A Design of Remote IoT-Based Non-Obstructive Temperature Monitoring System for Electric Vehicle Smart Chargers

Mousa Alwaraki   
Qatar University, Doha, Qatar  
[mousa.alwaraki@gmail.com](mailto:mousa.alwaraki@gmail.com)

Loay Ismail  
Qatar University, Doha, Qatar  
[loay.ismail@qu.edu.qa](mailto:loay.ismail@qu.edu.qa)

Ahmed Massoud  
Qatar University, Doha, Qatar  
[ahmed.massoud@qu.edu.qa](mailto:ahmed.massoud@qu.edu.qa)

*Abstract*— Electric Vehicles (EVs) are seen as a solution to the increasing carbon emissions due to gas-powered cars. Thus, extensive research has been done in optimizing the efficiency and ensuring reliable operation of EV chargers. However, there is a void in tackling temperature monitoring using non-obstructive methods and manipulating charging accordingly to protect the battery’s health and ensure no safety complications. This paper aims to suggest a prototype using an Internet of Things (IoT) system to create such an infrastructure. The paper suggests a low-cost prototype using a micro-controller and thermal camera that allows for completely remote temperature monitoring, and with some additions can allow manipulation of the smart charger, remotely while being connected to the internet. The research method of this paper is to create individual projects using different IoT services and determining how best to utilize them in terms of the prototype. Some aspects of the prototype can be run on a local network if no network with an internet connection exists. This may be beneficial for charges in less economically developed areas, where an active internet connection at charging stations is not feasible. The use of the prototype on a local network is undesired as monitoring and controlling of charger must be made on the same network, and so removes the complete remote monitoring aspect.

Keywords— Electric Vehicles, Temperature Monitoring, IoT, Low-cost micro-controller, ESP8266, Node-RED, Homebridge, Blynk

# Introduction

The steady rise in carbon emissions has been brought to the centerpiece of attention of the general public. This has led to a knock-on effect where carbon emissions have been at the forefront of many politicians’ policies, such as the UK government banning the sale of gas-powered car engines by 2040 and striving for an 80% reduction of CO2 emissions in the transport sector by 2050 [1]. This, in turn, leads to pressure on different markets to innovate to reduce carbon emissions. The transportation sector contributes around a quarter of the global greenhouse gas emissions [2]. In California, it is estimated that if 77% of all transport miles were replaced with electric vehicles (EVs), there would be a 94% reduction in carbon emissions compared to the carbon emissions of 1990. [3]. This emphasises the importance of the transport sector, in terms of reducing carbon emissions. One of the markets that has made significant strides recently within the transportation sector is the automobile market. With soaring predicted demand for EVs [1], car manufacturers are racing to create EVs that fit different consumers' demands. The main energy storage system used for EVs is Lithium-ion (Li-ion) batteries, due mainly to their high energy density, low self-discharge rate, and lack of memory effect [4]–[7].

In recent years thermal predicaments from battery systems are believed to be the main cause of fatal fire accidents [8]. The monitoring and controlling of Li-ion batteries are crucial to protect the battery pack from exploding in extreme circumstances [6]. Thermal issues can also cause lower energy performance [4], storage failure [5], and shorter lifetime [9] in Li-ion batteries.

Another roadblock delaying the transfer of consumers to electric cars from conventional cars is the charging time, mainly due to regular interruptions to charging [10]. The introduction of intelligent chargers that detect charging interruptions has helped ease the concern of interruptions. Likewise, considerable advancements have been made in the issue of charging speed. With the introduction of fast chargers, charging time can be cut by as much as four times [11]. However, this raises serious concerns over the safety implication of fast charging on both battery lifetime and catastrophic failures [12].

Unlike previous research, in this paper, an IoT-based non-obstructive temperature monitoring system for EV smart chargers is presented, which can be monitored remotely. The prototype used takes advantage of low-cost microcontrollers and creates an IoT-based temperature monitoring system. The demand for IoT systems has seen a steady rise over the past decade, thanks to the fall of microcontrollers prices and the advancements in technology. This paper addresses the integration of multiple IoT services and presents an approach for utilizing that in temperature monitoring system for EV smart chargers. These different services include the Blynk platform, Firebase Cloud Storage, Node-RED, MQTT, and Homebridge.

