Advances in Artificial Intelligence and Machine Learning; Research 5 (2) 3784-3808 Received 08-03-2025; Accepted 09-06-2025; Published 16-06-2025

Development of an Intelligent Fault Diagnosis Tool for iPhone Motherboards: Power Consumption Analysis Using Deep Learning

P.D.K. Madhubhashana
Department of Information and Communication Technology, Faculty of Technology
University of Sri Jayewardenepura, Sri Lanka

H.D.N.V. Jayasekara
Department of Information and Communication Technology, Faculty of Technology
University of Sri Jayewardenepura, Sri Lanka

G.D.G.N. Jayawardena
Department of Information and Communication Technology, Faculty of Technology
University of Sri Jayewardenepura, Sri Lanka

B.N.S. Lankasena nalaka@sjp.ac.lk
Department of Information and Communication Technology, Faculty of Technology
University of Sri Jayewardenepura, Sri Lanka

B.M. Seneviratne *Department of Information and Communication Technology, Faculty of Technology*
University of Sri Jayewardenepura, Sri Lanka

Corresponding Author: B.N.S. Lankasena

Abstract

This study presents an intelligent microcontroller-based diagnostic tool and application designed to enhance fault detection accuracy and efficiency in iPhone motherboards, utilizing power consumption data and deep learning (DL) for real-time diagnostics. Integrating an RP2040 microcontroller and INA226 current sensor, the tool captures power patterns during boot-up, a method applicable across embedded systems and robotics for predictive fault analysis and maintenance. The tool, deployed in phone repair centers, has generated a comprehensive dataset of over 1,600 iPhone 6s devices with faults linked to 12 distinct power rails. Various deep learning models, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, were evaluated, with the LSTM achieving the highest accuracy (99%) and F1-score (0.997) for precise fault classification. This diagnostic application communicates with a central server, enabling a scalable and automated framework suitable for robotics and intelligent systems requiring power diagnostics. By introducing DL-based power consumption analysis, this study pioneers an approach with broad implications for intelligent maintenance in embedded and robotic systems. Our findings offer a foundation for faster, automated, and reliable diagnostics, potentially advancing fault management in robotic applications and other intelligent devices reliant on precise power monitoring and control.

Citation: P.D.K. Madhubhashana, et al. Development of an Intelligent Fault Diagnosis Tool for iPhone Motherboards: Power Consumption Analysis Using Deep Learning. Advances in Artificial Intelligence and Machine Learning. 2025;5(2):215.

Keywords: Deep learning, IPhone motherboard, Fault diagnosis, Power consumption, LSTM.

1. INTRODUCTION

The purpose of this research is to advance the understanding of mobile device fault diagnosis in order to facilitate the identification of motherboard defects in mobile devices, particularly iPhones. The investigation devised an effective approach for identifying the specific defective regions of the iPhone main motherboards with greater precision and in a shorter amount of time. iPhones that are unable to operate (not turning on) are the primary focus of this investigation. To realize the aforementioned goal, time series analysis in conjunction with deep learning (DL) techniques was undertaken to analyze the power consumption patterns. The research includes several crucial steps. Power consumption data for the iPhone 6 model was initially collected, which serves as the basis for the system's intelligence. Time series analysis plays a major role in the research by assisting in the comprehension of patterns in power consumption over time, particularly during the boot-up sequences of iPhone motherboards. In conjunction with time series analysis, DL was implemented to detect any abnormal or unhealthy motherboard conditions. These models were tested against real-world operational scenarios and tuned to handle the variabilities of the real-world. A user-friendly application and server backend were developed to minimize the technicians' efforts. The study delivers a diagnostic system for potential iPhones with faults, which improves the overall repair process by increasing reliability. This research anticipates bridging the gap between technology and practical applications, benefiting both iPhone users and technicians.

Advanced diagnostic techniques are necessary for the identification of faults in sophisticated devices that are frequently used in daily life, such as iPhones, due to their high electronic complexity. Manual diagnostic methods, including manual inspection and point testing, are often inaccurate and generally consume more time to complete on complex circuit boards like motherboards. Several innovative perspectives suggest the potential of power consumption data to address this issue through machine learning, particularly DL. Deep learning, a subcategory of artificial intelligence, is regarded as an advanced type of machine learning for fault diagnosis that aims to mine sophisticated patterns from power consumption data. The process of fault diagnosis is an essential sub-process in the maintenance of electronic equipment involving the systematic determination of the root causes of faults [1–3]. This process usually involves observing fault symptoms, isolating potential sources of problems, and verifying repairs through testing. According to Dahouk and Abu-Naser (2018) [4], expert systems can be implemented to resolve operational complications associated with desktop PCs. In their proposed system, the knowledge base was supported by rules collected from the expert's database that helped recognize different hardware problems. However, this approach was most effective when used with singular and specifically identified faults; it could hardly be successfully implemented in diagnosing complex motherboards with numerous components.

As previous studies have concluded based on the analysis of data, ML brings a more effective solution to the traditional rule-based system since it extricates patterns to find faults. In 2007 [5], Aminian used the idea of neural networks to find problems in analog circuits by being able to deal with component tolerance and non-linearity. However, training such networks, in general, requires enormous databases consisting of labelled data, which are hard to come by on complex systems. In recent studies, such as Chern et al. (2019) [6], multi-stage ANNs have been employed to enhance the diagnostic accuracy of intermittent faults, demonstrating the inherent versatility of such systems. Thus, power consumption analysis proves itself to be an effective approach to performing fault diagnosis, especially on mobile devices. Huang et al. (2010) [7], pointed out that the analysis of failure logs with power consumption fluctuations should be the key to diagnostics. It is noteworthy that power consumption changes could be obtained directly from the device. DL is especially useful due to its capability to work on high-dimensional data for these types of analyses. Ma et al. (2019) [8], have also shown that Graph Convolutional Networks are effective when used to analyze circuit netlists for testability analysis. Their approach, based on node embedding to represent the circuit topology, could extend to probing power consumption and performing fault-prone nodes identification on motherboards.

One of the disadvantages of time series data is that it is often oscillating and noisy, which limits the effectiveness of DL. Wavelet transform (WT), especially continuous wavelet transform (CWT) is an effective technique for decomposing a signal into its constituent frequencies, providing a time-frequency representation as well as providing a way of filtering noise and emphasizing significant structures [9]. A CNN is provided with enhanced features for classification based on the CWT transforms, which produce scalograms related to time-frequency patterns. The same approach can be naturally applied to the iPhone motherboard fault diagnosis, where CWT carries out feature extraction from fluctuations in power consumption, helping the deep learning model to establish the presence of fault patterns.

RNN performs very well when dealing with time-related sequential data therefore, it is suitable for identifying low-level patterns of power consumption that represent abnormalities. LSTM is particularly well-suited for time series data among the various types of DL models, including other variations of RNNs [10, 11]. Long-term dependencies can be modelled by LSTMs, which makes them effective in establishing patterns of power consumption records over extended periods. LSTM performs better than conventional techniques like Autoregressive Integrated Moving Average (ARIMA) for time series forecasting [12]. The LSTM has a ‘memory line’ and some gating mechanisms that allow it to successfully remember long sequences and capture complex and dependent relationships within time series data. This is especially useful in tasks where future data needs to be predicted since it carries information about the future from past knowledge [13]. In LSTM models the total number of complete training cycles against a dataset does not influence performance and a single epoch can suffice for effective model training.

Studies suggest that diagnostic faults can be detected by identifying load profile (electrical power demand or consumption) changes through the application of methods such as CWT along with CNNs and LSTMs [14–16]. Obtaining and annotating large datasets remains challenging alongside identifying power profile features and understanding DL decision-making processes. To fully harness DL capabilities for iPhone motherboard fault diagnosis a sophisticated strategy must be implemented to overcome existing limitations. Several unresolved matters remain in the application of DL to diagnose faults in iPhone motherboards. In general, these models face training constraints because they need large labeled datasets which track power consumption changes. The data acquisition and labelling procedure require both professional instruments and expert knowledge. The performance of X-sensitivity prediction models depends heavily on feature engineering as explained by Pradhan et al. (2018) [17]. A comprehensive analysis of power consumption fields makes the identification of related features from circuit structure challenging.

Under the above circumstances, the objectives set for the study are; (1) to develop an iPhone motherboard fault diagnosis system which uses power consumption patterns to identify faults efficiently (2) to develop a microcontroller-based data acquisition tool while creating a database of current consumption patterns for multiple iPhone models fault scenarios and (3) to optimize deep learning algorithm for fault diagnosis. The research addresses a major practical problem in fault detection through the creation of a user-friendly tool that simplifies iPhone motherboard diagnosis while enhancing the accuracy of fault detection. Mobile phone industry face a growing challenge to identify defects both quickly and accurately. The proposed method takes a fresh approach to motherboard power usage trends analysis through DL techniques. This approach enables specialists to swiftly locate malfunctioning areas which reduces diagnostic times along with disassembly and testing procedures. Traditional diagnostic methods necessitate disassembling almost the entire phone which increases the risk of damage to both the motherboard and the device itself. The results of this study reduce this risk by enabling the technicians to use the suggested system to locate issues non-invasively using power consumption pattern analysis. In addition to making defect detection easier, this suggested method lowers the possibility that a technician may accidentally break equipment when disassembling the product. This study utilizes the DL model to identify the distinctive power consumption patterns associated with 12 main faults that are widely recognized. The system was developed based on a thorough diagnostic approach combined with DL.

2. DATA AND METHODS

2.1 System Design Overview

FIGURE 1 shows an abstract overview of the implemented system. The diagram illustrates the system design for power consumption variation analysis and diagnosis. It involves data collection from iPhone devices, processing through a microcontroller and DL model, and outputting diagnosis results.

