The Environmental Temperature Effects on Common IoT Devices

1st Michael Barcelo
College of Engineering and Computing
Florida International University
Miami, FL, United States
mbarc024@fiu.edu

2nd Manikanta Nagulapalli
College of Engineering and Computing
Florida International University
Miami, FL, United States
mnagu005@fiu.edu

I. Introduction

A. Importance of Power Consumption Monitoring

The Internet of Things (IoT) has revolutionized how we interact with technology, offering a vast array of intelligent devices that can monitor and control our environment. However, while these devices are designed to be energy-efficient, they still require a constant power supply to operate. As the number of IoT devices continues to grow, so does the need to optimize their energy usage and sustainability. One critical aspect of this optimization is understanding how environmental temperature affects the power consumption of IoT devices.

The IoT has become a part of our everyday life, linking gadgets and machines to allow a wide range of applications. However, because batteries or other restricted power sources often supply these gadgets, their usefulness and dependability must be improved. As a result, reducing power consumption in IoT devices is critical for maximizing functionality and usability.

Increased battery life is one of the most significant benefits of optimizing power usage in IoT. Reduced energy usage allows IoT devices to run for extended periods without needing battery replacements or recharges, resulting in lower maintenance costs and an enhanced user experience. IoT devices are more practical since they can be depended on for prolonged periods.

Furthermore, power optimization in IoT might be cost-effective. Lowering energy usage can result in cheaper energy bills and longer battery life, resulting in cost savings for both people and enterprises. Reduced energy consumption can also help the environment by lowering the carbon footprint associated with IoT device use.

Moreover, optimizing power consumption in IoT devices can increase dependability. Power difficulties are a significant cause of device failure; however, by lowering energy usage, IoT devices may run more consistently and lessen the likelihood of such failures. This enhanced dependability can boost consumers' trust and confidence in IoT devices, leading to more adoption and usage. Scalability is another crucial benefit of optimizing power usage in IoT devices. Power-efficient IoT devices may be deployed in large-scale deployments without regular battery changes or recharge since they consume less energy. Due to cost savings and a more convenient user experience, IoT devices become a more appealing alternative for industrial and commercial applications.

Finally, improving security by optimizing power usage in IoT devices. IoT devices that use less power are less vulnerable to attacks that exploit power-related weaknesses, lowering the risk of security breaches. This is especially relevant in critical applications where security is paramount. Increasing battery life, cost-effectiveness, dependability, scalability, environmental friendliness, and security are all advantages of optimizing power usage in IoT devices. In addition, IoT devices run more efficiently, cost-effectively, and reliably if their energy consumption is reduced, making them more practical and appealing for various applications.

This article describes a study experiment examining the link between outdoor temperature and power consumption in three Raspberry Pi devices: the Raspberry Pi Zero, the Raspberry Pi 3b, and the Raspberry Pi 4b. The experiment evaluates each device's power consumption under different temperature circumstances, offering vital information into how temperature impacts energy usage, performance, and sustainability.

The results of this experiment can have significant implications for organizations that rely on IoT devices in their operations. Organizations can optimize their device usage, reduce energy costs, and improve device design and performance by identifying how environmental temperature affects power consumption, leading to more efficient and sustainable IoT solutions.

Section 2 discusses the significance of monitoring power usage in IoT devices and the link between temperature and power consumption. Section 3 outlines the experiment's experimental setup and methods. Section 4 gives the experiment's results, and Section 5 explores the consequences of these findings. Finally, Section 6 brings the work to a close by summarizing the findings and outlining future research objectives.

B. Experiment Description

Our experiment investigates the relationship between ambient temperature and power consumption in three Raspberry Pi devices: the Raspberry Pi Zero, the Raspberry Pi 3b, and the Raspberry Pi 4b. To do this, we developed a system that assesses each device's power consumption under changing environmental circumstances.

The setup consists of an INA219 sensor, which measures the voltage and current consumed by each Raspberry Pi device. The sensor is connected to a Raspberry Pi Model 3b, which logs the power consumption readings over time. To expose the devices to varying environmental temperatures, we will place each Raspberry Pi device outside and use a DHT22 temperature and humidity sensor to measure the temperature at each measurement point.

During the experiment, the Raspberry Pis will be exposed to a range of temperatures during the 12 hours starting at 8 AM EST in Miami, FL. The Raspberry Pis will report their temperature and voltage every minute, and at the end of the experiment, we will take an hourly average to express the data better.

