Smarter dam based on cyber-physical system utilizing Raspberry Pi4 and NodeMCU ESP8266

Mustafa Yassin Deab, Muayad Sadik Croock
Department of Control and Systems Engineering, University of Technology-Iraq, Baghdad, Iraq

Article Info

Article history:

Received Jul 5, 2022
Revised Aug 7, 2022
Accepted Aug 23, 2022

Keywords:

Cloud server
CPS
Dam controlling systems
NodeMCU ESP8266
Raspberry Pi4
WSN

ABSTRACT

The majority of dams still use traditional technologies to manage water gates. These strategies have a number of limitations, including those related to human error. This study describes an automated method for controlling and managing dams based on the structure of a cyber-physical system (CPS) in order to reduce work hazards and human efforts. The suggested system design consists of four ESP8266 nodes, each connected to a water level sensor and distributed around the main dam and its feeder sections. These nodes are linked to a central server (Raspberry Pi4). The data is gathered and retransmitted through the message queuing telemetry transfer (MQTT) protocol to the (Raspberry Pi4) central server via wireless sensor nodes (WSNs) distributed on the dam's various sides to regulate the water levels in the dam's main reservoir and the areas it feeds. Furthermore, the Raspberry Pi4 transmits data to the cloud server using internet media. A cloud-based dashboard with numerous tabs for each node has been constructed. The results of the experiments reveal that the proposed technique is superior to the ones currently in use for dam management.

Corresponding Author:

Mustafa Yassin Deab
Department Control and Systems Engineering
University of Technology, Iraq, Baghdad, alsinaea street
Email: cse.20.25@grad.uotechnology.edu.iq

1. INTRODUCTION

It is of the utmost importance to pay attention to water distribution, consumption, and the avoidance of waste floods given the fast development of the population as well as the expanding demand for water. If the water level unexpectedly rises owing to substantial floods from surrounding nations and strong rains in the north, drowning towns, cities, and agricultural areas close to dams, this intimate interaction can considerably raise the probability of making mistakes. As a result, the purpose of the research is to design an automated system that is built on the architecture of a cyber-physical system (CPS) and that fulfills all of the requirements. Within CPSs, natural and human systems (also known as physical space) are intricately linked to computer, communication, and control systems (also known as cyberspace) [1], [2].

Wireless sensor networks (WSNs) are used by collaborative computational entities in CPS to interface directly with hardware. The internet of things (IoT) is frequently characterized as a network of interconnected systems that communicate with one another and with people [3]. Actuation and sensing interact via the physical environment and between sensors and actuators via a node (node), many (i.e., distributed) computers, or intelligent control to produce the desired outcome or steady-state [4]. The CPS functions with entire autonomy or, in any case, aids the human in-loop mechanism as part of its semi-autonomous control tasks. Through its distributed closed-loop operation, the CPS may influence, automate, manage, and control various small, medium, and large artificial (but natural) systems. Because of its operational nature in most industrial control activities, CPS systems are sometimes known as Operational Technology Systems (i.e., OT Systems) [5]. WSNs are important components in the development of new electronic and physical systems (CPSs). Many unique applications have risen in popularity, including smart homes, military, and medical applications [6]-[7], WSN users may access sensor data and other resources from anywhere on the planet thanks to the internet [8], [9].

The WSN is used to regulate dams because to the difficulties of constructing and installing them in a confined area. Furthermore, while the CPS may be found in a variety of systems, it is most commonly seen in dam control systems. In this article, dam management and control systems are suggested and integrated into the CPS architecture. This technology works automatically and with minimum human participation, which reduces errors and difficulties. Furthermore, the proposed system features a cloud-based website for remote property control and management. The findings suggest that the proposed technology outperforms current dam management and control systems.

Because it is essential to maintain the dams as free of difficulties as possible, various research has focused on reducing and avoiding issues and discovering acceptable solutions. The authors of [10] proposed multi-level management and regulating architecture. Among these levels were water monitoring in the dam before and after operation, as well as gate control measures. Furthermore, the dam's output water pressure was monitored to keep the dam's wall secure. WSN was utilized to take readings and provide the necessary actions. The findings supported the author's premise that the dams could be controlled. A dam early warning system has been developed to evaluate the water discharge [11]. The suggested system was designed with three degrees of warning based on the water level and employed WSN to read the water level. Drones were also operated to provide photographs into the system for water level measurement. The authors of [12] proposed a modern tank management system that included pressure, water level, flow, outflow, tilt, and vibration sensors. Arduino sensors provide real-time data (Power BI transforms diverse data sources into coherent, compelling graphics and interactive insights). Both parameters are presented correctly in databases, webpages, and structured files (spreadsheets, CSV, XML, and JSON). Emergencies cause flooding. The remote site control room dashboard displays sensor data via a website.

However, because of the high energy consumption and short radio transmission distance, [13] the researcher advised using Lora for dam safety monitoring. He constructed a smart monitoring station using an ARM LPC1788 and Lora. Low power relay mode is employed with Lora and 4G transmission, and active loop recall and early warning are proposed. Integrating an intelligent monitoring terminal, a low-power relay gateway, and a cloud platform may improve system viability, stability, and data transmission. The WSN was used to manage dams in catastrophes by gathering the necessary data and assisting in reducing losses [14]. The data was collected and transferred to the central control unit through the local control portions. The dam gates were opened and closed using the central control unit. The authors of [15], [16] state that the WSN was used to automatically monitor and regulate dam gates based on water levels. The water level data was taken with WSN and communicated to the control unit, which automatically controlled the dam gates. The biggest problem with this method is that it responds quickly to changes in the water level without taking anything else into account.

This paper is organized as follows, with brief descriptions of each section: The second section of this section discusses relevant work, in which we respond to the findings of the researcher; the third section discusses our proposed management system for dams, which makes use of a wireless sensor network to manage and monitor environmental parameters such as water level, humidity, and temperature; and the fourth section discusses the findings. The conclusion is presented in the last section.

2. PROPOSED SYSTEM MODEL

The suggested dam management and control system is designed with the help of the CPS framework. The WSN, gate actuators, and other hardware are all included here. Subsections are provided here to facilitate in-depth discussion of the proposed system.

