DOI: 10.20472/IAC.2024.063.007

BOCHUN ZHU

School of Electrical and Electronic Engineering, Singapore Polytechnic, Singapore

FAN JULIE

School of Electrical and Electronic Engineering, Singapore Polytechnic, Singapore

YI FENG YEE

School of Electrical and Electronic Engineering, Singapore Polytechnic, Singapore

LOW-COST PROTOTYPE FOR REALTIME CONDITION MONITORING WITH NOSQL CLOUD NATIVE DATABASE

Abstract:

This paper introduces a low-cost prototype developed using open source hardware and software for realtime condition monitoring in areas, like Industry 4.0, etc. The prototype utilizes Arduino® UNO R4 WiFi with sensors to record, send and store machine / environment data timely to cloud native NoSQL database. The machine / environment data stored in NoSQL is visualized with Grafana from both desktop browser and mobile devices. Realtime alert is sent to personnel in charge for necessary actions to be taken when abnormal machine conditions / environment data are detected. By monitoring key components and environmental factors in a machine or system, detecting deviations from optimal operational parameters, and facilitating timely intervention using this prototype, overall equipment effectiveness could be enhanced, and downtime could be reduced. The adoption of open source software offers several advantages, including cost-effectiveness, flexibility, and community support. This study could provide valuable insights into the development and implementation of low-cost Industry 4.0 condition monitoring solutions using open source software and hardware, addressing the growing demand for affordable yet efficient condition monitoring and maintenance strategies in industrial settings.

Keywords:

Conditional Monitoring, IoT, Industry 4.0, NoSQL Database, Cloud Native Database, Open Source Hardware, Open Source Software

JEL Classification: L17, L86, O14

Introduction

The digital economy is booming. Internet of things (IoT) devices are projected to surge 2.5 times from 2023 to 39 billion by 2029 (UN Trade and Development, 2024). The global machine condition monitoring market size is expected to grow from USD 3.1 billion in 2024 to USD 4.7 billion by 2029 (MarketsandMarkets, 2024). A low-cost solution for realtime condition monitoring is essential for rapid development and mass deployment.

The computing capabilities and storage facilities of the cloud allow IoT device to store, handle, and compute vast data. The most utilized clouds for IoT include AWS (Amazon Web Services), Microsoft Azure, Google Cloud, Open IoT, etc. A cloud native database is a database service which is built, deployed and delivered through cloud platforms.

One of the best practices of cloud native databases is to utilize multicloud deployment, which offers enhanced security, increased storage capacity, effective collaboration, improved cost effectiveness. Deploying the cloud native database across multiple cloud platforms could avoid vendor lock-in, maximize flexibility, ensure redundancy, and mitigate the risk of service interruptions (Navdeep Singh Gill, 2023, Stephen Watts, 2023).

In the next section, a low-cost prototype using open source hardware and software for realtime condition monitoring with NoSQL cloud native database will be introduced.

System Design

The system consists of 3 main parts: an IoT device, an MQTT protocol, and a cloud native database. The overview of the system design is shown in Fig. 1.

