradient update and biases the model toward that board's thermal behavior.

To equalize this, each sample receives an inverse-variance weight that is inversely proportional to how spread out its target temperatures are. The intuition is straightforward: a training frame from a high-power test with large temperature differences across components would otherwise generate a disproportionately large error signal compared to a frame near steady state where all components sit at similar temperatures. Weighting by the inverse of this spread ensures that both frames contribute equally to the model's learning, preventing the high-power conditions from overshadowing the steady-state behavior.

Formally, for a batch of N training samples, let $\mathcal{M}^{(n)}$ denote the set of labeled ROI pixels for sample n—these are the pixels that fall within a component's SROI mask and have a corresponding thermocouple ground-truth value. The per-sample target variance $v^{(n)} = \mathrm{Var}\{y_p : p \in \mathcal{M}^{(n)}\}$ measures how spread out the normalized residual targets are across all instrumented components in that frame. The adaptive weight is:

$$w^{(n)} = \frac{(v^{(n)} + \epsilon)^{-1}}{\frac{1}{N}\sum_{m=1}^{N}(v^{(m)} + \epsilon)^{-1}}, \qquad \epsilon = 0.01 \tag{4.24}$$

Here $w^{(n)}$ is the normalized weight assigned to sample n, with larger weights given to samples whose target values are tightly clustered (low $v^{(n)}$) and smaller weights given to samples with widely spread targets (high $v^{(n)}$). The stabilization constant $\epsilon = 0.01$ prevents division by zero when a sample has nearly constant target values, such as during steady state where all components have converged to similar temperatures. The denominator normalizes the weights so their mean across the batch equals one, ensuring that the overall learning rate is unaffected—only the relative contribution of each sample changes. This weighting scheme

is inspired by the uncertainty-based multi-task loss in [53].

The primary per-sample loss is then:

$$\mathcal{L}_{\text{primary}}^{(n)} = \frac{w^{(n)}}{|\mathcal{M}^{(n)}|} \sum_{p \in \mathcal{M}^{(n)}} \left(y_p^{(n)} - \hat{y}_p^{(n)}\right)^2 \tag{4.25}$$

where $y_p^{(n)}$ is the normalized residual target (from (4.22)) and $\hat{y}_p^{(n)}$ is the model's sigmoid-bounded prediction at pixel p.

Smoothness Regularizer

The second loss component encourages physically plausible temperature fields in the background (non-ROI) region of the prediction, following the principle of embedding physical constraints into neural network loss functions [54]. On a thermally conductive surface away from active heat sources, steady-state temperature satisfies Laplace's equation $\nabla^2 T = 0$ –meaning neighboring pixels should have similar temperatures. The smoothness loss penalizes deviations from this condition:

$$\mathcal{L}_{\text{smooth}}^{(n)} = \frac{1}{|\bar{\mathcal{M}}^{(n)}|} \sum_{p \notin \mathcal{M}^{(n)}} \left(\nabla^2 \hat{y}_p^{(n)}\right)^2 \tag{4.26}$$

where $\bar{\mathcal{M}}^{(n)}$ is the set of unlabeled background pixels for sample n. The discrete Laplacian ∇^2 is computed by convolving the prediction with the standard 3×3 kernel:

$$\nabla^2 \text{ kernel} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

This kernel compares each pixel to its four immediate neighbors (same operation as Channel 3 from Section 4.1.1). Where the predicted temperature varies smoothly, the Laplacian is near zero and the penalty is small; where it varies abruptly—as would happen if the decoder produced noisy or discontinuous background predictions—the penalty is large.

Combined Loss and Board-Level Weighting

The per-sample total loss combines both components:

$$\mathcal{L}^{(n)} = \mathcal{L}^{(n)}_{\text{primary}} + \alpha \cdot \mathcal{L}^{(n)}_{\text{smooth}}, \qquad \alpha = 0.005 \tag{4.27}$$

The smoothness weight $\alpha = 0.005$ was chosen so that the regularizer gently discourages abrupt temperature jumps in the background without competing with the primary ROI loss—at this scale, the smoothness term contributes roughly 0.5% of the total loss during typical training, enough to prevent background artifacts without distorting the component-level predictions.

A final board-level weighting addresses the same imbalance problem at the board level. Each sample n from board b is weighted by the board's inverse-variance weight w_b, normalized so the mean weight across boards equals one:

$$\bar{\mathcal{L}} = \frac{1}{N} \sum_{n=1}^{N} w_b \cdot \mathcal{L}^{(n)} \tag{4.28}$$

Without this weighting, the board with higher overall temperature variance would dominate the gradient signal. With it, each DUT contributes equally to the optimizer's parameter updates despite their different thermal operating ranges.

In summary, the loss applies three levels of balancing:

1. **Per-pixel masking** restricts the primary loss to instrumented component ROIs.
2. **Per-sample inverse-variance weighting** ($w^{(n)}$) normalizes across samples with different temperature spreads within a batch.
3. **Per-board weighting** (w_b) equalizes gradient contributions between PCB designs with different overall thermal ranges.

This multi-level balancing is important for the cross-board generalization goal: without it, the model would preferentially fit whichever board or operating condition produces the largest loss values, at the expense of accuracy on the others.

With the model architecture, residual learning target, and loss function now fully defined, the remaining question is how the raw FLIR recordings and thermocouple logs from Chapter 3 are transformed into the 10-channel tensors, thermal state vectors, and residual target maps that the model requires. The following section describes this data pipeline, and Section 4.3 then covers the training strategy—data splitting, temporal sampling, hyperparameters, and augmentation—that governs how the model learns from the assembled dataset.

4.2 Data Pipeline: SROI to Training Datasets

The data pipeline bridges the SROI measurement framework (Chapter 3) and the training process described in Section 4.3, transforming raw thermal recordings into the ML-ready datasets that the model consumes. Referring to the pipeline stages in Fig. 3.1 of Chapter 3: Stage 1 (Altium export) and Stage 2 (coordinate transformation) produce the SROI file, Stage 3 (ResearchIR extraction) produces the per-frame thermal data, and Stage 4 (post-processing) assembles the ML-ready datasets described below.

Each thermal test session produces two synchronized data streams. The FLIR thermal video is recorded at 15-second intervals through ResearchIR; thermocouple contact measurements are logged at 0.5-second intervals (2 Hz) and downsampled to match the 15-second FLIR frame rate. The SROI file generated by the automated pipeline described in Section 3.1 of Chapter 3 defines the pixel coordinates for each component ROI, enabling ResearchIR to extract per-component temperature statistics (mean, min, max) from every recorded frame without manual intervention. The ROI pixel map CSV exported alongside the SROI binary (Section 3.1.3) provides the spatial index that maps component designators to their pixel locations in the 640×480 thermal frame—this mapping is used directly to construct Channel 4 (the PCB binary mask) and to compute the per-component FLIR mean $\bar{F}_c$ in the residual target (4.21).

Thermocouple data provides the ground-truth contact measurements at component locations (Section 3.2.2). The FLIR-thermocouple residual in (4.21) is then computed for each instrumented component at each timestep, producing the normalized target maps

described in Section 4.1.5. Components without thermocouple instrumentation are masked out in the loss function, as described in Section 4.1.6.

The aligned frame-level data is assembled into 10-channel input tensors ($480 \times 640 \times 10$) following the channel construction summarized in Table 4.1 and detailed in Section 4.1.1. Thermal state vectors $\mathbf{s} \in \mathbb{R}^6$ (4.9) are computed from each frame's timestamp and thermal statistics. These tensors, targets, and ROI masks are then packaged into HDF5 files—one per board per test condition—for efficient batch loading during model training. Each HDF5 file contains the complete temporal sequence for one board, with dataset keys for the image tensor, thermal state vector, target residual map, and ROI mask at each timestep.

4.3 Training Strategy

The HDF5 datasets assembled in Section 4.2 provide the inputs for training, but four decisions govern how the model learns from them. First, the data must be split in a way that prevents the model from memorizing frame sequences rather than learning thermal coupling patterns. Second, boards with different test durations must be resampled so that no single board dominates the gradient signal. Third, the optimizer, learning rate schedule, and initialization robustness evaluation must be configured to produce stable, reproducible results. Fourth, spatial augmentation must expand the effective training set to reduce the risk of the encoder overfitting to the fixed component layout of any single PCB. Section 4.3.1 addresses the first concern through component-level splitting. Section 4.3.2 addresses the second through non-uniform temporal resampling and per-board balancing. Section 4.3.3 covers the optimizer and robustness evaluation protocol. Section 4.3.4 describes the mirroring augmentation that quadruples the number of unique spatial arrangements available during training.

4.3.1 Component-Level Data Splitting

Model training combines data from DUTs with random component-level splitting using an 80/20 train-test partition. The split is performed at the component level rather than the frame level: all temporal frames for a given component are assigned entirely to training

or testing. This prevents temporal leakage where the model memorizes frame sequences rather than learning thermal coupling patterns. Validation components span all of the DUT designs included within training, ensuring the test set evaluates cross-PCB generalization.

4.3.2 Temporal Sampling and Dataset Balancing

Non-uniform temporal sampling addresses two related challenges: the information imbalance between transient and steady-state phases, and the sample-count imbalance between boards with different test durations. Left unaddressed, the board with the most frames would dominate gradient updates, biasing the model toward that board's thermal behavior. Each board's thermal test spans a different total duration at the same 15-second FLIR acquisition interval. DUT-1 and DUT-Test both cover 4,500 s of warm-up followed by a longer steady-state tail; DUT-2 covers 1,800 s of warm-up with a proportionally shorter tail. Raw frame counts therefore differ substantially across boards. A balanced dataset generation step equalizes per-board contributions prior to training by independently resampling each board into a common target count.

Resampling divides each board's timeline into a dense region (warm-up) and a sparse region (steady-state). Within the dense region, all boards are resampled to produce the same number of frames—N_{dense} samples—by adjusting the effective step size. Both N_{dense} and the overall per-board target N_{target} are tunable parameters that the user selects based on the available hardware resources (system RAM for dataset generation, GPU VRAM for training batch size) and the desired temporal resolution. Larger values of N_{dense} produce finer temporal coverage of the warm-up transient but increase memory consumption proportionally; smaller values reduce memory requirements at the cost of under-sampling rapid thermal changes.

$$\Delta t_{\text{dense}}^{(b)} = \frac{T_{\text{dense}}^{(b)}}{N_{\text{dense}} - 1} \tag{4.29}$$

where $T_{\text{dense}}^{(b)}$ is board b's dense-region duration and N_{dense} is the chosen dense target count. When the resampled timestamps do not coincide with original FLIR frames, linear interpolation between adjacent frames produces the intermediate thermal images.

Within the sparse region, a per-board step size is selected by grid search over integer multiples of the native 15 s interval:

$$m_b^* = \arg\min_{m \in \{1,\ldots,500\}} \left| \left\lfloor \frac{T_{\text{sparse}}^{(b)}}{m \cdot 15} \right\rfloor + 1 - N_{\text{sparse}} \right| \tag{4.30}$$

where $T_{\text{sparse}}^{(b)}$ is the duration of the sparse (cool-down) region for board b, m is the candidate multiplier of the 15 s native interval, m_b^* is the optimal multiplier for board b, and $N_{\text{sparse}} = N_{\text{target}} - N_{\text{dense}}$ is the remaining per-board sample budget after the dense region. This search ensures that all boards contribute within 5% of one another in total sample count, despite their different test durations. As a concrete example, Table 4.4 shows the resulting configuration when $N_{\text{dense}} = 121$ and $N_{\text{target}} \approx 500$. With these settings, DUT-1 and DUT-Test ($T_{\text{dense}} = 4{,}500$ s) produce a dense step of ≈37.5 s, while DUT-2 ($T_{\text{dense}} = 1{,}800$ s) retains the native 15 s interval. Different choices of N_{dense} and N_{target} shift these step sizes accordingly.

Table 4.4: Example balanced dataset configuration with $N_{\text{dense}} = 121$ and $N_{\text{target}} \approx 500$.

Board	**Dense Step**	N_{dense}	**Sparse Step**	N_{sparse}	**Total**
DUT-1 (HBridge)	37.5 s	121	285 s	377	498
DUT-2 (LoadShedding)	15 s	121	300 s	370	491
DUT-Test (TGD)	37.5 s	121	285 s	378	499

Balanced files are generated once during preprocessing and stored as HDF5 datasets with the same schema as the originals, allowing the training pipeline to load them interchangeably. An interpolation flag stored alongside each sample distinguishes native frames from interpolated ones, enabling post-hoc analysis of interpolation effects if needed. With spatial augmentation (Section 4.3.4), each of the 8 board variants receives its own balanced file, producing a total training pool of approximately $8 \times N_{\text{target}}$ samples before the 80/20 component-level split.