## Literature Review

Much research has been conducted pertinent to optimizing EV batteries to ensure efficiency, bigger storage and slower discharge[3][13][14], including work on State-of-Charge (SoC) estimation and its impact on battery health [7],[15]–[18]. As the importance of temperature monitoring of EV batteries is entering the mainstream, research in temperature monitoring is growing [19],[20]. The key temperature that needs monitoring is the core temperature of the battery and not the surface temperature; therefore, an estimation of the core temperature must be made using the surface temperature of the battery [21]. Publications within the scope of EV charging systems have been split into three main streams. The ideal placement of charging infrastructure [22], the creation of a reliable and safe charging infrastructure in relation to the national power grid [23], and lastly, attempting to reduce inconvenience of EV charging for customers[24]. This paper relates closer to the second research stream, as it looks to create a seamless customer experience with smart charging while also relating to the safety of the charging infrastructure in regard to EV smart chargers and battery health. The prototype relies mainly on the use of microcontrollers in order to alert users and read sensor measurements. There are many research papers on the use of microcontrollers with sensors and actuators, but they mainly focus on smart home automation [25]. This links with this paper as many of the concepts are ported into this project but with different applications.

# Experimental Work

## Employed Technologies

There are three main technologies that are used for this prototype, namely the Arduino Software, the ESP8266 board, and an AMG8833.

### Arduino Software

Arduino Software is an IDE that is used in this project to write the programs for the ESPs.

### ESP8266

The ESP8266 NodeMCU is a low-cost microcontroller. It is based on the ESP-12 but contains the NodeMCU firmware. The NodeMCU firmware is an open-source IoT platform that allows running different applications on the ESP8266 board.

The ESP8266 contains an 802.11 b/g/n WIFI transceiver [26]. Not only does this allow the ESP to communicate with the internet through a network, but it also allows the ESP to act as a Soft Access Point (AP). This means that the ESP can create a network of its own to allow other devices to connect to it, create a locally hosted webpage that allows other devices on the network to receive sensor readings from sensors, or manipulate actuators connected to the ESP. It can also act simultaneously; this is particularly crucial if an extensive network of ESPs is needed in an IoT project.

### AMG8833

The AMG8833 is an 8X8 low-cost thermal camera. It has an accuracy of ­±2.5°C [27]. However, as its resolution is low, it may not be effective on a large-scale charging station, where the charging station may need to monitor all charging points at once. This may also cause problems to arise with the placement of the AMG8833 in a smart charger. However, it serves the function in the scope of a prototype.

## IoT Applications

Several IoT services are available to work with the ESP to create an interface in which the user can receive sensor readings. Some of these IoT services require an active internet connection and therefore can work over any network as long as the network is connected to the internet, whilst some only work over a local area network (LAN).

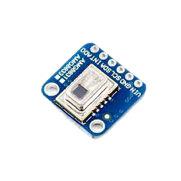
### Webserver

The AMG8833 readings are displayed on a local web server, hosted by the ESP8266 using the Soft AP WiFi mode. The readings are updated automatically at a chosen rate, using a JavaScript Function. This removes the need for a refreshing webpage to get newer readings.

Thermal Camera

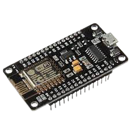
(AMG8833)

Passes thermal readings using Arduino



Microcontroller

(ESP8266)



Hosts LAN

Readings accessible to users on LAN network

Displays thermal map & readings

Webserver

### Blynk Application

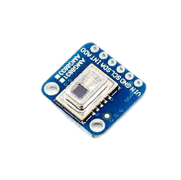
Blynk is an IoT service that helps in controlling hardware remotely using mobile applications to control IoT systems. It is connected to its private cloud server and can include data analytics and machine learning. In addition, it can be linked with hardware such as a Raspberry Pi or an ESP using libraries such as in Arduino.

Each Blynk project creates a unique Auth token used to verify user permission and connect the Blynk server to the correct hardware (and vice-versa). This creates a secure connection and so sensitive data and actuator manipulation can be used. As Blynk uses its own servers, the smartphone with the Blynk project does not need to be connected to the same network as the hardware connected to the project. This allows remote monitoring and manipulation if an active internet connection is available. However, users can host a local/private Blynk server which then removes the energy limitation. It cannot be controlled using different networks out of the box when hosted locally, but it is achievable using other software. Hosting a local Blynk server allows the creator to bypass the limit on widgets enforced. However, that is unneeded in the scope of the prototype.