The system consists of several key components, as shown in FIGURE 1. The system collects power consumption patterns and relevant data when the iPhone is connected to the microcontroller through the connectors. This connection enables the microcontroller to gather real-time power consumption readings while the iPhone undergoes its boot-up sequence. After the microcontroller has collected the power consumption pattern over one minute, a CSV containing the time series data is exported and sent to the server. Once the data is received by the server, the time series data is fed into the DL model. Based on the power consumption pattern in time series data, the model determines the power rail that is affected by the motherboard. After identifying the faulty area, the server generates diagnostic results indicating the problematic area of the iPhone motherboard. These results are transmitted back to the PC to be displayed to the technician or end-user. The system also includes mechanisms for providing feedback to the technician or user. This feedback may encompass comprehensive diagnostic reports or repair suggestions. Traditional iPhone motherboard diagnostics involve manual inspection, multimeter probing, and testing components using schematics. These methods time consuming and require expert knowledge. Their accuracy often depends on the technician's skill. Our research's design aims to overcome these limitations and make diagnosis more effective.

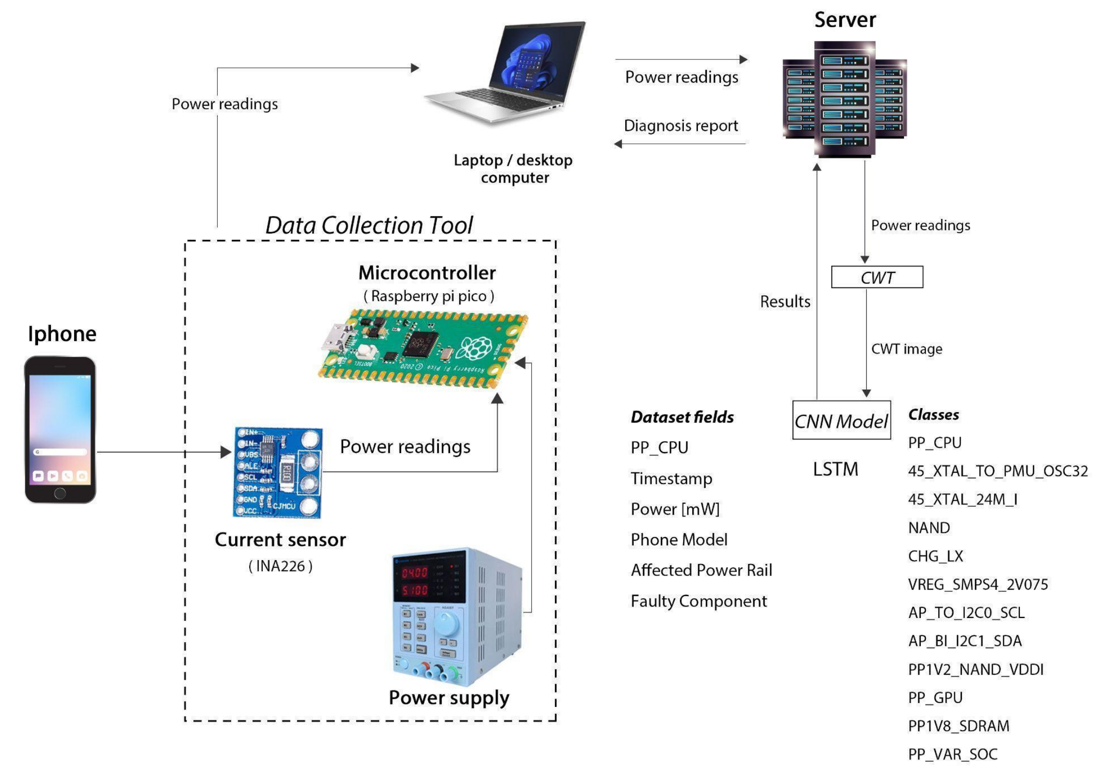


Figure 1: System design overview

2.2 Circuits for Data Collection

In this phase, a microcontroller-based data acquisition system was designed and constructed. The purpose of this system is to collect and process data related to the iPhone's current consumption patterns, which will subsequently be used for diagnostic purposes. The following FIGURE 2, shows the circuit with components including a Raspberry Pi Pico microcontroller, a current sensing module, a MOSFET, and an OLED display used for data collection. iPhone battery connectors are used to interface with the iPhone motherboard without causing any damage, enabling non-intrusive data collection.

Buck Converter: The current and voltage controller is a simple DC-DC converter that produces an output voltage lower than its input. In an ideal current and voltage controller, the output voltage is the product of the switching duty cycle and the supply voltage. Therefore, a DC-DC current and voltage controller module was integrated as a power supply module. The range of input voltage of this circuit (6~40V) is wide according to the various power sources. The output voltage is stable within a range of 1.2V to 36V, while the iPhones require 4.2V. This is a critical factor for maintaining the tool voltage's constant level, and it surely lowers the power loss.

INA226: The INA226 can measure both current and voltage. It monitors the voltage drop across a shunt resistor (for current measurement) and the bus supply voltage. Its extended range allows

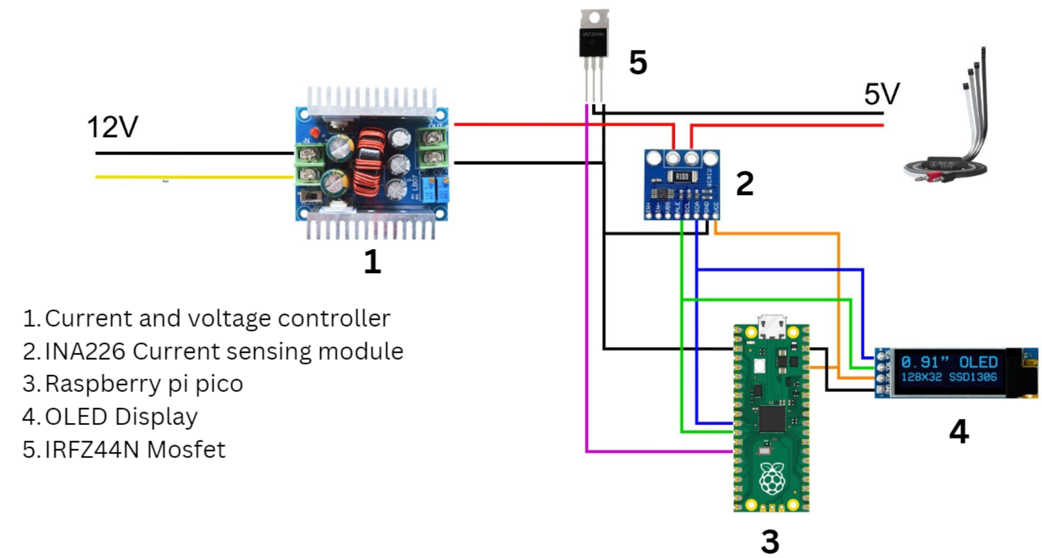


Figure 2: Circuit diagram of the data acquisition tool

it to measure voltages up to 36V and currents up to 20A. Additionally, it has high accuracy and configurable averaging and conversion times.

Raspberry Pi Pico: The Raspberry Pi Pico is a popular microcontroller board due to its affordability, simplicity, and flexible GPIO pins for various electronics projects. Three communication interfaces (SPI, I2C, UART) are available on the Raspberry Pi Pico powered by a dual-core Arm Cortex-M0+ processor. The development of this specific device employed the I2C communication technique.

IRFZ44N Mosfet: The IRFZ44N Mosfet excels as a powerful Mosfet because of its ability to handle large currents and switch quickly.

OLED Display: Because OLED displays have a slim profile and consume less power than LCDs they are ideal for portable devices.

The phase resulted in the completion of a special hardware device designed to extract data from iPhones which will act as an essential foundation for collecting data and performing diagnostic evaluations.

2.3 Data Collection

iPhone A1586 also known as the iPhone 6 was selected for the study due to many reasons. The main reason is since the model is outdated, it remains widely available at repair shops which facilitates access to extensive datasets. The model includes various specifications and features which could be used to explore the objectives of the research. Only one model was selected initially, as it enables a comprehensive examination of the faults within a single model before progressing to other models. Another factor that influenced the choice of an older model was its financial feasibility, allowing researchers to use their resources to develop the diagnostic system without worrying about the financial aspect. The study focuses on observing power consumption-related patterns on iPhone motherboards; therefore, the investigated area of the motherboard is narrowed down to a set of 12 power rails. Those power rails in the iPhone motherboard are where many issues can occur affecting the bootup process, thereby enabling us to associate unique power consumption patterns with distinctive power rails. Identifying power rails that are affecting the bootup process within the iPhone motherboard is a key aspect of this research since it permits us to investigate power consumption variations that are specific to particular components and functions in a detailed way. Each power rail is a provider of electricity to a particular circuit or subsystem within the motherboard, and any inconsistency in their consumption pattern is a warning sign of malfunction. Through the use of power rails, which are specific to individual lines, the proposed model can determine where the problem is with a high degree of accuracy, without extensive manual inspection or even without preventive repairing. In order to thoroughly evaluate the system's diagnostic capabilities, several problems with the 12 power rails were modelled on the iPhone 6. The possible motherboard problems covered by these scenarios ensure that the proposed model is reliable and capable in multiple fault diagnosing. The identified power rails are as follows:

- **PP_CPU**: Main power supply for the iPhone's central processing unit (CPU)
- **PP0V95_FIXED_SOC**: This is a 0.95V fixed power line to the CPU/SOC
- **45_XTAL_24M_I**: This represents a 24 MHz clock signal from a crystal oscillator, used for various components within the iPhone.
- **NAND**: Main power supply for the NAND flash memory, used for storage in the iPhone.
- **CHG_LX**: Related to the charging circuitry.
- **PP_LCM_BL_ANODE**: Power rail to the display backlight
- **AP_TO_I2C0_SCL**: Represents I2C (inter-integrated circuit) communication lines between the application processor (AP) and other components. SCL refers to the clock line.
- **PP1V2_SDRAM**: 1.2V power supply for the SDRAM (Synchronous Dynamic Random-Access Memory), which the main system memory in the iPhone.
- **PP1V2_NAND_VDDI**: A 1.2V power supply for the NAND flash memory, used for storage in the iPhone.
- **PP_GPU**: Similar to PP_CPU, this line is the power supply for the Graphics Processing Unit (GPU).
- **PP1V8_SDRAM**: Supply a 1.8V power supply for the SDRAM, the main system memory in the iPhone.
- **PP_VAR_SOC**: A variable voltage power supply for the System on a Chip (SOC), which typically includes the CPU, GPU, and other essential components.

