Milestone 2: ESP-IDF 32 Sensor Reading and Server Communication

Adam Damreh

University of North Carolina at Charlotte

***Abstract*— This document showcases the initial application of the ESP-IDF 32 board through sensor use and additional communication to a server.**

I. Introduction

In order to utilize the ADR server an understanding of the ESP board is needed. For this project, a ESP32-WROOM-32 board is utilized, with an AHT10 sensor, RGB screen, and UART communication. For the purpose of this milestone, we are utilizing the AHT10 humidity and temperature sensor, and then communicating that information to an https server.

II. Initial Sensor Readings

After connecting the board via Arduino IDE, necessary libraries and dependencies were installed to make sure the board works properly. Included is the initial code and output from the serial monitor to confirm the success of reading from the AHT10 sensor

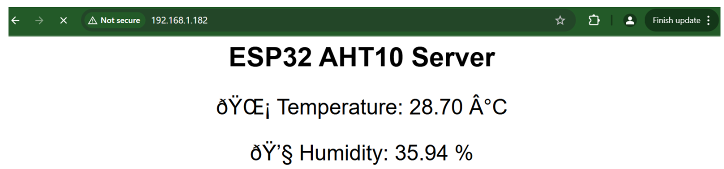
```
#include <Wire.h>
#include <Adafruit_AHTX0.h>

Adafruit_AHTX0 aht;

void setup() {
  Serial.begin(115200);
  Serial.println("Testing AHT Sensor...");

  Wire.begin(27, 33);  // SDA = IO27, SCL = IO33

  if (!aht.begin()) {
    Serial.println("AHT sensor not found! Check wiring.");
    while (1);
  }
}

void loop() {
  sensors_event_t humidity, temp;
  aht.getEvent(&humidity, &temp);

  Serial.printf("Temperature: %.2f °C, Humidity: %.2f %%\n", temp.temperature, humidity.relative_humidity);

  delay(2000);
}
```

```
Temperature: 27.43 °C, Humidity: 36.88 %
Temperature: 27.41 °C, Humidity: 36.89 %
Temperature: 27.40 °C, Humidity: 36.85 %
Temperature: 27.39 °C, Humidity: 36.84 %
Temperature: 27.41 °C, Humidity: 36.87 %
Temperature: 27.40 °C, Humidity: 36.87 %
Temperature: 27.39 °C, Humidity: 36.84 %
Temperature: 27.40 °C, Humidity: 36.84 %
```

III. Server implementation

After confirming the output of the sensor, the data must then be communicated via an https server. Here is the result:

```
🌡 Temperature: 28.47 °C, 💧 Humidity: 35.86 %
🌡 Temperature: 28.47 °C, 💧 Humidity: 35.85 %
🌡 Temperature: 28.39 °C, 💧 Humidity: 35.99 %
🌡 Temperature: 28.45 °C, 💧 Humidity: 35.83 %
🌡 Temperature: 28.46 °C, 💧 Humidity: 35.88 %
```



IV. Conclusion

In conclusion, the board was able to properly communicate with the initialized server and update to display temperature and humidity.

References

[1] "ESP32 DHT Web Server Project" Youtube, Uploaded by DIYTechRush, 16 December 2022, https://www.youtube.com/watch?v=_GohoygyCNE

Annotated Bibliography / Article Summaries

Augello, A., P. Gallo, E. R. Sanseverino, G. Sciumè, and M. Tornatore. "A Coexistence Analysis of Blockchain, SCADA Systems, and OpenADR for Energy Services Provision." *IEEE Access*, vol. 10, 2022, pp. 99088-99101. IEEE, https://doi.org/10.1109/ACCESS.2022.3205121 .

Interconnecting systems on the powergrid, due to the sheer scale, requires a combination of technologies. This paper discusses multiple technologies involving the optimization of our energy system. Some technologies discussed are Blockchain, Supervisory Control and Data Acquisition (SCADA) Systems, Internet of Things (IoT), and OpenADR. SCADA currently dominates this area of technology within the power grid but many modern ideas propose a shift to IoT where energy devices may become easier for systems to manage. A detailed description of how OpenADR works, its protocols, how it interacts with IoT and more are discussed in this paper. There are also examples given of systems such as an Italian Model for DR, SCADA systems, Blockchain, and Unified Architecture models. Overall this paper is quite in depth, but the discussion of the wide variety of technologies present in this field prove to be useful. Having a comprehensive understanding of not only OpenADR but the technologies it will work alongside is crucial in maximizing the performance of OpenADR. The paper was published in 2022 making it quite recent and relevant in the discussion of the current state of the OpenADR framework.