Thermal Camera

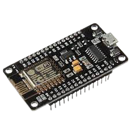
(AMG8833)

Passes thermal readings using Arduino



Microcontroller

(ESP8266)



Internet connection + Arduino Blynk library

Blynk Server

User accesses readings on Blynk App connected to internet

Blynk app



### NodeRED

Node-RED is an open-source browser-based editor that can be used to write applications with little-to-no programming easily. It uses visual flows with nodes to create sequential steps to run a program. It is then deployed to the host device; in this project, a Raspberry Pi is used. Node-RED has many advantages beyond code-less programming. Primarily, the libraries available, whether they are native libraries or third-party libraries. The most notable library which is used consistently throughout the majority of IoT systems is the Node-RED Dashboard. The Node-RED Dashboard allows the user to create dashboards of live data provided to the host device and actuators that can be controlled.

For Node-RED to work in an anticipated way, the host device is required to also act as an MQTT broker. MQTT (Message Queuing Telemetry Transport) is a publish-subscribe messaging transport service. It is lightweight and does not require extensive resources or network bandwidth, making it ideal for micro-controllers and IoT systems. In this case, the broker, Raspberry Pi, acts as a middleman between the publishers and the subscribers. Multiple devices can be connected as a publisher, subscriber, or both.

In the case of the prototype, the ESP is connected as a publisher for the readings provided from the ESSP8266 by the AMG8833. On the other side, Node-RED acts as a subscriber for the AMG8833 sensor readings. While MQTT acts on a local web page, the sensor readings and LED cannot be read or manipulated on different networks. Methods of connecting MQTT to the internet are possible by using two brokers with a bridge. However, that is beyond the scope of the prototype.

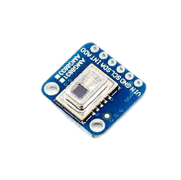
Diagram

Description automatically generated

Thermal Camera

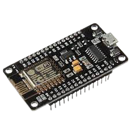
(AMG8833)

Passes thermal readings using Arduino



Microcontroller

(ESP8266)



MQTT hosted on a Raspberry Pi connected to a LAN network by ESP8266

Node-RED



User accesses readings on Node-RED Dashboard anywhere

MQTT subscribers and publishers

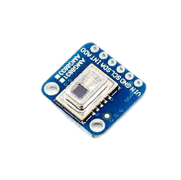
### Firebase

Firebase is a cloud storage service hosted by Google. However, it contains many more tools that can be used, particularly in the realm of application development. Firebase tools include authentication, notification handling, analytics, database, file storage, and many others. The Real-Time Database (RTD) provided in Firebase will be the only service used in this project. As seen in its name, the RTD is a database that is synced in real-time, so its contents can be extracted to multiple devices with it being entirely up to date. It requires an active internet connection and cannot be connected over a local network.

Thermal Camera

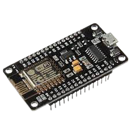
(AMG8833)

Passes thermal readings using Arduino



Microcontroller

(ESP8266)



Internet connection

Firebase

User accesses readings on apps built on a Firebase backend



Backend

Firebase is best utilized as a back end for applications. While the website can be used for readings and manipulating actuators, it is not user-friendly, and therefore is not recommended. Therefore, for the usage of this prototype, no application will be built to take advantage of Firebase. However, if the prototype is converted into a user product, an application built using firebase as a back end is recommended.

### Homebridge

Homebridge is a service that turns IoT system components into ‘smart’ devices that can be recognized using Apple’s Home, which allows for native Siri integration out of the box. It can be downloaded onto a host device, such as a Raspberry Pi, then integrated with Node-RED or any MQTT service.

In order to connect MQTT devices to Homebridge, a plugin named ‘Homebridge Mqttthing’ needs to be added. After the plugin is added and Homebridge is restarted, each device needs to be added to the plugin with the correct topic subscriptions. After that, the Homebridge website can be used on a local network using the Homebridge UI plugin, which runs like the Node-RED dashboard.

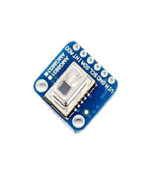
To use Homebridge to its capabilities, the host device's network must be connected to the internet. This allows for Siri and Home Application to be able to get readings and manipulate actuators. However, the iOS device must be connected to the same network as the host device.

One limitation of the Homebridge application is that Homebridge uses the MQTT server, Homebridge UI, and even the Home app/Siri have to be on the same network as the MQTT broker (the Raspberry Pi in this case) in order for them to function. This depletes the value of Homebridge, as the whole purpose of the project is to control actuators and get sensor readings at any location at any given time. However, there are a couple of ways to bypass this issue.