The parameters observed and data were recorded during the data collection process are Power [mW], Phone model, Affected power rail, Faulty component and Fault. Faulty iPhone devices were placed through a bootup process to create the dataset. During this cycle, the power consumption was monitored carefully. The data was collected at a momentum of 10 readings per second. A high frequency of sampling was employed to provide a detailed analysis of current consumption fluctuations during crucial periods, including the initialization and termination of system operation. Each device was allotted one minute of time to collect the data, and hence there were 600 records per device.

The collected data was processed and filtered according to its quality and suitability for the analysis criteria. Some of the elements that were not necessary or important were removed through techniques such as noise reduction, outlier analysis, smoothing, and filtering (Gaussian filter). The goal of this step was to improve the signal-to-noise ratio and ensure that all of the produced data accurately depicted the patterns. The process of creating the dataset involved determining power consumption under different situations, identifying any potential issues, and organizing and compiling the data collected.

2.4 Algorithm Selection and Model Development

Identification of the most suitable DL technique is one of the objectives of the study. CNN is particularly advantageous for image-event analysis and pattern recognition [18]. The time-frequency representation of CWT enables the detection of transient elements and the observation of changes in frequency components over time in the signal. This information is valuable for fault detection [19]. However, since power consumption data changes over time in a unique way, RNNs were found to be better at understanding the patterns in how the data changes. To improve the extracted features from the power consumption data, CWT was used due to proven better results [7]. A time-frequency representation produced by the CWT enables detection of transient elements and monitoring changes in frequency components throughout the signal duration.

We used an LSTM model to classify time series data, although neural networks do not work well with sequential data unless they are improved. In contrast, RNNs are developed to support such data by keeping a memory that contains past information to learn temporal dependencies. During the RNN implementation, experiments with LSTM have been conducted to determine which one yielded the most optimal outcomes for the given task. An extra step was taken to help the network learn on its own using time series data. This was done by using RNN architectures that were synchronized with structures like Penn Treebank or IMDB for transfer learning. This way, the network was able to gain some pre-specified knowledge that would allow it to recognize relevant features from the time series data, thereby yielding excellent results, as mentioned earlier, even with a very small training set, as proven by Nanduri et al. (2016) [20].

During the training process, an ’unfreezing’ procedure has been maintained, where the previously frozen layers from pre-training remain frozen initially. However, other layers were gradually and selectively unfreezed during training to enable the network to learn the distinctive features of our dataset while still retaining the features learned during pre-training. Adam optimizer was used for fine-tuning the model. Data augmentation was done using techniques such as scaling, jittering, and shifting to make the model less sensitive to noise and less prone to overfitting similar to various other studies [21, 22]. The model was enhanced by selecting the optimal parameters and fine-tuning the numerous hyperparameters on a validation set. The evaluation process included metrics like accuracy, precision, recall, and F1-score. The method utilizes time series data features and uses RNNs to identify temporal dependencies which allows it to diagnose power rail failures within the power consumption patterns of iPhone motherboards.

2.5 Desktop Application With Server Backend Development for Predictions

The prime objective of the study was to develop a desktop application with a comprehensible and convenient user interface to be utilized by users who undertake phone repairs. This will create a platform where non-specialists, such as technicians, can easily relate to the system and its instructions. The app was designed to be user-friendly, provide efficient navigation, and offer a clear explanation of its features and functions. The application contains a user interface that allows the technicians to input specific fields from the iPhone that has undergone initial inspection. This includes the model number of the phone, the observed consequences, or some other relevant information. The input should be uncomplicated and guided to eliminate mistakes and ensure the correctness of the diagnostic process.

The application has a mechanism for triggering the hardware tool to start the diagnostic process. This feature enables timely and on-demand assessments, providing flexibility in diagnosing iPhones as needed. This is an aspect of integrating with the database and ensuring that there is constant communication between the application and the server. The application was developed to enable future expansion, where new iPhone models can be added or there could be changes at the system level. The user interface was designed to support different operating systems and devices that are popular among technicians to ensure that the software is easily accessible.For implementing real-time fault detection, an independent server back end is created based on which the LSTM deep learning model is trained. The backend, where the LSTM model is implemented, predicts faults by taking the power consumption data from the desktop application and sending diagnosis results back to the desktop application. API calls between the application and the server are properly encrypted to allow only authorized access to the information while making it easily accessible and secure at the same time. This design architecture also permits real-time updating and enhancement of the models for precise diagnostic decisions for end-users.

The application utilizes the Electron framework for the desktop interface and the Django framework for the backend system. Both frameworks are well-suited for future enhancements and handling heavy traffic. Electron supports cross-platform compatibility, while Django offers robust features such as database optimization, caching, and asynchronous request handling. The backend system is hosted on Azure App Service, which provides automatic scaling, load balancing, and seamless deployment. Together, these technologies establish a future-proof foundation for the application. In conclusion, the desktop application with server communication provides an interface between technicians and the diagnostic system. It is easy to use, accepts the user's data input, initiates the diagnostic process, and emphasizes the secure connection with the encrypted server. The development process follows the standard guidelines for application design, focusing on both functionality and security.

2.6 Testing and Validation

Data acquisition tool validation: In order to establish an effective and reliable procedure for the data acquisition tool, we followed various testing procedures. Thus, we confirmed that it successfully sampled power consumption readings with a frequency of 10 reads per second and was continuously capturing data for one minute, which cumulatively equalled 600 data points in the recording session. We tested the tool with iPhones to see if it collects data at the right frequency and contains minimal noise.

Dataset validation: A good set of data is the primary component that determines the effectiveness of deep learning. We carefully built a dataset directed to the iPhone 6 because it is easier to access and there were more records gathered. The dataset comprises the power consumption profiles of twelve prominent power rails, all of which were exposed to the emulation of short circuits and component failures. It also made sure that the dataset was inherent in real-life faults and was suitable for training the DL models.

Deep learning model validation: We evaluated the performance of five deep learning models: ResNet50, ResNet101, VGG16, VGG19, and LSTM. All of these two models were trained for 100 iterations and tested for accuracy, loss, and F1-score using the prepared data. To perform model generalization, scaling, jittering, and shifting operations were conducted on the training set. Therefore, the Adam optimizer was used for fine-tuning the models because of its efficiency. The LSTM model was proven to be more accurate than other models, with a 99.57% accuracy rate. The mean F1-score extracted was 0.997, which indicates relatively high accuracy, and the minimum loss was 0.0008. This showed that LSTM's ability to model the temporal dynamics of the power consumption data surpassed the rest in detecting fault patterns.

Desktop application validation: To validate the developed desktop application, multiple tests were conducted on Windows and macOS operating systems. We concentrated on such aspects as the UI design, smooth work, and proper safety of data in the application. Functional and performance tests ensured the integrated working of account management and the input of data for the diagnostic process, as well as a properly designed display of the results. Data confidentiality was also protected during transmission and storage through encryption while communicating with the server through the application.

Real-world validation: As the final step in our research, we had to apply the entire system to actual service delivery. Mobile phone repair shops were used as partners in the current study, specifically to diagnose faults in iPhone 6 motherboards. A variety of faults were found on power rails, including shorts and component issues. The tool was successful in identifying the faulty power rails in this case. Further, the system was developed considering general software maintenance and evolution principles making it robust for future changes [23–26]. This real-world validation increased the practicality of the tool by proving that it has the potential to drastically cut down on the time taken to diagnose the problem and increase repair effectiveness.

The testing and validation process ensure that the developed system is not only effective in identifying faults in iPhone motherboards but also robust and reliable across various simulated scenarios.

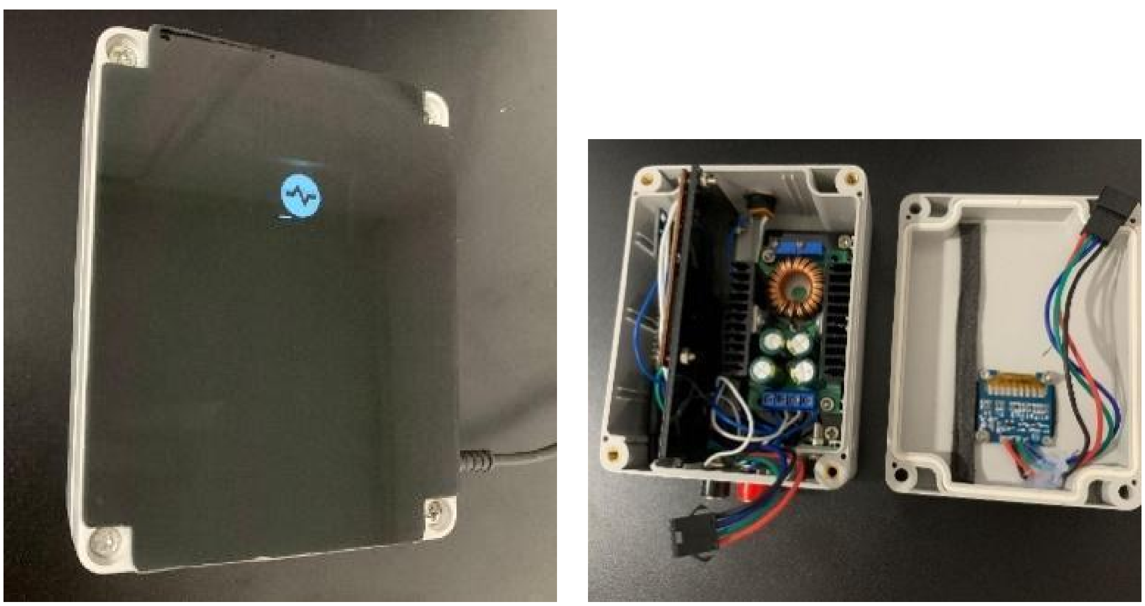
2.7 Scaling and Continuous Improvement

Provisions for scalability have been made to accommodate additional iPhone models and potential software updates. User feedback will be actively sought to identify areas for improvement and address evolving diagnostic needs. Regular updates and improvements will be implemented to ensure the continued effectiveness of the system.

