International Journal of Computational and Electronic Aspects in Engineering
Vol. 5, Issue 3, Sept 2024, pp 98-112
https://doi.org/10.26706/ijceae.5.3.20240804

Wireless Sensor Networks for Smart Gardening: ESP-NOW and Blynk IoT Integration for Water and Energy Optimization

Ibrahim A. Ameen[1,*] and Mustafa A. Al-Sheikh[2]

[1,2] Department of Mobile Communications and Computing, College of Engineering, University of Information Technology and communications, Baghdad, Iraq

Ibrahim.ameen@uoitc.edu.iq [1], Mustafa.alsheikh@uoitc.edu.iq [2]

Article – Peer Reviewed
Received: 26 July 2024
Accepted: 18 Sept 2024
Published: 30 Sept 2024

Cite this article: Ibrahim A. Ameen, and Mustafa A. Al-Sheikh, "Wireless Sensor Networks for Smart Gardening: ESP-NOW and Blynk IoT Integration for Water and Energy Optimization", *International Journal of Computational and Electronic Aspects in Engineering*, RAME Publishers, vol. 5, issue 3, pp. 98-112, 2024.
https://doi.org/10.26706/ijceae.5.3.20240804

Abstract: Conventional gardening and farming approaches struggle with freshwater shortage, wasting water and fertilizers, quality of plants, energy, and costs, which are mitigated by employing smart agricultural systems that utilize wireless sensor networks and IoT technologies. A bidirectional wireless sensor network is essential to monitor and control a water management system for medium and large areas of land. Also, the IoT provides the mobility for that smart system by accessing it from anywhere in the world using the internet connectivity. Toward minimizing drawbacks and reinforcing farmers' efforts to gain optimal results from their gardens and farms. This research aims to investigate the use of ESP-NOW to create a star-topology wireless sensor network that includes four end nodes communicating bi-directionally with a gateway node. Also, this paper discusses the nodes' energy consumption optimization by applying two operation modes of ESP boards: a fully active mode and a hybrid mode (active and deep sleep). In addition to uploading air temperature and humidity, soil moisture, and light intensity to a Blynk server and controlling four water valves distributed with the nodes. As a result, we implemented Blynk web and mobile dashboards to visualize and control the system. We achieved a manageable irrigation system by setting upper and lower thresholds of soil moisture. Finally, we made the network nodes live eight times longer when we applied the hybrid operating mode instead of fully active mode.

Keywords: Blynk IoT; ESP-NOW; Internet of Things (IoT); Wireless sensor networks; Smart gardening; IoT node energy optimization.

1. Introduction

Conventional agricultural systems face challenges and difficulties such as freshwater shortages, world population increases, food production demands, and climate change effects. Also, a huge amount of water wasted every year due to inefficient traditional approaches of irrigation. This situation leads us to spot the light on issues that related to agricultural systems and work on to improve them. At the same time, technology always is used to conduct solutions for any challenge that is related to humans' life to make it easier and confident, which leads to minimizing defects and difficulties. One of these technologies is an Internet of Thigs-IoT. This technology becomes attractive since it provides remote monitoring and controlling which may make the distances among the monitored, the controlled sites and user locations thousands of kilometers [1]. All of this can be achieved with low power consumption, low cost, and using public wireless communication protocols.

Smart agriculture system based on wireless communications and IoT technology can provide promising solutions. It can irrigate farms and gardens efficiently, remotely, automatically as well as avoiding under-watering or over-watering situations. This improves the quality and quantity of farm products, and this can reduce water wastage and energy consumption. Also, it is minimizing human interaction and exitance in their lands. LoRa technology is used in smart agricultural projects instead of Wi-Fi and GSM for large scale coverage area, low power consumption, and cheaper choice. In [2], one sensing node is implemented to send the collected data via LoRaWAN to an actuating microcontroller to control (turning ON/OFF) water pump.

While, in [3], a star wireless network topology was built to collect data from two different farms. Each farm included one sensing node to monitor: soil moisture, temperature, humidity, rain sensor, water level sensors. The distance between a gateway (master) node and a sensing node1 and a sensing node2 were 700 and 500 meters respectively. Also, in [4] LoRaWAN was used in star topology by employing number of nodes equipped with LoRa SX1278 LoRa 433 MHz module. The collected data went via LoRa to the gateway (master) node which is ESP8266 Node MCU board to upload the data via the internet to a google sheet. There, the data can be monitored and the number of nodes can be configured by turning ON/ OFF them from the cloud. Furthermore, Raspberry Pi board was used as a gateway device of the network and MySQL techniques were employed to manage the database with Apache software [5]. Some projects add a solar cell in addition to LoRaWAN to achieve low power and self-powered project. In [6], Each end node powered by solar cell 7cm x 6.5 cm to charge lithium battery to monitor soil temperature, moisture, electrical conductivity, and CO2. There were eight nodes in a large scale garden with wireless communication range 3,422 meters and current consumption 13mA of each node.

RF modules are the other tools that utilized to build wireless sensor networks for large scale farm or garden. In [7], HC-12 module was employed to create a tree topology wireless sensor network to extend the distance that can be covered by the network. The distance between two nodes is 250 Meters and can be increased by adding intermediate nodes to reach up to 750 meters. Also in this system each node included a solar cell and a recharge unit. On the other hand, ZigBee wireless communication protocol can be employed to create medium coverage range of wireless sensor networks. This protocol can perform very well in complex networks. In [8], ZigBee technology is utilized in star topology wireless network which included four sensing/Actuating nodes. Each of those four nodes was configured as a router device in the ZigBee network, while the fifth node which is the central node was configured as a coordinator device in the ZigBee network. Arduino UNO boards were the microcontroller boards which were connected to ZigBee modules in the router nodes, while Raspberry pi board was used in the central node as a processing unit to send controlling commands back to the router nodes and as a gateway to upload the collected data to a Thingspeak IoT platform for data visualizing purpose. Whereas, ESP-NOW wireless communication protocol was utilized for small scale farms, low power, low latency, creating complex wireless sensor networks. In [9], built a network of four nodes communicated via ESP-NOW. Three of them end nodes collected (soil moisture, and air temperature and humidity) , while the fourth nodes was a central (sink) node (gateway) to upload the data to a firebase database and the sink node included SD data logger to record the collected data to have access to the data manually web-based user interface was designed for users to create their personal accounts. This website was linked with the database to visualize the data. The whole system used NodeMCU ESP8266 boards. Finally, in [10], Raspberry Pi3 board was used as a central (gateway) node and four sensor nodes were built in star topology. Each sensor node includes Arduino Mega, ESP8266 Wi-Fi module, MQ2 gas, DHT11, and water level sensors. The data collected from the sensor nodes goes to the central node via the Wi-Fi and it was uploaded to the cloud through it.

Two nodes only were used to implement wireless communication between the Sensing/Actuating node and the central (gateway) node. In [11], two nodes were used one for sensing/ actuating purposes used Arduino UNO board and the second node is the base station node used ESP8266 board to upload the data to the cloud. NRF24L01 modules were on both nodes to provide wireless communication between the two nodes. In contrast, in [12], two NodeMCU ESP8266 boards were employed. One was a sensing and the other was a gateway and the communication between them secured by Wi-Fi.