"Automated Demand Response." *Medford's Go Green Initiative!*, City of Medford, https://medfordenergy.org/gogreen/automated-demand-response/. Accessed 4 Feb. 2025.

The power grid is a fluid system that goes through peaks and valleys. Peak times such as hot summer days or cold winter nights result in high energy consumption. Large energy consumption days result in not only higher electricity bills but can potentially cause blackouts. Automated Demand Response (ADR) works to increase the power grids efficiency and capacity by optimizing energy using devices during these peak energy usage days. Through signaling connected systems to turn on or off the energy load is able to be decreased, not only at peak times but at all times. Medford, MA has several energy efficient buildings with energy sources such as solar panels. These buildings not only are able to function off of the grid due to their energy generation but are also beginning to implement ADR systems. The usage of ADR systems can better support off grid systems and create more confidence in the ability to independently power buildings.

Electric Power Research Institute (EPRI). *OpenADR-Virtual-Top-Node.* GitHub, February 4, 2025. https://github.com/epri-dev/OpenADR-Virtual-Top-Node.

Implementation example of a virtual top node (TVP) as used in OpenADR.

Fu, Xinwen. *IoT.* GitHub, Accessed 4 Feb. 2025, https://github.com/xinwenfu/IoT.

Implementation example of setting up IoT Security demo on PCB or Breadboard hardware.

"OpenADR & Communications | Building Technology and Urban Systems." *Lawrence Berkeley National Laboratory*, 2019, https://buildings.lbl.gov/openadr-communications. Accessed 4 Feb. 2025.

During the 2002 California electricity crisis OpenADR was developed at

Berkeley Lab. OpenADR is one of 16 Smart Grid Standards supported by the U.S. Department of Energy the National Institute of Standards and Technology Smart Grid Interoperability Standards effort. Specific examples of messages OpenADR uses to communicate between the VTN and VEN are included. These messages show how the system might communicate for identifying the current electrical loads. The OpenADR framework is accepted as an international standard and is presently used across the United States as well as around the world.

Park, M., Kang, M., and Choi, J.Y. "The Research on Vulnerability Analysis in OpenADR for Smart Grid." Data Analytics for Renewable Energy Integration, edited by Woon W., Z. Aung, and S. Madnick, vol. 8817, Springer, 2014, https://doi.org/10.1007/978-3-319-13290-7_4.

As society's dependence on electric power continues to grow, technology continues to look into accommodating the demand. Demand Reduction (DR) is used to manage and reduce the demand for electricity. OpenADR is an open source protocol of DR that is meant to be used within the context of a Smart Grid. This article discusses security weaknesses within OpenADR which largely surround improper coding practices. CERT Java is a set of secure coding rules that identify violations relating to static, complexity, and data flow. Rule OBJ01-J, for example, requires data members be declared as private and made accessible with wrapper methods. At the time of this article, OpenADR was found to violate this rule more than 720 times. Insecurities of this nature may lead to detrimental consequences with real world applications. This article was published in 2014 and OpenADR has likely fixed many of these insecurities. However, the importance of ensuring quality and security from open source software can still be highlighted. The LDRA Testbed and the set of CERT Java policies are excellent examples of tools that can be used to ensure secure and high quality code.

Yassine, A. "Implementation Challenges of Automatic Demand Response for Households in Smart Grids." *Proceedings of the 3rd International Conference on Renewable Energies for Developing Countries (REDEC)*, 2016, pp. 1-6. IEEE, https://doi.org/10.1109/REDEC.2016.7577546.

Demand Response (DR) systems have largely been used in industrial contexts where electrical draw can be extremely large. As Smart Grid systems advance and the number of residential consumers increases on the power grid DR systems become more common in residential settings. OpenADR utilizes VEN gateways that communicate between the consumer and Demand Response Service Providers (VTN) which supplies energy to consumers. With OpenADR, DR systems are able to be automated and remove the need to manually contact users about energy consumption. This article discusses the benefits of OpenADR such as customer control, supporting household contribution to energy conservation, the integration of renewable energy sources, and more. Challenges with implementing OpenADR on a large scale are also discussed, the main four being: privacy, security, scalability, and user acceptance. While an individual household does not have much impact on the power grid the conglomeration of all households does. OpenADR has the potential to have a large impact on energy consumption from residential

households making the challenges
worth overcoming.