3. RESULTS AND DISCUSSION

3.1 Develop a Microcontroller-Based Data Acquisition Tool:

We created the data acquisition tool in FIGURE 3a and FIGURE 3b, from scratch and tested it. The image shows the physical device, which includes the components for the data acquisition from the iPhones. The device can collect power consumption readings at the rate of 10 records per second collectively, and it can record 600 records per minute, which is well-suited for the intended purpose. Even though we gathered information from all three iPhone models, the study continued with the iPhone 6 since it has the most records overall. The number of records for the models that collected data are; iPhone 6-1600, iPhone 7-420 and iPhone 8-170.



(a). Outside view **(b).** Inside view

Figure 3: Data acquisition tool

3.2 Dataset of Current Consumption Patterns for Multiple iPhone Models and Fault Scenarios:

Under this objective, we collected the power consumption of iPhone motherboards using the developed microcontroller and created a time series dataset showing the power consumption of the motherboard when the device is turned on. Due to the limitations of records, the dataset was created only for the iPhone 6, as shown in FIGURE 4.

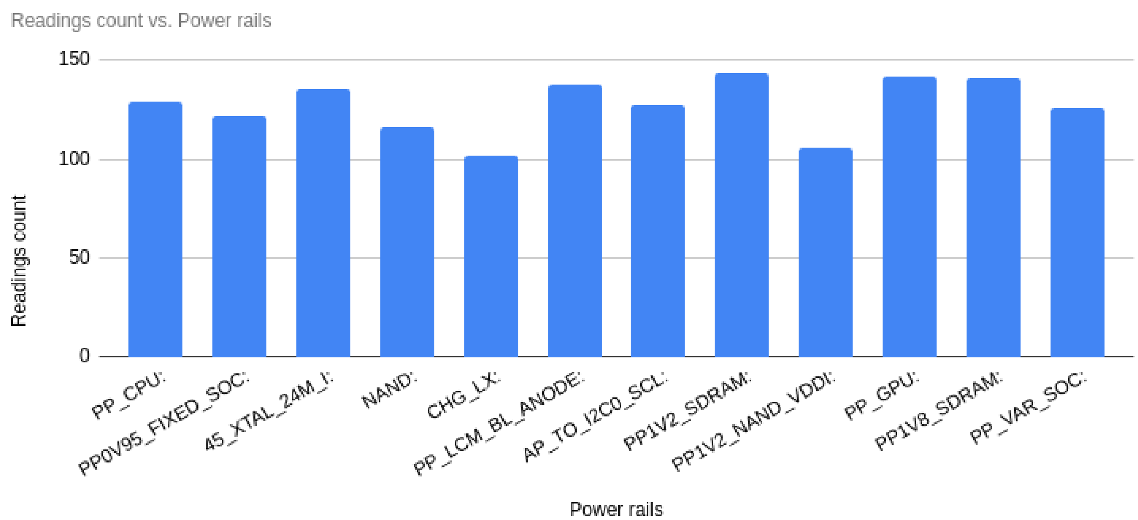


Figure 4: iPhone 6 dataset

3.3 To Develop and Optimize a Machine Learning Algorithm for Fault Diagnosis:

When the power consumption pattern is plotted against time, it reveals distinct patterns for each defect, as illustrated in FIGURE 5 (refer to Appendix A for all images). The record of each device is then turned into a CWT image, as shown in FIGURE 6, and these images were utilized to carry out the defect diagnostic procedure, which uses an optimized pre-trained CNN model to recognise patterns in the images.

The figure shows the line graphs for the different power rails to show the significant difference between each fault.

The above figure shows the CWT images for some of the faulty power rails to show the significant difference between each faulty rail (refer to Appendix B for all the images).

3.4 Deep Learning Models:

The four CNNs (ResNet50, ResNet101, VGG16, and VGG19) were tested for their capacity to classify CWT images based on the affected power rail. Each model was initialized with pre-trained

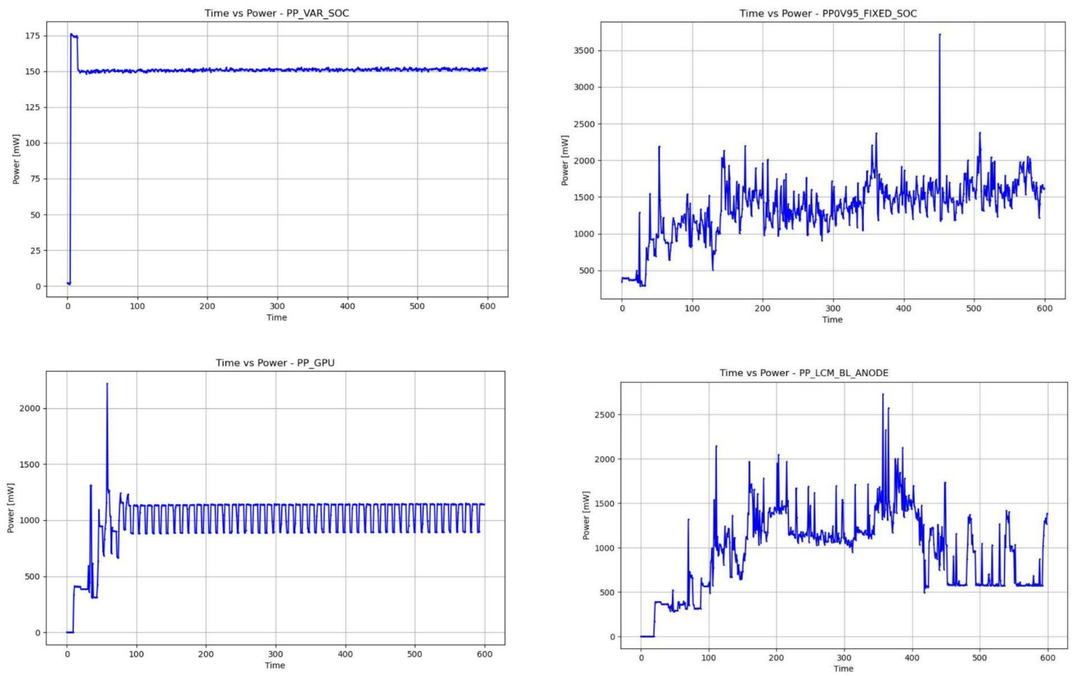


Figure 5: Faulty power consumption pattern of power rails

weights from ImageNet and then fine-tuned on our dataset. The training process involved two stages: initial training with most convolutional layers frozen, followed by a fine-tuning stage where deeper layers were gradually unfrozen.

The performance of each model was assessed using the metrics Accuracy, Loss and F1-Score. The training process was monitored over 100 epochs. Out of the ResNet101, ResNet50, VGG16, VGG19, the LSTM model proved to be extremely efficient, with a test accuracy of 0.99. The model training was also successful, with a test loss of 0.0008. The F1-score of 0.997 is much higher compared to the rest of the models trained. Although it's stated that ResNet50 is the best-performing model for CNN architectures, the LSTM model significantly performs better than ResNet50 in terms of accuracy and loss. As a result, the test accuracy of approximately 99% and the low loss rate indicate that the LSTM model is suitable for power consumption analysis in this application. This finding supports the use of LSTM models for time series analysis for determining iPhone motherboard failure diagnosis. TABLE 1 summarizes the performance of the 5 models after fine-tuning.

The results demonstrate that all four models achieved high accuracy and F1 scores, indicating their effectiveness in classifying CWT images. ResNet50 exhibited the best performance from CNN models, achieving the highest F1-Score and accuracy, along with a relatively low loss from CNN models. The LSTM model exhibited the best overall performance when compared with CNN models.

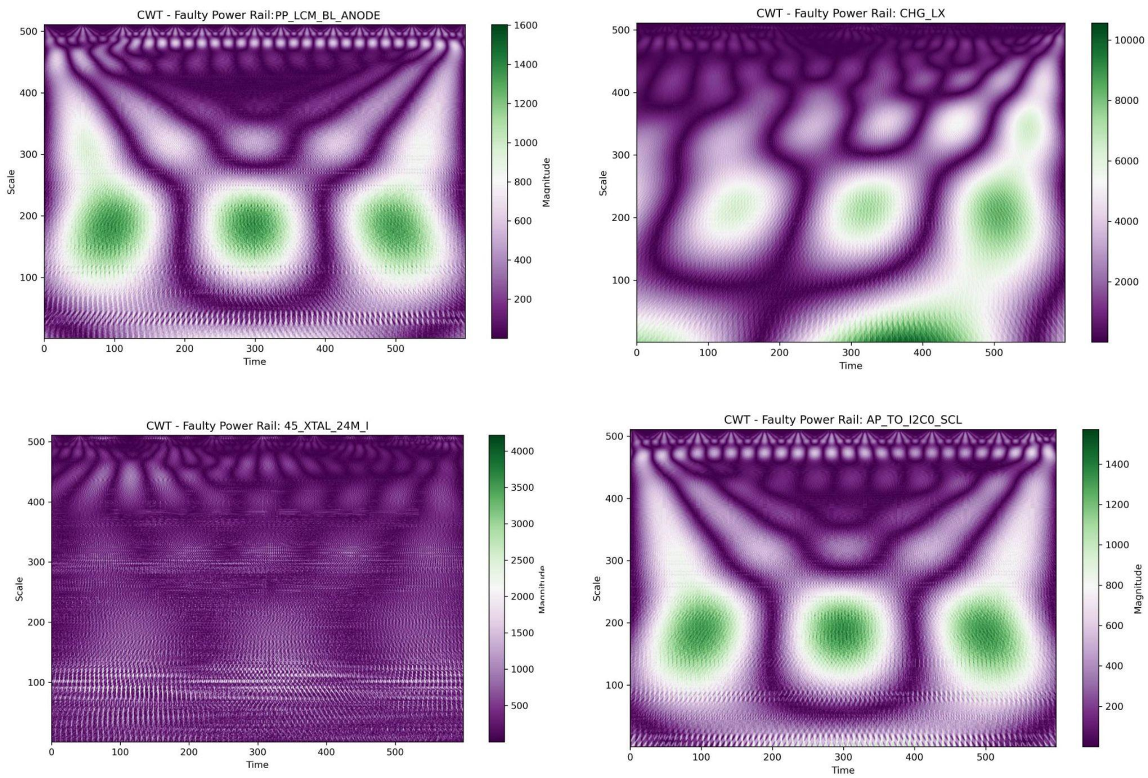


Figure 6: CWT images for faulty power rail patterns

Table 1: Comparison of model performance

Model	F1-Score	Loss	Accuracy
ResNet101	0.9777598287	0.7595050334	0.9785714149
ResNet50	0.9838679385	0.7789865732	0.9857142567
VGG16	0.9704659015	1.2690948247	0.9714285731
VGG19	0.9704659015	1.2001744508	0.9714285731
LSTM	0.9970347708	0.0008827138	0.9957143477