Thermal Camera

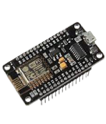
(AMG8833)

Passes thermal readings using Arduino



Microcontroller

(ESP8266)



MQTT hosted on a LAN network by ESP8266

Home-Bridge

MQTT hosted on a Raspberry Pi connected to LAN and internet

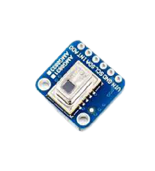
User accesses readings via Siri and iOS Home App only on same Network as Raspberry Pi



One way to bypass the need to be on the same network is to have an apple device (iPad, Apple TV, or HomePod devices) connected as a ‘Home Hub’ on the MQTT network, which will then allow the control from Apple devices/Siri to work on any network.

Thermal Camera

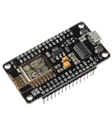
(AMG8833)



Passes thermal readings using Arduino

Microcontroller

(ESP8266)



MQTT hosted on a LAN network by ESP8266

Home-Bridge

User accesses readings via Siri and iOS Home App with internet connection

MQTT hosted on a Raspberry Pi connected to LAN and internet

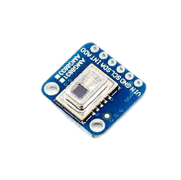


Home-Hub Device

The other way to bypass the network issue is to use the Google Home app. In order to be able to use Google Home, another plugin needs to be added to Homebridge, called ‘Homebridge Google Smart Home’. The google home app does not require a hub to be connected, and so will work out of the box with google assistant.

Thermal Camera

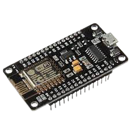
(AMG8833)



Passes thermal readings using Arduino

Microcontroller

(ESP8266)



MQTT hosted on a LAN network by ESP8266

Home-Bridge w/ ‘Google Smart Home’ plugin



MQTT hosted on a Raspberry Pi connected to LAN and internet

User accesses readings via Google Assistant and Google Home App with internet connection

# Prototype

The prototype includes all the IoT applications mentioned in the report to create a complete and robust system. As seen above in Figure 14, most of the applications can be run on any network. The webserver is the only one requiring the user to be on a local network. However, the data displayed in the webserver is being mirrored on Node-RED.

Of the IoT applications in the prototype, Firebase is the only one that requires an internet connection to function. Blynk can be ran on a local server. However, Blynk will run on the Blynk server for this prototype and require an internet connection.

The prototype currently does not notify the user when parameters are met, such as increasing the battery's temperature. However, this capability can be easily added using Firebase for mobile notifications or by connecting an actuator to create an audio or visual warning, depending on the preferences of the charging station.

ESP8266 + AMG8833

LAN

Internet

Webserver

Blynk

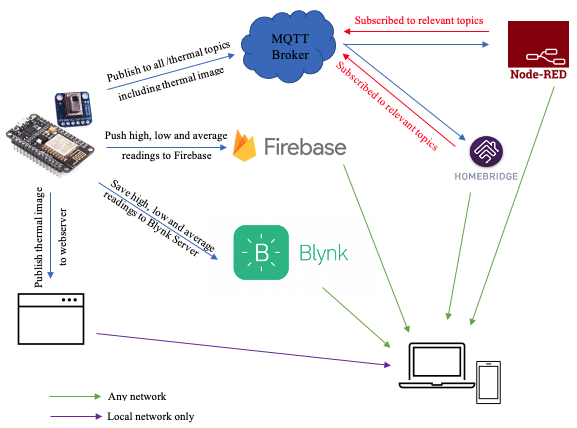
Firebase

MQTT

Node-RED

Homebridge

Devices



# Conclusion

The principal software prototype is completed and can be duplicated using Arduino Software, the IoT applications, an ESP8266 board, and an AMG8833. However, the prototype must be expanded, and the correct hardware placements must be made before the prototype could be implemented on a charging station.

One of the main issues of the prototype is that the thermal camera used (AMG8833) is of low resolution, which could cause multiple issues to arise when the prototype is to be implemented. Therefore, a higher resolution camera is suggested to be used at the implementation stage. Additionally, an internet connection should be present at the charging stations for the prototype to be functioning to its maximum capability, with all the desired IoT applications.

