DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

Design of Smart Irrigation System for Vegetable Farms Based on Efficient Wireless Sensor Network

Wid Badee Abdulaziz[1], Muayad Sadik Croock[2]
[1]*Computer Engineering Department, University of Technology, Baghdad, Iraq*
[2]*Control and Systems Engineering Department, University of Technology, Baghdad, Iraq*
[1]*ce.19.16@grad.uotechnology.edu.iq*, [2]*muayad.s.croock@uotechnology.edu.iq*

Abstract*— Designing an efficient irrigation system is a crucial issue in agriculture, due to water scarcity problem around the world with the need of increasing agricultural production to satisfy the demands of the enlargement of population. Therefore, to design a smart irrigation system, a real monitoring of field's information that affects the watering status is required which can be achieved with Wireless Sensor Networks (WSN). In this paper, an irrigation system based WSN is proposed to save water, power, labor, and as a result, saving cost with production and profit increase. Sensor nodes collect field data to be sent to the Raspberry pi, as a main controller, to make optimal decisions about irrigation process. The field data includes the sensor readings of temperature and soil moisture. Crop evapotranspiration is also considered; thus, the required amount of water is estimated with a particular irrigation time to avoid over irrigation that hurts the plants growth and yields quality. The obtained results show the efficiency of the proposed system operation and controlling on the irrigation process. These results are taken for tomato plant as a case study. The monitoring tools are used to verify the suggested algorithm effectiveness in irrigation scheduling.*

Index Terms*— Evapotranspiration, Irrigation systems, Raspberry pi, WSN.*

I. INTRODUCTION

Irrigation in agriculture consumes about 85% of the total consumed water in the world and this will increase rapidly with the increase of population [1]. Therefore, developing an optimal irrigation system is necessary to reduce cost, power and water consumption depending on real physical information instead of predicted one [2]. Nowadays, Wireless Sensor Networks (WSN) have a significant role in many application, such as environmental monitoring, medical, agricultural and others due to its flexibility and ability of processing data in different environments with an easy programming [3]-[4]. In agriculture, natural resources can be utilized in an efficient way using WSNs to gather real-time information on soil, weather, and plant growth. Therefore, farmers use this data to make smart decisions regarding to planting to get better yields and increase profits [5].

Many studies proposed smart irrigation systems based on WSNs with limited sensors' readings (soil moisture, air temperature and humidity sensors only), without referring to other climatic conditions such as wind speed, sun radiation. In addition, most of these researches used simulation models to verify the suggested system's operation. In [6], an Internet of Things (IoT) based Smart AgroTech system was proposed to improve farm production using efficient monitoring and scheduling irrigation process using WSNs with ESP8226 controller. The collected data from sensors (soil moisture, temperature, and humidity) compared with real data using comparative analysis and results showed an acceptable error on the charge of lacking in coverage area. While in [7], authors developed a smart drip irrigation system

Received 30/July/2021; Accepted 7/November/2021
 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

depending on real-time soil moisture data in a greenhouse tomato's crop. A comparative study between the proposed system and other irrigation systems was done with results approved that the giving system is efficient to deal with food security and water scarcity issues. In [8], a smart agriculture system was developed with an open source and not expensive hardware components like built a WSN using Arduino and Raspberry Pi to get a cost-effective system. The fuzzy logic control was used as a decision method for the amount and duration of irrigation. An automated irrigation system based on IOT was presented in [9], this was done by distributing different sensors in the farm (soil moisture, temperature and humidity) and the reads are entered to a machine learning algorithm. Authors in [10] proposed an automated drip irrigation system using real-time fuzzy inference method and WSN implemented in Laboratory Virtual Instrument Workbench platform (LABVIEW). In [11], a monitoring and control irrigation system was proposed based on multi agent system and artificial intelligence. The WSN gathered data from environment to evaluate the actual need of water every five minutes and weather map used to calculate the potential evapotranspiration precisely. Moreover, authors in [12] presented an automated drip irrigation with fertigation system based on WSN. It utilized soil moisture data and evapotranspiration for prediction of required water to irrigate the area. This system proposed also a transmission method between cluster head and a base station based on Travelling Salesman Protocol (TSP) to reduce power consumption in data transmission. The results approved that the proposed system used water efficiently. In [13], a developed real-time agriculture application was presented using multi-layered architecture containing three technologies (Internet of Things, cloud computing and context awareness).

At the other hand, authors in the smart drip irrigation system proposed in [14] used smartphone to capture soil image. Calculations were made on the image to know the level of wetness and send the data to a microcontroller to take an irrigation decision depending on the captured image in smartphone and sensors output. Authors of [15] proposed a control system to control environmental parameters that affect the crop production. Three parts included in the control system: hardware component, web application and mobile application. Authors of [16] presented a Solar Photovoltaic Watering System (SPVWS) with multithreading design and virtual timers. Raspberry pi was used due to its efficient performance with real time applications. By multithreading programming used in the system, controlling the system and HTTP server job working concurrently to give remote control of the irrigation system and online monitoring of data. Authors in [17] suggested a smart farm management system based on WSN to schedule irrigation, feed animals, and manage farm production. For irrigation scheduling, an Arduino UNO microcontroller controls soil moisture, temperature and humidity sensors which were used to gather real-time data relating water status in soil and air temperature and humidity. The microcontroller sends the collected data to control center via Bluetooth for processing and storing in data base. Control information is sent from control center to the Arduino to manage irrigation process.