The experiment's data will give important insights into the link between external temperature and power usage in IoT devices. By examining the data, we can determine how temperature impacts power consumption, performance, and sustainability in each Raspberry Pi device. These data may assist enterprises in optimizing device utilization, lowering energy costs, and improving device design and performance, resulting in more efficient and sustainable IoT solutions.

II. Background

A. *Monitoring Consumption Challenges*

Monitoring power usage may be complex, and present tools and methodologies have significant drawbacks. The lack of uniformity in monitoring power consumption is a severe difficulty, as different devices may have varied power profiles and energy usage patterns. This can make comparing power consumption statistics across devices and establishing uniform benchmarking measures challenging.

Another challenge is the need for high-precision and high-resolution measurement tools, as slight variations in power consumption can significantly impact the performance and longevity of IoT devices. In addition, existing tools such as multimeters and power meters may need to be more sensitive to accurately measure power consumption, especially for devices that consume very little power.

There are also challenges associated with integrating power monitoring into the design of IoT devices. This requires careful consideration of factors such as power supply design, component selection, and software optimization, which can be time-consuming and resource-intensive.

Finally, there are difficulties in analyzing and interpreting power usage statistics. Because of the sheer number and complexity of data created by IoT devices, identifying patterns and trends in power usage can be challenging, especially when data is collected over lengthy periods.

To solve these difficulties, researchers and developers are investigating novel power monitoring and management methods, such as using machine learning algorithms to discover energy use trends and developing new sensors and measurement tools better suited for monitoring IoT devices. These projects aim to improve the efficiency and dependability of IoT devices while ensuring that they can continue functioning successfully in the face of shifting energy requirements and restrictions.

B. *Why We Chose the Raspberry Pi*

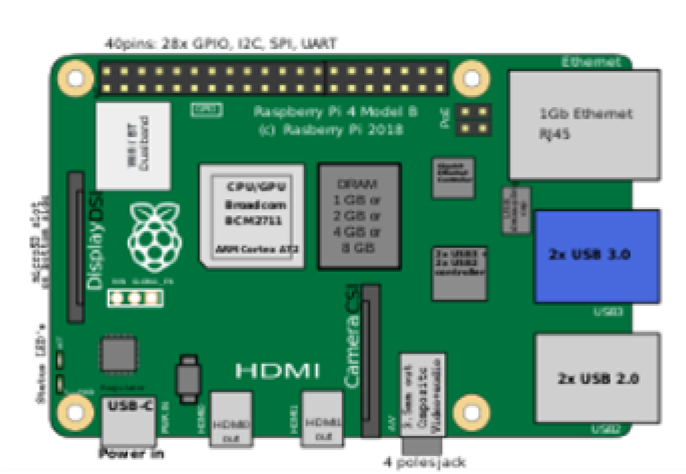

Fig. 1. Raspberrypi.

We chose Raspberry Pis because they are affordable, small, and power-efficient computers that can efficiently run various operating systems, including Linux. Additionally, they have a large community of users, which means a wealth of information is available on how to use and configure them for various applications.

In our scenario, we wanted a device that could connect to the DHT22 and INA219 sensors while running Python code. Therefore, the Raspberry Pi was a wise choice since it has a lot of GPIO ports that can be used to connect to sensors and execute Python programs natively. Furthermore, the Raspberry Pi's low power consumption made it suitable for our experiment, necessitating long-term power usage monitoring.

C. *Why We Chose the INA219 and DHT22*

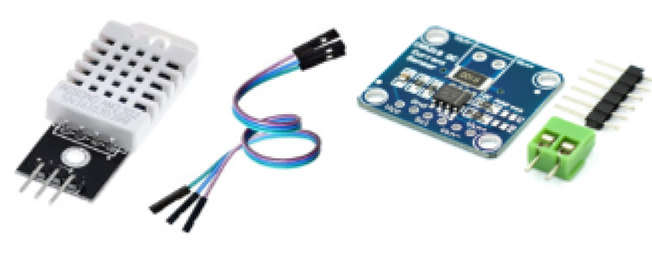

Fig. 2. DHT22 and INA219.

Because of their usability and availability, the INA219 and DHT22 sensors were utilized in the experiment. The INA219 sensor precisely detects the voltage and current, making it an excellent choice for monitoring power consumption. It also works well with the Raspberry Pi and has a Python library for interacting with it. The DHT22 sensor, on the other hand, was chosen because it can monitor temperature and humidity, two crucial environmental factors that may influence power usage. In addition, it is straightforward to use with the Raspberry Pi and has a Python library.