The study employed a Bidirectional Long Short-Term Memory (BiLSTM) model to classify time series data of power consumption patterns. To improve the model robustness and generalization, the algorithm used data augmentation techniques, specifically time shifting (±30 units), power magnitude shifting (±100 mW), and combinations, which doubled the training dataset. The network architecture was designed with two sequential BiLSTM layers, each containing 100 units and processing input sequences of 600 time steps in both forward and backward directions to capture comprehensive temporal dependencies. Dropout layers with a rate of 0.3 were applied after each BiLSTM layer for regularization. This includes a dense layer with 50 units and ReLU activation and a final Softmax output layer for classification. Training utilized a two-phase strategy: initial training with the Adam optimizer at a default learning rate and then fine-tuning with Adam at a reduced learning rate of 1e-5. Both trained for a maximum of 50 epochs with early stopping (patience of 10, monitoring validation loss) and a batch size of 32. This approach resulted exceptional performance, achieving an accuracy of 99.57%, a macro-averaged F1-score of 0.997, precision of 0.9949, and recall of 0.9951, with loss of 0.0008, demonstrating its high effectiveness. A confusion matrix, presented in FIGURE 7, further details the classification performance across different fault types.

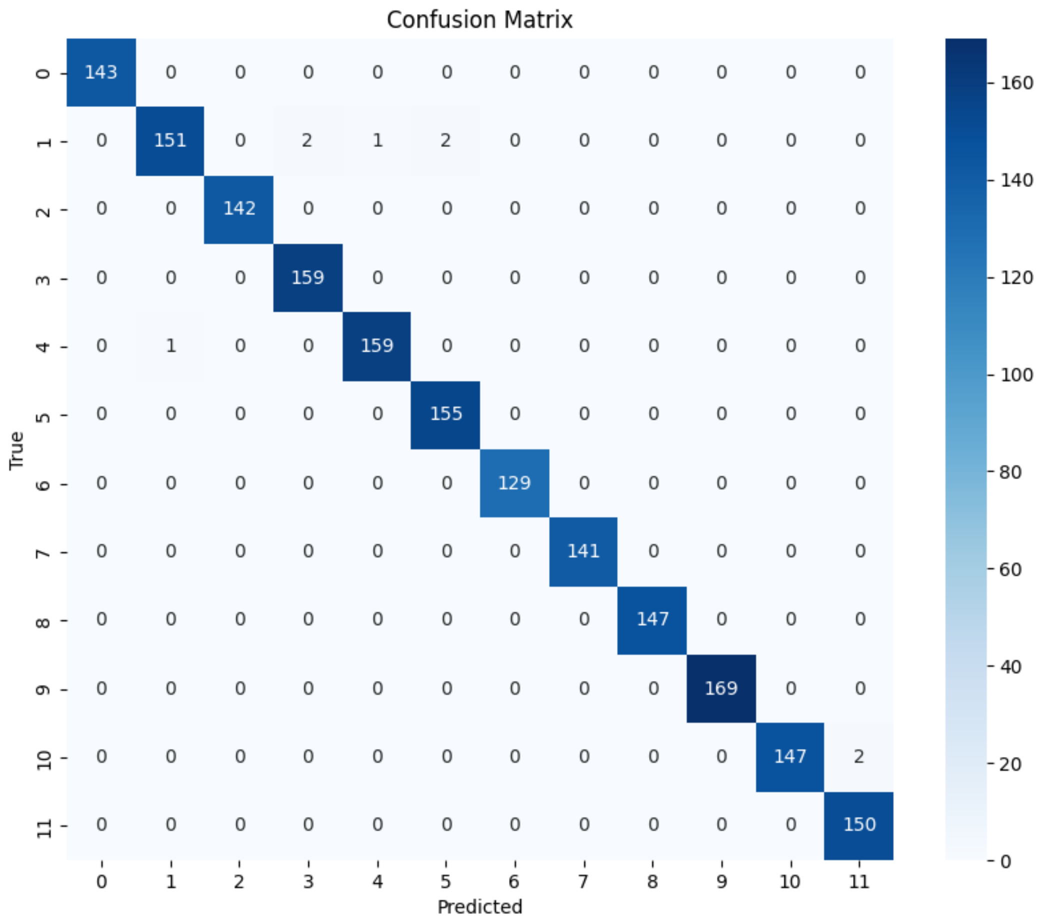
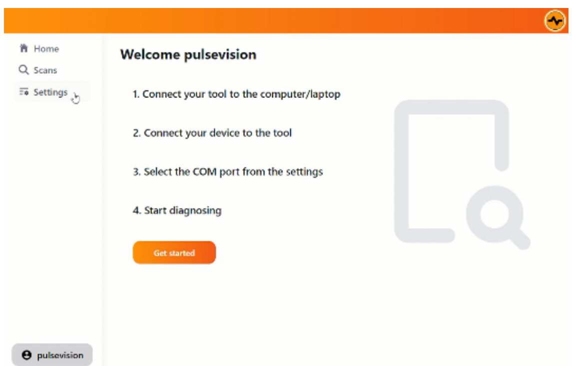


Figure 7: Confusion matrix of LTSM model

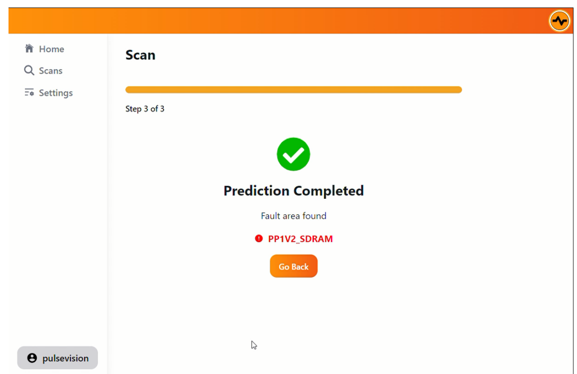
Results suggest that the LSTM architecture is well-suited for extracting relevant features from the Time series data and making accurate predictions regarding the affected power rail. The training graphs show a consistent trend of decreasing loss and increasing accuracy over the training epochs. The fine-tuning stage, where deeper layers were unfrozen, led to further performance improvements. These observations validate the effectiveness of transfer learning and the fine-tuning strategy employed in this work.

3.5 Development of a User-Friendly Desktop Application:

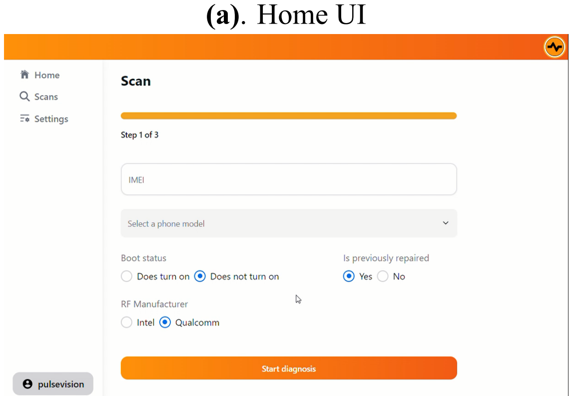
The desktop application offers intuitive user interfaces that enable the user to initiate the diagnosis process, view the results, and access critical interfaces for both user account management and diagnosis (FIGURE 8). FIGURE 8(a) shows the home user interface (UI), where the user will be redirected after logging in to the application. Step-by-step instructions are given to the user by this interface to start their diagnosis process. FIGURE 8(b) shows the scan completion UI, which will be shown after the diagnosis process, and the prediction completed, mentioning the predicted power rail. FIGURE 8(c) shows the scan-initiating UI, where the user will have to enter the basic details about the device that they are scanning. Finally, FIGURE 8(d) shows the list of scans that the user has carried out previously.



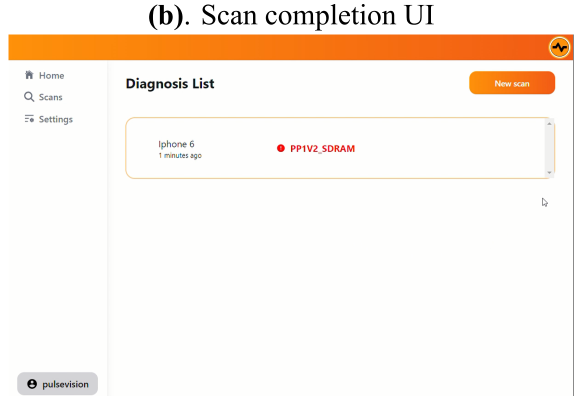
(a). Home UI



(b). Scan completion UI



(c). Scan initiating UI



(d). Diagnosis history

Figure 8: GUIs of the desktop application

4. DISCUSSION

The study presents an innovative approach to fault diagnosis for iPhones based on the variation of the power consumption of the motherboard, employing deep learning. There were studies carried out generally exploring the fault diagnosis for circuit fault diagnosis using artificial intelligence but not particularly regarding the field of mobile phones narrowing iPhone motherboards [7]. As per the results, the study produced a microcontroller-based data acquisition tool, dataset, deep learning model for fault pattern recognition, and desktop application for the end user.