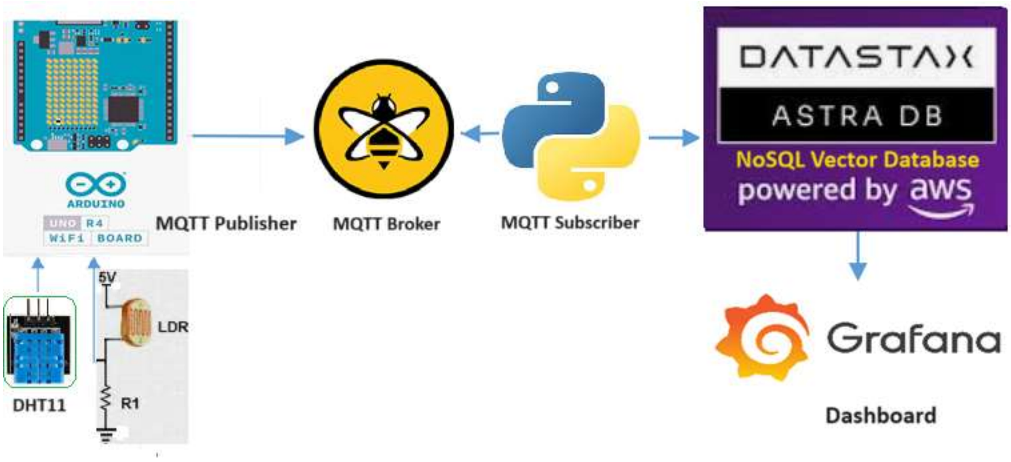


Fig 1. System Design

The Arduino UNO R4 WiFi, which was introduced in Mar 2023, is designed around the 32-bit microcontroller RA4M1 from Renesas with high-performance Arm® Cortex®-M4 core while also featuring an ESP32-S3 for Wi-Fi® and Bluetooth® connectivity. On the hardware side, the compatibility, pinout, voltage, and form factor of Arduino UNO R4 WiFi are unchanged from UNO R3, ensuring maximum hardware and electrical compatibility with existing shields and projects. On the software side, a big effort is being made to maximize backwards compatibility with the most popular Arduino libraries so that users will be able to rely on existing code examples and tutorials. In most cases, libraries and examples will work out-of-the-box, but a few of them which were optimized for the AVR architecture used in R3 will need to be ported (Arduino Team, 2023, Arduino.cc, 2024).

The popularity of Arduino makes Arduino UNO R4 WiFi an ideal choice for building a low cost IoT project. Fig 2. shows the PIN layout of Arduino UNO R4 WiFi board.

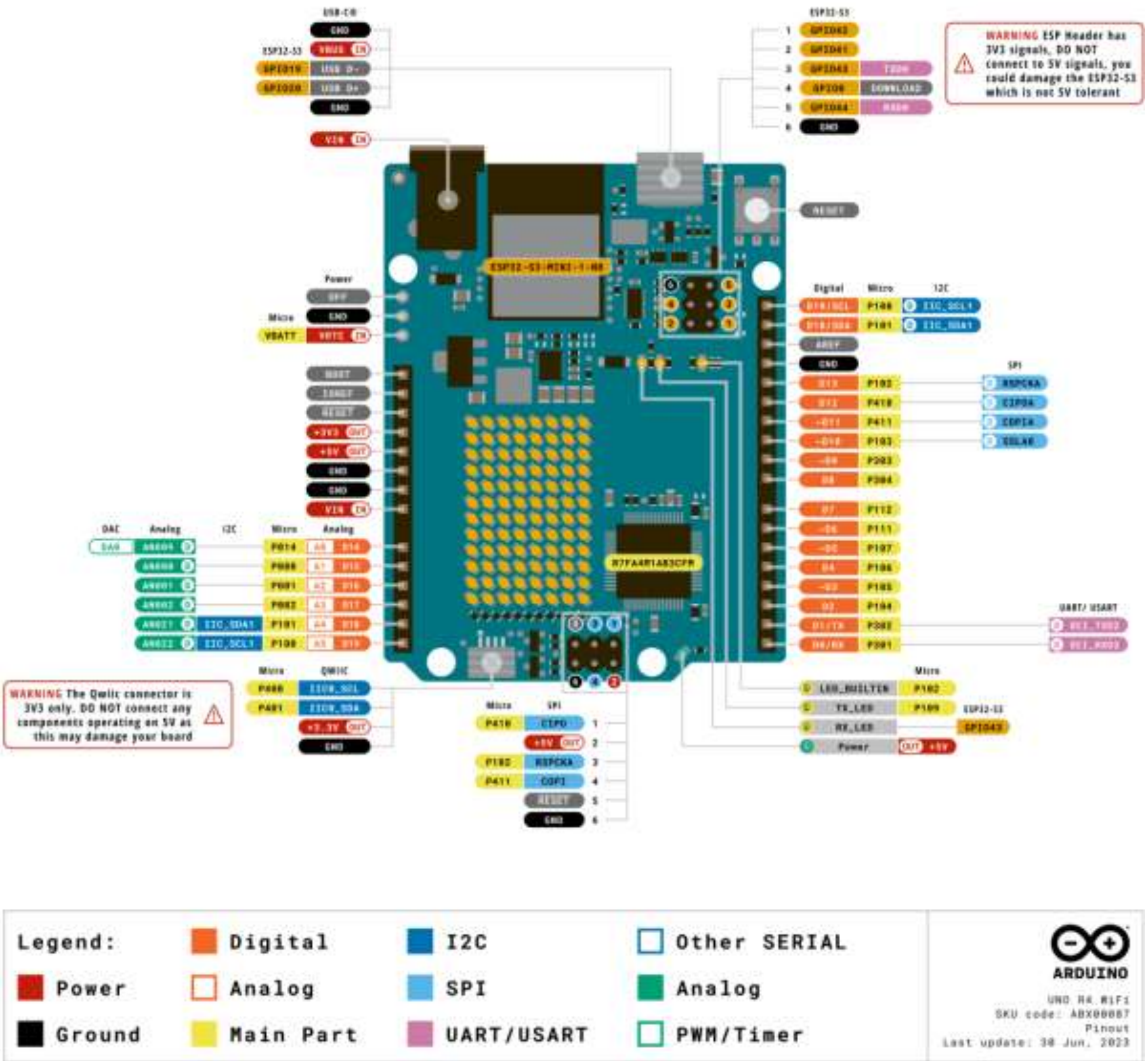
DHT11, the commonly used temperature and humidity sensor, is used to detect environment temperature and humidity. LDR (Light Dependent Resistor) is used to detect ambient brightness.