The proposed system in [18] used IOT to automate controlling and monitoring of irrigation process. Sensor devices collected environmental data from the field and regression algorithm used this data to determine water requirements in a day. Authors of [19] proposed a watering system with future prediction of soil moisture content using Neural Network. The expectation strategy utilized the collected data from sensors such as soil and weather parameters. Authors of [1] proposed an automated irrigation system based on WSN to gather data for irrigation process decision. A web application in smartphones also used to give notifications when the soil is dry or the level of the water in the tank is low. The WSN connected with the server using 4G-Long-Term Evolution (LTE) technology due to its flexibility. In addition, authors in [20] presented a simulation model to a smart irrigation system based on WSN. In the proposed model, WSN used cluster topology to ensure a scalable network and increase sensors life time. The proposed system used fuzzy inference system to irrigation control. Simulation results showed that the system performance is accurate with economical use of water and power.

Received 30/July/2021; Accepted 7/November/2021

ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

In this paper, a smart on-demand irrigation system is proposed based on WSN. Two sensor nodes are used to detect watering status, and send real-time data to raspberry pi for optimizing the right decision about starting or stopping irrigation process. In addition, the Raspberry pi determines the required amount of water depending on climatic conditions and crop data, and irrigation run time is also calculated to specify how long the particular plant should irrigate. As a case study, the tomato plant data is chosen to verify the operation of the proposed system. Table I shows a short brief of the work in the previous papers.

TABLE I. BREIF OF LITERATURE REVIEW

No.	Author	Sensors and Devices	Software	protocols/techniques	Notes
11	A. K. Podder *et al.* (2021) [6]	DHT11, soil moisture sensor, ESP8266, water pumps	Arduino Integrated Development Environment (Arduino -IDE)	Internet Of Things (IOT)	-Model has not considered weather parameters -No irrigation scheduling presented
22	R.Liao *et al*. (2021) [7]	Wireless soil moisture sensors,	Not determined	Not determined	-No clear information about the central irrigation controller hardware and software
33	B. Et-taibi, M. R. Abid, I. Boumhidi, and D. Benhaddou (2020) [8]	DHT11, soil moisture sensor, water level sensor, Arduino UNO, Raspberry pi 3, relays	Not determined	General Packet Radio Service (GPRS), Big Data analysis	-No monitoring for systems operation - No information about the programming languages used. - Model has not considered weather parameters
44	A.Vij *et al.* (2020) [9]	soil moisture, temp. & humidity, MQ2 gas sensor, Arduino Mega, Raspberry pi 3	Python	Internet Of Thing (IOT), Machine Learning	-Proposed an automated irrigation system without real implementation.
55	S. Jaiswal and M. S. Ballal (2020) [10]	soil moisture sensor, temperature sensor, humidity sensor, water level sensor, relays.	Laboratory Virtual Instrument Workbench platform (LabVIEW)	Fuzzy inference, General Packet Radio Service (GPRS)/Global System for Mobile Communications (GSM)	-Using of fuzzy logic in controlling irrigation system gives approximate solutions (not accurate)
66	O. Debauche *et al*. (2020) [11]	soil moisture sensor, temperature sensor, humidity sensor, ESP32	CropWat	Artificial intelligent	-Proposed an automated irrigation system without real implementation.
77	K. Karunanithy and B. Velusamy (2020) [12]	soil moisture, temperature, humidity, wind, sun shine sensors	Network simulator	Wireless Sensor Network (WSN), Message Queuing Telemetry Transport (MQTT), Zigbee	-No clear information about the hardware components used in the proposed system
88	R. Dobrescu, D. Merezeanu, and S. Mocanu (2019) [13]	Soil sensors (moisture and ph), environmental sensors (temperature and humidity)	CodeWarrior Bluemix Structured Query Language (SQL)	IOT, cloud computing, context awareness, MQTT, Zigbee	-Paper does not mention any algorithm used to control irrigation process - Model has not considered weather parameters

Received 30/July/2021; Accepted 7/November/2021

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

99	S. R. Barkunan, V. Bhanumathi, and J. Sethuram (2019) [14]	Temperature sensor, humidity sensor, light sensor, rain sensor, ARM microcontroller, motor	Propose an irrigation application on mobile	Global System for Mobile Communications (GSM), Short Message Service (SMS)	-Image processing was used to show soil moisture status which is not accurate as sensors reading - No irrigation scheduling proposed
110	J. Muangprathub *et al*. (2019) [15]	DHT22, soil moisture sensor, NodeMCU, relay	Web-application, mobile application	WSN, WiFi, data mining	-Model has not considered weather parameters - Paper does not mention any algorithm used to control irrigation process -No irrigation scheduling proposed
111	A. Selmani *et al.* (2018) [16]	DHT22, Soil moisture sensor, water level sensor, Raspberry pi, relay	Jave, MySQL	Multithreading, fuzzy logic control, Hypertext Transfer Protocol (HTTP)	-Real implementation of the proposed system is not included - No clear information about the proposed irrigation algorithm
112	M. Al-Janabi, Z. Faris, and A. Taqi (2018) [17]	DHT22, Soil moisture sensor, Arduino UNO, Bluetooth HC-06, relay	Visual Studio C#, SQL	Bluetooth with WSN	-Using Bluetooth for short range wireless transmission data is efficient for prototype model only, since the control center location must be near sensor nodes when using Bluetooth.
113	A. Kumar *et al.* (2017) [18]	soil moisture sensor, temperature sensor, water flow sensor, rain drop sensor, Raspberry pi	Mobile application	Regression algorithm, cloud computing, WiFi	-No clear information about programming languages used in the system - irrigation decision depends on soil moisture value only
114	A. G. Mohapatra and S. K. Lenka (2016) [19]	Soil sensors (moisture and ph), environmental sensors (temperature, humidity, wind speed, and light intensity), Raspberry pi.	Not determined	Neural network, fuzzy logic controller, Zigbee, WiFi	-No clear information about any software used in the system - No real implementation of the proposed system - No irrigation algorithm included
115	B. Khelifa *et al.* (2015) [1]	Soil moisture sensor, water level sensor	Cooja simulator	Zigbee, 4G- Long-Term Evolution (LTE), Constrained Application Protocol (CoAP), HTTPs, IOT	- No real implementation of the proposed system -Human irrigation decision (not smart system) - No clear information about irrigation algorithm
116	M. N. Abdullah, M. S. Croock, and A. K. Mousa (2015) [20]	Temperature sensor, Air humidity sensor, Wind speed sensor, Radiation sensor, Soil moisture sensor	Objective Modular Network Testbed (OMNeT++)	Fuzzy logic inference, IEEE 802.15.4 Wireless Personal Area Network (WPAN)	Proposed a simulation model without real implementation