4.3.3 Training Hyperparameters

Training employs the Adam optimizer [55] with an initial learning rate $\eta = 1 \times 10^{-3}$ and a customizable batch size. Two callbacks govern training duration and learning rate adaptation, both monitoring validation loss to prevent overfitting. The first callback, early stopping [48, 56], terminates training if validation loss does not improve for the configured number of consecutive epochs and restores the model weights to the best-performing epoch. The second callback, a ReduceLROnPlateau scheduler [57], reduces the learning rate by a factor of 0.5 whenever validation loss plateaus for 5 consecutive epochs:

$$\eta_{k+1} = \begin{cases} 0.5\,\eta_k & \text{if val\_loss has not improved for 5 epochs} \\ \eta_k & \text{otherwise} \end{cases} \tag{4.31}$$

where k is the epoch index and η_k is the learning rate at epoch k. A floor of $\eta_{\min} = 1 \times 10^{-6}$ prevents the learning rate from collapsing. Because the differential learning rate multiplier (Section 4.3.6) scales FiLM gradients by a constant factor m, the scheduler reduction propagates proportionally to both the backbone (η_k) and FiLM ($m \cdot \eta_k$) pathways. Initialization robustness testing across a configurable number of random weight initializations assesses sensitivity to initialization [58], with performance quantified using coefficient of determination (R^2), root mean squared error (RMSE), and mean absolute error (MAE) [48] computed at ROI pixel locations.

4.3.4 Spatial Data Augmentation

With a small number of training boards, the convolutional encoder risks memorizing the fixed spatial arrangement of components on each PCB rather than learning generalizable thermal coupling patterns. Spatial data augmentation [59] addresses this risk by mirroring each board's dataset along the horizontal (X), vertical (Y), and combined (XY) axes prior to training, producing three additional geometric variants per board and quadrupling the number of unique thermal spatial patterns available for training.

Mirroring is applied during preprocessing rather than on-the-fly. A dedicated script

reads each board's canonical HDF5 dataset and writes three new files—one per flip variant—containing the transformed data. For each variant, every 480×640 FLIR frame is flipped using NumPy array slicing: vertical flipping reverses the row axis, horizontal flipping reverses the column axis, and combined flipping applies both operations sequentially. The per-component ROI binary masks undergo the identical spatial transformation, ensuring that thermocouple-to-pixel alignment is preserved after flipping. A post-generation verification step confirms that each flipped ROI mask retains the same pixel count as its original, guaranteeing no spatial information is lost or misaligned. Fig. 4.3 illustrates this verification step for the four spatial orientations of DUT-2.

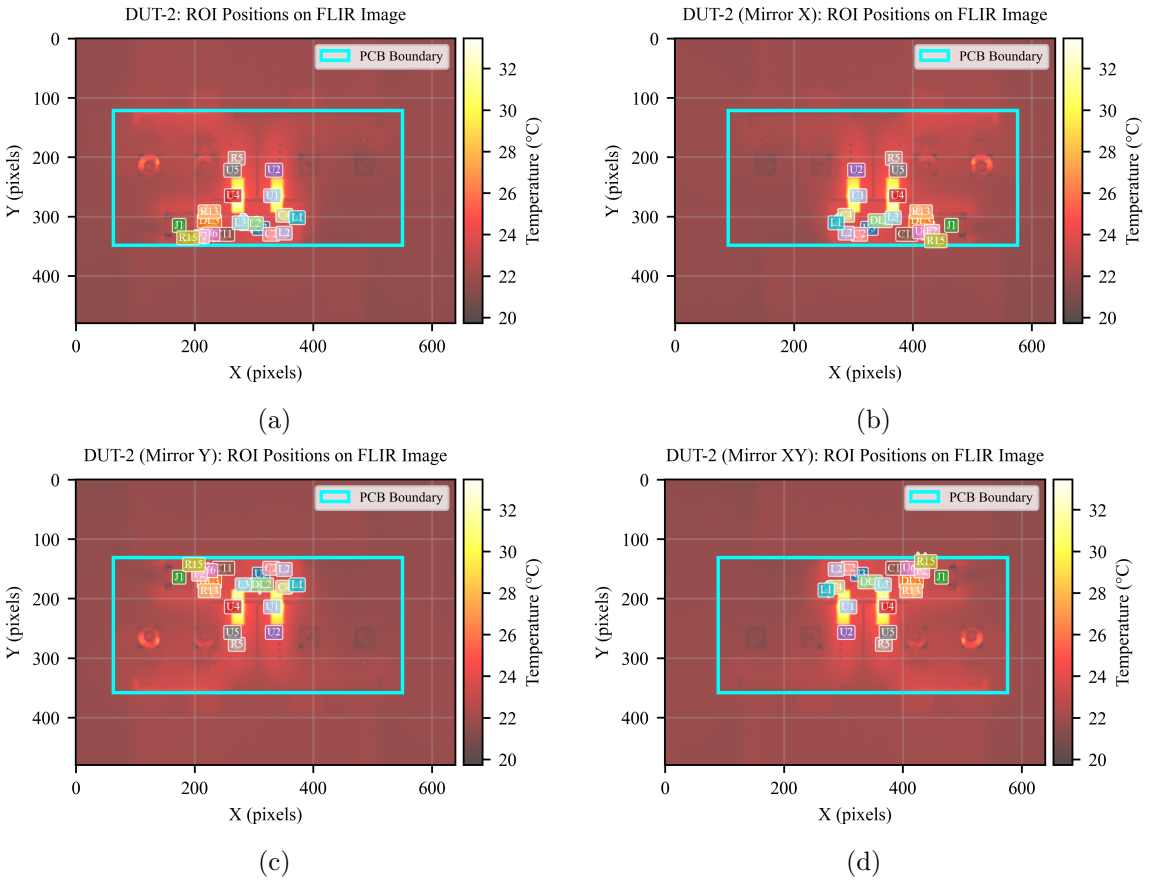


Fig. 4.3: Spatial data augmentation through mirroring for DUT-2 (LoadShedding): (a) original orientation, (b) horizontal flip (X-axis), (c) vertical flip (Y-axis), (d) combined flip (XY). ROI overlay positions confirm that component masks are flipped consistently with the FLIR frame.

Scalar quantities that are not spatially dependent—thermocouple temperatures, timestamps, and frame indices—are copied unchanged, since flipping a PCB image does not alter temperature values or temporal information. The PCB corner coordinates stored in the dataset metadata are recalculated to reflect the new bounding box after flipping, which ensures that downstream spatial features (e.g., the PCB mask in Channel 4 and the gradient features in Channels 1–3) are computed correctly from the transformed frames.

Each mirrored variant is assigned a unique board identifier (e.g., board_id 3.0 for DUT-1 flipX, 4.0 for DUT-1 flipY, etc.), which propagates into the domain descriptor channels Ch8 and Ch9. This means the FiLM conditioning pathway treats each variant as a distinct board with its own thermal operating point. Additionally, each variant receives an independently re-randomized 80/20 component-level train/validation split, so different components are held out across variants. This prevents the model from seeing all components of a given board geometry across variants and forces it to generalize across component subsets as well as spatial orientations. The result is $N_{\text{boards}} \times 4$ total training variants ($N_{\text{boards}} = 2$ yields 8: 2 original + 6 mirrored), each with a distinct spatial layout, unique board identifier, and independent validation split.

4.3.5 Domain Augmentation

Training sets containing only a small number of boards (DUT-1 and DUT-2 only) each occupy a narrow region in the Ch8–Ch9 domain descriptor space. Without augmentation, the FiLM conditioning pathway sees only two discrete domain points during training, which allows the affine projections to learn a sharp two-class step function that does not extrapolate to unseen boards. Domain augmentation addresses this limitation by perturbing the board-level domain descriptors during training, expanding the two discrete domain points into a continuous cloud that covers a broader range of thermal operating conditions. The augmentation operates only during training, after computing Ch8/Ch9 and $\mathbf{s}$, and before the model processes the data. It does not modify the model architecture or loss function.

The first augmentation technique is Gaussian jitter applied to the domain descriptors—a form of noise injection regularization [48]. For each training sample, independent Gaussian

perturbations are added:

$$\mathrm{Ch8}_{\mathrm{aug}} = \mathrm{clip}\big(\mathrm{Ch8} + \varepsilon_8,\ 0,\ 1\big), \quad \varepsilon_8 \sim \mathcal{N}(0, \sigma^2_{\mathrm{aug}}) \tag{4.32}$$

$$\mathrm{Ch9}_{\mathrm{aug}} = \mathrm{clip}\big(\mathrm{Ch9} + \varepsilon_9,\ 0,\ 1\big), \quad \varepsilon_9 \sim \mathcal{N}(0, \sigma^2_{\mathrm{aug}}) \tag{4.33}$$

with $\sigma_{\mathrm{aug}} = 0.50$. Because Ch8 and Ch9 are produced by the sigmoid function they already lie in $(0, 1)$; the clip operation is therefore rarely active but provides a safety bound after additive noise. The augmented values replace both the spatial channels $(x_p^{(8)}, x_p^{(9)})$ and the corresponding entries in the thermal state vector $\mathbf{s}$, maintaining consistency between the encoder and FiLM inputs. The perturbation propagates through the entire FiLM chain: noise in $\mathbf{s}$ produces proportional perturbations in the embedding $\mathbf{z}$, which in turn perturb $\boldsymbol{\gamma}_\ell$ and $\boldsymbol{\beta}_\ell$ at each decoder level. This forces the FiLM projections to learn smooth mappings across the domain parameter space rather than memorizing two discrete points.

The second technique is cross-domain mixing, which creates virtual training samples whose domain descriptors fall *between* existing boards. Gaussian jitter spreads each board's domain point into a cloud centered on its original location, but it cannot produce descriptors that lie in the gap between boards—for example, a hypothetical board whose thermal amplitude is midway between DUT-1 and DUT-2. Cross-domain mixing fills this gap by blending the domain statistics of two real boards, following the MixUp [60] and MixStyle [61] frameworks for constructing interpolated training samples.

During training, each sample has a $p_{\mathrm{mix}} = 0.30$ probability of undergoing cross-domain mixing. When mixing is triggered, a sample from board A has its domain descriptors interpolated with those of a randomly selected board B:

$$\mathrm{Ch9}_{\mathrm{mix}} = \lambda \cdot \mathrm{Ch9}_A + (1 - \lambda) \cdot \mathrm{Ch9}_B, \qquad \lambda \sim \mathrm{Beta}(\alpha_{\mathrm{mix}}, \alpha_{\mathrm{mix}}) \tag{4.34}$$

Here $\mathrm{Ch9}_A$ and $\mathrm{Ch9}_B$ are the domain descriptor values for the two boards, and the mixing coefficient $\lambda \in [0, 1]$ controls how much of each board's descriptor is used (Ch8 is blended analogously). The Beta distribution with shape parameter $\alpha_{\mathrm{mix}} = 0.40$ concentrates λ near

0 and 1, so most mixed samples stay close to one board's descriptors rather than landing halfway between them—this keeps the domain identity roughly consistent with board A's unmixed spatial pattern while still exposing the FiLM pathway to intermediate operating points.

The effect is that a sample keeps board A's pixel-level thermal pattern but receives domain descriptors shifted toward board B, preventing the FiLM pathway from learning a brittle association between each board's exact descriptor values and its thermal behavior. Both techniques are applied together during training, with Gaussian jitter following cross-domain mixing.

4.3.6 Differential Learning Rate

The FiLM conditioning pathway comprises 64,736 parameters out of the total 7.8M ($< 0.9\%$), yet these parameters are the sole pathway through which the model conditions on temporal and domain state. Because FiLM gradients must pass through the small-magnitude embedding MLP, the conditioning pathway trains slowly relative to the convolutional encoder-decoder.

A differential learning rate—a technique introduced for discriminative fine-tuning in [62]—addresses this imbalance by scaling the gradients for all FiLM and embedding variables by a multiplier $m = 10$:

$$\tilde{g}_v = \begin{cases} m \cdot g_v & \text{if } v \in \mathcal{V}_{\text{FiLM}} \\ g_v & \text{otherwise} \end{cases} \tag{4.35}$$

where v denotes any trainable parameter, $g_v = \partial \mathcal{L} / \partial v$ is the raw gradient of the loss with respect to that parameter, $\tilde{g}_v$ is the scaled gradient passed to the optimizer, and $\mathcal{V}_{\text{FiLM}}$ includes all kernel and bias parameters from the embedding MLP and the four pairs of FiLM affine projections. The effective learning rate for FiLM parameters becomes $\eta_{\text{FiLM}} = m \cdot \eta$, while the backbone retains the base rate η. When the learning rate scheduler reduces η, both pathways decay proportionally, preserving the ratio m throughout training.