Considering the multi-cloud supportive nature (Datastax.com, 2024), DataStax NoSQL Astra DB is used to store realtime data in the realtime low-cost prototype for condition monitoring.

DataStax Astra DB is built on Apache Cassandra, and could be deployed on AWS, Google GCP and Microsoft Azure. It maintains compatibility with Apache Cassandra, an open source NoSQL distributed database. In this paper, AWS is used to deploy Astra DB with free tier of usage.

Grafana, an open-source analytics and monitoring solution for databases (Grafana Labs, 2024), is used to visualize the data stored in Astra DB conveniently, for the purpose of analytics, etc.

MQTT (Message Queuing Telemetry Transport) is the de facto data exchange protocol for IoT messaging. Standardized by OASIS and ISO, the MQTT publish / subscribe protocol provides a lightweight, scalable, and reliable way to connect devices, including industrial automation devices in Industry 4.0 over the Internet, (HiveMQ, 2024; PTC, 2024). HiveMQ CE (Community Edition), an open source and free serverless MQTT broker, is used in the system.



Source: https://docs.arduino.cc/resources/datasheets/ABX00087-datasheet.pdf

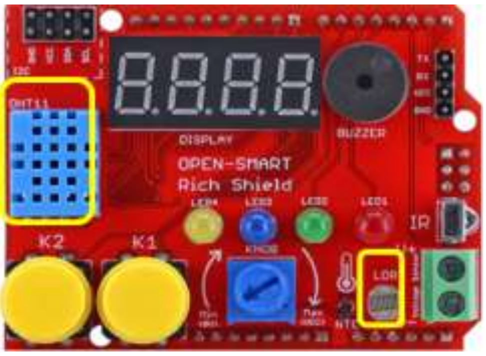
Fig 2. Arduino UNO R4 WiFi Pin Layout

System Implementation

Hardware – The IoT Device

An I/O board Rich Shield (Getmicros, 2024) from Open-Smart shown in Fig. 3, which hosts both DHT11 and LDR, is plugged onto Arduino UNO R4 WiFi board to become the IoT device. The LDR is connected to Arduino UNO R4 WiFi board analogue PIN A2. The DHT11 sensor's data pin is connected to Arduino UNO R4 WiFi board digital PIN D12.

As shown in Fig. 4, the IoT device is neat without additional wires. Only a USB-C cable is needed to connect to Arduino UNO R4 WiFi for loading and debugging the software on the IoT device. Once the program is tested OK for the IoT device, only a USB-C cable or a barrel jack is needed to power the IoT device.