Received 30/July/2021; Accepted 7/November/2021

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

117	This work	DS18B20 sensor, Soil moisture sensor, ESP8266 microcontroller, Raspberry pi 4	Arduino IDE, Python	WSN, MQTT	-Proposed an irrigation system with real sensor readings and optimal decisions by Raspberry pi related to irrigation demands depending on accurate climatic information and crops data (no learning is required since calculations are done in Raspberry pi every time irrigation is required) -using NodeMCU with Wi-Fi transmission reduces the need for extra transmission hardware. - system is implemented in hardware with results verified its desired operation.

II. THE PROPOSED IRRIGATION SYSTEM

A smart control and monitoring irrigation system based on WSNs is proposed in this work to detect the plants need of water and schedule the irrigation process accordingly. An overview of the suggested system is presented in *Fig. 1*. For each node, soil moisture and temperature sensors are connected with ESP8266 microcontroller and used to collect data from the field. Soil moisture sensor is used to detect the witness of the soil, while temperature sensor is used to detect air temperature. Sensor readings are sent from the controller to Raspberry pi using WiFi connection and MQTT protocol to make decision about starting or stopping the irrigation process. Raspberry pi sends control commands to ESP8266 microcontroller to switch on/off of water pump with the aid of WiFi connection and MQTT protocol. In addition, the amount of the water and the time required to run the irrigation process is determined by Raspberry pi using climatic data stored in a server data base to calculate reference evapotranspiration.

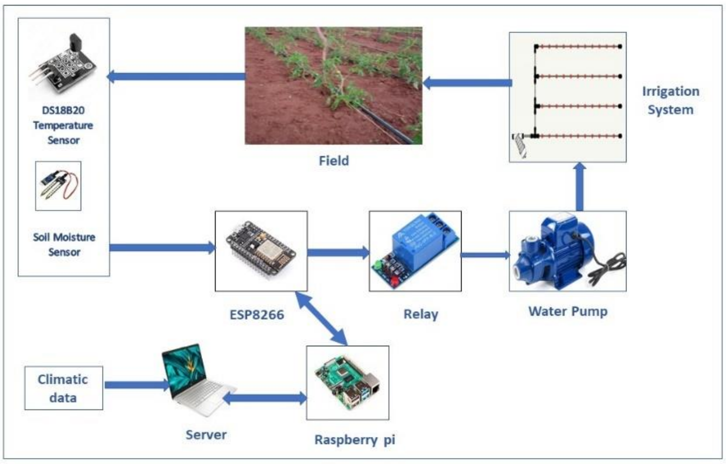


FIG. 1. AN OVERVIEW OF THE PROPOSED SYSTEM

Received 30/July/2021; Accepted 7/November/2021
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

Reference evapotranspiration (Et_0) is calculated in (1) with Penman–Monteith equation [11][21], and [22]:

$$ETo = \frac{0.408\Delta(\mathrm{Rn} - \mathrm{G}) + \gamma(900/\mathrm{T} + 273)\mathrm{U2}(\mathrm{es} - \mathrm{ea})}{\Delta + \gamma(1 + 0.34\mathrm{U2})} \quad (1)$$

Where, $\boldsymbol{ET_0}$ is the potential evapotranspiration [mm.day−1], ***Rn*** is the net radiation at the crop surface [MJ.m2.day−1], ***G*** is the soil heat flux density [MJ.m−2.day−1], ***T*** is the mean daily air temperature at 2m height of soil [◦C], ***U2*** is the wind speed at 2m height of soil [m.s−1], ***es − ea*** is the saturation vapor pressure [kPa], Δ is the slope of the vapor pressure curve [kPa.◦C−1], and $\boldsymbol{\gamma}$ is the psychrometric constant [kPa.◦C−1].