III. Methods

A. Hardware Used

The Raspberry Pi Zero, Raspberry Pi 3b, and Raspberry Pi 4b are all single-board computers used for IoT applications. Each device has features and specs which affect how much power they consume.

The Raspberry Pi Zero is the smallest and most basic of the three devices. It has a 1 GHz single-core CPU, 512 MB of RAM, and supports Bluetooth and Wi-Fi connectivity. Due to its small size and limited processing power, the Raspberry Pi Zero has a relatively low power consumption compared to the other models. Its power consumption is typically around 100-150 mA when idle and up to 500 mA under heavy load.

The Raspberry Pi 3b, released in 2016, is famous for IoT applications due to its balance of processing power and energy efficiency. It has a 1.2 GHz quad-core CPU and 1 GB of RAM and supports Bluetooth and Wi-Fi connectivity. However, its power consumption is higher than the Raspberry Pi Zero, typically around 200-300 mA when idle and up to 1.34 A under heavy load.

The Raspberry Pi 4b, released in 2019, is the most powerful of the three devices. It has a 1.5 GHz quad-core CPU and 4 GB of RAM and supports Bluetooth 5.0 and Wi-Fi 802.11ac connectivity. It also includes two micro-HDMI ports, enabling dual 4K displays. However, its power consumption is higher than the previous models, typically around 600-800 mA when idle and up to 3 A under heavy load.

The INA219 is a high-side current and power monitor that allows for precise measurement of a device's voltage, current, and power consumption. It uses the I2C protocol and has a 12-bit ADC for high-resolution measurements. The INA219 is designed to be easy to use and can measure current up to 3.2 A and voltage up to 26 V with an accuracy of 1 percent. It also has programmable alert thresholds and can be configured to power down the system in an overcurrent condition automatically. Overall, the INA219 is a versatile and reliable sensor for monitoring power consumption in various applications.

The DHT22 sensor is a digital temperature and humidity sensor that monitors the relative humidity and temperature of the surrounding air using a capacitive humidity sensor and a thermistor. It communicates through a single-wire digital interface and gives precise and reliable measurements with an accuracy of 2 percent RH and 0.5°C. In addition, the DHT22 is simple to use and operates across a wide voltage range (3-5.5 V). It also consumes less power, making it suited for use in IoT devices that run on batteries. Overall, the DHT22 is a popular choice for monitoring temperature and humidity in various contexts, including homes, workplaces, greenhouses, and industrial settings.

The Raspberry Pi is a popular single-board computer used in IoT applications due to its low cost and adaptability. It is frequently used as a central hub for IoT devices, offering a data gathering, processing, and transmission platform. The DHT22 sensor is a common sensor for measuring temperature and humidity in IoT applications. It gives reliable readings and is reasonably affordable, making it a popular choice for amateur and commercial applications. The sensor may be readily connected to a Raspberry Pi through GPIO pins, allowing the Pi to gather and interpret temperature and humidity data. The INA219 is a high-precision current sensor that measures both DC and AC. It is frequently used in IoT applications to monitor power usage, enabling power management and efficiency. Using an I2C connection, the sensor may be attached to a Raspberry Pi, allowing the Pi to monitor and analyze power usage statistics. The Raspberry Pi, DHT22 sensor, and INA219 sensor are all useful IoT tools. The Raspberry Pi is a flexible data-gathering and processing platform. The DHT22 and INA219 sensors may be used to monitor and analyze temperature, humidity, and power consumption data, allowing for improved optimization and efficiency in IoT devices.

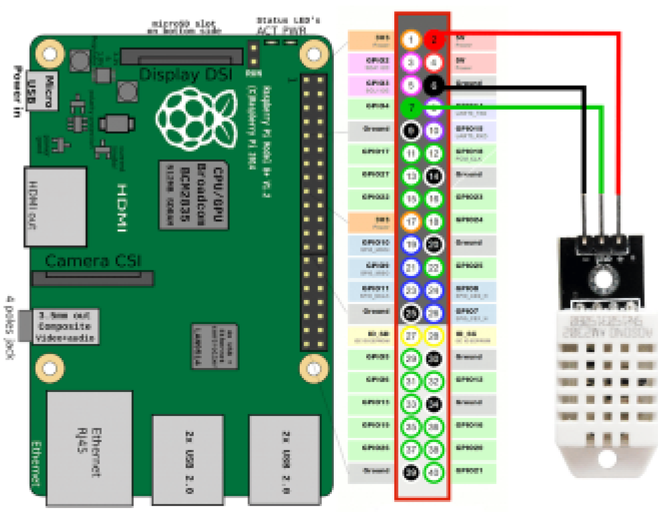

Fig. 3. Wire diagram of DHT22.