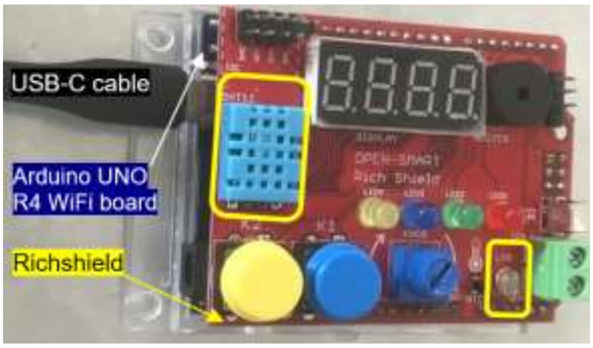
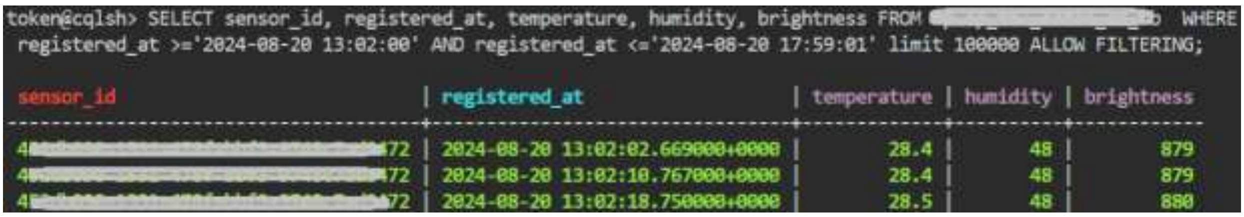


Fig 3. I/O board Rich Shield from Open-Smart Fig 4. IoT device consisting of Arduino UNO R4 WiFi and Rich Shield

Infrastructure – Set up of Cloud Native NoSQL Astra DB Database on AWS

As Astra DB is a cloud native database, creating Astra DB database in AWS is quite straightforward. The CQL (Cassandra Query Language) console could be used easily to test whether the Astra DB database is created successfully, and whether data is inserted into and could be retrieved from it successfully. This will be helpful in later part of software debugging and data visualization using Grafana.



```
token@cqlsh> SELECT sensor_id, registered_at, temperature, humidity, brightness FROM [illegible] WHERE
 registered_at >='2024-08-20 13:02:00' AND registered_at <='2024-08-20 17:59:01' limit 100000 ALLOW FILTERING;

 sensor_id                                | registered_at                   | temperature | humidity | brightness
------------------------------------------+---------------------------------+-------------+----------+------------
 4[illegible]472                          | 2024-08-20 13:02:02.669000+0000 |        28.4 |       48 |        879
 4[illegible]472                          | 2024-08-20 13:02:10.767000+0000 |        28.4 |       48 |        879
 4[illegible]72                           | 2024-08-20 13:02:18.750000+0000 |        28.5 |       48 |        880
```

Fig. 5 CQL console for Astra DB database

When generating application token, "Database Administrator" role is to be chosen, as shown in Fig. 6 below:

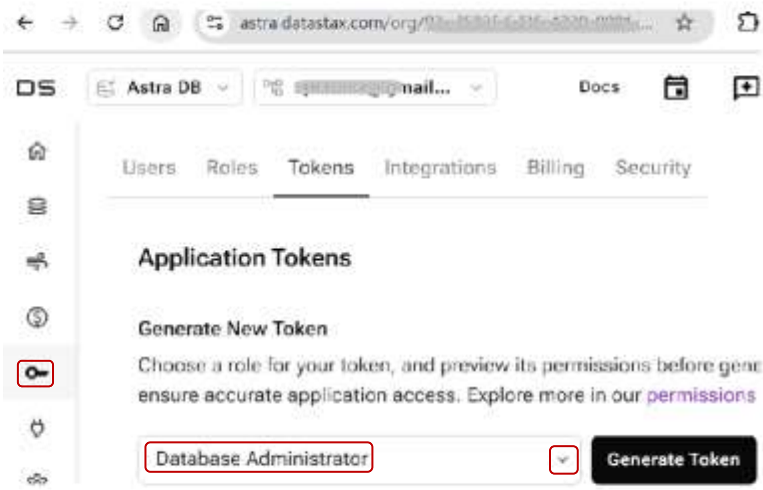


Fig. 6. Generating Application Token for Asra DB database

Software – Software Program Run on IoT Device and MQTT Client

The software program running on IoT device has two main functions: (i) to read environment data collected by the sensors, and (ii) to publish MQTT message on a topic with the sensor data MQTT broker.