The climatic data needed for (1) is obtained using CLIMWAT software provided by Food and Agriculture Organization (FAO). The crop water requirements (ETc) could be estimated according to (2) [12][21][22], and [23]:

$$ETc = ETo \times Kc \quad (2)$$

Where, ***ETc*** is the crop water requirements [mm day−1], ***ET0*** is the reference evapotranspiration [mm day−1], and ***Kc*** is the crop coefficient. Total Available Water (TAW) which is the total available water in the root zone is calculated according to (3) [22][17]:

$$TAW = 1000(\theta\mathrm{FC} - \theta\mathrm{WP}) \times \mathrm{Zr} \quad (3)$$

Where, ***TAW*** is the total available soil water in the root zone [mm], $\boldsymbol{\theta_{FC}}$ ***i***s the water content at field capacity [m3 m-3], $\boldsymbol{\theta_{WP}}$ is the water content at wilting point [m3 m-3], and ***Zr*** is the rooting depth [m]. Moreover, the Readily Available Water (RAW) which is a portion of TAW can be estimated using (4) [17]:

$$RAW = \rho\, TAW \quad (4)$$

Where, RAW is the readily available soil water in the root zone [mm], and ρ is the average fraction of TAW that can be depleted from the root zone before moisture stress (reduction in ET) occurs. Thus, the required net irrigation amount can be evaluated using (5) [17] and [24]:

$$dn = (\theta\mathrm{FC} - \theta\mathrm{Cm}) \times \mathrm{Zr} \quad (5)$$

Where, ***dn*** is current net irrigation amount [mm], and $\boldsymbol{\theta_{Cm}}$ is current soil moisture [m3 m-3]. Finally, irrigation application time (time for one irrigation) with drip irrigation is estimated by (6) [17]:

$$T = {}^{dn}\!/_{(q * N * E)} \quad (6)$$

Where, ***T*** is irrigation time [minute], $\boldsymbol{d_n}$ is net irrigation [mm], E is system efficiency %, q is nozzle discharge rate [l/s], and **N** is number of nozzles.

The flow chart of the proposed algorithm that controls the operation of the presented irrigation system is shown in *Fig. 2*.

Received 30/July/2021; Accepted 7/November/2021

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

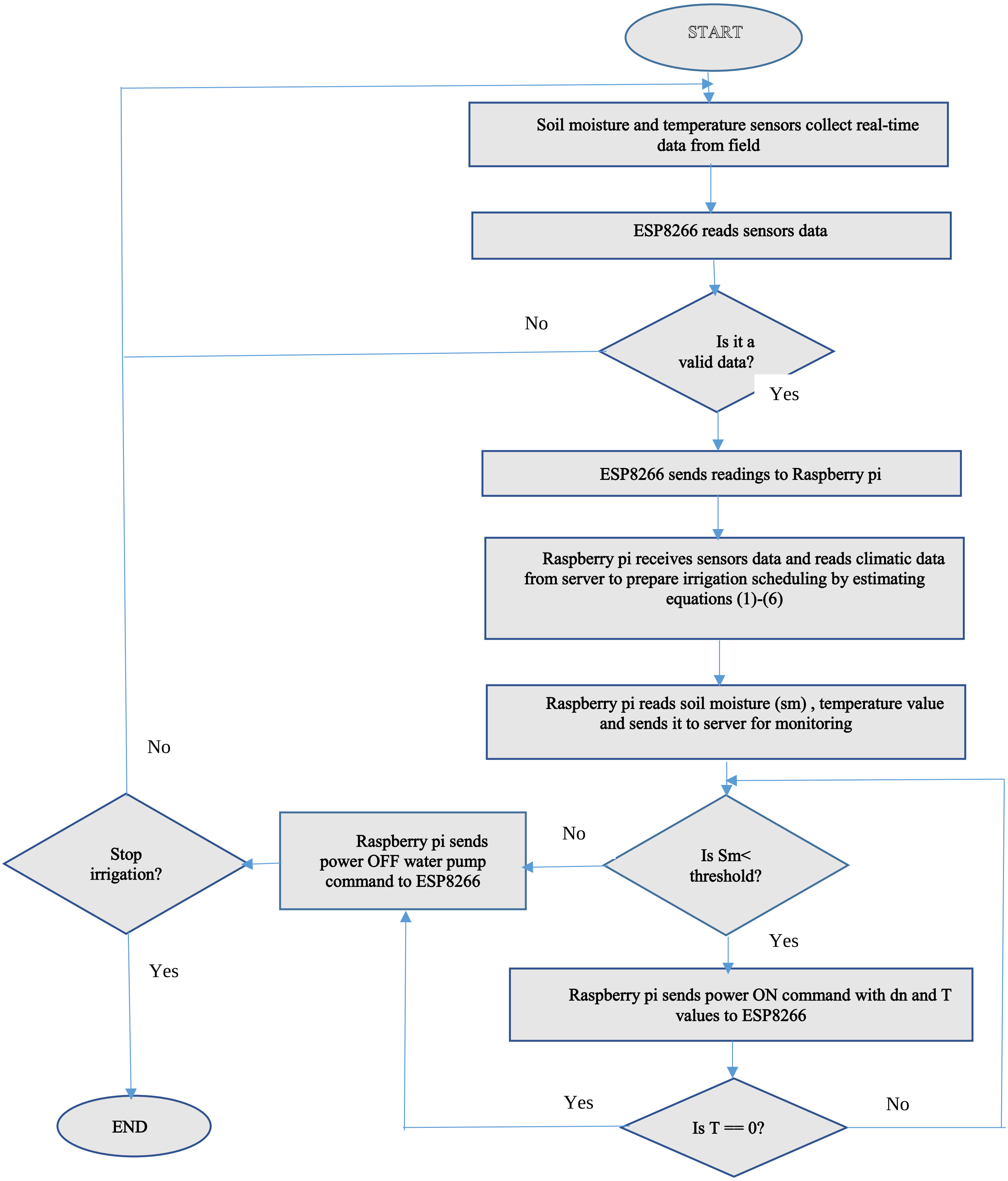


FIG. 2. FLOW CHART OF THE PROPOSED SYSTEM

Received 30/July/2021; Accepted 7/November/2021

ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

The work flow of the proposed algorithm can be explained in the following steps:

1. After powering up the system, ESP8266 microcontroller sends sensors reading to Raspberry pi with WiFi connection and MQTT protocol.
2. The Raspberry pi, as a base station in the system, is responsible of making irrigation decision and scheduling process. So, it receives sensors reading, sends it to server for monitoring using Node-RED tool.
3. The Raspberry pi makes the necessary calculations to estimate crop water requirements as well as irrigation run time using equations (1) – (6), while the microcontroller continues reading sensors at the same time.
4. The decision of starting irrigation or not depends on soil moisture sensor reading. If the reading below a preferred threshold value for a particular crop, then irrigation is started by sending power on command from Raspberry pi to ESP8266 to power on water pump for a specific time with a continuous check of soil moisture value to avoid over irrigation that harms plant growth and crop production. When sensor reading is above the determined threshold value or irrigation run time ends, Raspberry pi sends power off command to ESP8266 to stop water pump and starts the operation again.

In the implementation of the proposed irrigation system as a prototype, different hardware components are utilized including:

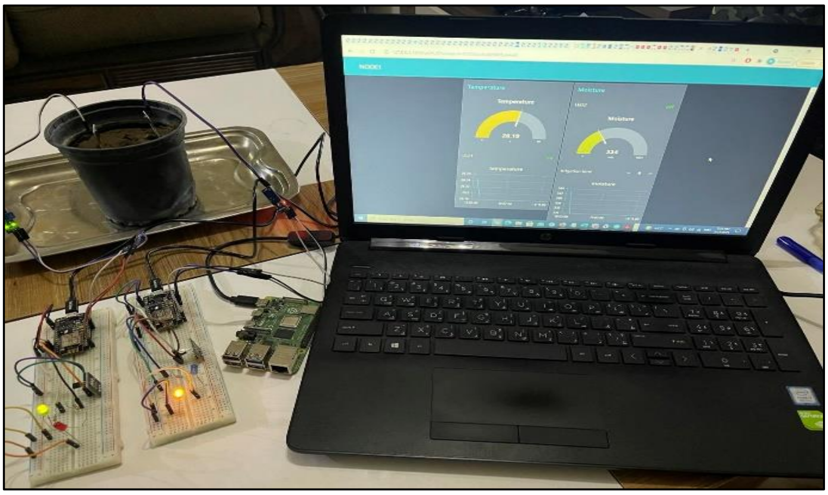
- Laptop
- Raspberry pi 4 Model B [25]
- ESP8266 microcontroller[26]
- FC-28 soil moisture sensor [27]
- DS18B20 temperature sensor [28]
- Light Emitting Diodes (LEDs)
- 15-ohm resistors.

These hardware components are shown in *Fig. 3*.

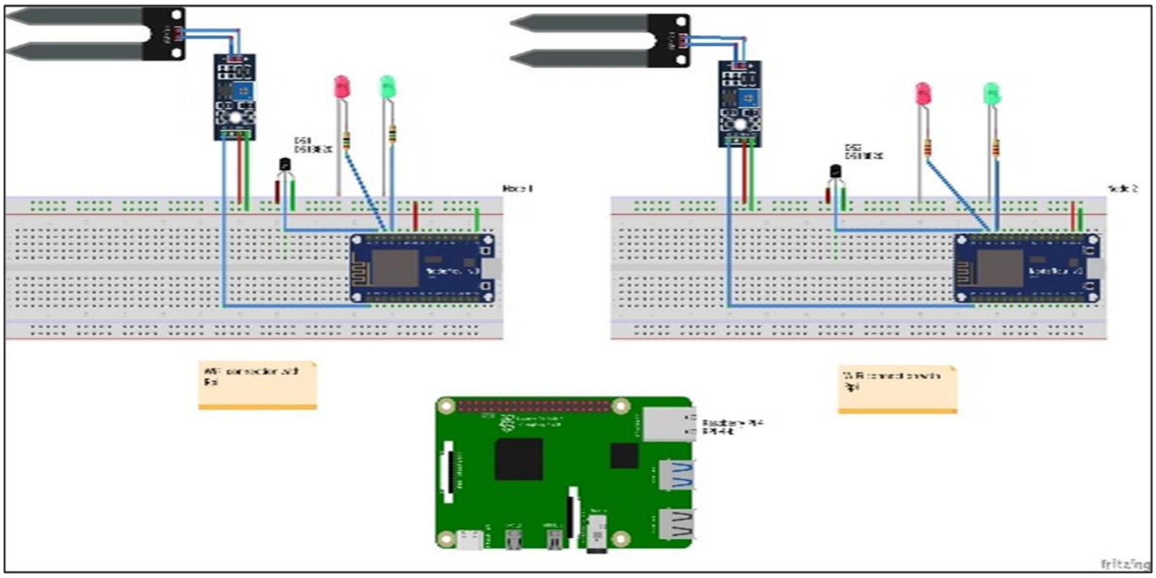
Received 30/July/2021; Accepted 7/November/2021

ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8



A. USED HARDWARE COMPONENTS



B. SYSTEM'S SCHEMATIC DIAGRAM

FIG. 3. HARDWARE COMPONENTS OF THE PROPOSED SYSTEM

III. RESULTS

In order to test the proposed irrigation system, the tomato crop is chosen as a case study. This is due to the high variance of irrigation factors of different plants. Tomato crop has 4 development stages with 90-150 total days: initial, crop development, mid-season, and late season. Every stage has different water requirements related to climate and crop coefficients change as illustrated in Table II [30]. Baghdad is selected as a planting region, the soil type in Baghdad is loam, which is preferred for tomato planting and the suitable soil moisture percentage is 70-80% [21], and [31]. Moreover, the best irrigation strategy for tomato is drip irrigation [29].