One node only was utilized for sensing, actuating, and uploading to cloud. For example, in [13], Arduino UNO with Ethernet shield were used for internet connectivity to upload soil moisture, air humidity and temperature to ThingSpeak IoT platform. This platform was also involved in [14, 15] to monitor soil moisture, fire and motion that may occur in the farm. Adafruit.io platform can be alternative choice to visualize the data [16]. While in [17], Arduino UNO with GSM module were employed to achieve the internet connectivity to upload temperature, humidity, motion, and water level. The GSM module was used to upload data via mobile communications networks Also to send waring SMS messages for specific phone numbers. Additionally, in [18], Arduino UNO with NodeMCU board were used to accomplish internet connectivity. The data (barometric pressure, air temperature and humidity, soil moisture, and light intensity) uploaded to a Blynk server and the user interface to visualize the data for monitoring and controlling purposes on a Blynk mobile application. Also, in [19], the NodeMCU ESP8266 was used as a main board to collect data and upload it to the Blynk, where the data monitored and in the Blynk app a slider widget used to set soil moisture thresholds that the watering depended on. Whereas, Raspberry Pi 3 board was used as a gateway to upload soil moisture, air temperature and humidity to a Microsoft Azure IoT platform the processing of the data occurred in the IoT platform by using an irrigation scheduling model. This determined when and for which time the water pump should be ON compared to thresholds set by users. Moreover, users were able to control the water pump through voice commands via Android mobile application [20]. In addition, in [21], the Raspberry Pi3 collected Nitrogen-Phosphorus-Potassium (NPK) sensor, LDR photoresistor light intensity sensor, the data uploaded to google cloud database. A fuzzy logic is used to detect the level of nutrients based on the uploaded data so the system sends SMS message for the farmer about the fertilizer that required.

Some kind of plants and crops require special growth environments. For example, in [22], sensors for light, humidity, temperature, and water level to monitor coriander vegetable growth from seeding to fully grown. Height, trunk width, and leaf width of the vegetable were compared between a controlled farm environment and a conventional farm environment. The results showed better vegetable and ready to harvest for shorter time in the controlled one. Mushrooms also is one of the crops that require specific range of temperature and humidity to grow up [23]. Furthermore, in [24], irrigation and fertilizer systems were implemented for taking care of chili plant. By monitoring Soil moisture and PH level and by using fuzzy logic to control the water flowrate, alkali, and acid solutions, the soil was kept within the desired ranges for that specific plant. The chili plants under the controlled environment were better than others after 8 weeks of monitoring. Also, Brassica rapa, Lactuca sativa, and Brassica integrifolia planted inside a greenhouse and the light intensity, air temperature and humidity, and PH level sensors were installed for monitoring purposes. If the sensors values cross some thresholds fan, light, and evaporative cooler are turned on to keep the environment within limitations. Raspberry Pi used to perform image processing program to compare between the plants that implanted under the controlled greenhouse and others. The comparison shows that the plants under the control had better growth [25]. Finally, other research studied the effects of relaying on smart agricultural system on crop productivity and this study [26] showed the productivity increased 10 % over the traditional methods.

The problems that are targeted by this paper are wasting fresh water and fertilizers, random irrigating intervals and quantities which may lead to have dried or overwatered lands, optimize wireless network nodes energy consumption, farmers and gardeners must stay close to their lands since they have to interact directly with. As a result, this paper provides promised solutions for the mentioned issues and assist in recognizing plants' needs and providing an adequate care. Additionally, it can help to schedule faming duties such as planting, seeding, irrigating, soil fertilizing, and harvesting based on different seasons and weather conditions.

2. Methodology

2.1. Proposed System Overview

The methodology of this project depends on monitoring and watering a house's outdoor garden as a case study. The garden's dimensions are 6 X 4 m2, and it is divided into four quarters (zones). There are two kinds of sensors (light intensity and soil moisture sensors) are distributed in each zone to collect then send data to a central node. The central node is equipped with a weather temperature and humidity sensor, so it uploads all the data to a Blynk server. At the same time, the irrigation process of those four zones is managed by two different methods. First, by each node's microcontroller which depends on the soil moisture sensor readings to take automatic decision. Second, by using the Blynk IoT platform user interface to switch on or off.

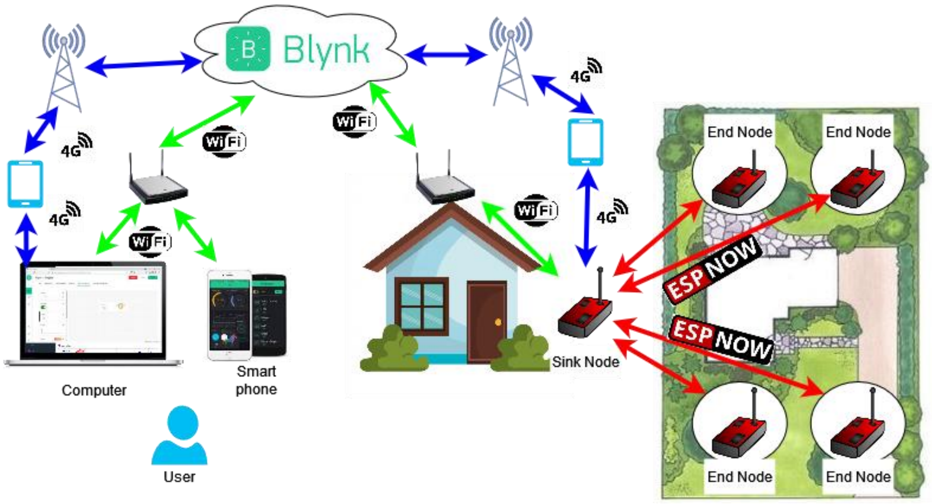


Fig.1 overview of the project

Fig. 1, shows the overview of the proposed system. A star topology of wireless sensors network is established bi-directional communications between the four end-nodes and the central node using the wireless communication protocol “ESP-NOW”. Next, the sink (central) node communicates with the cloud via the Wi-Fi or the cellular network (4G or 4G LTE). On the other side, users can monitor and control parameters via a smart phone and/or a computer from anywhere in the world. This methodology includes seven sub-sections, as follows:

2.2 Four End Nodes (Sensing/Actuating Nodes)

All four nodes in the garden have the same hardware and software functionality. Fig. 2, presents the end node’s black diagram, which includes microcontroller board (NodeMCU V1.0), two sensors (Light and soil moisture sensors), relay, solendoid valve, DC power supply, DC- DC step down convertor, and ESP-NOW to communication with the central node.