B. Software Used

Python is a high-level, interpreted programming language that was first released in 1991. It is widely used for various applications, including web development, scientific computing, data analysis, artificial intelligence, and automation. Python is known for its simplicity, readability, and ease of use. Its syntax emphasizes code readability and allows programmers to express concepts in fewer lines of code than in other languages. It also has a large and active community of users and developers who have contributed to developing numerous libraries, frameworks, and tools, making it easier to build complex applications with Python. Overall, Python is a versatile and powerful language suitable for both beginners and experienced programmers and is widely used in various industries and fields.

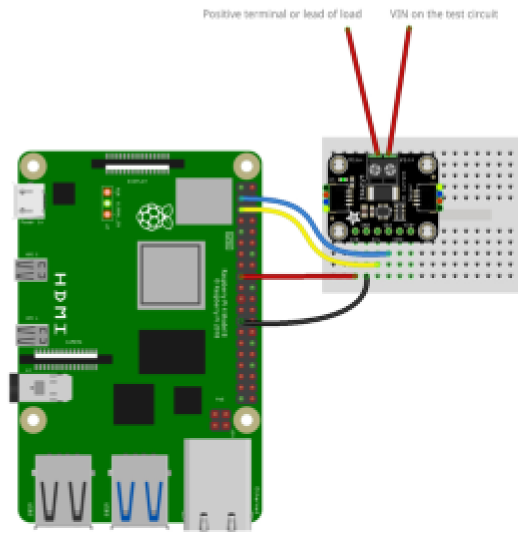

Fig. 4. Wire diagram of INA 219.

The INA219 Python library is a Python module that provides an interface to the INA219 high-side DC sensor. It allows developers to easily read and control the INA219 sensor within their Python applications. The library provides methods for configuring the sensor's parameters, such as the voltage and current ranges, and for reading the sensor's output, including the current, voltage, power, and shunt voltage. The INA219 library is compatible with various platforms, including Raspberry Pi, Arduino, and BeagleBone Black, and can be used in various applications, such as energy monitoring, battery management, and power regulation. In addition, the library is well-documented and includes examples that demonstrate how to use the library's methods and functions. Overall, the INA219 Python library is a powerful and easy-to-use tool for monitoring and controlling DC and voltage using the INA219 sensor in Python applications.

The adafruit-circuitpython-dht python library is a Python module that provides an interface to DHT series temperature and humidity sensors. It lets developers easily read and control DHT sensors within their Python applications. The library supports various DHT sensors, including DHT11, DHT22, and AM2302, and provides methods for reading the temperature and humidity data from the sensors. The library uses CircuitPython, a Python implementation optimized for microcontrollers, and can be used in various platforms, including Raspberry Pi, BeagleBone Black, and other single-board computers. In addition, the adafruit-circuitpython-dht library is well-documented and includes examples that demonstrate how to use the library's methods and functions. Overall, the adafruit-circuitpython-dht library is a powerful and easy-to-use tool for measuring temperature and humidity using DHT sensors in Python applications.

Mosquitto is an open-source message broker that implements the MQTT (Message Queuing Telemetry Transport) protocol, a lightweight and efficient communication protocol designed for IoT and M2M (Machine to Machine) applications. Mosquito is designed to be scalable, reliable, and easy to use, and it can be used to build complex IoT systems with multiple devices and sensors. It supports security features, such as Transport Layer Security (TLS) and access control lists (ACLs), to ensure secure and authenticated communication between devices. Mosquitto also provides various message filtering options, including topic-based filtering and message routing, to efficiently manage and route messages between devices. Mosquitto is a popular choice for building IoT systems and is widely used in various applications, including smart homes, industrial automation, and environmental monitoring.

ThingsBoard is an open-source, scalable, and customizable IoT platform for managing and analyzing data from connected devices. It provides various features, including device management, data collection, analytics, and visualization, that allow users to monitor and control their IoT systems easily. With ThingsBoard, users can create custom dashboards and widgets to display real-time data and analytics, set up rules and alerts to trigger actions based on certain events or conditions and integrate with third-party systems using various APIs and protocols. In addition, ThingsBoard supports data security and privacy with device authentication, encryption, and access control.