The ET_0 is estimated using (1) by CLIMWAT software and the results are shown in *Fig. 4.*

Received 30/July/2021; Accepted 7/November/2021
 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

TABLE II. MAIN CROP COEFFICIENTS USED FOR WATER MANAGEMENT[29]

	Development stage					Plant date
Crop characteristics	**initial**	**crop development**	**Mid-season**	**Late**	**Total**	
Stage length, days	**30**	**40**	**35**	**40**	**145**	**June**
Crop Coefficient,Kc	**0.6**	**0.6-1.15**	**1.15**	**0.7-0.9**		**June**
Root Depth, m	**0.25**	**0.25-1.0**	**1.0**	**1.0**		**June**
Depletion Coefficient, p:	**0.3**	**0.3-0.4**	**0.4**	**0.5**		**June**

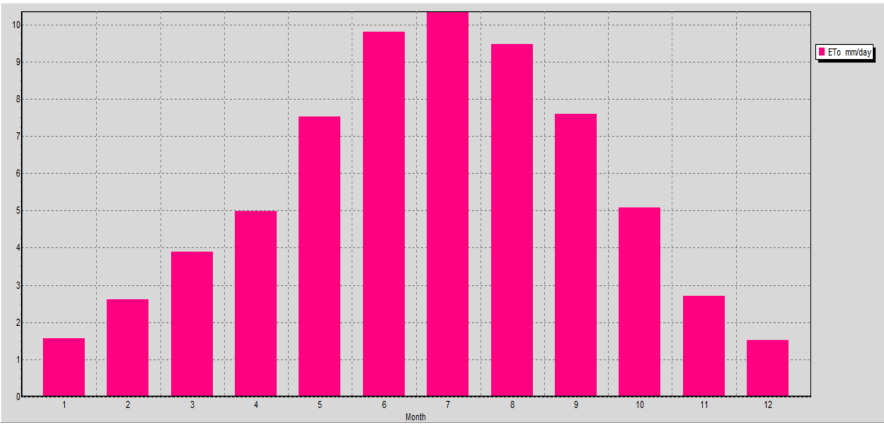


FIG. 4. ESTIMATED REFERENCE EVAPOTRANSPIRATION (ET0) VALUES IN BAGHDAD

Equations (2)-(5) are estimated using information in Table II and soil moisture curves shown in *Fig. 5* [30]. Irrigation application time in (6) is evaluated with properly suggested parameters: the efficiency range (E) of drip irrigation system is 70-90% [31], the nozzle discharge rate (q) for loam soil is 2.35 l/h which is equal to 0.6 gph [32], finaly, number of nozzles (N) is assumed to be 100 [33].

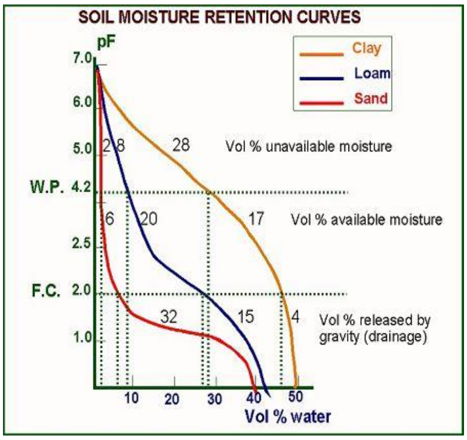


FIG. 5. SOIL MOISTURE RETENTION CURVE [32]

Received 30/July/2021; Accepted 7/November/2021
 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

The estimated crop water equirements for total development stages of tomato crop is illutrated in Table III, each stage is devided into decades.

TABLE III. TOMATO CROP WATER REQUIREMENTS

Stages	Kc	ET_0 mm/day	ET_C mm/day	d_n mm/day	T minutes
Initial	0.6	9.80	5.88	10.5	3.15
Initial	0.6	9.80	5.88	10.5	3.15
Initial	0.6	9.80	5.88	10.5	3.15
Crop development	0.63	10.35	6.52	16.8	5.04
Crop development	0.75	10.35	7.76	25.2	7.56
Crop development	0.9	10.35	9.315	44.8	13.45
Crop development	1.1	9.48	10.42	56	16.8
Mid-season	1.15	9.48	10.9	56	16.8
Mid-season	1.15	9.48	10.9	56	16.8
Mid-season	1.15	7.59	8.72	56	16.8
Mid-season	1.15	7.59	8.72	56	16.8
Late	1.15	7.59	8.72	70	21.02
Late	1.1	5.08	5.58	70	21.02
Late	0.9	5.08	4.57	70	21.02
Late	0.8	5.08	4.06	70	21.02

In this paper, LEDs are used to verify system's operation. When Raspberry pi sends power on command to ESP8266 microcontroller, LEDs are turned ON, for a particular time equals to irrigation run time determined by Raspberry pi. On the other hand, they are turnned OFF, with power off command.