4.4 Results: Initialization Robustness Validation

The training strategy described in Section 4.3 produces a trained model evaluated here across two configurations that separate best-case within-training performance from held-out cross-design generalization. Section 4.4.1 describes the experimental platforms, evaluation metrics, and seed selection protocol applied consistently across both configurations.

4.4.1 Test Setup

The thermal datasets used for model training and evaluation are produced by the validation framework described in Section 3.2 of Chapter 3. Two PCBs serve as experimental platforms: DUT-1 (TABSRC sensing interface [63], 22 monitored components) and DUT-2 (Load Shedding switch board [64], 20 monitored components). Both DUTs undergo 60–90 minute thermal tests in air and sand environments, with a FLIR T5590 camera recording 640×480 thermal frames at 15-second intervals (Section 3.2.1) and thermocouples logging contact temperatures at each instrumented component (Section 3.2.2). The resulting datasets contain approximately 200–400 FLIR frames per test session with 42 total monitored components across both designs. Because the model's inputs and targets are derived entirely from this measurement framework—FLIR surface images, thermocouple contact readings, and SROI-defined ROI masks—the prediction accuracy reported below reflects the model's ability to learn the FLIR-to-thermocouple correction, independent of the thermal test conditions themselves.

Two training configurations are evaluated: (1) DUT-Test excluded from training (two-board model) and (2) DUT-Test included in training (three-board model). Unless otherwise noted, all reported metrics correspond to the best seed from the initialization robustness evaluation, selected by lowest validation loss. The two-board model uses seed 51 (R^2 = 0.963, MAE = 1.63°C) and the three-board model uses seed 45 (R^2 = 0.969, MAE = 1.53°C), each selected from 10 seeds (42–51) by lowest validation loss.

The results are organized into three subsections corresponding to three distinct evaluation configurations. Section 4.4.2 presents the three-board model in which DUT-Test is included in the training set with an 80/20 component-level split, providing the best-case

performance baseline. Section 4.4.3 presents the two-board model in which DUT-Test is excluded entirely from training and evaluated as an unseen board, testing cross-design generalization. Section 4.4.4 presents an initialization robustness analysis of the two-board configuration across 10 random weight initializations, quantifying sensitivity to initialization.

4.4.2 Three-Board Model: DUT-Test Included in Training

In this configuration, all three boards—DUT-1, DUT-2, and DUT-Test—are included in the training set with an 80/20 component-level split. This represents the best-case scenario where thermal data is available for every board of interest, and the model has the opportunity to observe each board's thermal patterns during training.

Fig. 4.4 shows the per-component error distributions across all three DUTs. DUT-2 (LoadShedding) has the tightest error spread (MAE = 0.76°C), reflecting its lower thermal gradients, while DUT-1 (HBridge) shows the widest spread (MAE = 2.39°C) due to its higher power dissipation. DUT-Test (ThermalGateDriver) falls between the two (MAE = 1.36°C), confirming that the model captures its thermal patterns when included in training.

Fig. 4.5 shows the timeseries predictions for the best-performing components on each DUT alongside the actual thermocouple readings. Temporal predictions capture both the transient warm-up dynamics and steady-state equilibrium, confirming that FiLM conditioning enables accurate prediction across the full temporal range.

Fig. 4.6 shows the aggregate predicted versus actual temperature scatter. Points tightly clustered along the unity line ($y = x$) indicate low systematic bias, and the narrow shaded regions (first two standard deviations of MAE) confirm uniform accuracy across the temperature range.

The outlier predictions visible beyond the 2σ shaded region correspond to DUT-1 (H-Bridge), specifically the PS2 power supply component, which shows thermal coupling behavior distinct from the majority of components on that board. PS2 operates at a different thermal time constant and steady-state temperature relative to neighboring components,

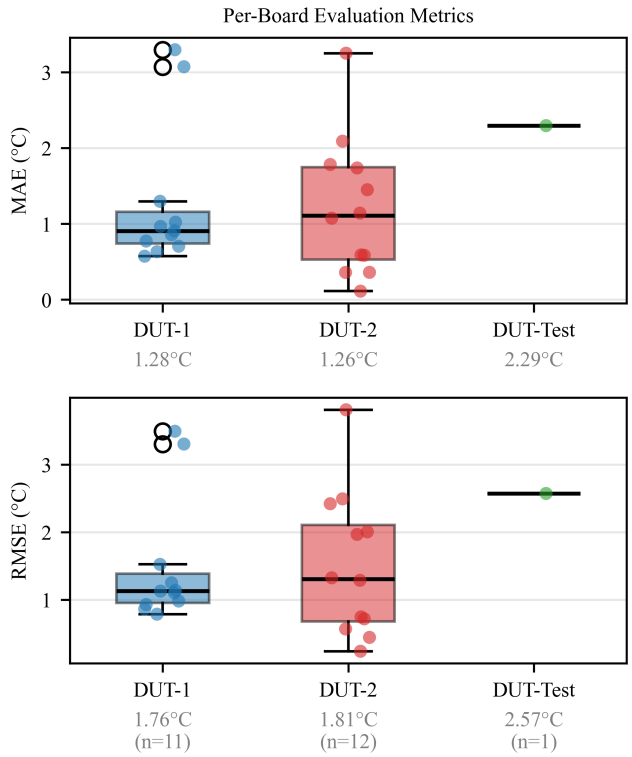


Fig. 4.4: Box-and-whisker comparison of per-component MAE and RMSE across all DUTs with DUT-Test included in the training set (80/20 component-level split).

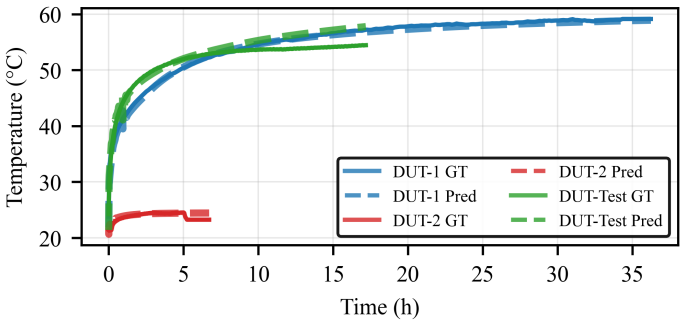


Fig. 4.5: Timeseries predictions of the best-performing components for each DUT alongside actual thermocouple readings, with DUT-Test included in training.

making it an atypical sample in the training distribution. Importantly, the model's prediction error on PS2 is an over-prediction—the model assigns a higher temperature than measured—which is a conservative outcome from a reliability standpoint: a design engineer relying on this prediction would apply more thermal margin to that component, not less. This behavior reflects the model appropriately flagging thermal uncertainty for an outlier

component rather than a systematic failure of prediction accuracy.

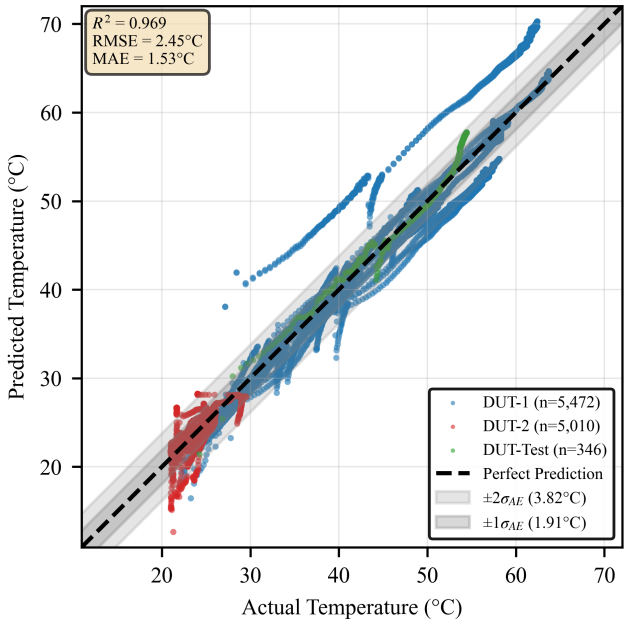


Fig. 4.6: Predicted vs. actual temperature scatter for the three-board model (DUT-Test included in training), with the first two σ of MAE shown as shaded regions around the unity line.

Table 4.5 summarizes the quantitative results for the three-board model, reporting per-board mixed-PCB validation metrics alongside the overall aggregate.

Table 4.5: Three-Board Model Results: Mixed-PCB Validation Performance (Best Seed 45).

Board	**Evaluation Type**	**R^2**	**RMSE (°C)**	**MAE (°C)**
DUT-1 (HBridge)	Mixed-PCB validation	0.829	3.98	2.39
DUT-2 (LoadShedding)	Mixed-PCB validation	0.787	0.95	0.76
DUT-Test (Gate Driver)	Mixed-PCB validation	0.934	1.69	1.36
Overall	**Mixed-PCB val**	**0.969**	**2.45**	**1.53**

The three-board model achieves an overall $R^2 = 0.969$, RMSE = 2.45°C, and MAE = 1.53°C across all validation components. DUT-2 has the lowest error (MAE = 0.76°C) owing to its moderate thermal gradients, while DUT-1 shows the highest (MAE = 2.39°C) due to

its larger power dissipation and steeper spatial gradients. DUT-Test achieves MAE = 1.36°C when included in training—a result that will serve as a baseline for comparison against the two-board held-out evaluation in Section 4.4.3.

Across the full dataset, the model does not exhibit a systematic tendency toward over- or under-prediction: the distribution of errors is approximately zero-mean for both boards and test conditions, as reflected in the tight scatter around the unity line. This symmetry is intentional—the model is trained with a symmetric loss function that treats under-prediction and over-prediction equivalently. From a reliability standpoint, a designer may prefer predictions that systematically err on the side of higher temperatures, ensuring that thermal margins are not underestimated. Approaches for introducing a conservative bias—including post-processing temperature offsets and asymmetric loss retraining—are discussed as a future direction in Section 5.4.

Fig. 4.7 presents the training curves for this configuration. The learning rate schedule, overfitting gap, and convergence trajectories confirm that the model trains stably with all three boards present.

4.4.3 Two-Board Model: DUT-Test Excluded from Training

In this configuration, only DUT-1 and DUT-2 are used for training (80/20 component-level split). DUT-Test is excluded entirely from the training set and evaluated as an unseen board, testing the practical deployment scenario where a new PCB design must be assessed without design-specific training data. This is the more demanding evaluation: the model must generalize its learned thermal coupling patterns to a board it has never observed.

Fig. 4.8 shows the per-component error distributions. Despite never appearing in training, DUT-Test achieves a held-out MAE = 1.24°C—lower than DUT-1's validation MAE of 2.94°C—demonstrating that the model's learned thermal coupling patterns transfer effectively to the unseen board design.

Fig. 4.9 shows the timeseries predictions for this configuration. Despite never observing DUT-Test during training, the model tracks both transient and steady-state behavior, though with wider prediction error bands compared to the three-board model. Fig. 4.10