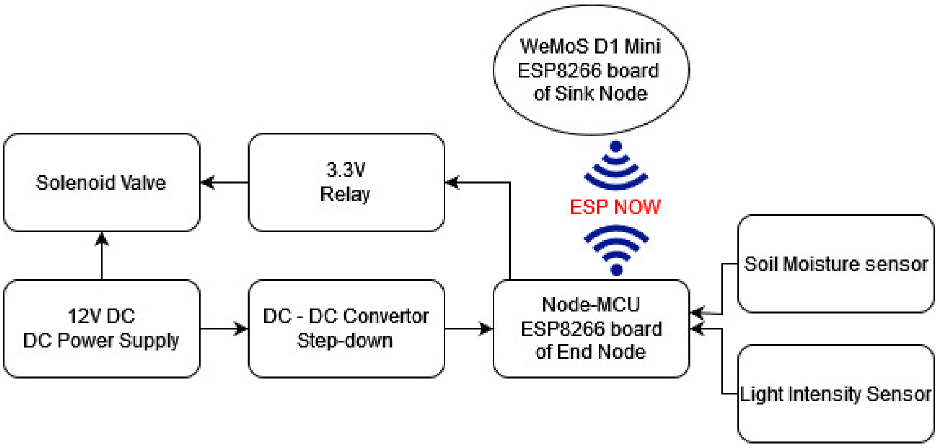


Fig. 2 Black diagram of the end node

Fig.3 shows the wiring diagram of the end node which includes photoresist or sensor (LDR) to detect the sun light intensity, FC-28 soil moisture sensor to measure the soil water content, power supply (four 18650 rechargeable lithium batteries (4800 mAh, 3.7 volts) connected in seriers), DC-DC step-down buck LM2596 to convert 14.8V to 3.3V, 12V DC (ZE-4F180) solenoid electric valve (normally closed) to open and shut the water, and 3.3V relay module. The two sesnors get their power from the microcontroller board, where it gets its power from the DC-DC step-down buck converter. The relay is powered and controlled by the microcontroller board while the solenoid valve is powered by the batteries through the relay.

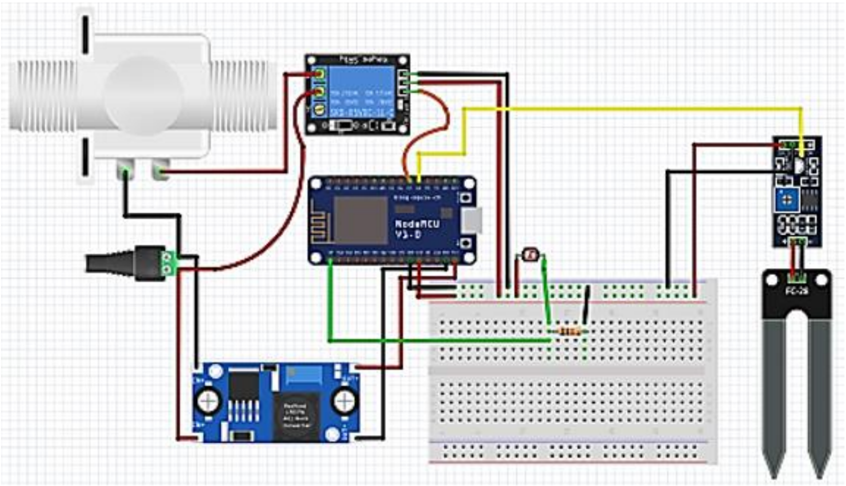


Fig. 3 Wiring diagram of the end node

Fig.4 illustrates the practical implementation of the end-node. It shows plastic cubic which includes all the components of the node. Also, it shows the soil moisture is inserted in the ground and valve is ready to be connected with water hoses.

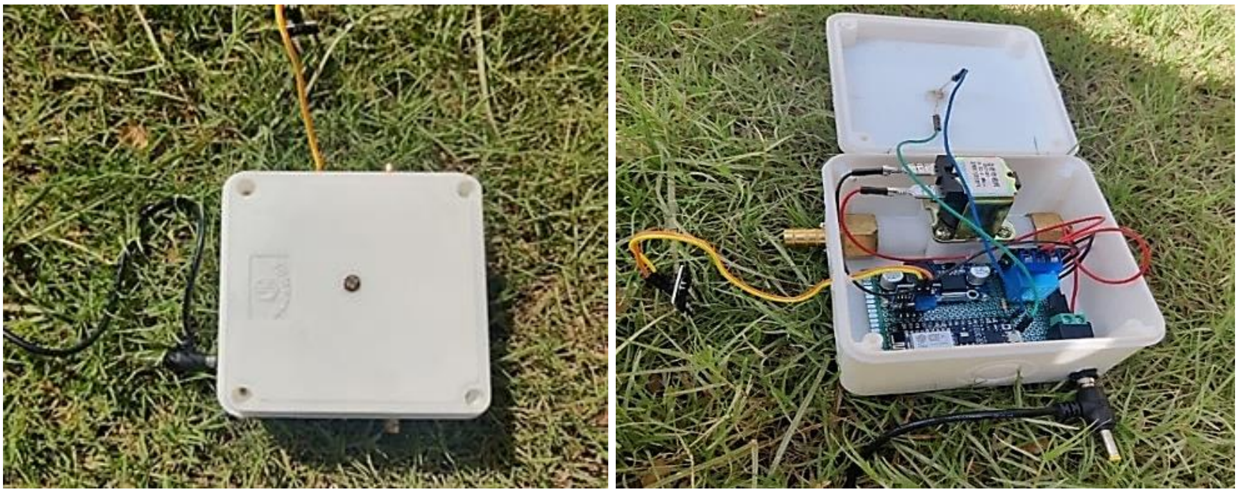


Fig. 4 The physical implementation of sensing & actuating node

Fig. 5, presents the algorithm flowchart of the end-node. It started with checking the controlling signal that comes from the IoT platform user interface. This signal is used to control a specific solenoid valve in a specific garden zone. If there is a signal, it is tested to see whether its value is HIGH or LOW. If there is no signal, the end-node continues its work which is clarified by the flowchart.

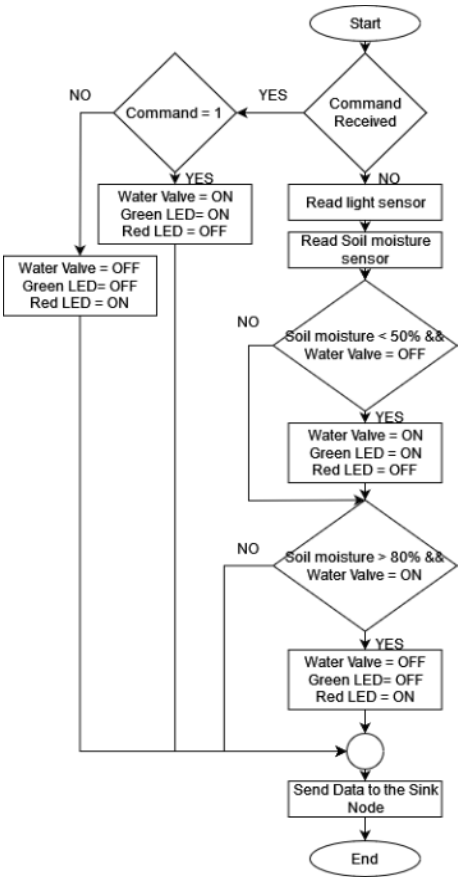


Fig. 5 End-node algorithm flowchart

2.3 Sink node (central node)

Fig. 6, shows the block diagram of the central node which represents the gateway of the wireless network between the four end-nodes and Blynk server. This block diagram presents the four end-node which equipped with ESP8266 or ESP32 microcontrollers communicate wirelessly with the sink node microcontroller via ESP-NOW protocol which it communicates with the server via Wi-Fi router access point. Additionally, there is a weather temperature and humidity sensor is attached to the sink node to measure the weather condition of the garden.