The flowchart of software program on IoT device is illustrated in Fig. 7.

On the IoT device, the software program will first connect to internet via Arduino Uno R4 WIFI's built-in Wi-Fi module. After internet connection is successful, it will connect to the MQTT broker. Once the MQTT connection is established, the software program will read the temperature and humidity data from the DHT11 sensor, and ambient brightness data from the LDR. It will then publish MQTT message on a topic with the three IoT data as payload to the MQTT broker. The MQTT broker receives published messages from MQTT publisher client. If the IoT data is detected as abnormal, an alert will be sent to the personnel in charge immediately for necessary action to be taken. In this system, if the environment temperature exceeds a threshold, an alert is sent out immediately to the person-in-charge.

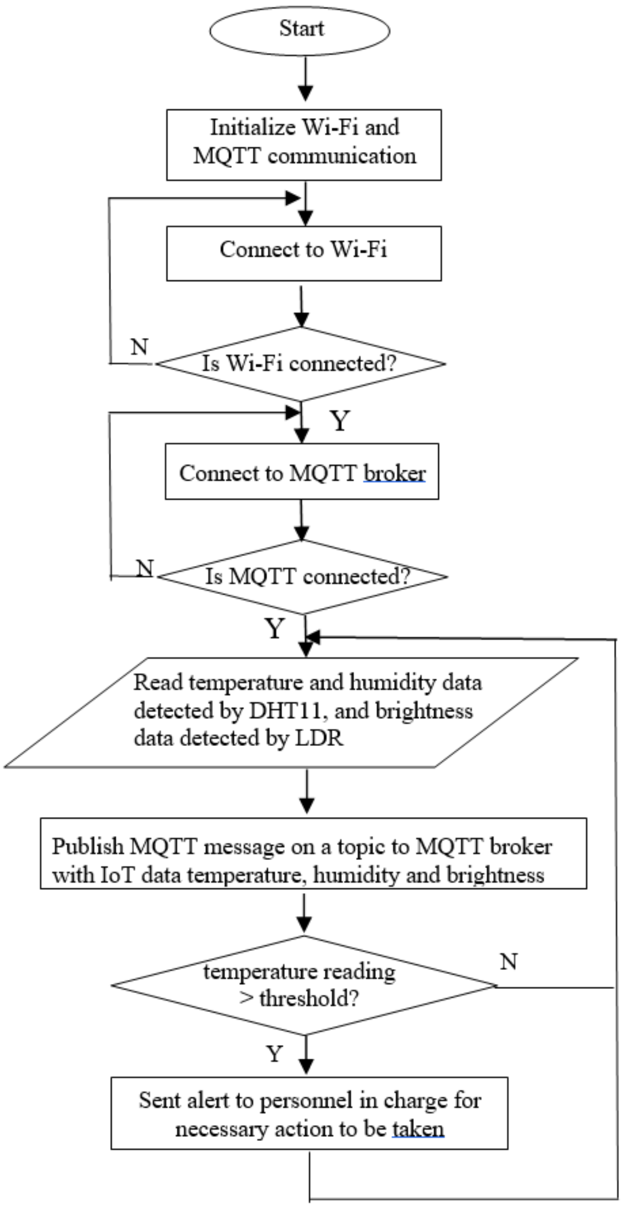


Fig. 7 Flowchart for software on IoT device

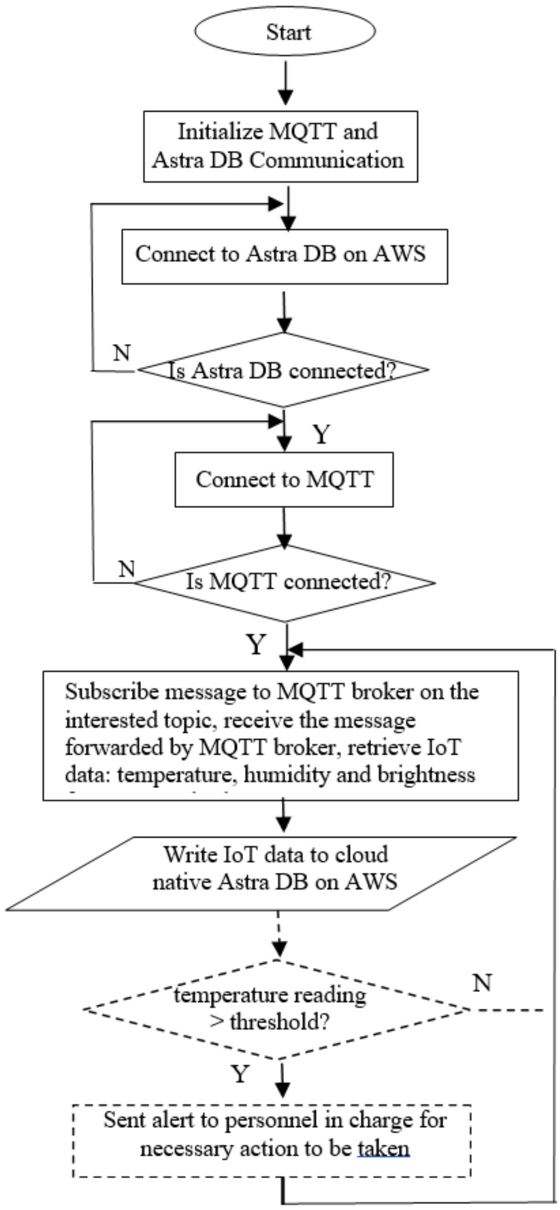


Fig. 8 Flowchart for MQTT subscriber

The MQTT subscriber client implemented here has the following two main functions: (ii) to subscribe to the interested topic and receive the IoT data from the MQTT broker, and (ii) to upload the received IoT data to cloud native database.

Fig. 8 illustrates the flowchart for the MQTT subscriber. After connection to Astra DB database on AWS and to MQTT broker are established successfully, it sends a subscribe message on the interested topic to MQTT broker. The MQTT broker forwards the message based on the interested topic to it. The MQTT subscriber retrieves the IoT data from received MQTT message on the subscribed topics which was published to MQTT broker by MQTT publisher client in the IoT device software program. Then it writes the IoT data to the cloud native Astra DB Database hosted on AWS. The alert could also be sent to the person-in-charge when the IoT data exceeds a threshold at this stage.

Result and Discussion

The system was set up in one of the campus offices. Fig. 9 shows the visualization of the office environment data collected by the IoT device in the morning of a typical working day, on both desktop computer and mobile device.

Fig. 9 Interactive Dashboard for Temperature of One of the Offices