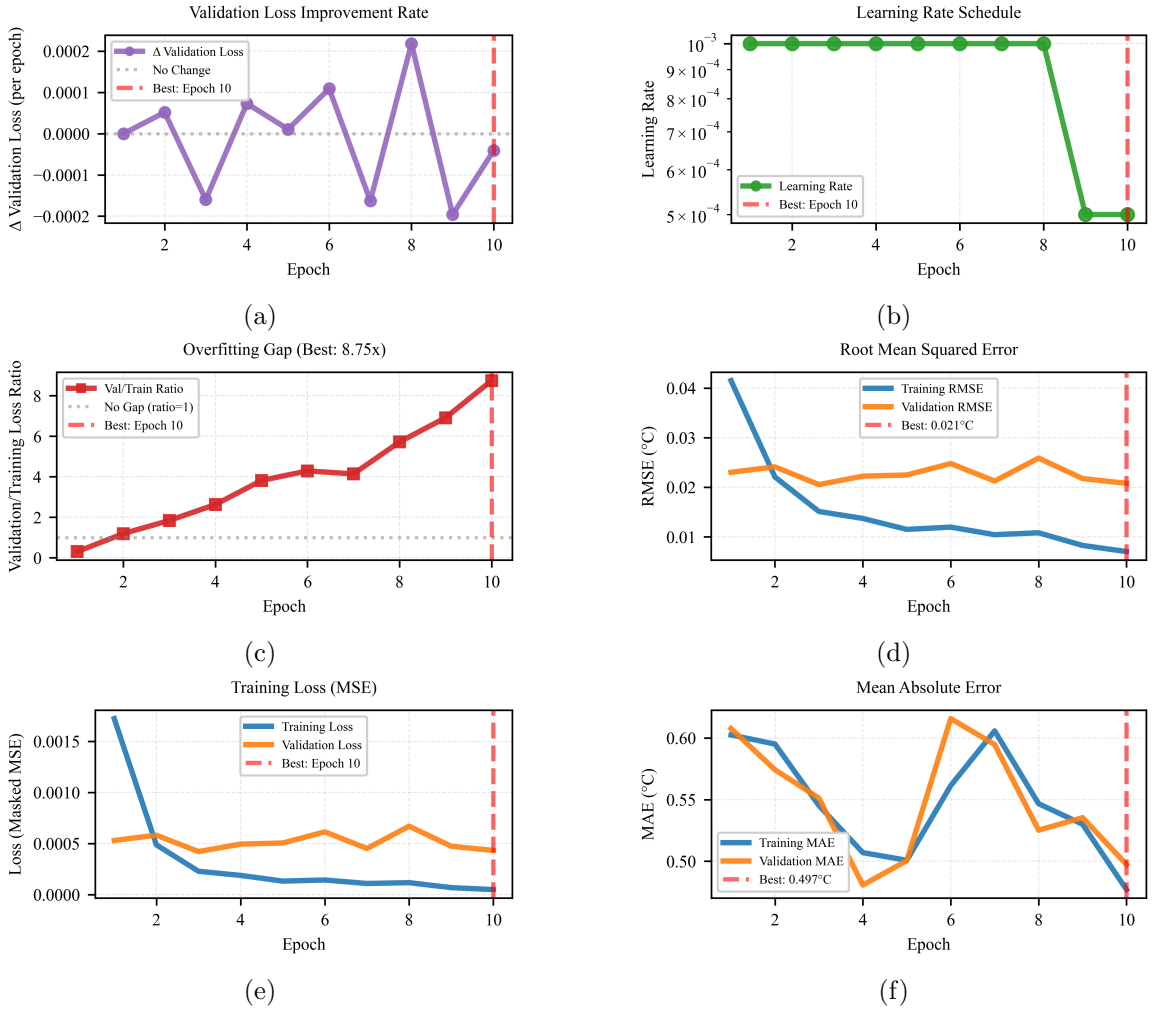


Fig. 4.7: Training curves with DUT-Test included in the training set: (a) improvement rate, (b) learning rate schedule, (c) overfitting gap, (d) RMSE, (e) loss, (f) MAE. Red lines indicate the best epoch as determined by lowest validation loss.

shows the predicted versus actual temperature scatter for the two-board model. Fig. 4.11 presents the training curves for this two-board configuration. Table 4.6 summarizes the quantitative results for the two-board model, comparing the mixed-PCB validation performance on training boards (DUT-1 and DUT-2) with the unseen-board evaluation on DUT-Test.

The DUT-Test unseen-board evaluation achieves R^2=0.949, RMSE=1.65°C, and MAE=1.24°C across 2,768 component-timestep pairs—a strong result given that the model has never observed DUT-Test data during training. The R^2=0.949 indicates that the model explains

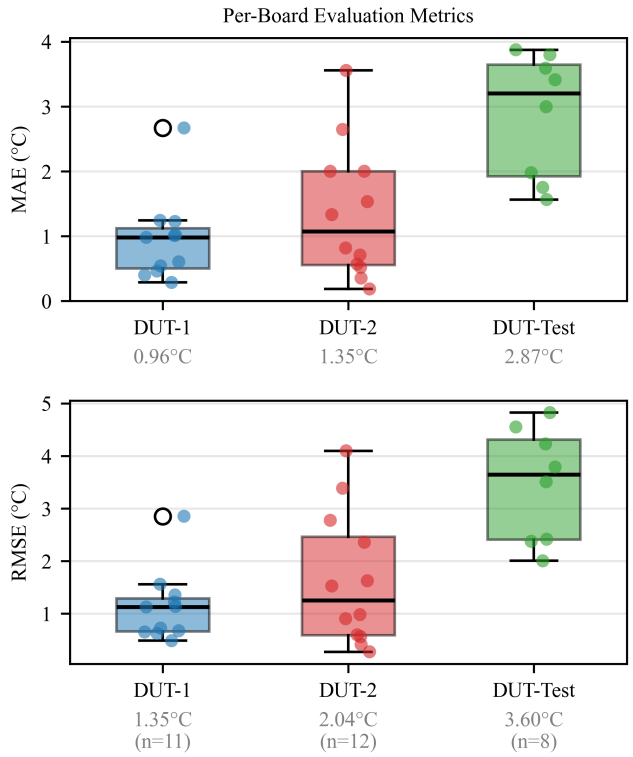


Fig. 4.8: Box-and-whisker comparison of per-component MAE and RMSE across all DUTs with DUT-Test excluded from training and evaluated as an unseen board.

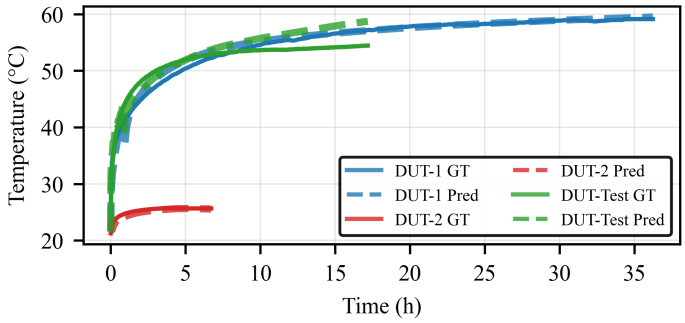


Fig. 4.9: Timeseries predictions of the best-performing components for each DUT alongside actual thermocouple readings, with DUT-Test excluded from training.

95% of the variance in DUT-Test component temperatures, confirming substantial transfer of thermal coupling patterns across PCB designs. The RMSE of 1.65°C is lower than the 2.70°C overall mixed-PCB validation error on training boards, indicating that the model generalizes effectively to this unseen design. For junction-temperature-limited components, rated maximums typically provide 10–25°C of margin above nominal operation. A 1.24°C

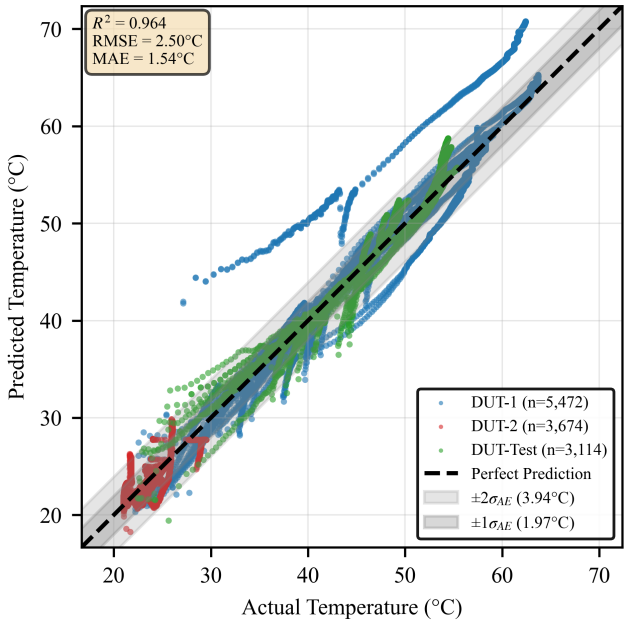


Fig. 4.10: Predicted vs. actual temperature scatter for the two-board model (DUT-Test excluded from training), with the first two σ of MAE shown as shaded regions around the unity line.

Table 4.6: Two-Board Model Results: Training Board Validation and Unseen-Board Generalization (Best Seed 51).

Board	**Evaluation Type**	**R^2**	**RMSE (°C)**	**MAE (°C)**
DUT-1 (HBridge)	Mixed-PCB validation	0.791	4.39	2.94
Overall	**Mixed-PCB val**	**0.963**	**2.70**	**1.63**
DUT-Test (Gate Driver)	Held-out (unseen-board)	0.949	1.65	1.24

MAE is well within this margin, enabling reliable identification of thermally at-risk components without per-design FEM setup.

4.4.4 Initialization Robustness Analysis

To assess whether the two-board model's performance is sensitive to the random weight initialization, initialization robustness testing across 10 random initializations is performed—a practice recommended in [58] to ensure reproducible benchmarking. All 10 seeds use the two-board configuration (DUT-Test excluded from training) with identical hyperparameters; only the random weight initialization differs. Table 4.7 summarizes the

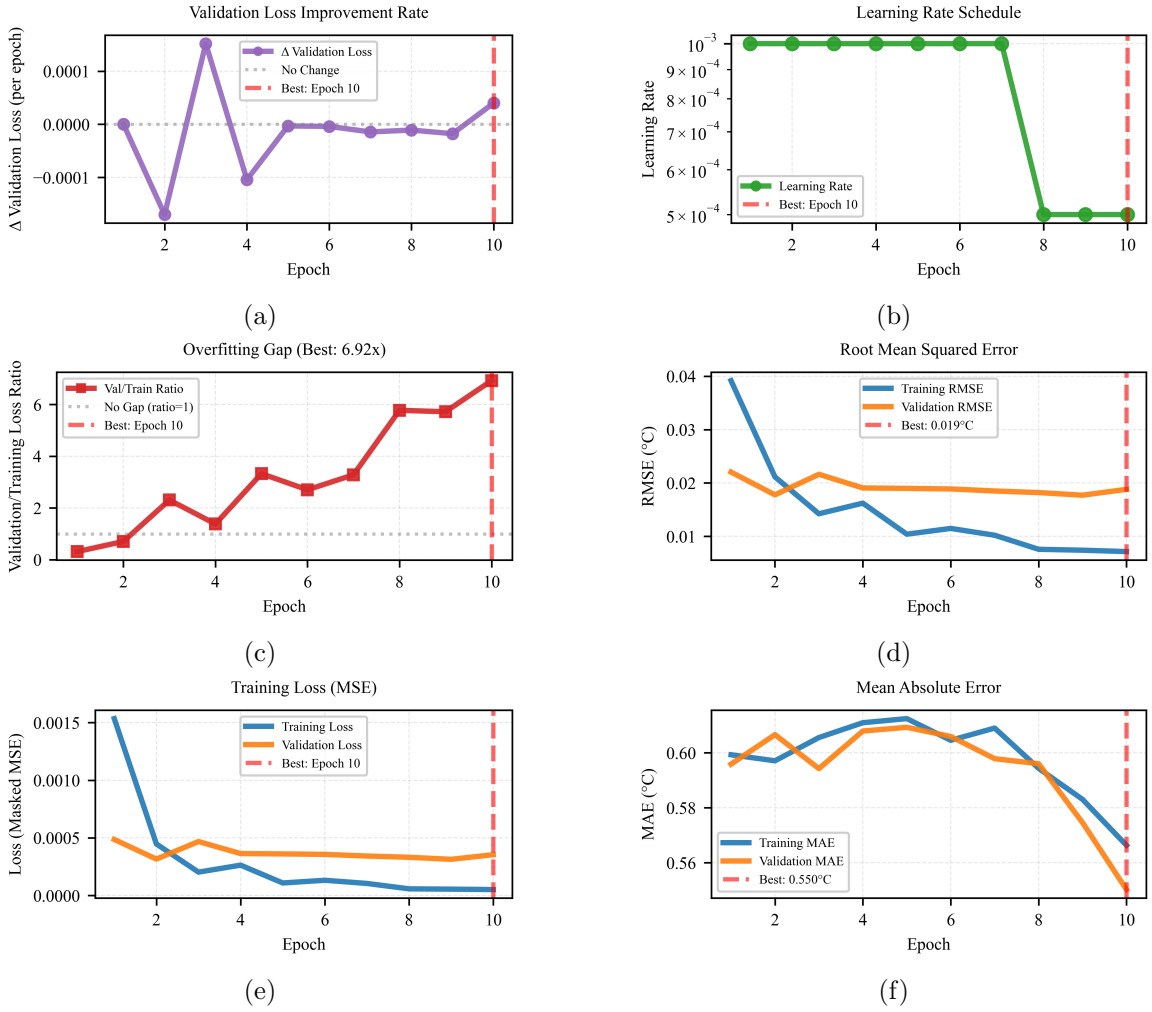


Fig. 4.11: Training curves with DUT-Test excluded from the training set: (a) improvement rate, (b) learning rate schedule, (c) overfitting gap, (d) RMSE, (e) loss, (f) MAE. Red lines indicate the best epoch as determined by lowest validation loss.

performance statistics across all 10 seeds.

The model achieves mean R^2=0.952 and MAE=1.91°C across all seeds, with standard deviation of 0.22°C for MAE and 0.33°C for RMSE, confirming good stability. The best seed (seed 51) achieves R^2=0.963 and MAE=1.63°C, while the worst seed (seed 48) maintains R^2=0.939 and MAE=2.31°C—a performance band indicating low sensitivity to initialization. Unless otherwise noted, all results reported in Sections 4.4.2 and 4.4.3 correspond to the best seed selected by lowest validation loss.