2.1. System structure

The proposed system design is shown in Figure 1. It consists of four ESP8266 nodes, each connected to a water level sensor and scattered around the main dam and the areas feeding it. These nodes are connected to the central server (Raspberry Pi4). He is responsible for collecting data from the contract and deciding when the dam gates will open and close. A message queuing telemetry transfer (MQTT) protocol [17] was used to transfer data between nodes and central servers. In addition, the Raspberry Pi4 sends data to the cloud server through a web portal, allowing continuous system monitoring and complete management of any process from anywhere in the world. Moreover, the data is saved to Google Cloud using Google Sheets and saved in Excel format for reading and analysis while monitoring in real-time. On any computer or mobile device, any authorized individual may perform the same operations by accessing the official homepage of Node-dam Red. This system offers an alternative to the differential protection method, monitoring, and regulation. System performance is not affected by the distance between the two sides of the device.

As previously stated, the system is made up of four nodes, each of which is a NodeMCU ESP8266 [18]. The central dam node connects all the sensors for water level, rain, water temperature, air temperature, and humidity. The remaining nodes are distributed across three places after the dam. Each node contains a water level sensor and a servo motor that regulates the opening and closing of one of the three gates independently. When using wireless control, the controller operates remotely while the operator performs the necessary tasks. The operating system is provided by the NodeMCU ESP8266 microcontroller.

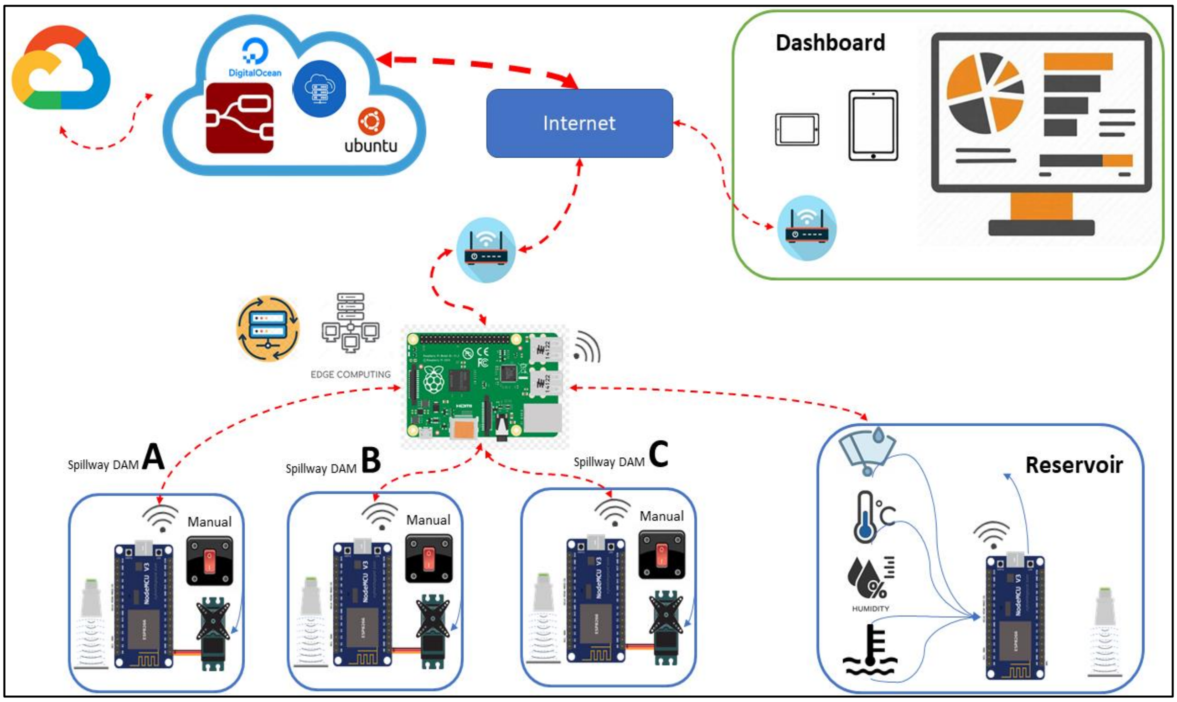


Figure 1. Block diagram of the proposed model

It adjusts the position of the gate in response to the amount of water by instructing the accompanying servo motor to spin at a specific angle. In the event of automated control, however, these nodes submit the data to the Raspberry Pi, which analyzes it and chooses the optimal course of action before returning the choice to the node for execution. In this study, each gate acts as a self-managing node, increasing the system's versatility.

In the proposed system, the Raspberry Pi 4 operates as a local server, receiving data from the controller nodes over the MQTT protocol, processing it, making a decision, and providing it back to the appropriate node to execute the order. Instead of a cloud server, the Raspberry Pi acquires and analyzes data using edge computing. Because of this technology, the system responded quickly to wireless gate control from locations distant from the dam site.

2.2. Cloud server

Cloud computing is a means of fulfilling the growing need for processing and storage. A cloud is a large, distributed network of computers and servers linked over the Internet. The concept is based on providing on-demand services via the Internet. This method is frequently built on charging for client consumption (i.e., calculated service) and having the freedom to develop and reduce resources as needed (i.e., flexible resources) [19]. Because the administrator requires remote monitoring and management in addition to data storage [20], we established a real server after reserving a certain amount of memory and processor on the cloud from DigitalOcean, an American cloud infrastructure provider with data centers all over the world [21]-[22]. The Node-red dashboard node was utilized in this study to create a graphical user interface (GUI) through which the server operator could deploy the monitoring and control system. Figure 2 depicts the reserved server, and the CPU handles the bandwidth and quantity of data.

Figure 2. Cloud server details

2.3. Dashboard design

In the proposed method [23], the Raspberry Pi unit is configured and built using Node-RED as a local server and a cloud server for setting up servers. Node-RED controls the ESP8266 with a simple web browser-based editor (NodeMCU). Figure 3 illustrates the process for joining knots in the Node-RED interface. To monitor and control the system, the server operator creates a graphical user interface (GUI) using Node-RED [24]. The dashboard, shown in Figure 4, is the system's ultimate user interface and is a web-based GUI [25]. It has controls that open the dam's gates at the proper angle and historical charts that display pertinent data, including water level, rainfall, and evaporation. With an internet connection, the user can use this dashboard to control the system from anywhere in the world.