Previous research has explored various approaches to fault diagnosis, including rule-based systems [27, 28], machine learning algorithms [29], and power consumption-based methods [30], but these have focused on general circuit diagnosis, not specifically on iPhone motherboards. Our contribution to the field is based on a new method that uses DL together with power consumption patterns to find motherboard problems in iPhones. Further, the present study's approach is much more efficient as it does not involve manual inspection or component testing as the DL can also perform a diagnosis of power consumption patterns in a very short time as compared to traditional methods [31].

This study presents a data acquisition tool, a deep learning model, and a desktop application integrated with a server for prediction, which enables this work to seamlessly plug into current iPhone repair processes. This approach provides a more efficient and easy method of identifying iPhone motherboard problems than other tools and techniques available. This study employs the LSTM model to extract rich time information from the power consumption data to predict complex trends and fluctuations. The developed microcontroller-based data acquisition tool is capable of acquiring power consumption data within the desired time duration. For example, the tool is capable of collecting the power consumption data within a minute by collecting 10 records per second, and hence collecting 600 records per minute at the moment from the connected device. These data can be used as time series data for the analysis process, either to add to the dataset or for fault diagnosis. Therefore, the resulting tool satisfies the objective of developing the microcontroller device, fulfilling all the requirements. Data collection was done in multiple mobile phone repair centers using the developed microcontroller-based tool. This allowed the tool to be verified against real-world usage and enabled efficient data collection. The collected data was then preprocessed to clean and remove unnecessary or null entries. As a result, a complete dataset was successfully created for the iPhone 6 model, which was used for training a model for fault diagnosis.

The research experimented with several DL models including ResNet101, ResNet50, VGG16, VGG19, and LSTM. Out of all the experiments, LSTM shows the best accuracy of 99.57%. The results indicate the suitability of LSTM in capturing long-term dependencies of power consumption series is critical for identifying more intricate fault patterns. When comparing ResNet with LSTM, it produced good accuracy in less time. This finding could suggest that accuracy and speed are compromised. On the other-hand, VGG models are not that suitable for capturing the intricate temporal structures present in the power consumption data. These comparisons show why LSTM is the right deep learning model to use, based on the data type and the balance between accuracy and speed.

The study also developed a user-friendly desktop application that communicates with the server for mobile diagnosis prediction. Additionally, user account management is provided by the application, allowing the user to safely store records of prior diagnoses and outcomes. The application's simple and user-friendly design allows users to quickly learn the diagnostic technique and apply it without extensive training. This innovative approach can drive the industry to develop faster component repair methods, improving customer satisfaction. The combination of deep learning and a data acquisition tool with a user interface greatly distinguishes the proposed system from conventional diagnostic methods. Traditional diagnostic methods for iPhone motherboard faults involve manual inspection, multimeter probing, and circuit-level analysis using schematics. These procedures are time-consuming, demand substantial technical expertise, and their effectiveness can vary significantly depending on the technician's experience. In contrast, the proposed AI-based diagnostic system automates fault identification by leveraging power consumption signatures and machine learning models, thereby reducing diagnosis time and enhancing consistency. Although a direct quantitative comparison with manual methods is constrained by the lack of standardized benchmarks and variability in human performance, our approach shows strong potential for improving efficiency, standardizing fault detection, and reducing dependence on expert technicians. A key advantage of the developed DL-based diagnostic tool when compared with traditional methods that this tool is ahead in enhanced speed, accuracy, and reduced invasiveness. Manual diagnostics, heavily reliant on technician experience, schematics, and iterative probing, can be exceptionally time-consuming and exhibit variability in success rates.

The project's impact could be increased by adapting the tool to support a wider range of iPhone models. The project's current primary constraint is the dataset's insufficient size. In addition, the dataset itself limits the model's ability to generalize effectively across a diverse range of scenarios. This is due to the fact that deep learning models are not trained on every possible fault pattern that can be found on the iPhone's motherboards. Additionally, another significant challenge is the particularly second-hand iPhones, that are plagued by numerous defects.

We assessed the reliability and performance of our iPhone motherboard fault diagnosis tool in real-world settings. We utilized our data acquisition tool to analyze various iPhone 6 units to collect power consumption data. Based on the gathered power consumption data, the tool was able to pinpoint the power rails that were impacted. This real-world validation proved the applicability of the tool in identifying the faulty power rails in iPhone 6 motherboards and illustrates the tool's effectiveness.

The primary focus of our study is on the iPhone 6 model, which allows researchers to conduct in-depth analysis and develop proof-of-concept. However, there are limitations regarding the immediate generalizability of the findings to other iPhone models or newer devices. The iPhone 6, being an older model, provided an accessible platform for developing and validating our methodology; however, power consumption characteristics, common fault points, and component architectures can vary significantly across different hardware generations. Furthermore, while the system showed high accuracy for the 12 identified power rail faults, its adaptability to entirely novel or unencountered fault types within the iPhone 6, or its transferability to diagnosing faults in substantially different embedded systems without retraining or significant adaptation, remains an area requiring further investigation. The "intelligence" of the current system is highly specific to the patterns learned from the iPhone 6 dataset.

Utilizing a multi-tier approach and taking these constraints into account, the project can be further improved in the future. First and foremost, we should concentrate on the addition of a substantial number of records to the dataset. A series of strategies, including data augmentation techniques, synthetic data generation, and partnerships with other repair centers, enabled the implementation of this project. The effects of real scenarios are more plausible when the datasets are larger and more diverse, as the models are more capable of accurately depicting them and generalizing their behaviour in complex scenarios. Additionally, an unsupervised learning pathway that has been developed can identify and address fault patterns, even if they have not been previously observed. The traditional diagnosis approach is rigid, whereas this model becomes more intelligent and adaptive in its diagnosis by autonomously identifying faults within the collected data through the derivative methods. Future work should focus on developing a more adaptable diagnostic tool that can accurately identify a variety of device configurations and fault conditions, thereby improving its utility for mobile phone technicians dealing with diverse hardware issues. This involves validating and adapting the proposed methodology across newer iPhone models and hardware generations by analyzing differences in power consumption behavior, component layout, and fault patterns. Enhancing these aspects will improve the generalizability and practical relevance of the system across a broader range of devices.

5. CONCLUSION

This study addresses significant challenges in the field of iPhone motherboard fault diagnostics, specifically the identification of iPhone motherboard faults using deep learning techniques. It is time-consuming and inefficient to manually identify deficiencies in these complex components. By analyzing power usage, this research allows for more precise identification of both the problematic area and the damaged power rails. The development of a microcontroller-based data collection tool with a circuit layout design marks a significant advancement. The tool was used to effectively and reliably collect power consumption data from the affected motherboard. The study also used the Continuous Wavelet Transform to turn the collected time series data into images. These images were then turned into visual representations, which let the study use the power of CNN models that can recognize images. This research employs an LSTM model, a type of Recurrent Neural Network. The LSTM model detected complicated patterns within converted power consumption data with a greater accuracy of 99.57%. There is no existing tool that uses power consumption data with DL to analyze faults.

The user-friendly data gathering tool makes data collection easier, while the desktop application provides a simple and intuitive interface for analysis. This desktop application uses a server-based prediction model for its predictions. This combination prepares the solution for seamless integration into existing workflows in commercial mobile phone repair centers. This evaluation of the tool and its application in real-world scenarios resulted in a high level of accuracy. In summary, this investigation has the potential to significantly impact the field of mobile phone repair. This significantly reduces the time required for precise defect diagnosis. This proposed approach is intended to reduce system downtime for iPhone users and enhance repair efficiency. The study will establish a future in which iPhone repair is no longer a time-consuming endeavor but rather a straightforward, rapid, and precise one.

References

[1] Liu S, Wang Y, Fu H. Fault Diagnosis Based on AGA-LS-SVM for Analog Circuit. 2009 International Conference on Mechatronics and Automation. IEEE. 2009:681-685.

[2] Huang X, Gao J, Jiang H, Chen K. A Systematic Fault Root Causes Tracing Method for Process Systems. Proc Annu Reliab Maint Symp. IEEE. 2011;1-7.

[3] Nguyen V, Nguyen D, Nguyen N. Feature Selection Based on Eemd-Las and Optimized Bso-Lssvm Classifier Model in Bearing Fault Diagnosis. Int J Intell Eng Syst. 2021;14:247-258.

[4] Dahouk AW, Abu-Naser SS. A Proposed Knowledge Based System for Desktop PC Troubleshooting. Int J Acad Pedagog Res. 2018;2:1-8.

[5] Aminian M, Aminian F. A Modular Fault-Diagnostic System for Analog Electronic Circuits Using Neural Networks With Wavelet Transform as a Preprocessor. IEEE Trans Instrum Meas. 2007;56:1546-1554.

[6] Chern M, Lee SW, Huang SY, Huang Y, Veda G, et al. Improving Scan Chain Diagnostic Accuracy Using Multi-Stage Artificial Neural Networks. In: Proceedings of the 24th Asia and south Pacific design automation conference. ACM. 2019:341-346.

[7] Huang K, Stratigopoulos HG, Mir S. Fault Diagnosis of Analog Circuits Based on Machine Learning. In: Design automation & test in Europe conference & exhibition. Dresden Germany: IEEE. 2010:1761-1766.

[8] Ma Y, Ren H, Khailany B, Sikka H, Luo L, et al. High Performance Graph Convolutional Networks With Applications in Testability Analysis. In: Proceedings of the 56th annual design automation conference. ACM. 2019:1-6.

