Embedded System for Monitoring and Prevention of Hardware/Software Incidents in Data Center Devices

Alberto Pérez-Aboli, Carlos Dafonte, and Ángel Gómez

Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain
Centro de Investigación CITIC - Laboratorio Interdisciplinar de Aplicaciones de la Inteligencia Artificial (LIA2), Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain
Centro de Investigación CITIC - Laboratorio Interdisciplinar de Aplicaciones de la Inteligencia Artificial (LIA2), Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain
Correspondence: alberto.aboli@udc.es; carlos.dafonte@udc.es; angel.gomez@udc.es

DOI: *https://doi.org/10.17979/spudc.000024.25*

Abstract: Real-time monitoring of the status of the devices available in Data Centers (DC) is a critical task to try to provide quality services to customers, but also to detect possible hardware/-software incidents or to act against cyber-attacks. To perform these functions, there are specific system and network monitoring tools on the market that supervise different device functionalities (CPU, memory, storage, network, processes, services, etc.). However, these systems do not include the monitoring of basic elements for the correct operation of these devices, such as environmental parameters (power supply, temperature, humidity, noise, etc.). In this project we have designed and built a prototype embedded system, both hardware and software, to monitor in real time each device inside a rack and capable of integrating a wide variety of sensors. The hardware device is composed of: motherboard, power supply, infrared temperature sensor, noise level sensor, current consumption meter, 220VAC real input voltage meter, electronic module to switch on/off the 220VAC output and WiFi module to send the information.

1 Introduction

A data center or DC is the facility that centralizes the IT infrastructures of an organization. Within these infrastructures are the most important and critical devices for providing services to customers (Web, storage, computation, networks, etc.). The DCs are large facilities with high energy consumption due to the fact that they gather a large number of equipment in the smallest possible space, where temperature and cooling become important in order to maintain the DC in optimal working conditions. Having a more efficient and exhaustive control of the most critical parts of a data center is vital for any corporation.

For the software monitoring of devices and services there are many widely used monitoring tools, among them we can highlight open source solutions such as Nagios (Nagios, 2023) or Zabbix (Zabbix, 2023). This type of tool collects information from each piece of equipment by means of software agents installed in them. The information received is stored and processed in real time for proper display on the application's web interface. In this way, it is possible to see graphically any anomaly or incident detected in the operation of the monitored devices.

For hardware monitoring of the environment there are specific commercial products such as: humidity and temperature sensors (NetBotz APC, 2023a), vibration sensors (NetBotz APC, 2023b), integrated PDU (Power Distribution Unit) systems for voltage control in racks (NetShelter APC, 2023). These types of devices provide certain advantages but also have important limitations: they are closed products, their cost is high, and multiple devices are required to cover different types of measurements.

This project proposes the creation of an embedded device that can monitor the components of a DC individually by means of various types of sensors. In addition, this device will be designed following a modular architecture to facilitate future extensions, should be as small as possible and with a reduced manufacturing cost. The control of each of the equipment will be done in real time and in a non-invasive way: it will be connected directly using the base of the power input socket of the equipment to be monitored.

The system will provide relevant data for data center administrators such as measurements of the noise generated by the equipment's fans, changes in the temperature of the enclosure compared to that of its immediate surroundings, changes in consumption and voltage changes occurring in the 220VAC network. The monitoring of this information, collected in real time, will help in the management and prevention of incidents. The detection of anomalous operation such as changes in power consumption, unusual temperatures, strange noises or vibrations could reveal information about what is happening with the status of the device being monitored.

This device would also help to study the profitability of the equipment, when a device starts to consume too much energy and overheat due to wear and tear, it stops performing as expected at the computational level. This information would help in making decisions for equipment renewal.

2 Project description

Creating a device with the indicated characteristics is not a simple task; it requires both hardware and software design, involving the use of numerous professional tools and equipment to design and manufacture the hardware itself, which will be developed at each stage.

The hardware objective of this project is to create an electronic device that is easy to install, non-invasive and as simple as possible to set up. This device will send critical measurement data in real time on the status of each of the devices/servers within a data center. Among the functionalities that will be available, we can highlight:

- External infrared temperature sensor, which will provide the temperature of the case of the device to be monitored and the temperature of its immediate surroundings.
- Voltage sensor, which will measure the actual voltage at all times at the 220VAC input.
- Current consumption sensor, which will control the actual consumption in amperes (A) of the device to be monitored, including the consumption of the monitoring device itself.
- Microphone of very small size, which will measure the noise/vibration of the case of the device to be monitored, with the idea of capturing the noises of the fan bearings.
- Microcontroller with WiFi, which will be the core of this embedded system, will be in charge of the computational part and of sending telemetry data via WiFi.