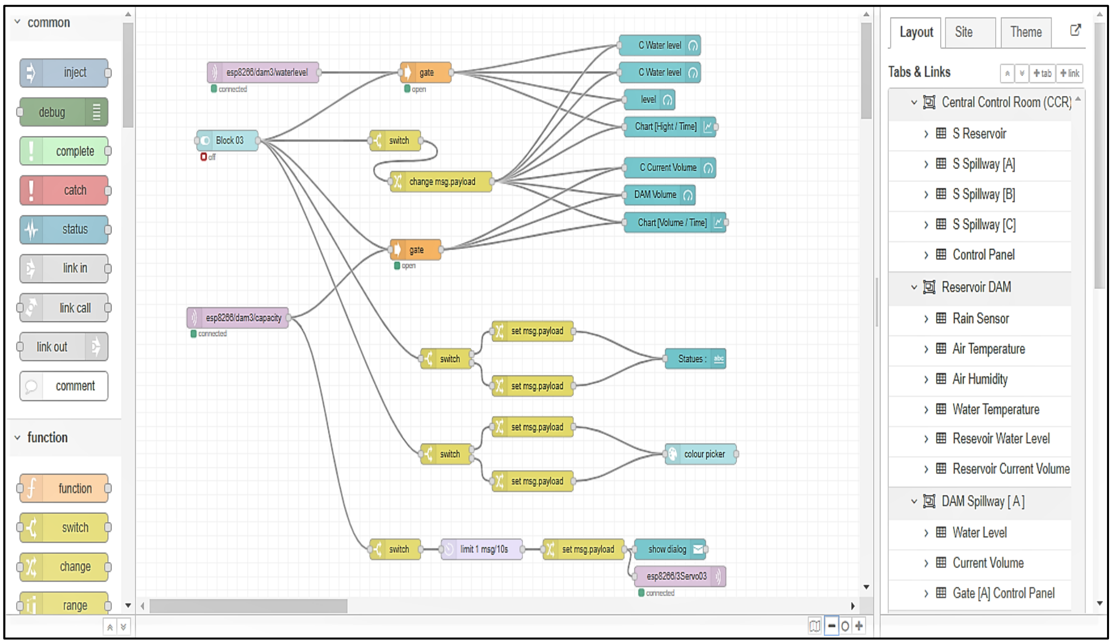


Figure 3. Node-red interface

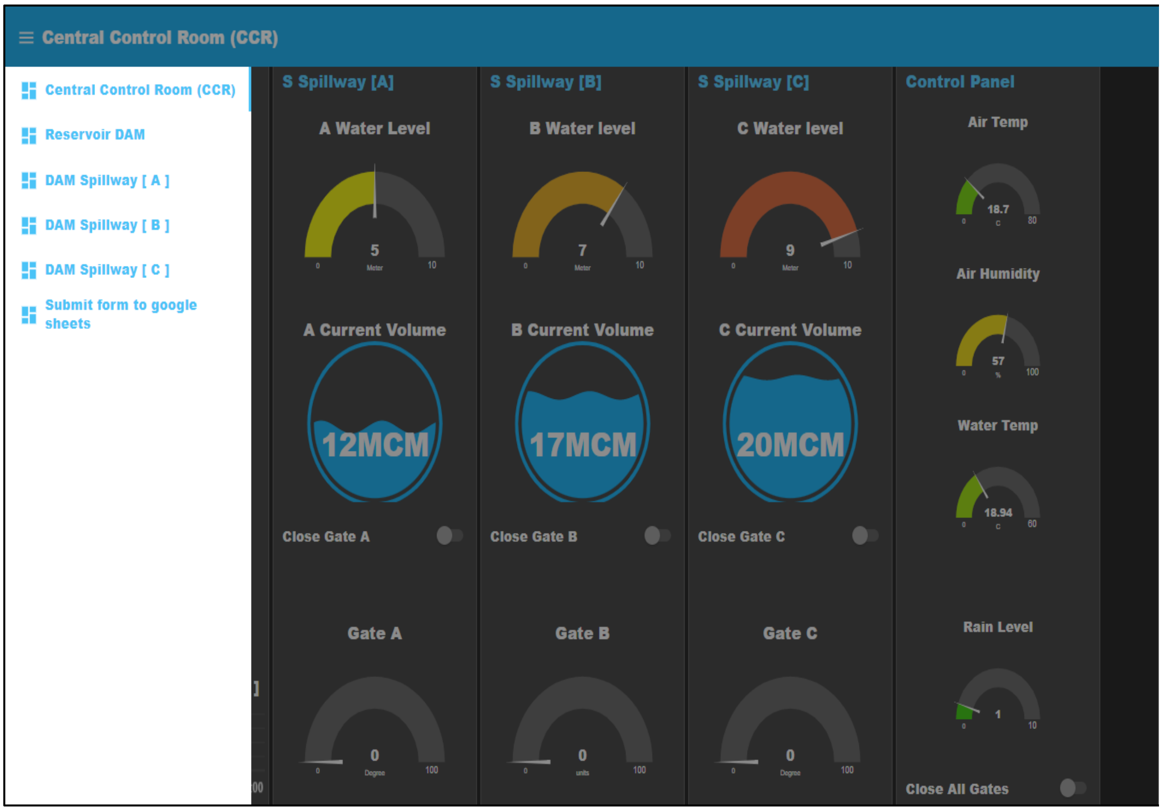


Figure 4. Node-red-dashboard

2.4. Proposed algorithm

A flowchart has been created to represent the suggested system algorithm. The following is a summary of the data and information flow between the dam and the proposed system in this proposed design:

a. The contract is set up by connecting the electric current to it, then looking for an Internet connection after the devices have been successfully connected.
b. The nodes start reading data from the sensors and sending it to the Raspberry Pi for analysis and the cloud server.
c. The water level sensor is the most important sensor for the system; it is roughly 15 cm above the main tank's surface. In this case, the sensor detects the difference of 12 cm between its level and the water's surface. Based on the system's capacity, a fixed quantity of water is allowed in the main tank and the other zones.