[9] Naseem S, Javed K, Jawad Khan M, Rubab S, Attique Khan M, et al. Integrated CWT-CNN for Epilepsy Detection Using Multiclass EEG Dataset. Comput Mater Continua. 2021;69:471-486.

[10] Ienco D, Gaetano R, Dupaquier C, Maurel P. Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks. IEEE Geosci Remote Sens Lett. 2017;14:1685-1689.

[11] Sun Z, Di L, Fang H. Using Long Short-Term Memory Recurrent Neural Network in Land Cover Classification on Landsat and Cropland Data Layer Time Series. Int J Remote Sens. 2019;40:593-614.

[12] Siami-Namini S, Tavakoli N, Siami Namin A. A Comparison of Arima and Lstm in Forecasting Time Series. 17th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE. 2018:1394-1401.

[13] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Comput. 1997;9:1735-1780.

[14] Guo Y, Zhou J, Dong Z, She H, Xu W. Research on Bearing Fault Diagnosis Based on Novel Mrsvd-Cwt and Improved CNN-LSTM. Meas Sci Technol. 2024;35:095003.

[15] Zhu S, Yang G, Song S, Du R, Yuan H. Joint-Module Health Status Recognition for an Unmanned Platform: A Time–Frequency Representation and Extraction Network-Based Approach. Machines. 2024;12:79.

[16] Tang C, Zhang Y, Wu F, Tang Z. An Improved Cnn-Bilstm Model for Power Load Prediction in Uncertain Power Systems. Energies. 2024;17:2312.

[17] Pradhan M, Bhattacharya BB, Chakrabarty K, Bhattacharya BB. Predicting X-Sensitivity of Circuit-Inputs on Test-Coverage: A Machine-Learning Approach. IEEE Trans Comput Aid Des Integr Circuits Syst. 2019;38:2343-2356.

[18] Gao D, Zhu Y, Wang X, Yan K, Hong J. A Fault Diagnosis Method of Rolling Bearing Based on Complex Morlet CWT and CNN. Prognostics and System Health Management Conference.IEEE. 2018:1101-1105.

[19] Gu J, Peng Y, Lu H, Chang X, Chen G. A Novel Fault Diagnosis Method of Rotating Machinery via VMD CWT and Improved CNN. Measurement. 2022;200:111635.

[20] Nanduri A, Sherry L. Anomaly Detection in Aircraft Data Using Recurrent Neural Networks (RNN). In: Integrated communications navigation and surveillance (ICNS). IEEE. 2016:5C2-1-5C2-8.

[21] Paul H, Udayangani H, Umesha K, Lankasena N, Liyanage C, et al. Maize Leaf Disease Detection Using Convolutional Neural Network: A Mobile Application Based on Pre-trained VGG16 Architecture. N Z J Crop Hortic Sci. 2025;53:367-383.

[22] Lankasena N, Nugara RN, Wisumperuma D, Seneviratne B, Chandranimal D, et al. Misidentifications in Ayurvedic Medicinal Plants: Convolutional Neural Network (CNN) to Overcome Identification Confusions. Comput Biol Med. 2024;183:109349.

[23] Dharmasiri N, Kodithuwakku J, Pallawala P, Lankasena B. The Impact of Agile Practices on Software Evolution in Startup Companies. 2023 Available at: https://www.researchgate.net/publication/384676606_The_Impact_of_Agile_Practices_on_Software_Evolution_in_Startup_Companies

[24] Pamunuwa V, Deraniyagala D, Kulasekara V, Thennakoon R, Lankasena B. Investigating the Impact of Software Maintenance Activities on Software Quality: Case Study. 2023. Available at: https://www.researchgate.net/publication/384675050_Investigating_the_Impact_of_Software_Maintenance_Activities_on_Software_Quality_Case_Study

[25] Wisidagama N, Karunarathne M, Paranagama P, Rathnayake R, Lankasena B. A Comprehensive Study on Software Evolution in Plan Driven and Agile Methodologies. 2023. Available at: https://www.researchgate.net/publication/377982523_A_Comprehensive_Study_on_Software_Evolution_in_Plan-Driven_and_Agile_Methodologies

[26] Divanjana I, Wickramanayake B, Naotunna D, Lankasena N, Liyanage H. Web-Based Visual Acuity Testing Under Low-Resource Settings. Adv Technol. 2025;4.

[27] Al-Shatnawi AM, Alsharafat WS, Al-Fawaz BM, Alshobul R. An Integrated Expert System for Mobile Phone Failure Diagnosis. Int J Comput Appl. 2014;98:1-7.

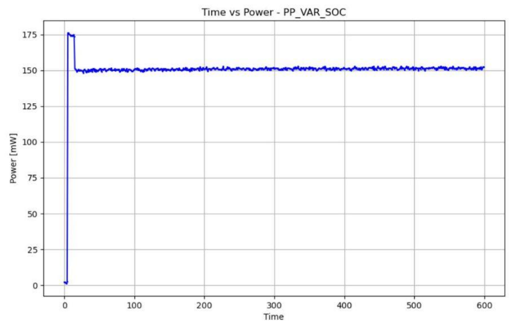
[28] Abidin SA, Suhaimi NS. Decision Support System for Diagnosing Mobile Phone Failure Using Rule-Based Technique. J Adv Res Comput Appl. 2019;17:14-19.

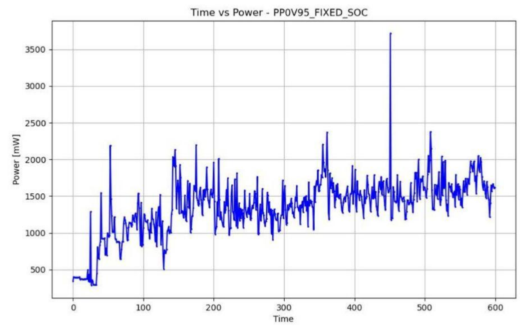
[29] Cui J, Wang Y. A Novel Approach of Analog Circuit Fault Diagnosis Using Support Vector Machines Classifier. Measurement. 2011;44:281-289.

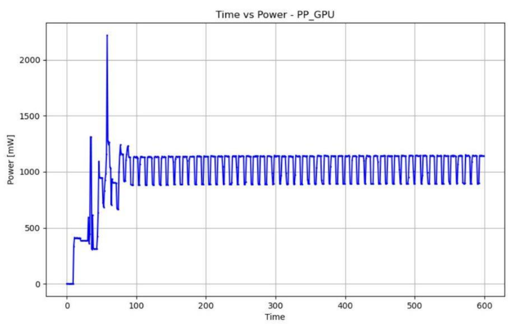
[30] Moser G, Costamagna P, De Giorgi A, Greco A, Magistri L, et al. Joint Feature and Model Selection for Svm Fault Diagnosis in Solid Oxide Fuel Cell Systems. Math Probl Eng. 2015;2015:1-12.

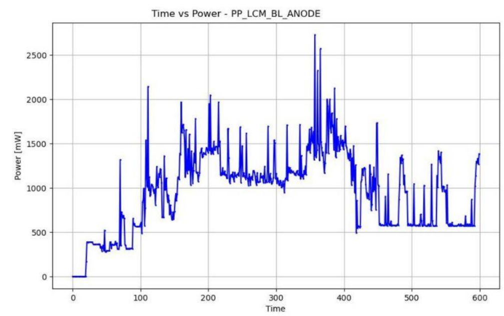
[31] Sabry AH, Nordin FH, Sabry AH, Abidin Ab Kadir MZ. Fault Detection and Diagnosis of Industrial Robot Based on Power Consumption Modeling. IEEE Trans Ind Electron. 2020;67:7929-7940.

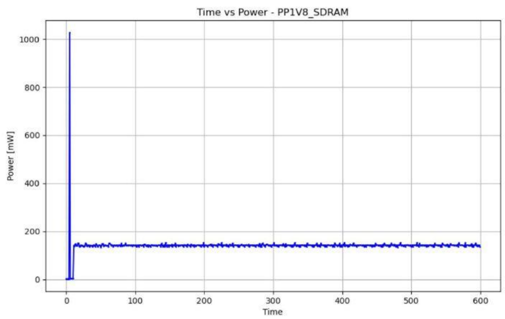
Appendix A: Faulty power consumption pattern of each power rail