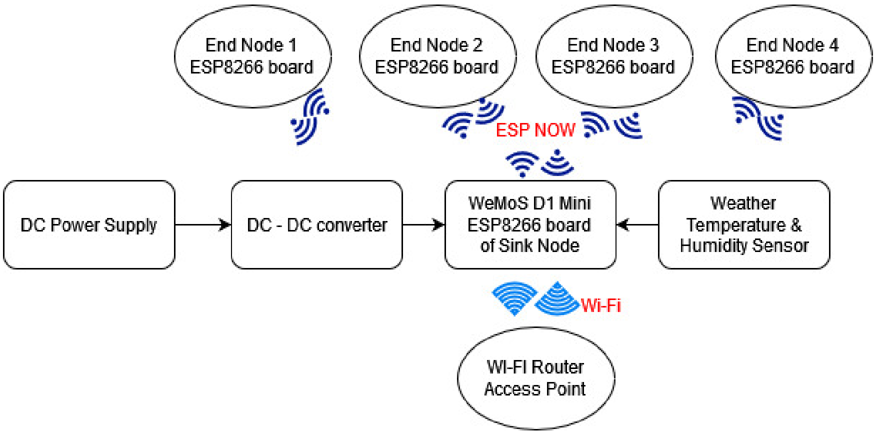


Fig. 6 The block diagram of the sink node

Fig.7. presents the sink (central) node's hardware connection which includes microcontroller board (WeMos D1 Mini), air temperature and humidity sensor (DHT22), and DC-DC Step Down Buck LM2596. The node is powered by two 18650 rechargeable lithium batteries (3800 mAh, 3.7 v). The microcontroller is powered by the DC-DC step-down converter, and the sensor gets its power from the WeMos D1 Mini.

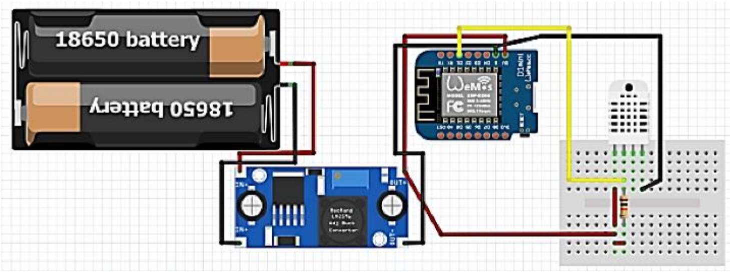


Fig. 7 Wiring diagram of the sink node

Fig 8, shows the plastic box which includes all the components of the sink node.

Fig. 8 The physical implementation of the sink node

The algorithm that is shown in Fig.9 explains how the central node deals with the collected data from the garden, and it checks whether there is a signal from the IoT user interface or not. Lastly, it uploads the data to the Blynk IoT platform.

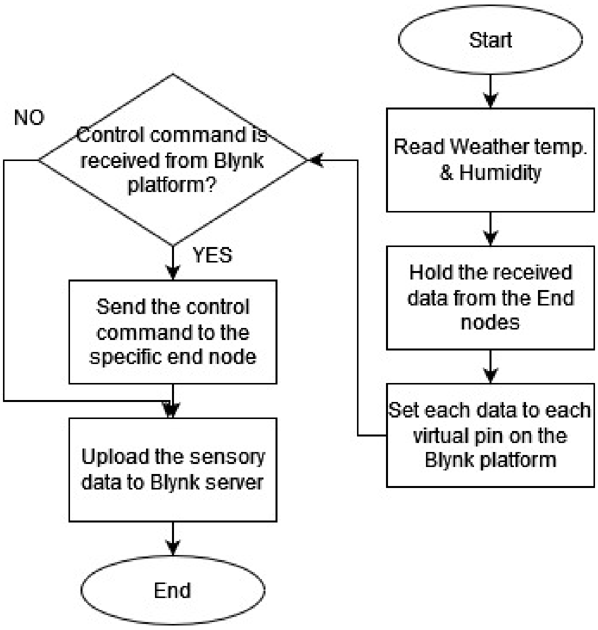


Fig. 9 Sink node algorithm flowchart

2.4 ESP-NOW protocol

ESP-NOW can be defined as a fast wireless communication protocol developed by “Espressif organization" to work for ESP boards family for IoT applications with low power and low latency. This communication requires initial paring and acknowledgement response. After pairing, the connection becomes secure and peer-to-peer with no need for handshake process. It allows each node to operate as a transmitter, receiver, or transceiver [27, 28]

ESP-NOW using the same frequency as the Wi-Fi 2.4GHz but it is not similar to its overhead. ESP-NOW is shorter. The data rate can be 250kbps as a maximum value, so it is suitable for sensory data and it is not suitable for big bandwidth data like media files. The peer numbers of this protocol depend on the type of the utilized ESP board. If the ESP32 board is used to build the network, there are up to 20 devices (nodes) can join the network. In contrast, if the ESP8266 devices are used, there are up to 10 devices (nodes) can join the network. the message between the two devices is encrypted so it adds security concern solution. The coverage area of the ESP-NOW is same as Wi-Fi coverage range 30-50 meters indoor and 100 meters outdoor because both are using 2.4GHz frequency band. This protocol offers flexible kinds of wireless sensor networks topologies such as star, point-to-point, mesh, tree, peer-to-peer, and hybrid topologies [29].

We can build a network without the need to a router device. It is an ideal solution for communication within local network. The transmitted data is encapsulated in a vender specific action frame which includes many fields such as MAC address of sender and receiver, data massage, and vender specific content. This field enable manufacturers and developers to add unique data to their devices which makes devices with same vender communicate without interfering with other devices using ESP-NOW with different vender from different manufacturer [27].One of the most important feature of ESP NOW is the boards of ESP development boards (ESP 8266 and ESP32) can communicate with each other via the ESP NOW protocol wirelessly. This solves the problem of using various board brands within the same ESP-NOW local network.

ESP NOW can secure two communication methodologies. First, one-way communication can be divided into two types “One-to-Many” and “Many-to- One”. The One-to-Many is suitable to construct a remote-control system, where one ESP board sends data to multiple ESP boards. While, the Many-to-One is suitable to collect data from multiple sensor nodes, where multiple ESP boards send data to one ESP board. Second, two-way communication can be used to build up complex wireless networks because each board can act as both a sender and a receiver (transceiver). So, it is suitable create a mesh network [28].