The analysis is given in Figure 5 after obtaining measurements from all nodes.

2.5. Implementation methodology

The dam's shape is printed in three sections on a 3D printer and then linked at the end, as seen in Figures 6 (a), (b), and (c). Figure 7 shows the finished dam, which has all the controls and sensors, the central basin, and the water fields outside.

The schematic diagram of the dam's central node, which houses the associated NodeMCU ESP8266 development board and several sensors, is shown in Figures 8 (a) and (b). These sensors are the base tank water level sensor (HC-SR04), rain sensor, water temperature sensor (DS18B20), and air temperature and humidity sensor (DHT11). This node learns critical data from the master dam and sends it back to the central server (Raspberry Pi4), using the MQTT protocol after connecting to the Internet via Wi-Fi.

The remaining three nodes are spread across the dam reservoirs, as shown in Figures 9(a) and (b). Each node is an ESP8266 controller connected to the water level sensor (HC-SR04) and servo motor (SG90) via pin D5 PWM to regulate opening and closing of the gate at the correct angle according to the flowchart design. Figure 10 shows how the main knot and additional nodes are arranged in the dam.

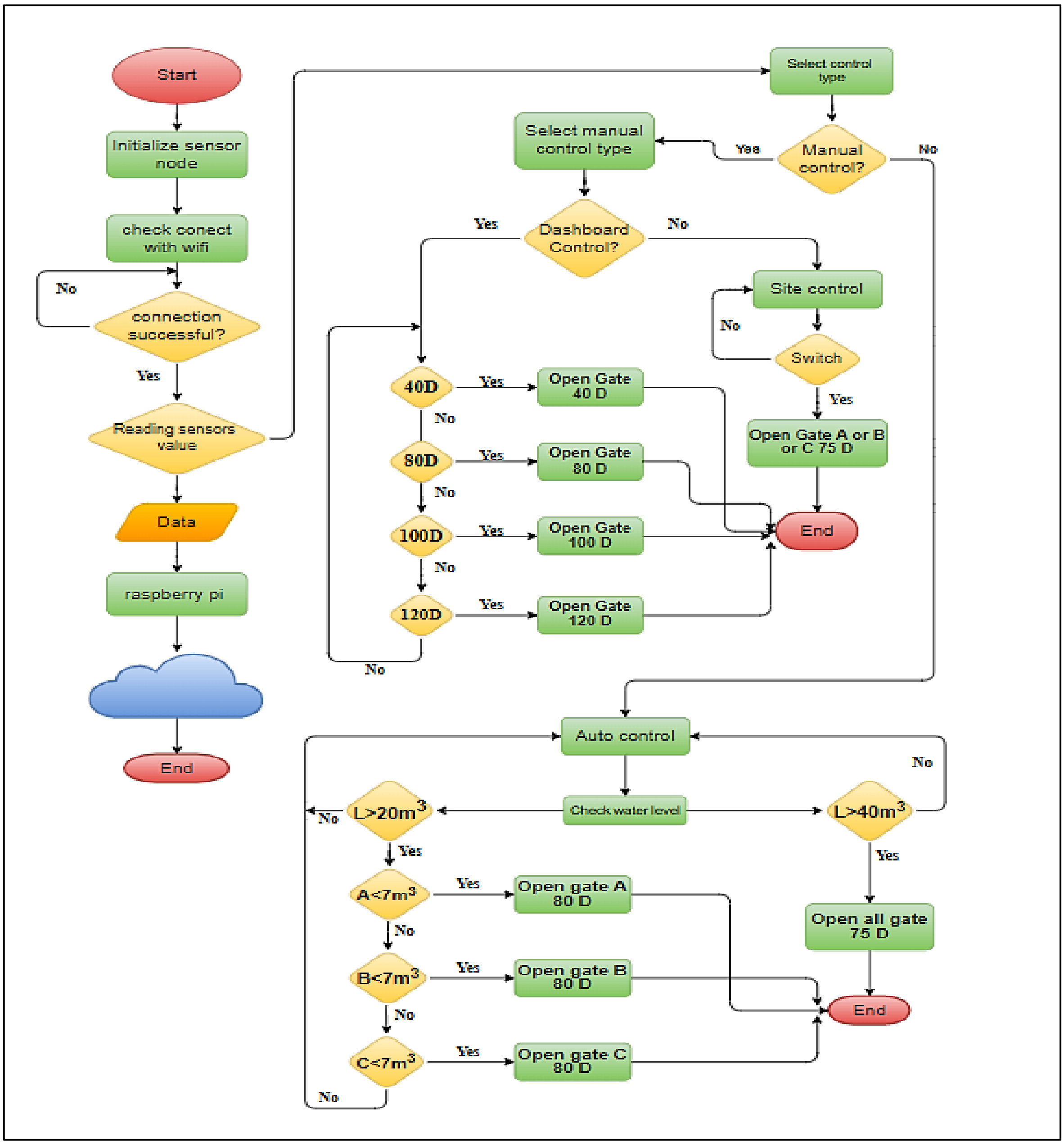
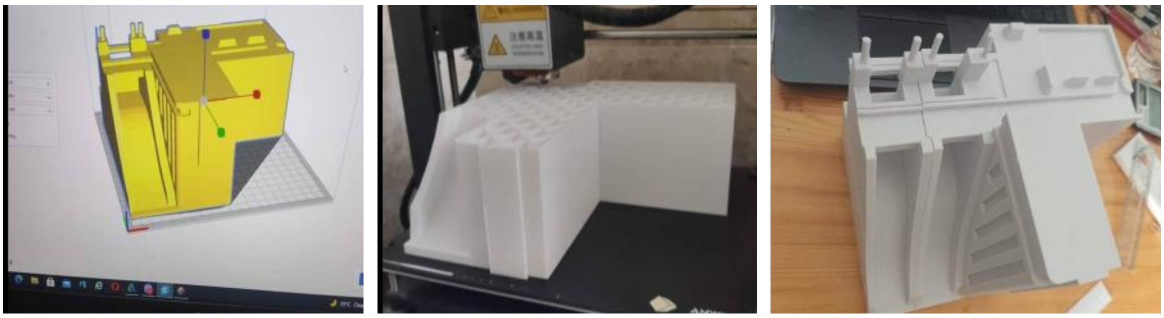