The held-out DUT-Test evaluation across all 10 seeds shows mean R^2=0.942 (std=0.039),

Table 4.7: Initialization Robustness Statistics (Two-Board Model, 10 Seeds).

	MAE (°C)	RMSE (°C)	R^2
Mean	1.91	3.05	0.952
Std	0.22	0.33	0.010
Q1	1.70	2.74	0.942
Median	1.90	3.10	0.951
Q3	2.07	3.37	0.962

mean RMSE=1.69°C (std=0.53°C), and mean MAE=1.21°C (std=0.51°C), confirming consistent cross-design generalization regardless of initialization. Table 4.8 presents the corresponding statistics for the three-board model (DUT-Test included in training), enabling a direct comparison of initialization robustness between the two configurations.

Table 4.8: Initialization Robustness Statistics (Three-Board Model, 10 Seeds).

	MAE (°C)	RMSE (°C)	R^2
Mean	1.91	2.99	0.953
Std	0.24	0.41	0.012
Q1	1.78	2.58	0.943
Median	1.85	3.26	0.945
Q3	1.95	3.32	0.966

The three-board model achieves comparable mean performance (R^2=0.953, MAE=1.91°C) to the two-board model, with the best seed (seed 45) reaching R^2=0.969 and MAE=1.53°C. Including DUT-Test in the training set improves the best-seed overall R^2 from 0.963 to 0.969 and reduces RMSE from 2.70°C to 2.45°C, while the initialization robustness variability remains similar (R^2 std of 0.012 vs 0.010). Fig. 4.12 shows the error distributions and metric spread across all 10 seeds for the two-board model. The tight clustering of R^2, MAE, and RMSE across seeds confirms that model performance is not sensitive to random weight initialization, a prerequisite for reproducible deployment. Despite never observing DUT-Test during training, the model achieves consistent cross-design generalization across initializations, with R^2 ranging from 0.851 to 0.978. Fig. 4.13 presents the corresponding

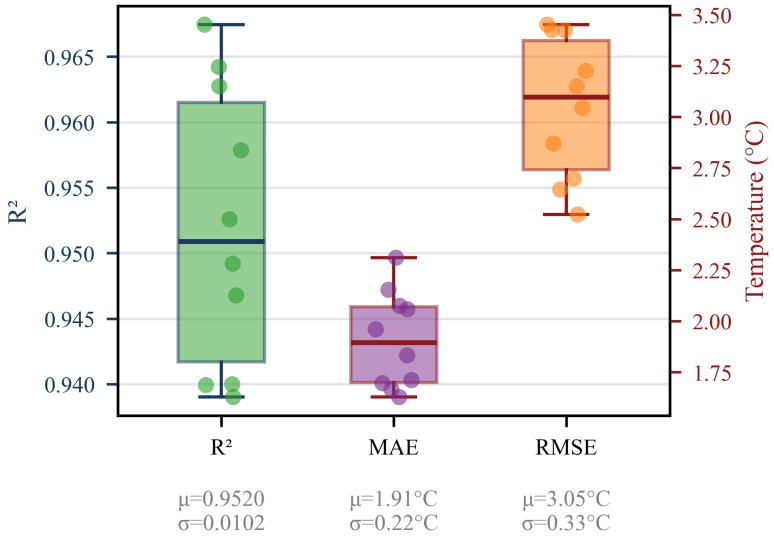


Fig. 4.12: Error distributions and metrics across 10 random seeds for the two-board model. The narrow standard deviation across seeds for R^2, MAE, and RMSE confirms that model performance is not sensitive to random weight initialization.

error distributions for the three-board model.

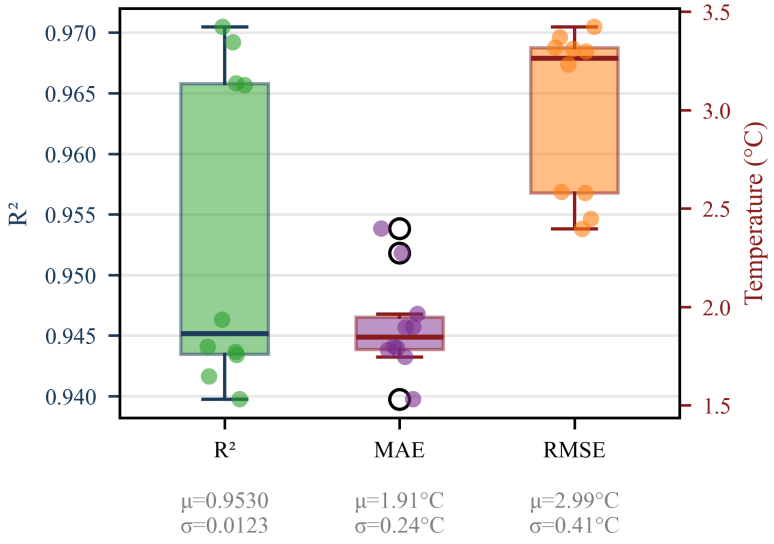


Fig. 4.13: Error distributions and metrics across 10 random seeds for the three-board model (DUT-Test included in training). The similar spread to the two-board model confirms that adding a third board does not destabilize training.

Validation components span diverse types including resistors, inductors, capacitors, and driver ICs. This component diversity confirms that the model learns generalizable spatial thermal coupling patterns rather than component-type-specific behaviors, supporting the cross-design generalization assessment in Section 4.5.

4.5 Generalization and Held-Out Evaluation

Cross-design generalization is the property that distinguishes a practical thermal screening tool from a design-specific regression model. In iterative research programs with evolving design requirements—where inverter boards, compensation network boards, and protection circuitry undergo concurrent development—a model that transfers learned thermal patterns across designs enables engineers to screen proposed layouts before committing to full FEM analysis or physical prototyping. Without cross-design generalization, each new board requires its own training dataset, its own model calibration, and its own validation cycle—reintroducing the per-design bottleneck that makes thermal iteration impractical under concurrent development timelines. A generalizable model eliminates this bottleneck by amortizing training cost across designs, transforming the ML model from a research result into a practical design tool within the thermally informed framework developed in this thesis.

The model trained on mixed DUT-1 and DUT-2 data demonstrates this capability: it predicts temperatures on held-out components from both designs without design-specific retraining. DUT-1 (TABSRC interface) and DUT-2 (Load Shedding board) represent distinct PCB topologies with different component layouts, power densities, and thermal coupling patterns. Mixed-component validation across these designs confirms that the model captures generalizable thermal coupling physics rather than design-specific spatial memorization.

Four architectural choices are central to this generalization capability. First, the z-score normalization (4.1) removes the board-level absolute temperature from the encoder input, forcing the convolutional pathway to learn spatial patterns that are invariant to which board produced the frame. Second, FiLM conditioning at all four decoder levels

injects board-level context through the thermal state vector, allowing the decoder to reconstruct board-appropriate predictions from domain-agnostic spatial features. Third, domain augmentation (Section 4.3.5) expands the two discrete training domain points into a continuous cloud, forcing the FiLM projections to learn smooth mappings that interpolate and extrapolate across the domain parameter space rather than memorizing a two-class step function. Fourth, spatial data augmentation (Section 4.3.4) mirrors each board along three axes, quadrupling the number of unique spatial layouts and preventing the encoder from associating fixed pixel positions with specific component identities.

4.6 Feature Importance Analysis

To determine which of the model's input features contribute most to prediction accuracy, permutation feature importance [65] is computed separately for each board. For each feature, its values are randomly shuffled across samples (preserving the marginal distribution while destroying the correlation with the target), and the model is re-evaluated. The change in RMSE (ΔRMSE) quantifies each feature's contribution: large positive ΔRMSE indicates a critical feature, near-zero indicates a neutral feature, and negative ΔRMSE indicates a feature whose removal improves predictions. Two separate feature importance analyses are conducted: one using the two-board model (seed 51) and one using the three-board model (seed 45), both evaluated on all three boards. Comparing the per-board importance profiles between the two models reveals how including additional training boards changes which features the model learns to exploit.

4.6.1 Two-Board Model Feature Importance (DUT-Test Held Out)

Table 4.9 presents the permutation feature importance results for all three boards using the two-board model (seed 51), sorted by $|\Delta$RMSE$|$. The baseline model achieves R^2=0.956 and RMSE=2.17°C on DUT-1, R^2=0.312 and RMSE=1.66°C on DUT-2, and R^2=0.949 and RMSE=1.65°C on the held-out DUT-Test before any feature permutation.

Channel 7 (log-normalized time) is the only feature classified as critical: shuffling it degrades RMSE by +10.84°C on DUT-1 and +7.76°C on the held-out DUT-Test, effectively

Table 4.9: Permutation Feature Importance by Board (Two-Board Model, Seed 51).

Feature	**ΔRMSE (°C)**			**Class**
	DUT-1	**DUT-2**	**DUT-Test**	
t_{norm} log-time	+10.84	+0.30	+7.76	Critical
Z-score temperature	+0.03	+0.13	+0.03	Neutral
FiLM[0] t_{norm}	−0.05	−0.04	−0.03	Neutral
$\dot{T}$ heating rate	<0.01	<0.01	<0.01	Neutral

lowering the model's predictive capability on both boards. The impact on DUT-2 is notably smaller (+0.30°C), reflecting its narrower thermal operating range where temporal position carries less discriminative weight. This result is consistent with the model architecture: Channel 7 encodes the temporal position that drives FiLM conditioning throughout the decoder, and it is the only spatial channel that changes monotonically across the thermal transient.

All other features contribute ΔRMSE < 0.13°C individually, classifying as neutral. The FLIR temperature channel (Ch0) shows its largest contribution on DUT-2 (ΔRMSE=+0.13°C), where the absolute temperature pattern provides relatively more information than the temporal trajectory. The FiLM time normalization feature (FiLM[0]) shows a modest *negative* ΔRMSE on all three boards (−0.03 to −0.05°C), indicating that shuffling it slightly improves predictions—suggesting partial redundancy with Channel 7 in the two-board configuration. Channel 5 ($\dot{T}$) contributes <0.01°C on all boards, indicating the model has not yet learned to exploit heating rate information with only two training boards. Fig. 4.14a visualizes the per-board importance magnitudes for this configuration, showing the dominant Channel 7 bar relative to all other features. The contrast between the negative FiLM[0] and negligible Channel 5 values here and their positive contributions in the three-board model (Table 4.10) confirms how additional training diversity changes the features the model relies on.

4.6.2 Three-Board Model Feature Importance (DUT-Test Included)

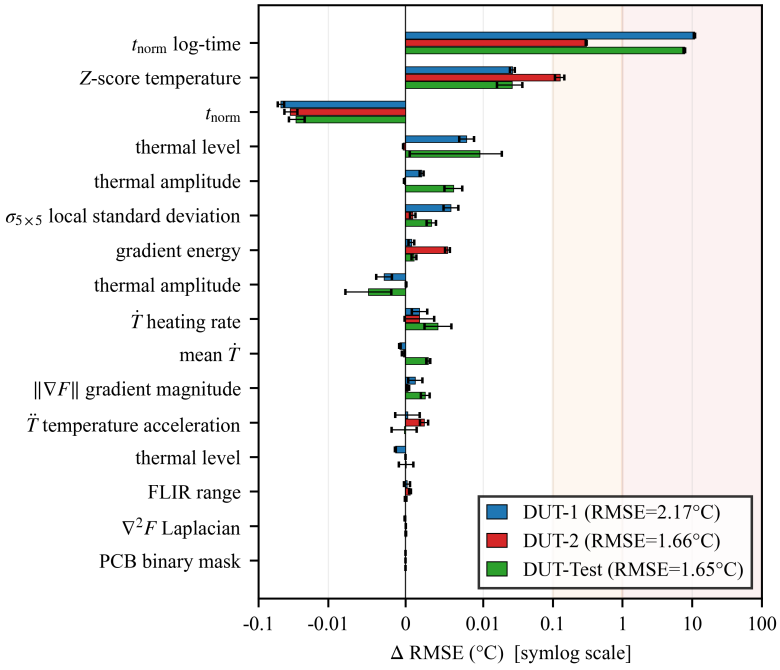
When DUT-Test is included in the training set, the feature importance landscape shifts. Table 4.10 presents the results for all three boards using the three-board model (seed 45), using the same per-board format as Table 4.9 to enable direct comparison. The baseline model achieves R^2=0.968 and RMSE=1.83°C on DUT-1, R^2=0.776 and RMSE=0.95°C on DUT-2, and R^2=0.981 and RMSE=1.00°C on the held-out DUT-Test before any feature permutation.

Table 4.10: Permutation Feature Importance by Board (Three-Board Model, Seed 45).