2.5 Project’s sensors

First, light plays a vital role in the plant's development cycle. Since light is one of the main factors in the photosynthesis process and day and seasonal time sensations, it also influences the plant's growth pattern. Additionally, different plants kinds require various levels of light intensity. Therefore, light intensity is monitored from four spots in the garden. Second, the soil water content is a significant element for plants growth and productivity. Also, different kinds of plants require different levels of wet. Therefore, this parameter is monitored and controlled in this project to prevent overwatered soil and dry soil situations. Finally, monitoring weather conditions (air temperature and humidity) is crucial to taking care of plants and cultivating the right plant species. Also, it helps farmers to predict watering schedules, soil fertilizer needs, harvesting processes, and managing their farming duties. Consequently, these sensors assist to achieve healthy and productive farm.

2.6 Blynk IoT platform.

There are two versions of the Blynk application. The older one is called “Blynk legacy” (Blynk 0.1) which has stopped being supported by Blynk’s authors. Therefore, the newer version (Blynk 2.0) is employed in this project. This copy has a new mobile application which requires an Android 5 or later version and iOS 14.1 or later to operate on. Also, it allows users to create a web dashboard in addition to a mobile dashboard. In contrast, the older version offered the mobile dashboard only. All the uploaded data to the Blynk server is displayed on both the mobile and web dashboards simultaneously. At the same time, users can control the water valves by using either the mobile or the web dashboard, and the status of the pressed switch is updated on both dashboards concurrently.

2.6.1 Blynk web application dashboard

At first, a free account can be created on the “https://blynk.io/” website. Then, a project template should be built to design the desired web dashboard. Where, each widget represents a monitored or a controlled parameter, and it should be assigned to a certain virtual pin. Fig. 10, shows the web dashboard, which includes gauge widgets to display the weather and the light intensity sensor values. Also, labeled value widgets reveal the soil moisture sensor statuses. Slide switch widgets control the solenoid valves. Finally, red LEDs and green LEDs are used to indicate whether the soil moisture status is below or above the specific threshold.

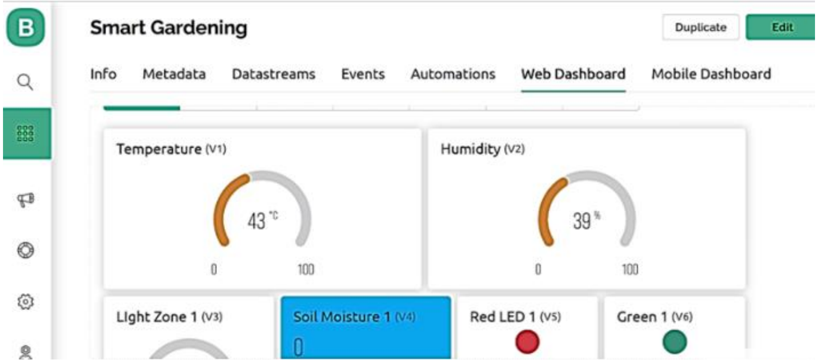


Fig. 10 Blynk web application dashboard template

2.6.2 Blynk mobile application dashboard

First, the Blynk IoT mobile application should be downloaded and then installed. Next, a developer mode should be accessed via pressing on a wrench icon, where all the created templates in the web console exist. After that, the desired template should be selected to open a canvas, where IoT systems developers must add the same widgets as in the web dashboard to make both dashboards have similar functionality. See Fig. 11.

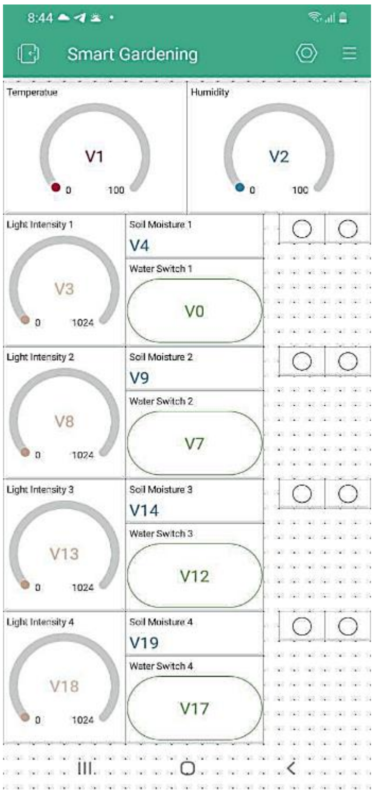


Fig. 11 Mobile application dashboard template

2.7 Energy Consumption Optimization

ESP8266 and ESP32 have different power consumption modes. So we depend on in our proposed project. At the beginning, the active mode was used in (Node MCU and Wemos D1 Mini) boards. Unfortunately, we recognized that mode is energy hunger. Therefore, we switched to a deep sleep mode by connecting the RST pin to the D0 pin of ESP8266 microcontrollers and using ESP.deepSleep (μs) function to identify the sleeping time. The boards get into deep sleep mode based on a timer, and wake up when a LOW signal goes from D0 pin to RST pin. Additionally, for more current consumption reduction, we supplied power to the sensors from the digital pins of the boards to turn them off completely during the deep sleep mode.

We relied on three mathematical equations to analyze and optimize the battery lifespan of end nodes and sink node.

$$Battery\ Life(h) = \frac{Battery\ capacity\ (mAh)}{Load\ current\ consumption\ (mA)} \qquad (1)$$

$$Total\ current\ cunsumption\ per\ cycle = \sum_n In\Delta tn \ (2)$$

Where In: the current consumption for a period Δtn.

$$Number\ of\ Cycles = \frac{Battery\ capacity}{Total\ current\ consumption\ per\ cycle} \quad (3)$$

2.7.1 End Nodes Energy Consumption Optimization

Each end-node is powered by using four batteries model, No. 18650 UltraFire, 4,800 mAh 3.7V Li-ion, which are connected in series to achieve 14.8V. Table1 presents one end-node components' current consumption in active and deep sleep modes.

Table 1. the current consumption of one end-node during active and deep sleep modes.

Component name	NodeMCU ESP8266 board	Soil moisture sensor	Light intensity sensor	Solenoid valve	Relay 3.3V	**Total consumption**
The current consumption in active mode	109mA (active mode)	4.25mA	0.5 mA	450mA	47mA	**610.75 mA**
The current consumption in deep-sleep mode	100 µA (deep sleep mode)	0 mA	0 mA	0 mA	0 mA	**100 µA**

Using eq.1, the battery lifespan in active mode is estimated as follow:$4800\text{mAh} / 610.75\text{mA} \simeq 8$ hours. However, the battery lifespan in deep-sleep mode is estimated as follow: $4800mAh / 100\mu A = 48{,}000\ hours$.

We optimized the system's energy by utilizing active-sleep cycle every 2 hours. One hour and forty-five minutes is in the deep sleep mode and fifteen minutes only is in the wake-up (active) mode.