Figure 5. Flowchart of system



(a) (b) (c)

Figure 6. Prototype design steps (a) design dam (b) 3D printer print design and (c) final form

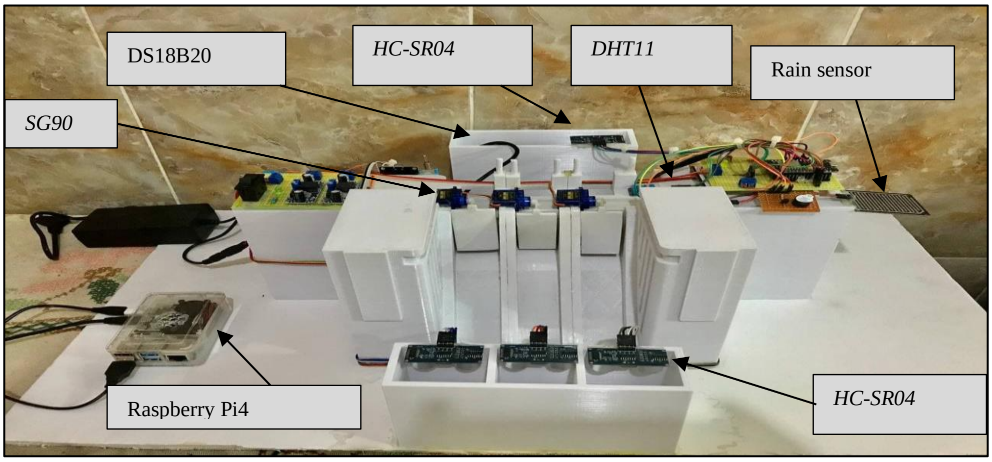
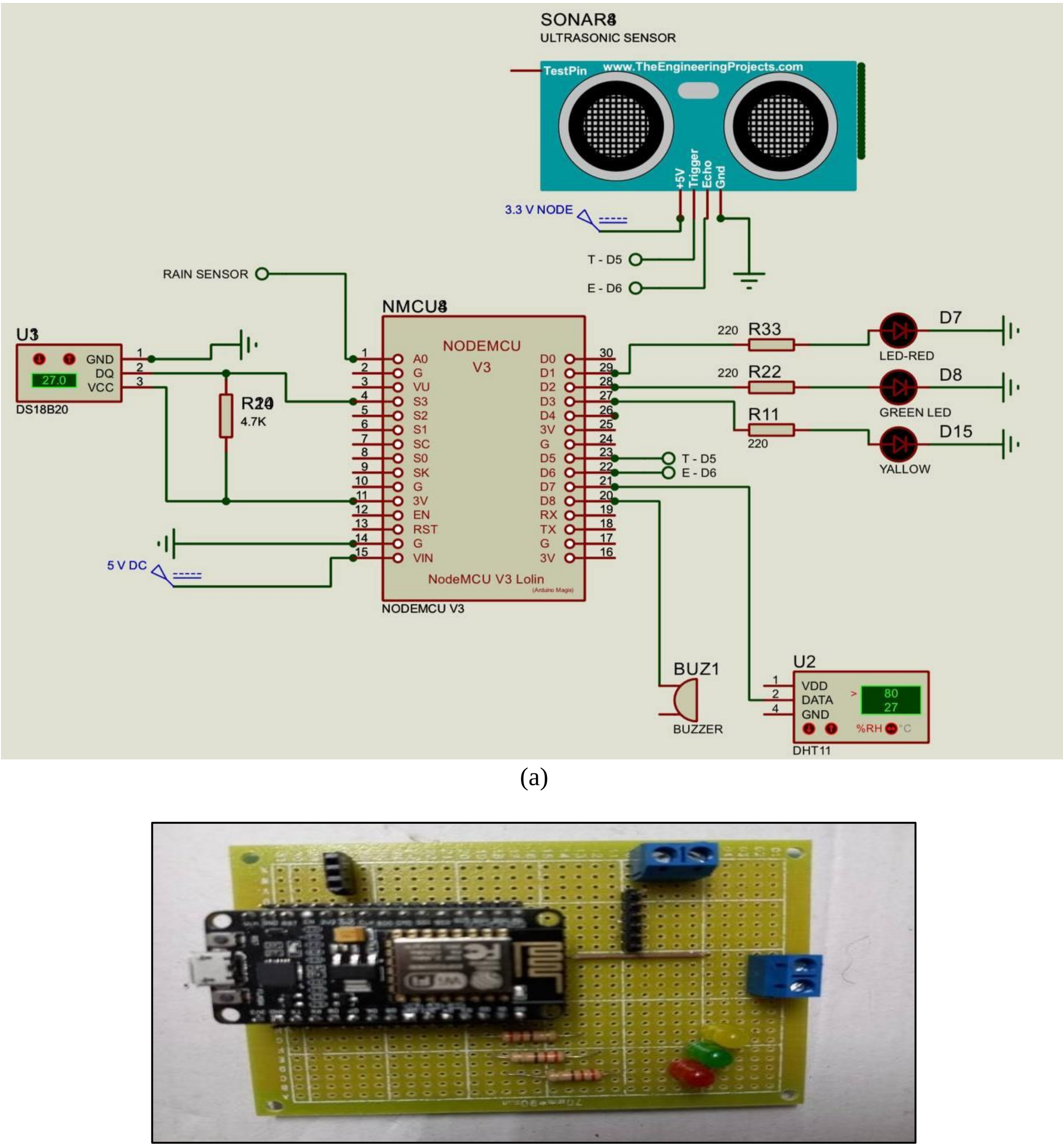