A function that needs to be added to the prototype is the need to alert the user or the charging stations at specific parameters. So

as the highest temperature, average temperature, or the temperature of one of the pixels of the thermal camera exceeds a pre-set temperature, a notification is sent to the user device or the charging station. Another notifying method could be an audio or visual method to alert the user or the charging station of the potential need of disconnecting the charger. This is a relatively easy addition but is vital to ensuring overall safety.

Finally, an addition that could be made for the prototype is the control of the charging actuator. Be that to change charging mode, from slow to fast charge, or completely stop or start charging. It can also be used to show the battery status, voltage, and amplitude being received by the battery and other desired parameters. However, achieving this is beyond the scope of this paper.

# References

[1] T. A. Vincent and J. Marco, "Development of Smart Battery Cell Monitoring System and Characterization on a Small-Module Through In-Vehicle Power Line Communication," in IEEE Access, vol. 8, pp. 220658-220671, 2020. doi: 10.1109/ACCESS.2020.3043657.

[2] F. Princiotta and S. Sikdar, "Trends in Transportation Greenhouse Gas Emissions," in *Advances in Carbon Management Technologies*, CRC Press, 2020, pp. 359-375.

[3] H. Kim and K. G. Shin, "DESA: Dependable, Efficient, Scalable Architecture for Management of Large-Scale Batteries," in IEEE Transactions on Industrial Informatics, vol. 8, no. 2, pp. 406-417, May 2012. doi: 10.1109/TII.2011.2166771.

[4] Y. Xiao, "Model-Based Virtual Thermal Sensors for Lithium-Ion Battery in EV Applications," in IEEE Transactions on Industrial Electronics, vol. 62, no. 5, pp. 3112-3122, May 2015. doi: 10.1109/TIE.2014.2386793.

[5] C. Zhu, Y. Shang, F. Lu, Y. Jiang, C. Cheng and C. Mi, "Core Temperature Estimation for Self-Heating Automotive Lithium-Ion Batteries in Cold Climates," in IEEE Transactions on Industrial Informatics, vol. 16, no. 5, pp. 3366-3375, May 2020. doi: 10.1109/TII.2019.2960833.

[6] M. A. Hannan, M. M. Hoque, A. Hussain, Y. Yusof and P. J. Ker, "State-of-the-Art and Energy Management System of Lithium-Ion Batteries in Electric Vehicle Applications: Issues and Recommendations," in IEEE Access, vol. 6, pp. 19362-19378, 2018. doi: 10.1109/ACCESS.2018.2817655.

[7] R. Klein, N. A. Chaturvedi, J. Christensen, J. Ahmed, R. Findeisen and A. Kojic, "Electrochemical Model Based Observer Design for a Lithium-Ion Battery," in IEEE Transactions on Control Systems Technology, vol. 21, no. 2, pp. 289-301, March 2013. doi: 10.1109/TCST.2011.2178604.

[8] J. Zhang, L. Zhang, F.Sun and Z. Wang “An Overview on Thermal Safety Issues of Lithium-ion Batteries for Electric Vehicle Application” in *IEEE Access,* vol. 6, pp. 23848-23863, May 2018. Doi: 10.1109/ACCESS.2018.2824838

[9] M. Ben-Marzouk, S. Pelissier, G. Clerc, A. Sari and P. Venet, "Generation of a real-life battery usage pattern for electrical vehicle application and aging comparison with the WLTC profile," in IEEE Transactions on Vehicular Technology. doi: 10.1109/TVT.2021.3077671.

[10] D. Kettles, "Electric Vehicle Charging Technology Analysis And Standards," *Electric Vehicle Transportation Center,* 2015, FSEC Report Number: FSEC-CR-1996-15

[11] M. Christopher, A. Shabbir, B. Ira, B. Andrew, C. Barney, D. Fernando, D. Eric J, J. Andrew N, K. Matthew, M. Anothony, M. Andrew, M. Manish, P. Ahmad, S. Don, and S. Matthew, "Enabling fast charging – Introduction and overview," *Journal of Power Sources,* vol. 367, pp. 214-215, 2017.

[12] Sebastian, SS, Dong, B, Zerrin, T, et al. “Adaptive fast charging methodology for commercial Li-ion batteries based on the internal resistance spectrum.” *Energy Storage*. 2020; 2:e141. doi: 10.1002/est2.141

[13] H. Oman, "Battery developments that will make electric vehicles practical," in IEEE Aerospace and Electronic Systems Magazine, vol. 15, no. 8, pp. 11-21, Aug. 2000. doi: 10.1109/MAES.2000.861768.