The software objective of this project is to obtain all the monitoring data from the aforementioned sensors and send them via WiFi, so that each user can decide how to integrate them into their control system. Specifically, we can cite the following objectives:

- Use a free software focused on IoT to manage this device.
- Develop custom firmware for the sensors of each device module.
- Access the device via IP through the Web browser, using a Web interface; the device can be configured and the monitoring data can be viewed at the same time as it is sent via

WiFi. In the Web interface, the user can configure certain parameters such as the data of his MQTT or HTTP server to receive the data.

3 Development

The main device of the embedded system is the processing unit, a device with built-in WiFi has been selected, all in a single component called SoC. The SoC model selected for our system was the Espressif ESP32-S3-WROOM-1U, 32-bit MCU 2.4 GHz Wi-Fi Bluetooth 5 (LE) dual-core (Espressif, 2023). It offers high performance at a very competitive price. In addition, there are open source solutions for IoT applications that can be used with this SoC type device. The Expressif ESP-S3-DevKitC-1U development board (Figure 1) contains the chosen SoC and will be used to support design development.

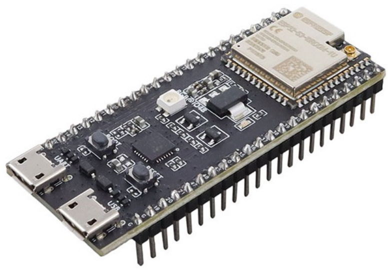
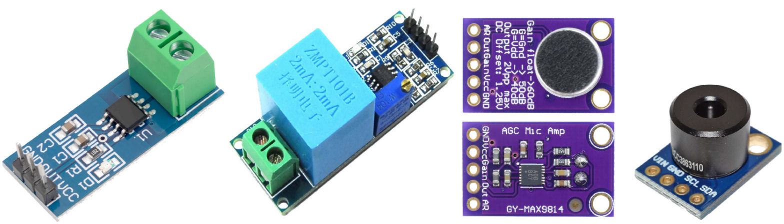


Figure 1: Expressif ESP-S3-DevKitC-1U development board.

In this project we need to implement a series of modules with the corresponding sensors for each functionality. Some of them are the following (Figure 2): a sensor to measure the current consumption in amperes (model ACS712-20A [1]), a sensor to measure the actual value of the AC voltage at the input (model ZMPT101B [2]), a sensor to measure the noise level (model MAX9814ETD+T [3]), an infrared temperature sensor (model MLX90614 [4]).



(a) ACS712-20A (b) ZMPT101B (c) MAX9814 (d) MLX90614

Figure 2: Selected sensor development boards and models.

In summary, the final hardware device will be composed of: motherboard, power supply, infrared temperature sensor, noise level sensor, current consumption meter, actual 220VAC input

---

[1] Allegro ACS712-20A, datasheet available at: *https://www.farnell.com/datasheets/1759100.pdf*

[2] InnovatorsGuru Módulo-ZMPT101B, datasheet available at: *https://www.circuits-diy.com/zmpt101b-ac-single-phase-voltage-sensor-module*

[3] ANALOGDEVICES MAX9814ETD+T, datasheet available at: *https://www.mouser.es/datasheet/2/609/MAX9814-3131693.pdf*

[4] MELEXIS MLX90614, datasheet available at: *https://www.farnell.com/datasheets/3959635.pdf*

voltage meter, 220VAC output on/off electronic module and WiFi module to send the information. During the development of the device it is very important to carry out an adequate design in order to place the different components in such a way that they occupy the smallest possible size without affecting its correct operation. Figure 3 shows the final design of the device and an image of the prototype in the firmware programming phase (Figure 4).

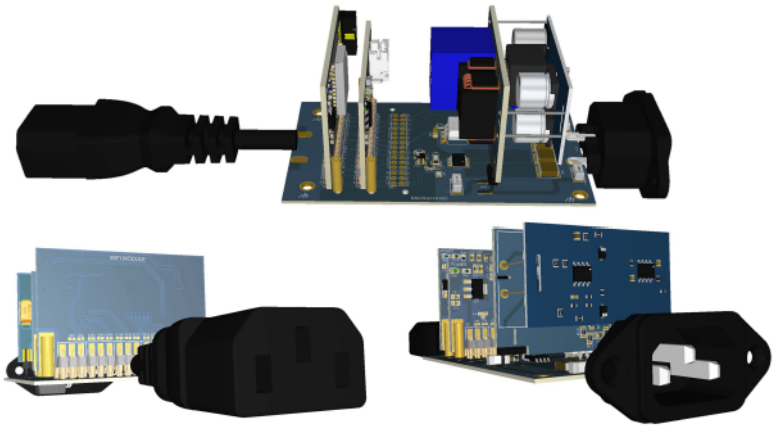


Figure 3: Hardware device design.