(a) from Computer (b) from Mobile Device

Fig. 10 Alert Sent to Relevant Staff When the Temperature is above Threshold

Fig. 10 shows the alert sent to the relevant staff when the room temperature is higher than 28.5°C.

All the three environmental data: brightness, humidity and temperature could be visualized in Grafana. 28.5°C is set as temperature threshold, because above this temperature, more staff's productivity, health, and job satisfaction might be affected (Panasonic, 2024). The office room temperature is centrally controlled to 26°C in summer as a response to the global sustainability challenge. The actual room temperature value recorded varies depending on the location in the office, e.g. if the staff uses multiple electronic devices concurrently such as using more than one laptop, the temperature taken from this staff cubical will be higher than a cubical where the staff only uses one laptop, etc. Staff, as human beings, generates heat as well. In this example, the system shows that to maintain the office temperature to 26°C, the office temperature might need to be centrally set to below 26°C.

As the open source software, free tier of usage provided by the cloud / cloud native database / data visualization platform are used, the only cost of the full system is the IoT device consisting of Arduino Uno R4 WiFi board with Richshield plugin, which is less than €35. Neither a separate Wi-Fi module nor a separated hardware gateway is needed.

In the Industry 4.0 setting, the IoT data could come from PLC (Programmable Logic Controller), which could be uploaded to cloud native database via OPC-UA (OPC Unified Architecture, a machine-to-machine communication protocol) over MQTT.

Conclusions and Further Exploration

The prototype presented in this paper shows that by utilizing open source hardware and software, free tier usage provided by the cloud / cloud native database / data visualization platform, including open-source MQTT broker, etc., low-cost realtime condition monitoring in areas, like industry 4.0, etc. is possible. The low-cost solution for realtime condition monitoring is essential for rapid development and gaining a competitive advantage in today's data-driven world.