Fig. 6 shows system's monitoring on server with Node-RED dashboards. A temperature guage and chart are used *in Fig. 6*.a to mointor temperature sensors reading, in addition to LED status, which is turnned ON when air temperature exceeds 25 C (according to Raspberry pi commands) and it is turnned OFF otherwise. *Fig. 6*.b-d show soil moisture monitoring windows, soil moisture sensor reading range is between 0 and 1024 with 0 represent 100% moisture and 1024 represent 0%. As mentioned earliar, the preferred soil moisture value for tomato crop is between 70-80%. After mapping the reading of the proposed sensor, the results show that a 70% moisture is equal to 430. Therefore in system design, when a Raspberry pi receives a moisture sensor reading above 430, a power ON command must be sent to ESP8266 to turn ON the specific water pump for starting irrigation process with a particular irrigation time. In the moisture monitoring window, a LED repersents a water pump status, it is turnned ON with timer when irrigation is required and it is turnned OFF otherwise. ESP8266 is programmed with C programming language and Arduino-IDE open source softwate, while Raspberry pi is configured with Python programming language.

Received 30/July/2021; Accepted 7/November/2021

ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

A B

C D

FIG. 6. SYSTEM MONITORING WITH NODE-RED TOOL

Fig. 7 shows the two nodes run program on serial port with Arduino IDE software.

```
Node1
.........................................
Temperature= 25.88c
Moisture= 1024
Percentage: 0%
(Received) Turn led1 ON
(Received) Turn led2 ON
.........................................
Node1
.........................................
Temperature= 25.88c
Moisture= 1024
Percentage: 0%
(Received) Turn led1 ON
(Received) Turn led2 ON
```

```
Node1
.........................................
Temperature= 26.00c
Moisture= 238
Percentage: 92%
(Received) Turn led1 ON
(Received) Turn led2 OFF
.........................................
Node1
.........................................
Temperature= 26.00c
Moisture= 238
Percentage: 92%
(Received) Turn led1 ON
(Received) Turn led2 OFF
```

A. NODE1 SERIAL PORT OUTPUT

Received 30/July/2021; Accepted 7/November/2021

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

```
Node2

Temperature= 26.69c
Moisture= 1024
(Received) Turn led1 ON
(Received) Turn led2 ON

Node2

Temperature= 26.69c
Moisture= 1024
(Received) Turn led1 ON
(Received) Turn led2 ON
```

```
Node2

Temperature= 26.50c
Moisture= 278
(Received) Turn led1 ON
(Received) Turn led2 OFF

Node2

Temperature= 26.56c
Moisture= 270
(Received) Turn led1 ON
(Received) Turn led2 OFF
```

.B. NODE2 SERIAL PORT OUTPUT

FIG. 7. NODES INPUT/OUTPUT READINGS

IV. CONCLUSION

In this work, a smart irrigation system based WSNs was proposed to enhance crops production by maintaining an efficient amount of water that is necessary to get a good quality crop. Sensors reading as well as weather information are used as inputs to the system to make an accurate irrigation decision. Raspberry pi is used as a base station that receives these inputs from ESP8266 ccontroller for making watering decisions, and sends commands to nodes. WiFi technology and MQTT protocol were used to exchange the data and control commands between nodes and Raspberry pi. Watering decisions were not only for starting or ending irrigation process, but also the required amount of water was determined according to crop evapotranspiration. Moreover, irrigation run time was estimated depending on drip irrigation strategy and its suggested tools. Tomato crop was chosen as a case study, due to its importance in agricultural production around the world. Finally, monitoring of system operation approved the effeciency of the proposed system in making appropriate irrigation decisions that lead to reduce water and power consumption in irrigation process with imrovement in quality and quantity of yields.

REFERENCES

[1] B. Khelifa, D. Amel, B. Amel, C. Mohamed, and B. Tarek, “Smart irrigation using internet of things,” in *2015 Fourth International Conference on Future Generation Communication Technology (FGCT)*, 2015, pp. 1–6, doi: 10.1109/FGCT.2015.7300252.

[2] H. K. Sodhi and U. Saxena, “Low-Cost Water and Energy Efficient Futuristic Irrigation System Using IBM Watson with Bayesian Analysis,” in *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, 2020, pp. 243–246, doi: 10.1109/ICRITO48877.2020.9197921.

[3] N. A. Hassan and A. K. Farhan, “Security Improve in ZigBee Protocol Based on RSA Public Algorithm in WSN,” *Eng. Technol. J.*, vol. 37, no. 3 B, pp. 67–73, 2019.

[4] Mahmood Farhan, “Hardware Implementation of Wireless Sensor Network Using Arduino and Zigbee Protocol,” vol. 34, no. January, pp. 816–829, 2018.

[5] N. M. Tiglao, M. Alipio, J. V. Balanay, E. Saldivar, and J. L. Tiston, “Agrinex: A low-cost wireless mesh-based smart irrigation system,” *Measurement*, vol. 161, p. 107874, 2020, doi: https://doi.org/10.1016/j.measurement.2020.107874.

[6] A. K. Podder *et al.*, “IoT based smart agrotech system for verification of Urban farming parameters,” *Microprocess. Microsyst.*, vol. 82, no. January, p. 104025, 2021, doi: 10.1016/j.micpro.2021.104025.

[7] R. Liao, S. Zhang, X. Zhang, M. Wang, H. Wu, and L. Zhangzhong, “Developement of smart irrigation systems based on real-time soil moisture data in a greenhouse: Proof of concept,” *Agric. Water Manag.*, vol. 245, no. November, p. 106632, 2021, doi: 10.1016/j.agwat.2020.106632.

[8] B. Et-taibi, M. R. Abid, I. Boumhidi, and D. Benhaddou, “Smart Agriculture as a Cyber Physical System: A Real-World Deployment,” in *2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS)*, 2020, pp. 1–7, doi: 10.1109/ICDS50568.2020.9268734.