C. Setup

To carry out our experiment, we needed to attach the DHT22 and INA219 sensors to each Raspberry Pi device. We connected the VCC pin to a 3.3V pin on the Pi, the GND pin to a ground pin on the Pi, and the data pin to a GPIO pin on the Pi for the DHT22 sensor. It was critical to write down the PIN we utilized because it was required in the code.

To connect the INA219 sensor to each Raspberry Pi device, we first connected the INA219 breakout board's VCC pin to either a 3.3V or 5V pin on the Pi, depending on the voltage range we wished to monitor. We then connected the breakout board's GND pin to a ground wire on the Pi. Finally, we connected the breakout board's SCL and SDA pins to the Pi's equivalent SCL and SDA wires. The PINs for SCL and SDA may differ based on the Raspberry Pi model. For example, the SCL and SDA pins of the Pi Zero are GPIO3 and GPIO2, respectively, although, on other versions, they may be GPIO2 and GPIO3 or vice versa.

We could read and process data from the sensors using Python code. We used Python libraries such as the adafruit-circuitpython-dht library and the INA219 python library, which provided easy-to-use functions for interacting with the sensors. It was important to refer to the sensor datasheets and code examples to ensure proper configuration and usage of the sensors. By successfully connecting the DHT22 and INA219 sensors to the Raspberry Pi devices and using Python code to read and process data, we collected the necessary data for our experiment.

After successfully collecting the necessary data from the DHT22 and INA219 sensors using the Raspberry Pi devices and Python code, we needed to transmit the data to our data management platform, ThingsBoard. To do this, we used

Mosquitto, an open-source message broker that implements the MQTT protocol.

```
import energyusage

# user function to be evaluated
def recursive_fib(n):
    if (n <= 2): return 1
    else: return recursive_fib(n-1) + recursive_fib(n-2)

energyusage.evaluate(recursive_fib, 40, pdf=True)
# returns 102,334,155
```

Fig. 5. Flow diagram of Data.

First, we had to install and configure Mosquitto on the Raspberry Pi devices. We then wrote Python code to publish the data from the sensors to a specific MQTT topic. Finally, we specified the ThingsBoard MQTT broker as the endpoint for the messages. ThingsBoard subscribes to the specified MQTT topic and automatically receives the data published by the Raspberry Pi devices.

Once ThingsBoard receives the data, it is processed and stored in our data management system. We could visualize the data in real-time using the ThingsBoard dashboard, allowing us to monitor the power consumption and temperature data during our experiment. Overall, using Mosquitto and ThingsBoard allowed us to easily and efficiently transmit and manage the data collected from our sensors.

IV. Results

A. Presenting the Findings

According to the experiment results, there is a significant relationship between ambient temperature and power utilization for the IoT devices under consideration. When the temperature climbed, so did the electricity use. However, the temperature did not appear to influence power utilization with the Raspberry Pi Zero.

Time of Day	Average Temp (F)	APC for RPi 4b	APC for RPi 3b	APC for RPi Zero
8:00	72	600 mA	380 mA	140 mA
9:00	76	608 mA	384 mA	142 mA
10:00	78	612 mA	385 mA	140 mA
11:00	80	612 mA	385 mA	139 mA
12:00	81	613 mA	388 mA	140 mA
13:00	80	612 mA	387 mA	140 mA
14:00	80	611 mA	388 mA	140 mA
15:00	82	613 mA	390 mA	139 mA
16:00	81	612 mA	387 mA	141 mA
17:00	79	611 mA	388 mA	140 mA
18:00	78	610 mA	385 mA	140 mA
19:00	77	609 mA	384 mA	141 mA
20:00	77	609 mA	385 mA	142 mA

*Humidity was not taken into account

Fig. 6. Table of Comparing Results.

Several reasons might explain the relationship between temperature and electricity usage. For example, the circuitry's resistance increases as the temperature rises, resulting in more significant power usage. Alternatively, greater temperatures may make components work harder, producing higher power demand. However, more research would be required to determine the particular causes of this association.

In addition to the correlation between environmental temperature and power consumption, we also observed variations between the different models of Raspberry Pi. For example, the Raspberry Pi 3B+ and 4B exhibited similar power consumption patterns, gradually increasing power consumption as the temperature increased. However, the Raspberry Pi Zero showed no significant change in power consumption with varying temperatures.