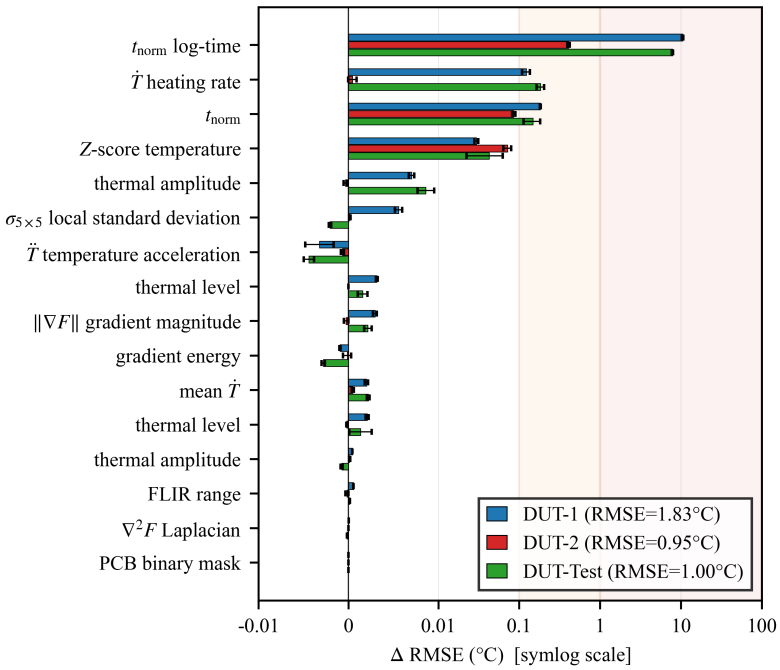
Feature	**ΔRMSE (°C)**			**Class**
	DUT-1	**DUT-2**	**DUT-Test**	
t_{norm} log-time	+10.51	+0.41	+7.84	Critical
$\dot{T}$ heating rate	+0.12	<0.01	+0.18	Helpful
FiLM[0] t_{norm}	+0.18	+0.09	+0.15	Helpful
Z-score temperature	+0.03	+0.07	+0.04	Neutral

With DUT-Test included in training, two features emerge as helpful beyond the critical Channel 7: FiLM[0] (time normalization conditioning, ΔRMSE=+0.15–0.18°C) and Channel 5 ($\dot{T}$, ΔRMSE=+0.12–0.18°C on DUT-1 and DUT-Test). The temporal derivative Channel 5 shows meaningful importance for the first time, suggesting that with three boards providing greater thermal diversity, the model learns to exploit heating rate information that was previously redundant with the time broadcast.

Fig. 4.14b visualizes the per-board importance magnitudes for the three-board configuration. Comparing it with Fig. 4.14a, the reduction in Channel 7 bar magnitude and the emergence of positive contributions from FiLM[0] and Channel 5 confirm that the model distributes reliance across more features as training diversity increases. This shift indicates that additional PCB designs enable the model to extract thermal patterns beyond the monotonic temporal position signal, which is expected to improve prediction robustness as more boards become available for training.



(a)



(b)

Fig. 4.14: Combined permutation feature importance across the two-board (a) and three-board (b) thermal-gate driver training configurations.

4.6.3 Feature Importance Summary

The Channel 7 dominance on DUT-2 (LoadShedding) is notably lower (ΔRMSE=+0.30–0.41°C) compared to DUT-1 and DUT-Test (+7–11°C). This reflects DUT-2's narrower thermal operating range: with a smaller temperature swing during testing, the temporal position carries less discriminative power. The FLIR temperature channel (Ch0) shows its largest contribution on DUT-2 (ΔRMSE=+0.07–0.13°C), where the absolute temperature pattern provides relatively more information than the temporal trajectory.

These results confirm that the model's prediction accuracy relies primarily on the temporal conditioning pathway (Channel 7 $\rightarrow$ FiLM) rather than on the spatial feature channels. The emergence of Channel 5 ($\dot{T}$) as a helpful feature in the three-board model suggests that adding training boards enables the model to extract richer temporal information beyond the monotonic time signal. The model's architecture—with its multi-scale encoder, skip connections, and FiLM conditioning at all decoder levels—is designed to exploit additional spatial and temporal diversity as more PCB designs become available for training.

The results presented in this chapter—cross-design generalization across two and three training boards, unseen-board prediction on a held-out third design achieving R^2=0.949, and feature importance analysis identifying the temporal conditioning pathway as the dominant contributor—demonstrate that the ML framework can provide the rapid thermal screening capability identified as the missing piece in the methods comparison of Chapter 2. Chapter 5 synthesizes these results with the simulation and experimental contributions into a practical design workflow, discusses the scope and limitations of the current framework, and outlines the path toward prototype-free thermal prediction.

CHAPTER 5

SUMMARY AND FUTURE WORK

This thesis has examined the thermal implications of embedding dynamic wireless power transfer (DWPT) power electronics within roadway environments. Building on prior research [13], which documented the thermal failure of an embedded DWPT gate driver board and established FEM-validated thermal modeling for embedded compensation capacitors (Chapter 2, Section 2.3), this work develops a practical methodology for thermal design and rapid screening of embedded modules spanning analytical estimation, simulation, and machine learning.

5.1 Summary of Contributions

The first contribution is a variable-speed PLECS simulation framework for multi-pad DWPT systems. The simulation captures time-varying coupling between transmitter and receiver coils as a function of vehicle speed and position, enabling system-level loss prediction across realistic driving scenarios. This framework replaces the single-speed steady-state analysis common in prior work with a dynamic model that reflects the pulsed, variable-coupling nature of roadway DWPT operation. By quantifying per-component losses under realistic pulsed utilization, this framework reveals the overrating margin between rated component envelopes and actual thermal loading—establishing that embedded systems *can* be derated and defining the design space for properly rated operation. This gap represents the first barrier to moving beyond conservative overrating.

The second contribution is a systematic comparison of thermal analysis methods for embedded power electronics. Analytical loss models using the improved generalized Steinmetz equation (IGSE) for inductor core loss and dielectric loss calculation for capacitors provide fast estimation of component-level power dissipation, which is the starting point for thermal analysis on a per-component basis. RC thermal circuit analysis was applied to a

resistor load assembly to assess the applicability of lumped-parameter models to embedded geometries, revealing that the uncertain thermal interface properties of embedding media limit analytical accuracy. This comparison of methods identifies the gap that motivates the experimental and data-driven approaches in subsequent chapters. By quantifying the accuracy-speed tradeoff, this contribution establishes that no single existing approach provides the fidelity and throughput needed for iterative thermal design of embedded DWPT electronics.

The third contribution is an automated thermal validation framework spanning SROI file generation from Altium PCB data and FLIR thermal imaging with thermocouple calibration. The SROI pipeline eliminates manual ROI placement for thermal imaging, enabling reproducible component-level temperature extraction across multiple PCB designs. Dual-environment testing in air and sand provides progressive validation, first in open air and then under thermal conditions representative of roadway embedment, and the thermocouple-calibrated FLIR measurement framework generates the training datasets required by the FiLM-conditioned U-Net. By automating data acquisition and component-level temperature extraction, this pipeline removes the scalability bottleneck that previously limited thermal characterization to one board at a time—enabling the multi-board training data that cross-design ML generalization requires.

The fourth contribution is a FiLM-conditioned U-Net thermal prediction model for component-level temperature estimation from FLIR thermal images. Two training configurations are evaluated: a three-board model in which DUT-Test is included in training with an 80/20 component-level split, and a two-board model in which DUT-Test is excluded entirely and evaluated as an unseen board. The two-board model achieves R^2=0.963 and MAE=1.63°C on mixed-PCB validation across DUT-1 and DUT-2, and R^2=0.949 with MAE=1.24°C on the unseen DUT-Test held-out board, demonstrating that a single model trained on mixed-board data can predict component-level temperatures on both seen and unseen designs. Initialization robustness testing across 10 random initializations confirms low sensitivity to weight initialization: mean R^2=0.952 and MAE=1.91°C with standard

deviations of 0.22°C for MAE and 0.33°C for RMSE. Several design choices contribute to this result: residual learning decouples prediction from the absolute temperature scale, the 10-channel input separates spatial structure, temporal dynamics, and domain identity into distinct representations, and domain augmentation with a differential learning rate addresses the challenge of training on only two boards. Feature importance analysis reveals that cross-design generalization relies predominantly on the temporal conditioning pathway (Channel 7 and FiLM), with spatial feature channels contributing marginally in the current two-board training set.

This approach addresses a practical gap in embedded DWPT thermal design: traditional FEM requires hours of setup per design iteration, and dense thermocouple instrumentation adds hardware cost with limited spatial coverage. The ML model produces component-level thermal predictions from a single FLIR frame and the thermal state vector, enabling fast thermal assessment during design iteration. The cross-design generalization result is particularly relevant in iterative research programs with evolving design requirements, where several PCB designs undergo development in parallel, and the DUT-Test unseen-board evaluation demonstrates prediction errors within 3°C MAE—sufficient for early-stage thermal screening.

5.2 Scope and Limitations

The framework presented in this thesis demonstrates the viability of ML-based thermal prediction for embedded DWPT power electronics, but the current implementation operates within specific constraints that define the scope of its validated applicability. The training dataset comprises two PCB designs (DUT-1 and DUT-2) with DUT-Test evaluated as a held-out board. Two training boards provide sufficient spatial diversity to demonstrate cross-design generalization but constrain the breadth of component types, board sizes, and power densities the model has encountered. The framework is designed to scale: each additional board tested through the Chapter 3 pipeline adds training data without architectural modification, and the feature importance results suggest that spatial channel contributions will strengthen as the training set grows.

The current model requires a FLIR thermal camera image as its primary spatial input, limiting deployment to laboratory environments with IR camera access. Similarly, ground-truth training labels rely on thermocouple instrumentation of the test boards. These requirements are inherent to the supervised learning approach and are addressed by the dual split-encoder architecture proposed in Section 5.4, which targets prototype-free prediction from design data alone. The SROI pipeline that generates training data expects PCB component locations in the Altium Pick&Place fabrication output format. Boards designed in other EDA tools would require format conversion or pipeline adaptation to produce equivalent coordinate data.

Embedded thermal validation uses sand as the embedding medium. Sand provides a cost-effective surrogate for roadway thermal boundary conditions and matches the thermal conductivity range of common pavement materials, but testing in epoxy, concrete, or composite embedments has not been performed. Different embedding media will produce different steady-state temperature distributions and transient dynamics.

The model's operating-point coverage is limited to the power levels represented in the training data. Extrapolation to power levels substantially outside the training range—particularly at higher power densities—has not been validated and would benefit from additional testing at expanded operating conditions. Each of these constraints represents a bounded scope rather than a fundamental limitation of the framework. The architecture, pipeline, and methodology are designed to accommodate expanded data, additional board designs, and alternative testing conditions as they become available.

A related limitation concerns operating point coverage. Each PCB was tested and training data collected at a single operating point—a specific power level, switching condition, and thermal boundary configuration. The model has not been validated for generalization to different operating points of the same PCB design, such as partial load, varying switching frequency, or different utilization duty cycles. For DWPT systems, pulsed utilization means that component temperatures during service depend on duty cycle and vehicle speed. Thermal behavior at 30% utilization may differ substantially from the training condition.

Engineers applying this framework to boards with variable operating conditions should treat the training operating point as a specific characterization point until multi-condition validation is completed.

5.3 Practical Design Recommendations

The methods developed in Chapters 2–4 combine into a staged workflow for evaluating and improving the thermal performance of embedded DWPT power electronics—particularly for systems not originally designed for the embedded thermal environment. The following steps describe the recommended design flow, from initial loss characterization through iterative ML-assisted layout optimization.

1. **Characterize component losses under pulsed utilization.** Use the variable-speed simulation framework (Chapter 2) to compute per-component losses across the expected range of vehicle speeds and pad configurations. Compare the pulsed loss profiles against continuous full-load ratings to identify the overrating margin—the gap between the rated component envelope and the actual thermal loading under roadway utilization. This step establishes that the system *can* be derated and defines the design space for properly rated operation.

2. **Prototype the PCB and test in air and sand environments.** Fabricate the PCB with thermocouple instrumentation on critical components and test using the automated SROI pipeline and FLIR/thermocouple methodology (Chapter 3). Air testing verifies baseline thermal behavior; sand testing provides embedded thermal boundary conditions representative of roadway deployment. The SROI pipeline automates ROI placement and component-level temperature extraction, producing the training datasets that feed the ML model.

3. **Apply the trained ML model for rapid thermal screening.** Use the FiLM-conditioned U-Net (Chapter 4) to predict component-level temperatures across the full

board from FLIR image inputs. The model identifies thermally at-risk components—those approaching or exceeding their rated temperature limits—without requiring per-design FEM setup. For boards represented in the training set, predictions achieve sub-2°C MAE; for unseen designs, held-out predictions provide screening-level accuracy sufficient to flag critical components.