We use eq.2 to find out the current consumption of the one end-node for one working cycle (1hour and 45 minutes' deep sleep time) + (15 minutes' active mode).
1) The current consumption during deep sleep mode: $100\mu A \times 1.75\ h = 175\mu Ah = 0.175mAh$
2) The current consumption during active mode: $610.75mA \times 0.25\ h = 152.688mAh$
3) The current consumption per cycle = $0.175 + 152.688 = 152.863mAh$

Using eq.3 to calculate the number of cycles that the battery can power the end-node before replacing or recharging the battery.
1) $Number\ of\ cycles = 4800mAh / 152.863mAh = 31\ cycle$ approximately, and each cycle represents 2 hours.
So, the battery lifespan 31 X 2 = 62 hours. Means two and half days. However, if we do not use deep sleep mode the battery energy discharged during only 8 hours as it is clarified earlier.

2.7.2 Sink Node Energy Consumption Optimization

We used two batteries of model No. 18650, 3800 mAh, 3.7V Li-ion, to power the central node. Those batteries were connected in series to achieve 7.5V.
Table 2 presents all the components of the central node (sink node) and their current consumption during both of ESP8266 working modes. Also, to emphasis the effectiveness of using deep sleep mode to increase the life span of the node.

Table 2. The current consumption of the sink-node during active mode and deep sleep mode.

Component name	WeMos D1 Mini ESP8266 board	DHT22 sensor	**Total consumption**
The current consumption in active mode	185 mA (active mode)	0.35 mA	**185.35 mA**
The current consumption in deep-sleep mode	93µA (deep sleep mode)	0 mA	**93 µA**

Using eq.1, the battery lifespan in active mode is estimated as follow: $3800mAh / 185.35 \cong 20.50\ h$. However, the battery lifespan in deep-sleep mode is estimated as follow:
$3800mAh / 93\mu A = 40{,}860\ h$.

We use eq.2 to find out the current consumption of the sink node for one working cycle (1hour and 45 minutes' deep sleep time) + (15 minutes' active mode).
1) The current consumption during deep sleep mode: $93\mu A \times 1.75\ h = 163.75\ \mu Ah = 0.16275\ mAh$
2) The current consumption during active mode: $185.35mA \times 0.25\ h = 46.338\ mAh.$
3) The current consumption per cycle = $0.16275 + 46.338 = 46.5\ mAh.$

Using eq.3 to calculate the number of cycles that the battery can power the sink node before replacing or recharging the battery.

1) $Number\ of\ cycles = 3800mAh\ /\ 46.5\,mAh\ =\ 81.7\,cycle$ approximately, and each cycle is 2 hours, so the battery can survive 81.7 × 2 = 163.4 hours. Means the battery life span is six and half days. However, if we do not use deep sleep mode the battery energy is discharged during only 21 hours as it is mentioned earlier.

3. Results and Discussion

The proposed smart agricultural system monitors in real-time values of the soil moisture and the light intensity of four nodes which are installed in different spots in the outdoor garden. Also, the air temperature and humidity of the garden is monitored by the central node. Furthermore, this system provides us the capability of controlling four water valves to irrigate the garden remotely, automatically, and manually.

The communication between the cloud and the system is secured by using the Wi-Fi connection or by using the mobile communication networks (4G or 4G LTE). When the system loses both of those connections the gathered data cannot be uploaded to the cloud. However, it can keep controlling the soil moisture levels because the upper and lower thresholds are set in each microcontroller of end nodes to control the irrigation of each zone.

3.1 Results of Controlling Water Valves

First, each end node is able to shut and open the water automatically and independently. The upper threshold is set in the whole end-nodes to 80% of soil moisture while the lower threshold of all of them are 50%. The experiment started at soil moisture = 0% which means the soil was completely dry. See Table 3. When the digital reading of the sensor becomes (1) that means the soil moisture value is less than the lower threshold, while when the sensor reading becomes (0) that means the soil moisture value is above the lower threshold. The sensor readings are from 0-1024 we make mapping using map function to make it 0-100 (0 means dry and 100 means completely wet) we use the potentiometer on the sensor to adjust the lower threshold by the comparator LM393 (the sensor give LOW when the sensor analog reading cross the threshold). The ADC of the microcontroller is 10-bits so the reading range between 0-1023. The moisture value is in range 0-100 percentage so we calculate the moisture percentage by using eq.4:

$$Soil\ moisture\ \% = \frac{sensor\ value}{1023}\ x\ 100\% \qquad (4)$$

Table 3 presents the status of one end-node based on time intervals, and we can notice that the irrigation process is started when the soil moisture level was less than the lower threshold (50%). Then it is stopped, when the soil moisture level reached to the upper threshold (80%). After that, the water valve shut until the soil moisture decreases and reaches to the lower threshold again to open water again and so on. Second, we can notice that the slide switch in the IoT platform (see Fig. 12) was in OFF position during all the processes. However, when the user presses the switch to make it ON, the water valve becomes open and irrigates the specific land without any consideration to the soil moisture sensor readings.

Table 3. Results of the water valve situations

Time (hour)	Soil moisture (%)	Soil moisture sensor digital output	The slide switch on the platform	Water valve condition
0	0%	1	Off	Open
0.30	64%	0	Off	Open
1	79%	0	Off	Open
11	86%	0	Off	Close
19	73%	0	Off	Close
24	65%	0	Off	Close
30	57%	0	Off	Close
35	45%	1	Off	Open

3.2 Results of light intensity monitoring

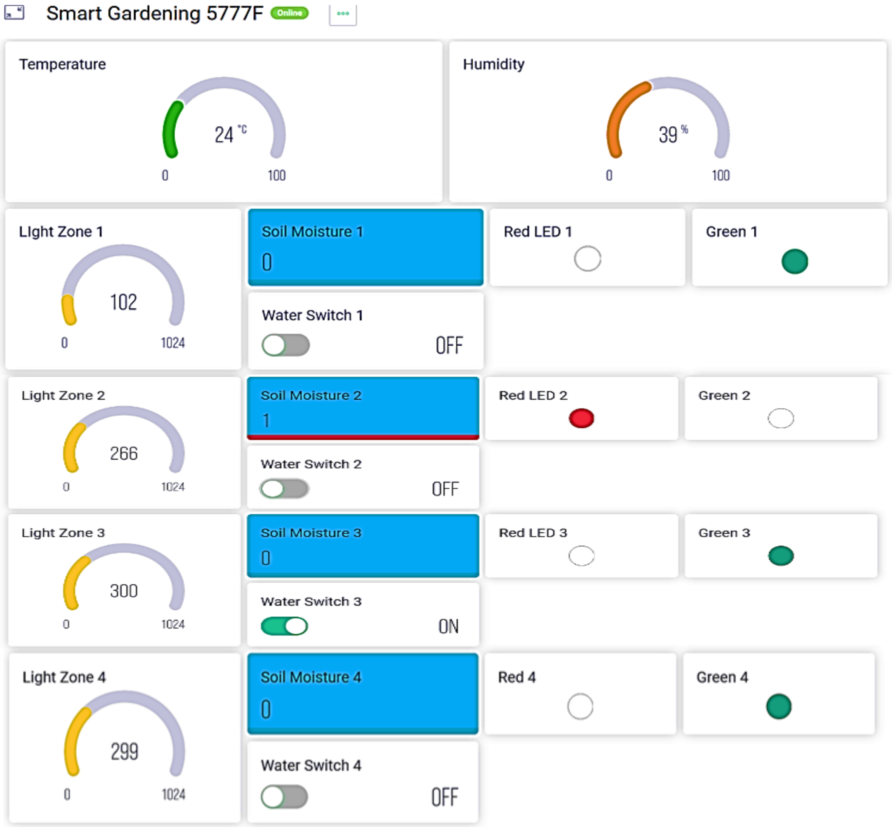
Table 4 shows the light intensity readings of the four nodes during the entire daytime. We can recognize that node 1 and 2 receives higher light intensity than nodes 3 and 4. The benefit of those records, we become able to know the distribution patterns of light intensity that hit the garden. So, the garden’s caregivers can depend on to plant suitable plants within the appropriate spot in the garden.