As DataStax Astra DB provides the tools, including APIs, which developers need to create robust AI applications, realtime data processing and integrations, further improvement could be tapping the AI tools and API provided by Astra DB, seamless integrating the low-cost condition monitoring system with AWS AI / ML services, etc.

The ESP32-S3 hosted on the Arduino Uno R4 WiFi is a powerful AI SoC (System on a Chip) integrating Wi-Fi 4 and Bluetooth 5 (LE), with rich peripherals, designed for AIoT applications. ESP32-S3 has additional support for vector instructions in the MCU, which provides acceleration for neural network computing and signal processing workloads. ESP32-S3 provides all the necessary security requirements for building securely connected devices, without requiring any external components (Espressif, 2024). Further exploration could be building low cost secured AIoT application with AI at the edge by tapping the capability of ESP32-S3, the dual-core XTensa LX7 MCU.

Acknowledgment

The project was funded by Ideafarm funding, Singapore Polytechnic. Thanks to support from the management of School of Electrical & Electronic Engineering, Department of Technology, Innovation and Enterprise, Singapore Polytechnic.

References

UN Trade and Development (2024) *Digital economy report 2024,* Retrieved from https://unctad.org/publication/digital-economy-report-2024

MarketsandMarkets (2024), *Machine Condition Monitoring Market Size, Share, Statistics and Industry Growth Analysis Report by Technique (Vibration Monitoring, Thermography, Oil Analysis, Ultrasound Emission), Offering (Vibration Sensors, Infrared Sensors, Spectrometers, Corrosion Probes, Spectrum Analyzers), Region - Global Forecast to 2029,* Retrieved from: https://www.marketsandmarkets.com/Market-Reports/machine-health-monitoring-market-29627363.html

Navdeep Singh Gill, 30 October (2023), Enterprise Digital Platform, *Cloud Native Databases on Docker and Kubernetes* | *Quick Guide,* Retrieved from https://www.xenonstack.com/insights/cloud-native-databases#xenonstack1

Stephen Watts, 13 November (2023), *What is Multicloud? An Introduction,* Retrieved from https://www.splunk.com/en_us/blog/learn/multicloud.html

Snehal Naik, 19 March (2024), olibr.com, *Top 12 Cloud Databases in 2024*, Retrieved from https://olibr.com/blog/top-12-cloud-databases/

Datastax.com (2024)*, Serverless Database Built on Apache Cassandra*, Retrieved from https://www.datastax.com/products/datastax-astra/features

Datastax.com (2024)*, Intro to Astra DB Serverless,* Retrieved from https://docs.datastax.com/en/astra-db-serverless/get-started/astra-db-introduction.html

Arduino Team, March 25th, (2023), *Arduino UNO R4 is a giant leap forward for an open source community of millions*, Retrieved from https://blog.arduino.cc/2023/03/25/arduino-uno-r4/

Arduino.cc (2024), *UNO R4 WiFi*, Retrieved from https://docs.arduino.cc/hardware/uno-r4-wifi/

Espressif (2024), *ESP32-S3 Designed for AIoT applications*, Retrieved from https://www.espressif.com/en/products/socs/esp32-s3

Grafana Labs (2024), *Visualize your data, optimize your performance,* Retrieved from https://grafana.com/oss/grafana/

HiveMQ (2024), *MQTT Essentials: The Ultimate Guide to MQTT for Beginners and Experts,* Retrieved from https://www.hivemq.com/mqtt/

PTC (2024), *MQTT Client*, Retrieved from https://www.ptc.com/en/store/kepware/drivers/mqtt-client

Getmicros (2024), *A Quick Look at the Rich Shield for the Arduino,* Retrieved on 18 Aug 2024 from http://www.getmicros.net/a-quick-look-at-the-rich-shield-for-the-arduino.php

Panasonic (2024), *The key to comfortable living: perfect room temperature and humidity*, Retrieved on 18 Aug 2024 from https://www.panasonic.com/global/hvac/nanoe/stories/comfortable-temperature-and-humidity.html#1