4. **Iterate the PCB layout based on thermal screening results.** Relocate hot-spot components, adjust copper pours and thermal via patterns, and modify component spacing to improve thermal margins. Re-screen each layout revision with the ML model to verify that modifications produce the intended thermal improvement. This iteration cycle operates on the timescale of minutes per evaluation rather than the days to weeks required for FEM re-setup, enabling fast design convergence.

5. **Validate critical components with targeted FEM or physical testing.** For components identified as thermally marginal during ML screening, perform targeted ANSYS Icepak analysis or additional physical testing to confirm thermal margins at full geometric fidelity. This step provides final design confidence for the components that matter most, while the ML screening in step 3 ensures that only the critical few—rather than the entire board—require detailed validation.

This workflow exploits the complementary strengths of each method: simulation provides system-level loss context, the automated pipeline produces training data at scale, and the ML model enables fast iteration. The result is a thermal design process in which component ratings reflect actual embedded operating conditions rather than worst-case continuous assumptions.

Design speed and reliability are complementary objectives in this framework, not competing ones. Faster iteration is one benefit, but it is not the primary motivation. The central reliability benefit is thermal margin quantification: the framework enables engineers to determine the actual margin for each component before a board is committed to roadway deployment. Conservative overrating alone cannot provide this—it substitutes a

cost penalty for a safety margin without quantifying that margin. A board may still fail if the overrating assumption was incorrect for the specific embedded boundary condition. Accurate thermal prediction converts an unknown margin into a measured one, enabling designs that are reliably validated and appropriately sized. The speed benefit compounds this: one thermal evaluation takes minutes rather than weeks, allowing engineers to explore more design variants and converge on a reliable design before fabrication.

5.4 Future Work

Expanded PCB testing represents the most immediate path to improving the ML thermal prediction model. Undergraduate and graduate students at USU are currently working towards testing additional PCB designs from ongoing DWPT research, and each new design adds training data to the FiLM model. Increasing the number and diversity of PCB topologies in the training set will improve prediction accuracy and extend cross-design generalization across a wider range of power electronics architectures.

Physics-based thermal modeling offers a complementary approach to the data-driven ML methods presented in Chapter 4. Application of thermo-dynamics modeling techniques—such as model-order-reduced thermal networks derived from FEM—could be combined with the U-Net predictions to provide physically grounded constraints on the learned model. This hybrid approach would leverage the speed of ML inference while maintaining the interpretability and extrapolation capability of physics-based models.

Removing the FLIR camera requirement is a priority for practical deployment of the ML prediction framework. The current model requires a thermal image as input, which limits its use to laboratory environments with IR camera access. Future work aims to replace the spatial FLIR input with sparse thermocouple measurements as the primary input, enabling thermal field reconstruction from a small number of contact sensors. This would extend the prediction capability to field-deployed systems where IR imaging is impractical.

A longer-term architectural direction is a dual split-encoder design that removes the prototype requirement entirely, as shown in Fig. 5.1. One encoder branch accepts a power dissipation map constructed from component datasheets, operating-point calculations, and a

lookup table built from initial FLIR steady-state tests on representative PCBs with relevant components. This branch learns to predict the expected thermal transient within air for each component, producing a proxy thermal image analogous to the air-test FLIR frames used in the current pipeline.

The second encoder branch processes a 3D representation of the PCB copper layers extracted from fabrication outputs (Gerber or ODB++ files), masked to indicate the presence or absence of copper at each spatial location. Both encoder branches feed into a shared FiLM-conditioned decoder—architecturally similar to the current model—that fuses the power-dissipation and copper-geometry features into a proxy thermal image. This proxy image then serves as the input to the existing trained model, which predicts component-level temperatures within an embedded material. By decoupling the model from physical FLIR measurements, this architecture would enable thermal screening during the design phase, before a prototype exists, substantially reducing design iteration cost.

5.4.1 Operating Point Parametrization

The current model implicitly encodes a single operating point in each training board's thermal statistics. Extending the framework to predict across multiple operating conditions would require the operating point as an explicit model input—for example, a normalized power-level scalar or duty-cycle vector appended to the thermal state vector. Training across multiple operating points per board, with domain augmentation spanning the utilization space, would allow a single model to cover the full range of conditions relevant to pulsed DWPT operation. Physics-informed normalization of the input channels by predicted steady-state temperature could further reduce distribution shift between conditions, improving generalization without proportional increases in training data requirements.

5.4.2 Parameterized RC Thermal Model Calibration

The RC lumped-parameter thermal model presented in Chapter 2 was constructed using material properties from component datasheets and published literature values; no calibration against the measured thermocouple records was performed. A structured param-

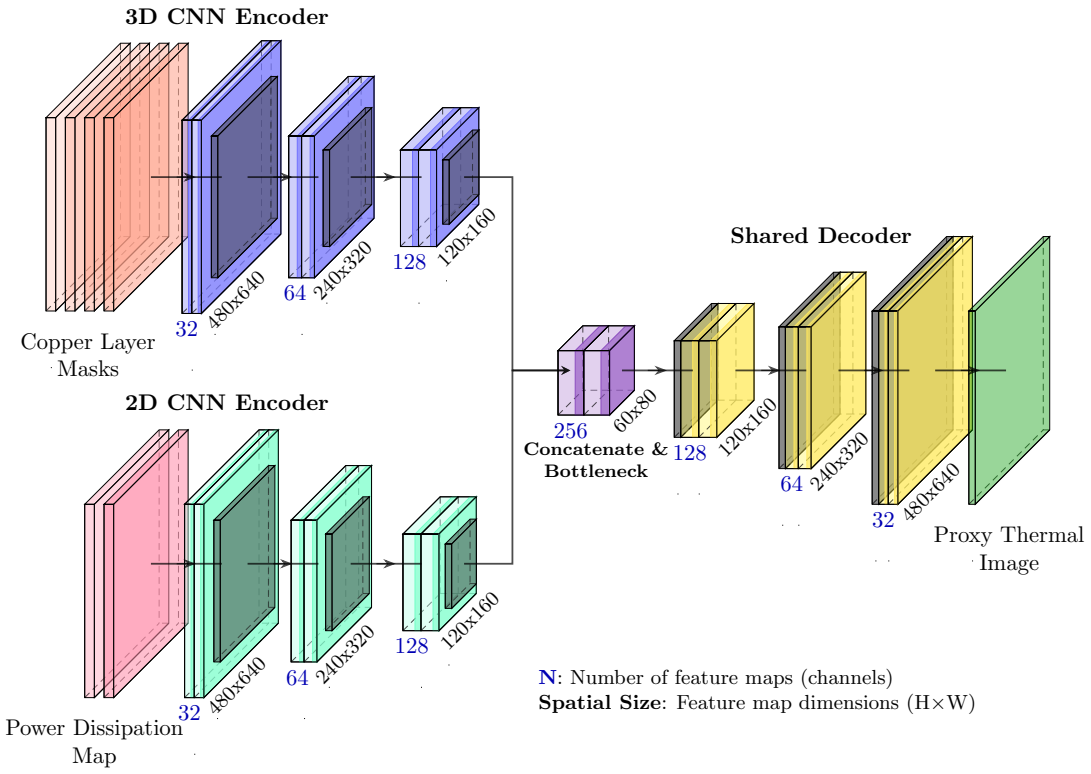


Fig. 5.1: Proposed dual split-encoder architecture for prototype-free thermal prediction. The power dissipation encoder and copper geometry encoder feed into a shared FiLM-conditioned decoder to produce proxy thermal images.

eter sweep—varying the thermal interface paste conductivity, the convective heat transfer coefficient at the resistor and heatsink surfaces, and the sand thermal conductivity and specific heat for the buried test configuration—against the experimental thermocouple records would identify the best-fit parameter set for each test environment. This calibration would quantify the model's sensitivity to each input and establish confidence bounds on temperature predictions derived from the RC circuit framework.

5.4.3 Broader Application Domains

The thermal prediction framework is not specific to roadway DWPT; the SROI pipeline, FLIR imaging protocol, and FiLM-conditioned model architecture apply to any application where thermal imaging and thermocouple ground-truth can be acquired during a qualification test. Three application domains are particularly direct extensions. Data center power

electronics—server power supply units, voltage regulators, and bus converters operating in both air-cooled and liquid-cooled rack environments—present an analogous problem: dense PCBs with heterogeneous components, difficult-to-instrument hot spots, and tight thermal budgets. Aerospace and space power converters in conduction-cooled channel assemblies present an even more constrained cooling environment, where the consequence of thermal design errors is higher and FEM per-design turnaround is equally impractical under concurrent development schedules. For each of these domains, extending the framework requires new training boards tested in representative thermal environments; the pipeline and model architecture require no modification.

5.4.4 Integration with Physics-of-Failure Reliability Analysis

The thermal profiles measured during framework validation provide a component-level characterization of actual temperatures under representative operating conditions. These measured profiles could serve as inputs to reliability analysis tools such as Ansys Sherlock for newly designed PCBs that incorporate the same components. Rather than relying on datasheet worst-case assumptions, a designer could apply the measured thermal profile of each component as a starting-point boundary condition in a Sherlock analysis. This provides a field-validated thermal baseline, improving the accuracy of component-level lifetime and fatigue predictions for new PCB designs before physical testing is conducted.

5.4.5 Real-Time Monitoring and Sparse-Sensor Augmentation

The model's fast inference enables two distinct real-time deployment scenarios. In the first, a FLIR camera streams thermal images of the DUT during laboratory qualification testing, and the model produces component-level temperature predictions in near-real-time. A technician can monitor thermal hot spots across all components simultaneously without waiting for post-processing, reducing the risk of missing a thermal failure during qualification. In the second scenario—suited to PCBs deployed in the field—a small number of thermocouples placed at the thermally critical points of the PCB replace the FLIR camera as the primary data source. These measurements would be incorporated into the thermal

state vector as additional conditioning inputs, allowing the model to produce a full spatial prediction from point measurements alone. This architecture reduces instrumentation cost for in-service health monitoring, where a full FLIR camera is impractical but a few thermocouples embedded alongside the PCB are feasible.

5.4.6 Conservative Prediction Bias for Reliability-Oriented Design

The current model is trained with a symmetric loss function that treats over-prediction and under-prediction equivalently. This produces near-zero systematic bias across the training distribution. For reliability-critical applications, an engineer may prefer predictions that systematically err toward higher temperatures, ensuring that thermal margins are never underestimated. Two approaches are proposed. The simpler applies a user-selectable positive offset to all component temperature predictions: $T_{\text{conservative}} = \hat{T} + k\,\sigma_{\text{MAE}}$. Here k is a conservatism factor based on the required reliability confidence level, and σ_{MAE} is the per-board prediction uncertainty from the initialization robustness evaluation.

This offset can be applied without retraining, and the conservatism level is adjustable independently for each design decision. A more fundamental approach would retrain the model with an asymmetric loss function that penalizes under-predictions more heavily than over-predictions. Candidate formulations include a modified Huber loss with asymmetric thresholds or a quantile regression loss targeting the 90th percentile of the prediction distribution. This embeds conservatism directly into the model's learned behavior, producing predictions that serve as a reliable upper bound rather than a best-estimate mean.

The current model also has practical limitations that constrain its prediction accuracy. The dominance of Channel 7 (log-normalized time) in the feature importance analysis suggests that the spatial encoder has not yet learned to exploit the spatial feature channels fully, likely because two training boards provide insufficient spatial diversity. Expanding the training set with additional PCB designs at various operating points would strengthen the spatial channels' contribution and improve unseen-board generalization. The model also requires operation at specific power levels represented in the training data; extending coverage to a wider range of operating conditions is needed for robust deployment across

the range of power levels encountered in DWPT systems.

5.5 Conclusion

Full-scale roadway validation should extend the experimental results to longer roadway sections and multi-module configurations under realistic conditions. These studies should incorporate drive-cycle power profiles reflecting actual traffic patterns, pavement wear and degradation effects on thermal boundary conditions, and weather exposure including seasonal temperature variation and moisture infiltration. The combination of expanded ML training data, physics-informed modeling, and field-scale testing will advance the embedded DWPT framework toward deployment-ready thermal design tools. With additional training boards and the proposed prototype-free prediction architecture, this framework can deliver the thermal design confidence that first-generation embedded DWPT systems have lacked—enabling component ratings sized to actual embedded conditions rather than continuous full-load assumptions. By reducing the time and cost of thermal evaluation from weeks of per-design engineering effort to minutes of ML screening, this work contributes to making widespread roadway-embedded wireless charging infrastructure economically and practically feasible.