Table 4. Light intensity sensors (photoresistors) readings

Time	Light intensity Nodes 1 & 2	Light intensity Nodes 3 & 4
9:00	984	956
11:00	991	959
14:00	991	960
16:00	985	954
18:00	983	943

3.3 Results of Blynk web-based user interface

Fig.12, exhibits the web-based dashboard user interface which provides monitoring and controlling facilities. The two big gauge widgets display the outdoor weather temperature (0-100 °C) and humidity (0-100 %). The four gauges (on the left side) present the light intensity readings of the four garden zones, and the labeled value widgets reveal the soil moisture values (0 or 1) of each garden spot. Furthermore, the green indicator LED becomes ON of that the specific garden zone, when the soil moisture value is (0) while the red indicator LED become ON, when the soil moisture value is (1). Finally, users can control the solenoid valve of any garden zone via the slide switch without considering the soil moisture values.



Smart Gardening 5777F Online

Temperature 24 °C

Humidity 39 %

Light Zone 1: 102

Soil Moisture 1: 0

Water Switch 1: OFF

Light Zone 2: 266

Soil Moisture 2: 1

Water Switch 2: OFF

Light Zone 3: 300

Soil Moisture 3: 0

Water Switch 3: ON

Light Zone 4: 299

Soil Moisture 4: 0

Water Switch 4: OFF

Fig. 12 The web dashboard user interface

3.4 Results of Blynk Mobile Application User Interface

Fig. 13, presents the Blynk mobile application dashboard, which has similar properties and functionalities as the web application dashboard. The most significant thing to mention here that the Blynk platform on both the mobile application and the web application dashboards are working simultaneously and synchronously. This means any change happen on one dashboard, the other one is updated in real time.

Fig. 13 The Mobile Dashboard User Interface

3.5 Results of Energy Consumption Optimization

When we compare the system operation in fully active mode and hybrid mode (mixed active and deep sleep mode), we recognize that the end nodes in the hybrid mode live longer than in the only active mode by 8 times approximately so the improvement in power optimization is 675% . Also, the sink node in the hybrid mode lives longer than in the only active mode by 8 times approximately so the improvement in power optimization is 678%.

The limitation of using deep sleep mode is that the board's CPU turns off the Wi-Fi module during the deep sleep period, so the controlling signal from the Blynk platform cannot be received by the sink node microcontroller.

4. Conclusion and Future Work

The classic agricultural methodologies and techniques are facing challenges about wasting resources and crop qualities. So, we need to involve some sort of monitoring and controlling techniques to help. At the same time, it is not feasible to look after a whole piece of land by observing and controlling one spot. Therefore, a wireless sensor network is crucial to performing this duty. This project aimed to construct a smart agricultural system by establishing a star topology wireless sensor network using ESP-NOW and accessing the internet using the Wi-Fi or the cellular network to be monitored and controlled worldwide. As a result, the land's watering system is controlled based on the soil moisture rate and the user's desire to keep the wet level within an acceptable range. Also, the light intensity and the weather's temperature and humidity are monitored. The monitoring and controlling have been done by two types of user interfaces: web and mobile Blynk IoT dashboards. Also, this research discussed the deep sleep mode as an energy optimization approach to make the wireless sensor network nodes live longer before the recharging is required. The project limitations are that, during the deep sleep mode, the ESP8266 turns off its Wi-Fi module, so the users cannot control the water valves via the Blynk IoT platform. Second, the end node batteries discharge within two days and a half, which requires a local recharge system such as a solar cell unit beside each node. For future work, more sensors can be added and machine learning algorithms can be involved in it to obtain an entire controlled plant care system. Finally, this research provided practical solutions for the smart gardening sector.

References

[1] Prajwal Khupat, Usha Barapatre, Nikita Khadatkar, Sneha Badge, Pooja Meharkar, Shweta Bhelonde, Vaishali C. Asole, Samiksha S. Thul, "IOT Based ECG Monitoring and Analysis System with Machine Learning", *International Journal of Computational and Electronic Aspects in Engineering*, RAME Publishers, vol. 2, issue 3, pp. 51-56, 2021. https://doi.org/10.26706/ijceae.2.3.20210602

[2] Catak, M., 2021. IoT Based Smart Irrigation System with LoRa. In *Progress in Intelligent Decision Science: Proceeding of IDS 2020* (pp. 703-709). Springer International Publishing. https://doi.org/10.1007/978-3-030-66501-2_57

[3] Khoa, T.A., Man, M.M., Nguyen, T.Y., Nguyen, V. and Nam, N.H., 2019. Smart agriculture using IoT multi-sensors: a novel watering management system. *Journal of Sensor and Actuator Networks*, *8*(3), p.45. https://doi.org/10.3390/jsan8030045

[4] Hoan, T.N.K., Qui, N.C., Truong, N.P. and Danh, L.V.Q., 2021, February. A reconfigurable IoT-based monitoring and control system for small-scale agriculture. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1088, No. 1, p. 012047). IOP Publishing. DOI: 10.1088/1757-899X/1088/1/012047

[5] Wang, C.Y., Tsai, C.H., Wang, S.C., Wen, C.Y., Chang, R.C.H. and Fan, C.P., 2021. Design and implementation of LoRa-based wireless sensor network with embedded system for smart agricultural recycling rapid processing factory. *IEICE TRANSACTIONS on Information and Systems*, *104*(5), pp.563-574. DOI: 10.1587/transinf.2020NTI0001

[6] Ramson, S.J., León-Salas, W.D., Brecheisen, Z., Foster, E.J., Johnston, C.T., Schulze, D.G., Filley, T., Rahimi, R., Soto, M.J.C.V., Bolivar, J.A.L. and Malaga, M.P., 2021. A self-powered, real-time, LoRaWAN IoT-based soil health monitoring system. *IEEE Internet of Things Journal*, *8*(11), pp.9278-9293. DOI: 10.1109/JIOT.2021.3056586

[7] Saqib, M., Almohamad, T.A. and Mehmood, R.M., 2020. A low-cost information monitoring system for smart farming applications. *Sensors*, *20*(8), p.2367. https://doi.org/10.3390/s20082367