These findings indicate that temperature influences the power consumption of IoT devices, especially in warmer locations. This has significant consequences for the design and deployment of IoT devices, as power consumption is critical in determining battery-powered device lifespan. Therefore, further study should examine how temperature impacts power consumption and ways to minimize temperature's influence on IoT devices.

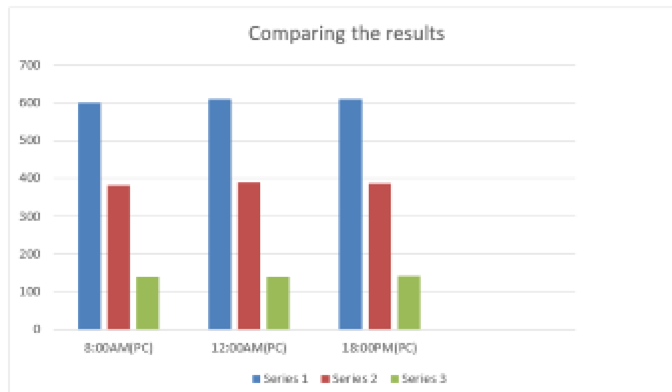

Fig. 7. Table of Comparing Results.

Overall, the findings have significant implications for IoT devices' real-world design and operation. For example, designers and engineers may create more efficient and long-lasting devices capable of operating in various conditions if they better understand the link between environmental parameters and power consumption.

V. Conclusion

In conclusion, our experiment aimed to investigate the relationship between temperature and power consumption for IoT devices. We used DHT22 and INA219 sensors connected to Raspberry Pi devices to collect temperature and power consumption data. The data was then transmitted to ThingsBoard for further analysis.

Our results showed a clear correlation between temperature and power consumption, with power consumption increasing as temperature increases. However, it was interesting to note that for the Raspberry Pi Zero, the temperature did not affect power consumption. This may be due to the unique properties of the Raspberry Pi Zero or limitations in our experiment, such as wind or cloud coverage.

One of our experiment's shortcomings was the inability to control wind and cloud covering. Because the sensors were left outside in the elements, they were sensitive to wind and cloud coverage fluctuations, which may have influenced our results.

Despite these constraints, our experiment was a success in terms of fulfilling its objectives. We gathered and sent temperature and power usage data using Python programming and sensor packages. This information may help guide future research and enhance the design and performance of IoT devices.

Overall, our experiment highlights the significance of considering environmental elements when building and testing IoT devices. As the IoT devices grow more common and are incorporated into our daily lives, it is critical to understand how they interact with their surroundings and how environmental variables might impact their performance. Our investigation sheds light on these parameters and establishes the groundwork for future study.

VI. Future Work

In this experiment, we successfully monitored the power consumption and temperature of IoT devices using Raspberry Pis, DHT22 sensors, and INA219 sensors. However, there is still much work to be done in this field.

One limitation of our experiment was that we could not control wind and cloud coverage, which could have affected our results. Therefore, in future experiments, we could consider these variables and possibly use weather shields or other protective measures to shield the devices from the effects of wind.

Another area for future work would be to investigate using different sensors and techniques for monitoring power consumption. While the INA219 sensors provided accurate measurements for our experiment, other sensors or techniques could provide even more accurate or granular data. For example, it may be possible to use sensors that measure the power consumption of individual components within the IoT devices rather than just the overall power consumption.

Additionally, we used Raspberry Pis as our monitoring device, but other types of devices could also be used. For example, microcontrollers such as Arduino boards or ESP8266/ESP32 modules could be used instead. Each type of device has its strengths and weaknesses, and future work could investigate which devices are best suited for different monitoring applications.

In terms of data transmission and analysis, we used the Mosquitto broker and the ThingsBoard platform to transmit and visualize our data. Platforms like AWS IoT, Google Cloud IoT, or Microsoft Azure IoT could also be used. Additionally, machine learning techniques could be used to analyze the data and make predictions or recommendations based on the data.

The development of more energy-efficient IoT devices is another subject of future study. Our research found a strong link between temperature and power consumption, which has implications for the energy efficiency of IoT devices in real-world applications. Future studies might look into ways to improve the energy efficiency of IoT devices, such as adopting low-power components or enhancing thermal management techniques.

Overall, there is still much work to be done in IoT device monitoring and energy efficiency. Our experiment was just one small step in this direction, and there are many avenues for future research and development.