REFERENCES

[1] C. Pay, "Ev charging stations vs. gas stations: Comparing density in the u.s." Online, 2024, available: https://coastpay.com/ev-charging-stations-vs-gas-stations-comparing-density-in-the-u-s/, Accessed: Aug. 22, 2025.

[2] U. B. of Transportation Statistics, "Energy consumption by mode of transportation," Online, 2023, available: https://www.bts.gov/content/energy-consumption-mode-transportation, Accessed: Aug. 27, 2025.

[3] T. E. AG, "Ferrites and accessories — siferrit material n87," TDK Electronics AG, Munich, Germany, Datasheet N87, Jun. 2025, accessed: Aug. 27, 2025. [Online]. Available: https://product.tdk.com/en/catalog/datasheets/ferrite_N87.pdf

[4] Z. Imran, O. A. Moncada, C. Vickers, D. Aliprantis, J. E. Haddock, S. D. Pekarek, and A. Brovont, "Full-scale thermal and mechanical testing of pavement with embedded high-power dynamic wireless power transfer coil," *IEEE Transactions on Transportation Electrification*, vol. 11, no. 3, pp. 7615–7625, 2025.

[5] J. B. Larsen, A. Kamineni, N. Roberts, M. Halling, P. Vaikasi, and A. Barnes, "Test platform to evaluate pavement embedded wireless charging pads," in *2022 Wireless Power Week (WPW)*, 2022, pp. 861–866.

[6] B. J. Varghese, "Roadway-embedded transmitters and multi-pad receivers for high power dynamic wireless power transfer," Ph.D. dissertation, Utah State University, Logan, UT, 2021, all Graduate Theses and Dissertations, Paper 8232. [Online]. Available: https://digitalcommons.usu.edu/etd/8232

[7] I. D. of Transportation. Dynamic wireless power transfer. Indiana Department of Transportation. [Online]. Available: https://www.in.gov/indot/emerging-mobility/dynamic-wireless-power-transfer/

[8] I. E. Agency, "Electric car sales, 2012–2024," https://www.iea.org/data-and-statistics/charts/electric-car-sales-2012-2024, International Energy Agency, Paris, France, 2024, licence: CC BY 4.0; Accessed: Jul. 29, 2025.

[9] J. Kelly, A. Elgowainy, R. Isaac *et al.*, "Cradle-to-grave lifecycle analysis of u.s. light-duty vehicle-fuel pathways: A greenhouse gas emissions and economic assessment of current (2020) and future (2030–2035) technologies," Argonne National Laboratory, Lemont, IL, Technical Report ANL-22/27 Rev. 1, Nov. 2023, accessed: Jul. 29, 2025. [Online]. Available: https://publications.anl.gov/anl-22-27rev1

[10] U.S. Department of Transportation, "Charging electric vehicles," Alternative Fuels Data Center, 2024, accessed: 2025. [Online]. Available: https://afdc.energy.gov/fuels/electricity-charging

[11] ENRX. Electrifying the future of transportation: A collaborative journey toward sustainable mobility. ENRX. [Online]. Available: https://www.enrx.com/en/Company/Media/News/ASPIRE-Electric-Roadway-test-track---Electrifying-the-future-of-transportation

[12] Electreon. Aspire demonstration. Electreon. [Online]. Available: https://electreon.com/projects/aspire-demonstration-utah

[13] C. R. Sabin, "Thermal Management of Roadway-Embedded Power Electronics for Electric Vehicle Dynamic Wireless Charging Systems," Master's thesis, Utah State University, 2024. [Online]. Available: https://digitalcommons.usu.edu/etd2023/311

[14] "Wireless power transfer for light-duty plug-in electric vehicles and alignment methodology," SAE International, aug 2024, revised Aug. 2024; originally issued May 2016. [Online]. Available: https://saemobilus.sae.org/standards/j2954_202408-wireless-power-transfer-light-duty-plug-electric-vehicles-alignment-methodology

[15] J. Pries, V. P. N. Galigekere, O. C. Onar, and G.-J. Su, "A 50-kw three-phase wireless power transfer system using bipolar windings and series resonant networks for rotating magnetic fields," *IEEE Transactions on Power Electronics*, vol. 35, no. 5, pp. 4500–4517, 2020.

[16] S. Li, W. Li, J. Deng, T. D. Nguyen, and C. C. Mi, "A double-sided lcc compensation network and its tuning method for wireless power transfer," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 6, pp. 2261–2273, 2015.

[17] A. C. Bagchi, A. Kamineni, R. A. Zane, and R. Carlson, "Review and comparative analysis of topologies and control methods in dynamic wireless charging of electric vehicles," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 4, pp. 4947–4962, 2021.

[18] G. Reilly, A. Brovont, and T. E. Nantung. (2024) Indot dynamic wireless power transfer (dwpt) in-pavement ev charging pilot project. Purdue Road School. [Online]. Available: https://docs.lib.purdue.edu/roadschool/2024/presentations/53/

[19] B. J. Varghese, A. Kamineni, N. Roberts, M. Halling, D. J. Thrimawithana, and R. A. Zane, "Design considerations for 50 kw dynamic wireless charging with concrete-embedded coils," in *2020 IEEE PELS Workshop on Emerging Technologies: Wireless Power Transfer (WoW)*, 2020, pp. 40–44.

[20] B. J. Varghese, "Roadway-embedded transmitters and multi-pad receivers for high power dynamic wireless power transfer," 2021. [Online]. Available: https://doi.org/10.26076/6c1c-3335

[21] T. Dragičević, P. Wheeler, and F. Blaabjerg, "Artificial intelligence aided automated design for reliability of power electronic systems," *IEEE Transactions on Power Electronics*, vol. 34, no. 8, pp. 7161–7171, 2019.

[22] Y. Dou, "An improved prediction model of IGBT junction temperature based on back-propagation neural network and kalman filter," *Complexity*, vol. 2021, p. 5542889, 2021.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[24] R. Usamentiaga, P. Venegas, J. Guerediaga, L. Vega, J. Molleda, and F. G. Bulnes, "Infrared thermography for temperature measurement and non-destructive testing," *Sensors*, vol. 14, no. 7, pp. 12 305–12 348, 2014.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Springer International Publishing, 2015, pp. 234–241.

[26] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators," *Nature Machine Intelligence*, vol. 3, no. 3, pp. 218–229, 2021.

[27] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhatt, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," in *International Conference on Learning Representations (ICLR)*, 2021.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] W. Kirchgässner, O. Wallscheid, and J. Böcker, "Data-driven permanent magnet temperature estimation in synchronous motors with supervised machine learning: A benchmark," *IEEE Transactions on Energy Conversion*, vol. 36, pp. 2059–2067, 2021.

[30] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[31] D. Coenen, H. Oprins, and R. Degraeve, "Benchmarking of machine learning methods for multiscale thermal simulation of integrated circuits," in *IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*, 2022.

[32] E. Gurpinar, M. Mohammad, U. Kavimandan, E. Asa, V. P. Galigekere, B. Ozpineci, S. Mukherjee, L. Tolbert, H. Bai, and Y. Liu, "Failure modes and effects analysis for wireless and extreme fast charging," Oak Ridge National Laboratory and University of Tennessee, Knoxville, Knoxville, TN, USA, Technical Report, 2021.

[33] A. N. Barnes, "Thermal modeling and analysis of roadway embedded wireless power transfer modules," Master's thesis, Utah State University, Logan, Utah, 2020, all Graduate Theses and Dissertations, 7808. [Online]. Available: https://digitalcommons.usu.edu/etd/7808

[34] S. Inoue, S. Kiguthi, J. Newman, T. Goodale, C. R. Teeneti, B. Hesterman, A. Kamineni, and R. A. Zane, "50 kW reflexive tuning networks with low uncoupled transmitter currents for dynamic inductive power transfer systems," *IEEE Open Journal of Power Electronics*, 2024.

[35] S. Inoue, D. Goodrich, S. Saha, R. Nimri, A. Kamineni, and N. S. Flann, "Fast design optimization method utilizing a combination of artificial neural networks and genetic algorithms for dynamic inductive power transfer systems," *IEEE Open Journal of Power Electronics*, vol. 3, pp. 915–929, 2022.

[36] K. Venkatachalam, C. Sullivan, T. Abdallah, and H. Tacca, "Accurate prediction of ferrite core loss with nonsinusoidal waveforms using only steinmetz parameters," in *2002 IEEE Workshop on Computers in Power Electronics, 2002. Proceedings.*, 2002, pp. 36–41.

[37] P. Bradford, A. Zade, S. Gurudiwan, and H. Wang, "A novel thermal modeling analysis for liquid-cooled high-power EV chargers," in *2024 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2024, pp. 6593–6599.

[38] C. P. Steinmetz, "On the law of hysteresis," *Transactions of the American Institute of Electrical Engineers*, vol. IX, no. 1, pp. 1–64, 1892.

[39] W. Yang, Y. Xu, Q. Chen, J. Hu, B. Cui, Z. Yang, B. Zhao, and Q. Sun, "Improved igse core loss calculation method based on the magnetic density distribution characteristics of large-size cores," in *2023 IEEE Sustainable Power and Energy Conference (iSPEC)*, 2023, pp. 1–6.

[40] T. E. AG, "Ferrites and accessories — u 93/76/30 with i 93/28/30 cores (b67345)," TDK Electronics AG, Munich, Germany, Datasheet B67345, Oct. 2024, release 2024-02. Available: https://product.tdk.com/en/catalog/datasheets/ferrite_U937630.pdf, Accessed: Aug. 27, 2025.

[41] TDK Electronics AG, "Ferrite magnetic design tool," Online. Available: https://www.tdk-electronics.tdk.com/en/180490/design-support/design-tools/ferrite-magnetic-design-tool, accessed: Aug. 27, 2025.

[42] A. Bar-Cohen and W. M. Rohsenow, "Thermally optimum spacing of vertical, natural convection cooled, parallel plates," *Journal of Heat Transfer*, vol. 106, no. 1, pp. 116–123, 1984.

[43] S. W. Churchill and H. H. S. Chu, "Correlating equations for laminar and turbulent free convection from a vertical plate," *International Journal of Heat and Mass Transfer*, vol. 18, no. 11, pp. 1323–1329, 1975.

[44] A. Baig, A. Zahid, J. B. Larsen, A. Kamineni, and R. Zane, "Investigation of split lccl tuning network for high power wpt systems," in *2023 IEEE Wireless Power Technology Conference and Expo (WPTCE)*, 2023, pp. 1–6.

[45] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[46] S. Butterworth, "On the theory of filter amplifiers," *Wireless Engineer*, vol. 7, pp. 536–541, 1930.

[47] J. Athavale, M. Yoda, and Y. Joshi, "Comparison of data driven modeling approaches for temperature prediction in data centers," *International Journal of Heat and Mass Transfer*, vol. 135, pp. 1039–1052, 2019.

[48] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: https://www.deeplearningbook.org

[49] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2018.

[50] V. Dumoulin, E. Perez, N. Schucher, F. Strub, H. de Vries, A. Courville, and Y. Bengio, "Feature-wise transformations," *Distill*, 2018, survey and analysis of FiLM layers for conditional neural networks.

[51] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proceedings of the 4th International Conference on Learning Representations (ICLR)*, 2016, arXiv:1511.06434.

[52] TensorFlow Authors, "tf.keras.layers.conv2dtranspose," 2024, tensorFlow API documentation. [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/layers/Conv2DTranspose

[53] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7482–7491.

[54] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015, arXiv:1412.6980.

[56] Keras Team, "Earlystopping," 2024, keras 3 API documentation. [Online]. Available: https://keras.io/api/callbacks/early_stopping/

[57] ——, "Reducelronplateau," 2024, keras 3 API documentation. [Online]. Available: https://keras.io/api/callbacks/reduce_lr_on_plateau/

[58] X. Bouthillier, P. Delaunay, M. Bronzi, A. Trofimov, B. Nichyporuk, J. Szeto, N. Sepah, E. Raff, K. Madan, V. Voleti, S. E. Kahou, and C. Pal, "Accounting for variance in machine learning benchmarks," *Proceedings of Machine Learning and Systems (MLSys)*, vol. 3, pp. 747–769, 2021, arXiv:2103.03098.

[59] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[60] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.

[61] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," in *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021.

[62] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 328–339.

[63] M. Mansour, V. P. Galigekere, J. Pries, and O. C. Onar, "Triple active bridge soft-switching pwm dc-dc converter with integrated magnetics for dynamic wireless charging," in *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2021, pp. 1234–1241, tABSRC topology for DWPT HBridge converter - USU collaboration.

[64] A. Zade, C. R. Teeneti, M. Mansour, S. Gurudiwan, B. Hesterman, H. Wang, and R. Zane, "A 21-kw unfolding-based single-stage ac–dc converter for wireless charging applications," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 12, no. 1, pp. 8–27, 2024.

[65] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.