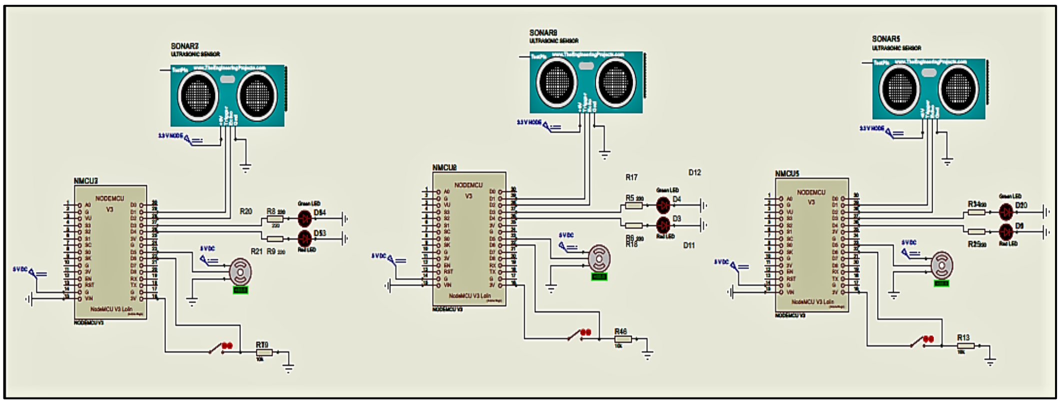
Figure 7. The final design which includes controls and sensors



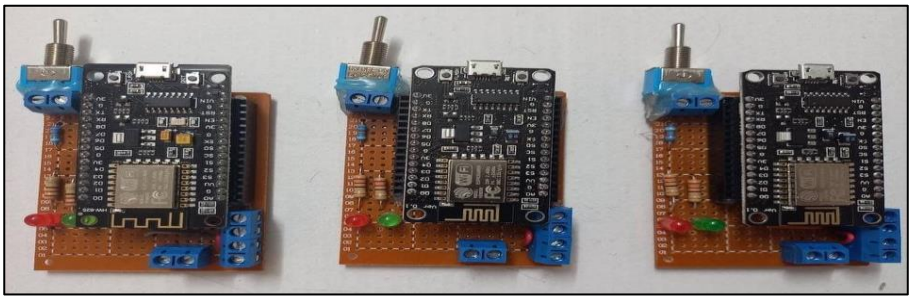
(a)

(b)

Figure 8. The dam's primary node (a) node schematic and (b) NodeMCU on board



(a)



(b)

Figure 9. Three nodes for three different regions (a) nodes schematic and (b) NodeMCU on board

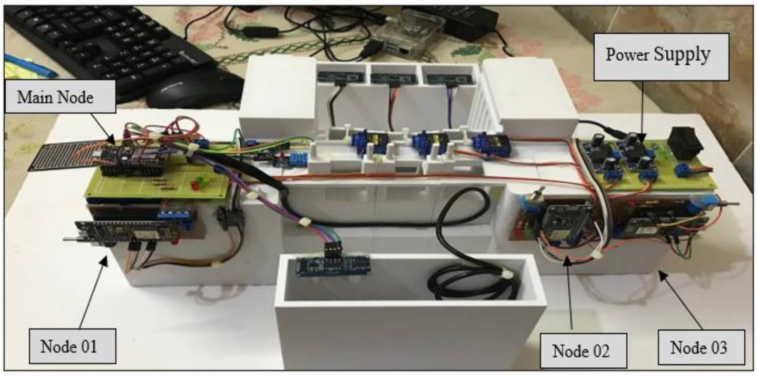


Figure 10. Nodes on prototype

3. RESULTS

Different case studies are evaluated for a week to assess the efficacy of the suggested method in regulating and managing the dam. The sensor readings are published using Node-RED on the Raspberry Pi's on-site server and the Digital Ocean platform's cloud server. The data is shown on the dashboard and is instantly saved in the Google Cloud as an Excel table using the Google Sheet. Any individual who is authorized to monitor the data may be added by providing their email address to get data updates every 10 minutes. The dashboard may also be seen from any tablet or computer around the globe through an interactive web page. The case studies given above will be described further down.

3.1. Case study 1

The first node sends data to the server for processing after detecting the water level in the primary tank. The server notifies all nodes in charge of the gates of the decision to open them automatically when the water level in the tank exceeds 40 (m^3). The system also displays a warning message on the dashboard at such time. The procedure is carried out once more till the water level drops to 40 m^3, after which the gates close. The findings of this case study are shown in Figures 11(a), (b) and (c).

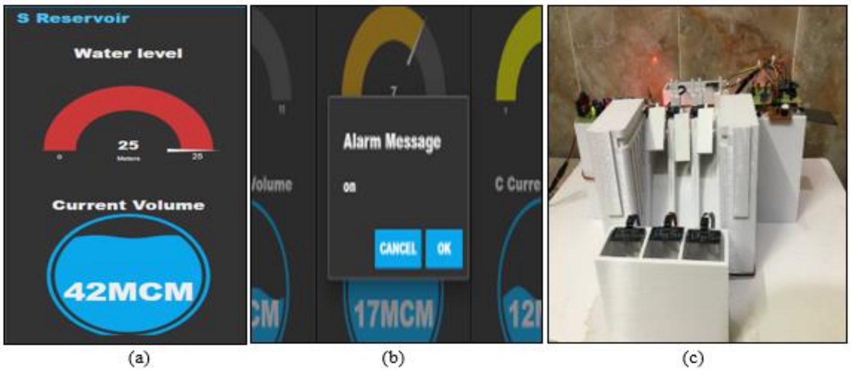
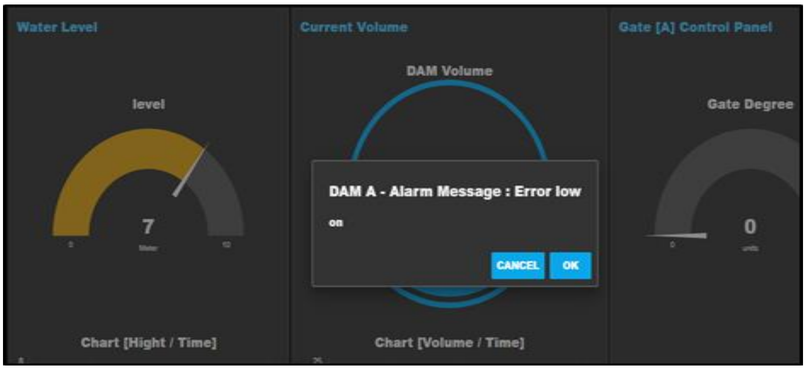