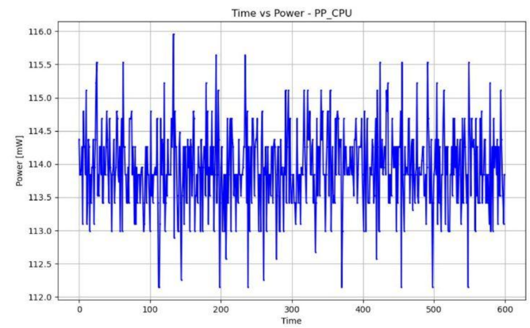


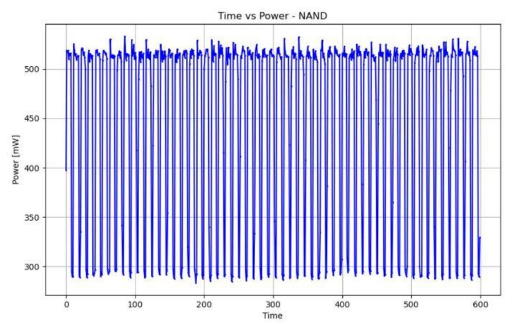


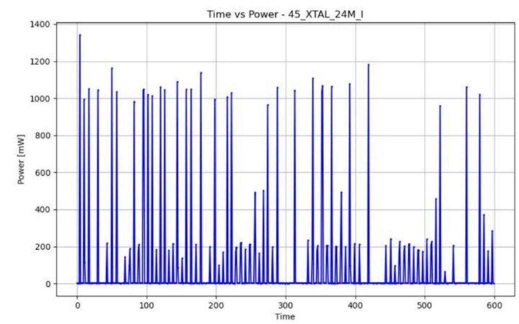


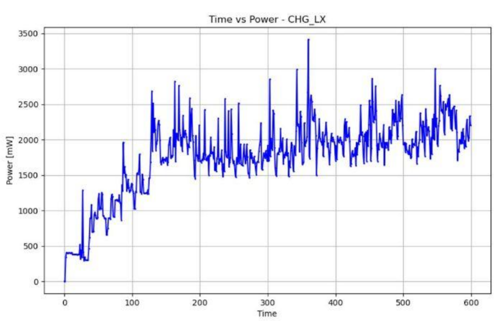


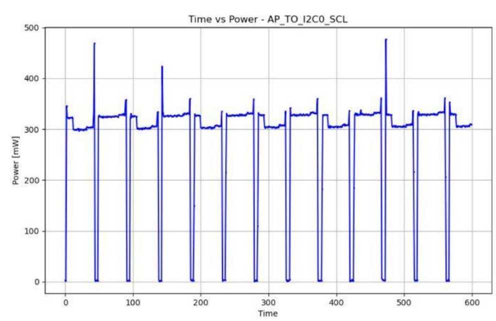

Time vs Power - PP1V8_SDRAM
Power [mW]
Time

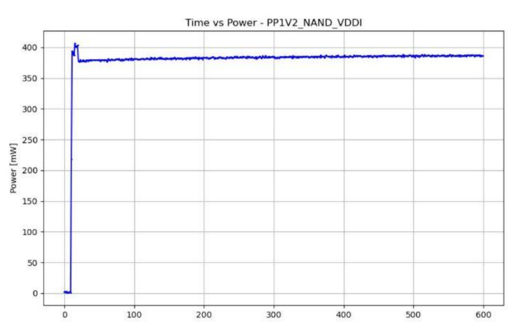

Time vs Power - PP_CPU
Power [mW]
Time

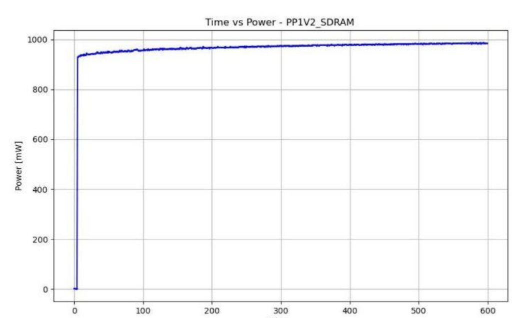

Time vs Power - NAND
Power [mW]
Time


Time vs Power - 45_XTAL_24M_I
Power [mW]
Time

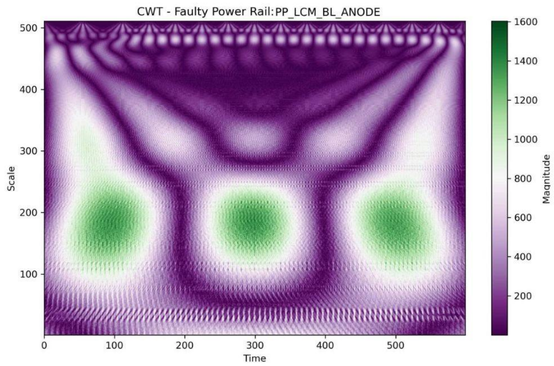

Time vs Power - CHG_LX
Power [mW]
Time

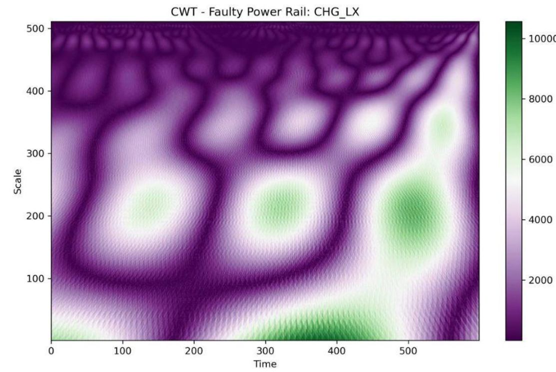

Time vs Power - AP_TO_I2C0_SCL
Power [mW]
Time

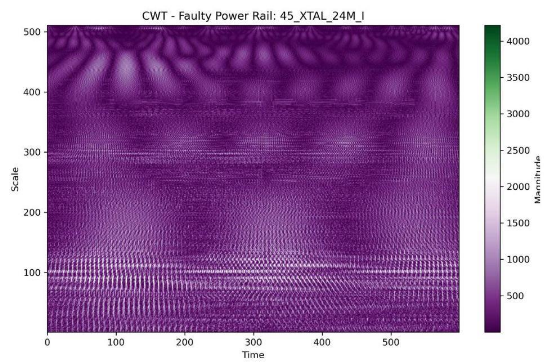

Time vs Power - PP1V2_NAND_VDDI
Power [mW]
Time

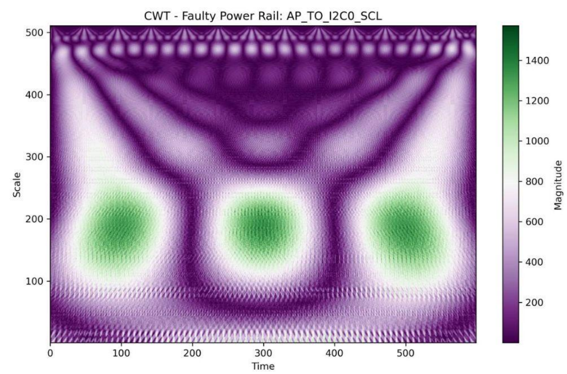

Time vs Power - PP1V2_SDRAM
Power [mW]
Time

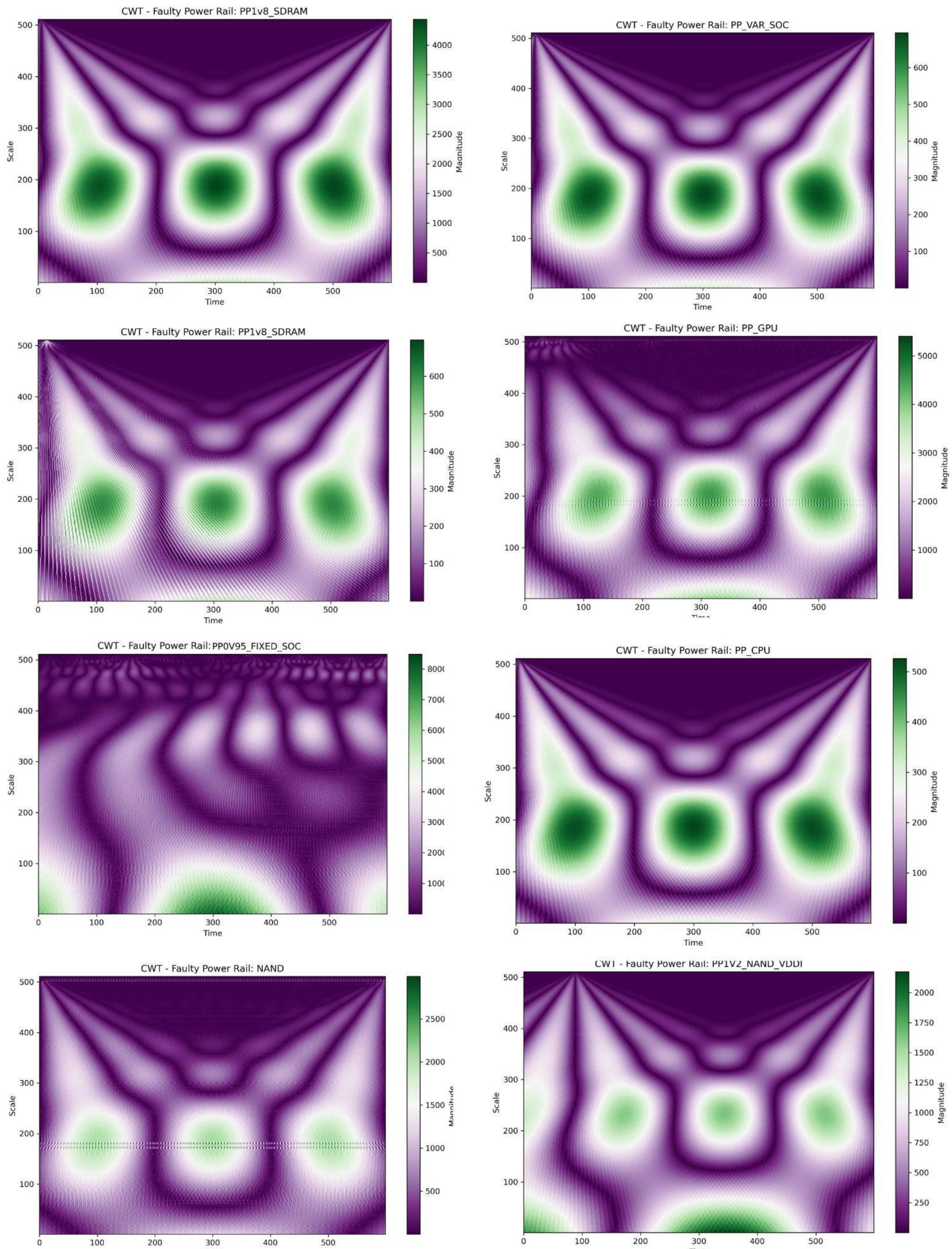
Appendix B: CWT images of 12 power rails










CWT - Faulty Power Rail: PP1v8_SDRAM
CWT - Faulty Power Rail: PP_VAR_SOC
CWT - Faulty Power Rail: PP1v8_SDRAM
CWT - Faulty Power Rail: PP_GPU
CWT - Faulty Power Rail: PP0V95_FIXED_SOC
CWT - Faulty Power Rail: PP_CPU
CWT - Faulty Power Rail: NAND
CWT - Faulty Power Rail: PP1V2_NAND_VDDI
Scale
Time
Magnitude