[8] Srivastava, A., Das, D.K. and Kumar, R., 2020, July. Monitoring of soil parameters and controlling of soil moisture through iot based smart agriculture. In *2020 IEEE Students Conference on Engineering & Systems (SCES)* (pp. 1-6). IEEE. DOI: 10.1109/SCES50439.2020.9236764

[9] Koushik, M.S., Srinivasan, M., Lavanya, R., Alfred, S. and Setty, S., 2021, April. Design and Development of Wireless Sensor Network based data logger with ESP-NOW protocol. In *2021 6th International Conference for Convergence in Technology (I2CT)* (pp. 1-5). IEEE. DOI: 10.1109/I2CT51068.2021.9417914

[10] Vij, A., Vijendra, S., Jain, A., Bajaj, S., Bassi, A. and Sharma, A., 2020. IoT and machine learning approaches for automation of farm irrigation system. *Procedia Computer Science*, *167*, pp.1250-1257. https://doi.org/10.1016/j.procs.2020.03.440

[11] Kamaruddin, F., Abd Malik, N.N.N., Murad, N.A., Latiff, N.M.A.A., Yusof, S.K.S. and Hamzah, S.A., 2019. IoT-based intelligent irrigation management and monitoring system using arduino. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *17*(5), pp.2378-2388. http://doi.org/10.12928/telkomnika.v17i5.12818

[12] Visconti, P., Giannoccaro, N.I., de Fazio, R., Strazzella, S. and Cafagna, D., 2020. IoT-oriented software platform applied to sensors-based farming facility with smartphone farmer app. *Bulletin of Electrical Engineering and Informatics*, *9*(3), pp.1095-1105. https://doi.org/10.11591/eei.v9i3.2177

[13] Sharma, S., Sharma, A., Goel, T., Deoli, R. and Mohan, S., 2020, July. Smart home gardening management system: A cloud-based internet-of-things (iot) application in vanet. In *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (pp. 1-5). IEEE. DOI: 10.1109/ICCCNT49239.2020.9225573

[14] Sharma, A., Soni, G. and Aggarwal, N., 2024. Implementation of Smart Agriculture System using IoT on ThingSpeak cloud and MATLAB. In *IOP Conference Series: Earth and Environmental Science* (Vol. 1285, No. 1, p. 012022). IOP Publishing. DOI; 10.1088/1755-1315/1285/1/012022

[15] Abba, S., Wadumi Namkusong, J., Lee, J.A. and Liz Crespo, M., 2019. Design and performance evaluation of a low-cost autonomous sensor interface for a smart iot-based irrigation monitoring and control system. *Sensors*, *19*(17), p.3643. https://doi.org/10.3390/s19173643

[16] Rohini Pochhi, Sandeep Thakre, Balwant Bansod, "Using wireless sensor network for monitoring and controlling of farm", International Journal of Computational and Electronic Aspects in Engineering, RAME Publishers, vol. 2, issue 1, pp. 22-25, 2021. https://doi.org/10.26706/ijceae.2.1.20210408

[17] Rajesh, T., Thrinayana, Y. and Srinivasulu, D., 2020. IoT based smart agriculture monitoring system. The International Journal of Innovative Technology and Exploring Engineering, 9(9), pp.325-328.

[18] bin Sadli, M.D.D., 2019, April. An IoT-based Smart Garden with Weather Station System. In *2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (pp. 38-43). IEEE. DOI: 10.1109/ISCAIE.2019.8743837

[19] Sheth, M. and Rupani, P., 2019, April. Smart gardening automation using IoT with BLYNK app. In *2019 3rd international conference on trends in electronics and informatics (ICOEI)* (pp. 266-270). IEEE. DOI: 10.1109/ICOEI.2019.8862591

[20] Abdelmoamen Ahmed, A., Al Omari, S., Awal, R., Fares, A. and Chouikha, M., 2021. A distributed system for supporting smart irrigation using Internet of Things technology. *Engineering Reports*, *3*(7), p.e12352. https://doi.org/10.1002/eng2.12352

[21] Lavanya, G., Rani, C. and GaneshKumar, P., 2020. An automated low cost IoT based Fertilizer Intimation System for smart agriculture. *Sustainable Computing: Informatics and Systems*, *28*, p.100300. https://doi.org/10.1016/j.suscom.2019.01.002

[22] Tangwannawit, P. and Saengkrajang, K., 2021. An internet of things ecosystem for planting of coriander (Coriandrum sativum L.). *International Journal of Electrical and Computer Engineering*, *11*(5), pp.4568-4576. DOI: 10.11591/ijece.v11i5.pp4568-4576

[23] Ariffin, M.A.M., Ramli, M.I., Amin, M.N.M., Ismail, M., Zainol, Z., Ahmad, N.D. and Jamil, N., 2020, November. Automatic climate control for mushroom cultivation using IoT approach. In *2020 IEEE 10th International Conference on System Engineering and Technology (ICSET)* (pp. 123-128). IEEE. DOI: 10.1109/ICSET51301.2020.9265383

[24] Pezol, N.S., Adnan, R. and Tajjudin, M., 2020, June. Design of an internet of things (iot) based smart irrigation and fertilization system using fuzzy logic for chili plant. In *2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)* (pp. 69-73). IEEE. DOI: 10.1109/I2CACIS49202.2020.9140199

[25] Tolentino, L.K.S., Fernandez, E.O., Amora, S.N.D., Bartolata, D.K.T., Sarucam, J.R.V., Sobrepeña, J.C.L. and Sombol, K.Y.P., 2020. Yield evaluation of brassica rapa, Lactuca Sativa, and brassica Integrifolia using image processing in an IoT-based Aquaponics with temperature-controlled greenhouse. *AGRIVITA Journal of Agricultural Science*, *42*(3), pp.393-410. http://doi.org/10.17503/agrivita.v42i3.2600

[26] Roy, S.K., Misra, S., Raghuwanshi, N.S. and Das, S.K., 2020. AgriSens: IoT-based dynamic irrigation scheduling system for water management of irrigated crops. *IEEE Internet of Things Journal*, *8*(6), pp.5023-5030. DOI: 10.1109/JIOT.2020.3036126

[27] Hoang, T.N., Van, S.T. and Nguyen, B.D., 2019, October. ESP-NOW based decentralized low cost voice communication systems for buildings. In *2019 international symposium on electrical and electronics engineering (ISEE)* (pp. 108-112). IEEE. DOI: 10.1109/ISEE2.2019.8921062

[28] Labib, M.I., ElGazzar, M., Ghalwash, A. and AbdulKader, S.N., 2021. An efficient networking solution for extending and controlling wireless sensor networks using low-energy technologies. *PeerJ Computer Science*, *7*, p.e780. https://doi.org/10.7717/peerj-cs.780

[29] Yukhimets, D., Sych, A. and Sakhnenko, A., 2020, September. Designing a method for constructing distributed open ACS based on the ESP-NOW wireless protocol. In *2020 International Russian Automation Conference (RusAutoCon)* (pp. 642-647). IEEE. DOI: 10.1109/RusAutoCon49822.2020.9208135