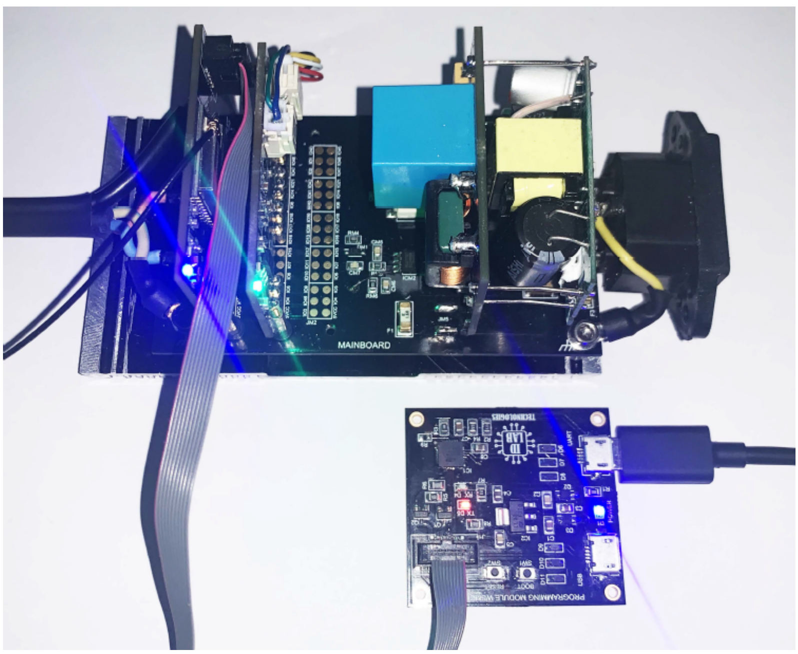


Figure 4: Final prototype in firmware programming phase.

For the development of the system software, we chose to use the open source Tasmota firmware (Tasmota, 2023), specially designed for microcontrollers manufactured by Espressif. Tasmota offers many possibilities of customized configuration, starting from the default configuration you can make modifications in the code to adapt it to your needs and generate the compiled firmware for your project. Among the functionalities it provides is the Web interface (WebUI), which allows access to the device via IP from the Web browser. Using Tasmota's WebUI you can customize some parameters, configure the WiFi or send monitoring data for centralization in environments such as Home Assistant or Zabbix. To control Tasmota devices the MQTT protocol (MQTT, 2023) is used, it is a standard for IoT messaging that works with publishings and subscriptions to a topic through a broker or agent that manages those publishings and subscriptions.

4 Conclusions

This paper presents an embedded system capable of monitoring the environmental parameters of the equipment in a data center individually and assisting technical personnel in the prevention of incidents. It is designed as a low-cost hardware device, small size and modular architecture, allowing future expansion and scalability. The modules that compose it are of expansion type, being able to remove, add or modify modules by configuring the GPIOs to be used. To install the developed device it is only necessary to insert the device between the two input and output sockets of the equipment to be monitored. The information collected by each device will be sent via WiFi to the DC monitoring server. Additionally, through a Web interface included in the firmware of the device, it is possible to access the configuration or visualize the monitoring information of each equipment in real time.

The preliminary results obtained have been very positive, but the implementation and testing in a real controlled environment is proposed as a line of future work, which would allow the results to be studied and compared with other measurement tools.

Acknowledgements

CITIC is funded by the Xunta de Galicia through the collaboration agreement between the Consellería de Cultura, Educación, Formación Profesional e Universidades and the Galician universities for the reinforcement of the research centres of the Galician University System (CIGUS).

Bibliography

Espressif. Espressif Systems: ESP32-S3-DevKitC1U. *https://docs.espressif.com/projects/esp-idf/en/latest/esp32s3/hw-reference/esp32s3/user-guide-devkitc-1.html*, 2023. [Online; accessed 1-September-2023].

MQTT. MQTT: The Standard for IoT Messaging. *https://mqtt.org*, 2023. [Online; accessed 1-September-2023].

Nagios. Nagios Open Source. *https://www.nagios.org/*, 2023. [Online; accessed 1-September-2023].

NetBotz APC. NetBotz Sensors: APC Temperature & Humidity Sensor. *https://www.apc.com/us/en/product/AP9335TH/apc-temperature-humidity-sensor*, 2023a. [Online; accessed 1-September-2023].

NetBotz APC. NetBotz Sensors: NetBotz Vibration Sensor - 12 ft. *https://www.apc.com/us/en/product/NBES0306/netbotz-vibration-sensor-12-ft-/*, 2023b. [Online; accessed 1-September-2023].

NetShelter APC. NetShelter: APC Rack PDU, 2G, metered, 0U, 30A, 200V and 208V, 36 C13 and 6 C19 sockets. *https://www.apc.com/us/en/product/AP8841/apc-rack-pdu-2g-metered-0u-30a-200v-and-208v-36-c13-and-6-c19-sockets/*, 2023. [Online; accessed 1-September-2023].

Tasmota. Tasmota: Open source firmware for ESP devices. *https://tasmota.github.io/docs/*, 2023. [Online; accessed 1-September-2023].

Zabbix. Zabbix - The Enterprise-Class Open Source Network Monitoring Solution. *https://www.zabbix.com/*, 2023. [Online; accessed 1-September-2023].