[14] K. W. E. Cheng, B. P. Divakar, H. Wu, K. Ding and H. F. Ho, "Battery-Management System (BMS) and SOC Development for Electrical Vehicles," in IEEE Transactions on Vehicular Technology, vol. 60, no. 1, pp. 76-88, Jan. 2011. doi: 10.1109/TVT.2010.2089647.

[15] Z. B. Omariba, L. Zhang and D. Sun, "Review of Battery Cell Balancing Methodologies for Optimizing Battery Pack Performance in Electric Vehicles," in IEEE Access, vol. 7, pp. 129335-129352, 2019. doi: 10.1109/ACCESS.2019.2940090.

[16] X. Hu, J. Jiang, D. Cao and B. Egardt, "Battery Health Prognosis for Electric Vehicles Using Sample Entropy and Sparse Bayesian Predictive Modeling," in IEEE Transactions on Industrial Electronics, vol. 63, no. 4, pp. 2645-2656, April 2016. doi: 10.1109/TIE.2015.2461523.

[17] J. S. Goud, K. R and B. Singh, "An Online Method of Estimating State of Health of a Li-Ion Battery," in IEEE Transactions on Energy Conversion, vol. 36, no. 1, pp. 111-119, March 2021. doi: 10.1109/TEC.2020.3008937.

[18] B. Rente et al., "Lithium-Ion Battery State-of-Charge Estimator Based on FBG-Based Strain Sensor and Employing Machine Learning," in IEEE Sensors Journal, vol. 21, no. 2, pp. 1453-1460, 15 Jan.15, 2021. doi: 10.1109/JSEN.2020.3016080.

[19] M. Salameh, S. Wilke, B. Schweitzer, P. Sveum, S. Al-Hallaj and M. Krishnamurthy, "Thermal State of Charge Estimation in Phase Change Composites for Passively Cooled Lithium-Ion Battery Packs," in IEEE Transactions on Industry Applications, vol. 54, no. 1, pp. 426-436, Jan.-Feb. 2018. doi: 10.1109/TIA.2017.2763588.

[20] K. Xu, H. Li and Z. Liu, "ISOMAP-Based Spatiotemporal Modeling for Lithium-Ion Battery Thermal Process," in IEEE Transactions on Industrial Informatics, vol. 14, no. 2, pp. 569-577, Feb. 2018. doi: 10.1109/TII.2017.2743260.

[21] D. Zou et al., "Temperature Estimation of Lithium-Ion Battery Based on an Improved Magnetic Nanoparticle Thermometer," in IEEE Access, vol. 8, pp. 135491-135498, 2020. doi: 10.1109/ACCESS.2020.3007932.

[22] F. Baouche, R. Billot, R. Trigui and N. El-Faouzi, “Efficient Allocation of Electric Vehicles Charging Stations: Optimization Model and Application to a Dense Urban Network”, in *IEEE Transactions on Intelligent Transportation Systems,* vol. 6, no. 3, pp. 33-43, September 2014. doi: 10.1109/MITS.2014.2324023

[23] S. R. Khutwad and S. Gaur, "Wireless charging system for electric vehicle," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 2016, pp. 441-445, doi: 10.1109/SCOPES.2016.7955869.

[24] C. Jin, J. Tang and P. Ghosh, "Optimizing Electric Vehicle Charging: A Customer's Perspective," in IEEE Transactions on Vehicular Technology, vol. 62, no. 7, pp. 2919-2927, Sept. 2013, doi: 10.1109/TVT.2013.2251023.

[25] M. Hasan, M. H. Anik and S. Islam, "Microcontroller Based Smart Home System with Enhanced Appliance Switching Capacity," 2018 Fifth HCT Information Technology Trends (ITT), 2018, pp. 364-367, doi: 10.1109/CTIT.2018.8649518.

[26] T. Thaker, "ESP8266 based implementation of wireless sensor network with Linux based web-server," 2016 Symposium on Colossal Data Analysis and Networking (CDAN), 2016, pp. 1-5, doi: 10.1109/CDAN.2016.7570919.

[27] V. Manuel Ionescu and F. Magda Enescu, "Low cost thermal sensor array for wide area monitoring," 2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2020, pp. 1-4, doi: 10.1109/ECAI50035.2020.9223193