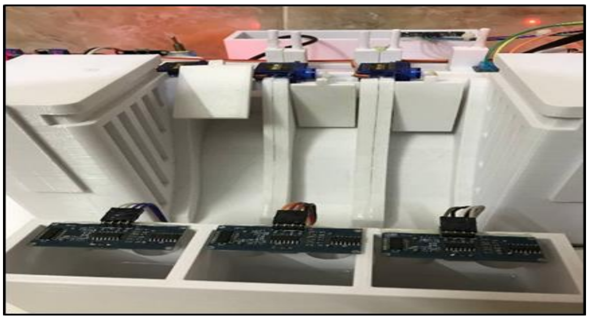
Figure 11. Case 3 implementation (a) water level on the dashboard (b) alert and (c) all gates open

3.2. Case study 2

The system sends out a warning and then chooses to open the first dam's gate to make up for the drop, as shown in Figures 12(a) and (b), when the water level in the first water fields 12(a), which is supplied by the main dam, decreases. The gate automatically shuts when the requisite amount of water is reached in that field. The gate won't open if the main tank's water level is below 20 m^3, even if field 12(a) needs extra water.



(a)



(b)

Figure 12. Case 2 implementation (a) dashboard alarm and (b) automated gate opening

3.3. Case study 3

The manual operation of the dam's gates by the authorized operators is investigated in this case. The procedure of wirelessly opening and closing gates A, B, and C to the required degree is explained and controlled by the dashboard of gate B, which is seen in Figure 13. This feature allows the dam's operators to manage the gates from anywhere in the globe using the dashboard, which can be accessed by IP and includes the login and password. As a result, only authorized users have access to it. For gates A and C, the same method applies.

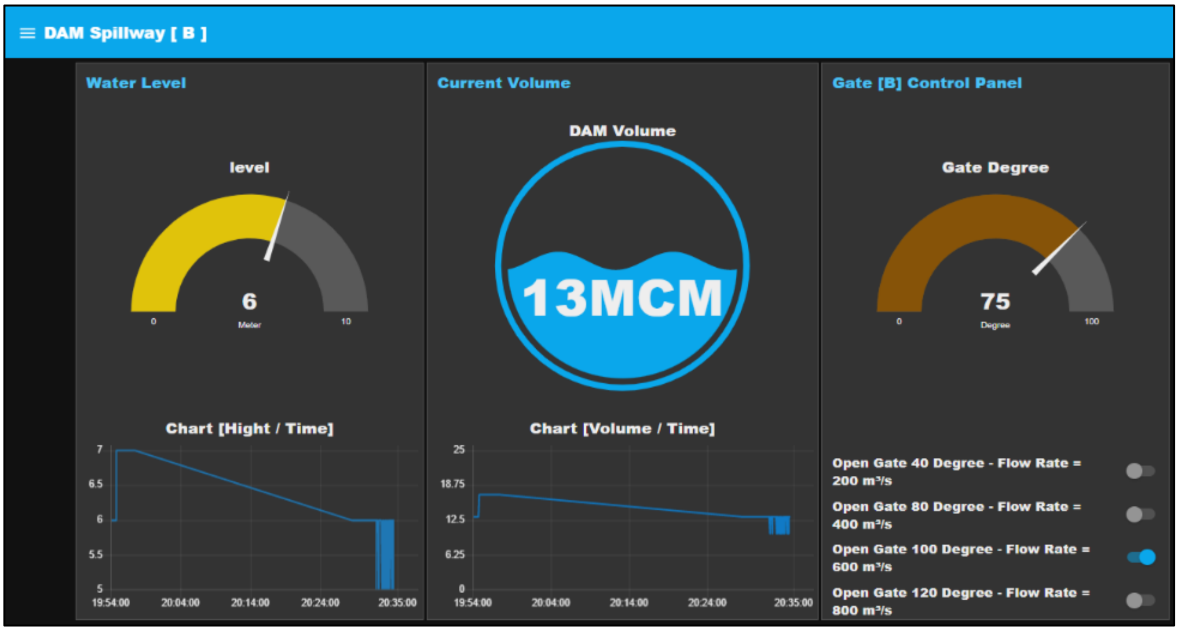


Figure 13. A dashboard for controlling the wireless gate position

4. CONCLUSION

The CPS structure offered a dam management and control system that was exceptionally trustworthy, adaptable, and low-cost. The plan was installed over three phases and featured remote control and monitoring capabilities. The dam's water level, temperature, rainfall, air temperature, humidity, and the water level of the dam-supplied fields were monitored using sensors in the first stage. These nodes used the MQTT protocol to communicate data to the central server. The second phase was setting up a Raspberry Pi server to gather information from the nodes, make the right choice, and then relay the decision to the node responsible for carrying it out. The next step was to construct a cloud server, to which the data would be linked via Raspberry Pi and shown on a dashboard for remote dam monitoring and control. The suggested technique was assessed using several case studies. The acquired findings demonstrated its capacity to reduce the labor force and the effectiveness of remote control, real-time monitoring, and proactive alerts for timely decision-making.

REFERENCES

[1] L. Monostori *et al.*, "Cyber-physical systems in manufacturing," *CIRP Annals*, vol. 65, no. 2, pp. 621–641, 2016, doi: 10.1016/j.cirp.2016.06.005.

[2] S. Davidson, "cyber-physical system design with sensor networking technologies," in *IEEE Design & Test*, vol. 34, no. 3, pp. 105-107, June 2017, doi: 10.1109/MDAT.2017.2655504.

[3] A. Rein, R. Rieke, M. Jäger, N. Kuntze, and L. Coppolino, "Trust establishment in cooperating cyber-physical systems," in *Lecture Notes in Computer Science*, pp. 31–47, 2015, doi: 10.1007/978-3-319-40385-4_3.

[4] M. S. Croock, S. D. Khuder, and Z. A. Hassan, "Self-checking method for fault tolerance solution in wireless sensor network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 4, pp. 4416-4425, 2020, doi: 10.11591/ijece.v10i4.pp4416-4425.

[5] A. Burg, A. Chattopadhyay and K. -Y. Lam, "Wireless Communication and Security Issues for Cyber–Physical Systems and the Internet-of-Things," in *Proceedings of the IEEE*, vol. 106, no. 1, pp. 38-60, Jan. 2018, doi: 10.1109/JPROC.2017.2780172.

[6] M. A. Kore and D. Kadam, "IoT Based Real Time Greenhouse monitoring system using Raspberry Pi," *International Research Journal of Engineering and Technology*, vol. 06, no. 4, pp. 3941-3945, Apr. 2019.

[7] S. Saponara and B. Neri, "Mm-wave integrated wireless transceiver: Enabling technology for high bandwidth short-range networking in cyber physical systems," *Microsystem Technologies*, vol. 22, no. 7, pp. 1893–1903, 2016, doi: 10.1007/s00542-016-2888-1.

[8] S. S. Wali and M. N. Abdullah, "Review of wireless body sensor networks," *International Journal of Engineering & Technology*, vol. 9, no. 4, pp. 863-870, 2020.

[9] D. D. Khudhur and M. S. Croock, "Application of self-managing system in greenhouse with Wireless Sensor Network," *Iraqi Journal of Computer, Communication, Control and System Engineering*, vol. 21, no. 1, pp. 53–61, 2021.

[10] A. H. M. Almawgani *et al*., "Smart monitoring system of Najran dam," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 4, pp. 3999–4007, 2020, doi: 10.11591/ijece.v10i4.pp3999-4007.

[11] P. Rajesh, N. Manjunathan, and A. Suresh, "Unmanned Dam Monitoring System Using Wireless Sensor Networks," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 5, pp. 343-347, 2019.

[12] S. Rohith P., R. Catherin, B. Priya R., P. Krishna K. and R. Kumar P., "IoT based State of the Art Dam Management System in Indian Scenario," *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2020, pp. 362-366, doi: 10.1109/I-SMAC49090.2020.9243503.

[13] C. Xie, "Research and application of key technologies for dam safety monitoring based on LoRa," *IOP Conference Series: Earth and Environmental Science*, vol. 768, no. 1, p. 012131, 2021, doi: 10.1088/1755-1315/768/1/012131.

[14] Anita, R. Singh, S. Choudhury, and B. Singh, "Wireless disaster monitoring and management system for dams," in *Procedia Computer Science*, vol. 48, pp. 381–386, 2015, doi: 10.1016/j.procs.2015.04.197.

[15] K. P. Indhumathi, G. K. Lekshmi, and Mahalakshmi, "Auto shutter control system for WLAN using reservoir (auto dam shutter)," *International Journal of Engineering Research & Technology (IJERT)*, vol. 7, no. 01, pp. 1-5, 2019, doi: 10.17577/IJERTCONV7IS01013.

[16] S. LM, A. Bajirao C, Y. Praksh J, R. Ramesh P, S. Suryawanshi S, and S. Shankarrao H, "Review on IoT based dam parameters monitoring system," *Journal of Information Technology & Software Engineering*, vol. 08, no. 05, 2018, doi: 10.4172/2165-7866.1000247.

[17] A. D. Salman and N. A-hussein, "IoT Monitoring System Based on MQTT Publisher/Subscriber Protocol," *Iraqi Journal of Computer, Communication, Control and Systems Engineering*, vol. 20, no. 3, pp. 75–83, Jul. 2020, doi: 10.33103/uot.ijccce.20.3.7.

[18] "ESP8266," *Espressif.com. [Online]*. Available: https://www.espressif.com/en/products/hardware/esp8266ex/overview. [Accessed: 07-Jan-2022].

[19] F. Firouzi, K. Chakrabarty, and S. Nassif, "Intelligent internet of things: From device to fog and cloud," *Springer International Publishing*, 2020, doi: 10.1007/978-3-030-30367-9.

[20] S. J. Siwan, W. F. Shareef, and A. R. Nasser, "Cloud Based Processing of Vibrational Signal for Oil Pipeline Monitoring System," *Iraqi Journal of Computers, Communications, Control and Systems Engineering*, vol. 21, no. 3, pp. 24-39, 2021, doi: 10.33103/uot.ijccce.21.3.3.

[21] "DigitalOcean–The developer cloud," *Digitalocean.com*. *[Online].* Available: https://www.digitalocean.com/. [Accessed: 16-Feb-2022].

[22] Bloomberg," *Bloomberg.com*. [Online]. Available: https://www.bloomberg.com/profile/company/0852730D:US. [Accessed: 17-Feb-2022].

[23] "Node-RED," *Nodered.org*. *[Online]*. Available: https://nodered.org/. [Accessed: 21-Jan-2022].

[24] "Node-red-dashboard," *Nodered.org*. [Online]. Available: https://flows.nodered.org/node/node-red-dashboard. [Accessed: 27-Jan-2022].

[25] Z. M. Resan and M. S. Croock, "Software engineering model based smart indoor localization system using deep-learning," *Telecommunication, Computing, Electronics and Control (TELKOMNIKA)*, vol. 18, no. 4, p. 1904, 2020, doi: 10.12928/telkomnika.v18i4.14318.

BIOGRAPHIES OF AUTHORS

Mustafa Yassin Deab received a B.Sc. from Diyala University/ Computer Engineering Department in 2019. I have two years of hands-on experience in programming controllers and wireless sensor networks to develop the Internet of Things and build intelligent applications. I look forward to gaining more experience. A student in the research phase for a master's degree in the Department of Control and Systems Engineering at the University of Technology. He can be contacted at email: cse.20.25@grad.uotechnology.edu.iq.

Muayad Sadik Croock earned his Ph.D. (2012) in Computer Engineering at Newcastle University, School of Electrical, Electronics and Computer Engineering. He finished his M.Sc. degree in Computer Engineering from University of Technology-Iraq (2003), and a B.Sc. degree in Computer Engineering from the University of Technology (1998). Dr. Muayad is a staff member at University of Technology since 2004, where he currently holds assistant professor in Software Engineering. His research interest include software engineering, security, and WSN. He can be contacted at email: muayad.s.croock@uotechnology.edu.iq.