Received 30/July/2021; Accepted 7/November/2021

DOI: https://doi.org/10.33103/uot.ijccce.22.1.8

[9] A. Vij, S. Vijendra, A. Jain, S. Bajaj, A. Bassi, and A. Sharma, "IoT and Machine Learning Approaches for Automation of Farm Irrigation System," *Procedia Comput. Sci.*, vol. 167, pp. 1250–1257, 2020, doi: https://doi.org/10.1016/j.procs.2020.03.440.

[10] S. Jaiswal and M. S. Ballal, "Fuzzy inference based irrigation controller for agricultural demand side management," *Comput. Electron. Agric.*, vol. 175, p. 105537, 2020, doi: https://doi.org/10.1016/j.compag.2020.105537.

[11] O. Debauche, S. A. S. Mahmoudi, M. Elmoulat, S. A. S. Mahmoudi, P. Manneback, and F. Lebeau, "Edge AI-IoT Pivot Irrigation, Plant Diseases, and Pests Identification," *Procedia Comput. Sci.*, vol. 177, no. November, pp. 40–48, 2020, doi: https://doi.org/10.1016/j.procs.2020.10.009.

[12] K. Karunanithy and B. Velusamy, "Energy efficient cluster and travelling salesman problem based data collection using WSNs for Intelligent water irrigation and fertigation," *Measurement*, vol. 161, p. 107835, 2020, doi: https://doi.org/10.1016/j.measurement.2020.107835.

[13] R. Dobrescu, D. Merezeanu, and S. Mocanu, "Context-aware control and monitoring system with IoT and cloud support," *Comput. Electron. Agric.*, vol. 160, pp. 91–99, 2019, doi: https://doi.org/10.1016/j.compag.2019.03.005.

[14] S. R. Barkunan, V. Bhanumathi, and J. Sethuram, "Smart sensor for automatic drip irrigation system for paddy cultivation," *Comput. Electr. Eng.*, vol. 73, pp. 180–193, 2019, doi: 10.1016/j.compeleceng.2018.11.013.

[15] J. Muangprathub, N. Boonnam, S. Kajornkasirat, N. Lekbangpong, A. Wanichsombat, and P. Nillaor, "IoT and agriculture data analysis for smart farm," *Comput. Electron. Agric.*, vol. 156, pp. 467–474, 2019, doi: https://doi.org/10.1016/j.compag.2018.12.011.

[16] A. Selmani *et al.*, "Multithreading design for an embedded irrigation system running on solar power," in *2018 4th International Conference on Optimization and Applications (ICOA)*, 2018, pp. 1–5, doi: 10.1109/ICOA.2018.8370519.

[17] M. Al-Janabi, Z. Faris, and A. Taqi, "Smar Farm Management System Based on Sensors Network," *Cienc. e Tec.*, vol. 33, no. 1, pp. 177–201, 2018, [Online]. Available: http://e-journal.uajy.ac.id/14649/1/JURNAL.pdf.

[18] A. Kumar, A. Surendra, H. Mohan, K. M. Valliappan, and N. Kirthika, "Internet of things based smart irrigation using regression algorithm," in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 2017, pp. 1652–1657, doi: 10.1109/ICICICT1.2017.8342819.

[19] A. G. Mohapatra and S. K. Lenka, "Neural Network Pattern Classification and Weather Dependent Fuzzy Logic Model for Irrigation Control in WSN Based Precision Agriculture," *Procedia Comput. Sci.*, vol. 78, pp. 499–506, 2016, doi: https://doi.org/10.1016/j.procs.2016.02.094.

[20] M. N. Abdullah, M. S. Croock, and A. K. Mousa, "Automation of smart irrigation system based on wireless sensor network," *Sens. Lett.*, vol. 13, no. 1, pp. 92–97, 2015, doi: 10.1166/sl.2015.3416.

[21] S. H. Ewaid, S. A. Abed, and N. Al-Ansari, "Crop water requirements and irrigation schedules for some major crops in southern Iraq," *Water (Switzerland)*, vol. 11, no. 4, 2019, doi: 10.3390/w11040756.

[22] K. Tilahun and L. John, "Evapotranspiration Estimation Using Soil Water Balance, Weather and Crop Data," *Evapotranspiration - Remote Sens. Model.*, no. May, 2012, doi: 10.5772/17489.

[23] M. S. Croock, M. N. Abdullah, and A. K. Mousa, "Optimal power consumption strategy for smart irrigation system using Lagrange multiplier," *Sens. Lett.*, vol. 13, no. 12, pp. 1044–1049, 2015, doi: 10.1166/sl.2015.3587.

[24] M. Allen, Richard G., PEREIRA, Luis S., RAES, Dirk and SMITH, "FAO Irrigation and Drainage Paper Crop by," *Irrig. Drain.*, vol. 300, no. 56, p. 300, 1998, [Online]. Available: http://www.kimberly.uidaho.edu/water/fao56/fao56.pdf.

[25] RaspberryPi, "Raspberry Pi 4," *Raspberry Pi Found.*, no. May, p. 1129409, 2020.

[26] "NodeMCU datasheet." https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet.

[27] "soil moisture sensor." https://www.elprocus.com/soil-moisture-sensor-working-and-applications/.

[28] Maxim Integrated, "DS18B20 Programmable Resolution 1-Wire Digital Thermometer," *System*, vol. 92, pp. 1–22, 2008.

[29] FAO, "Tomato crop database." http://www.fao.org/land-water/databases-and-software/crop-information/tomato/en/.

[30] "soil properties." http://www.fao.org/soils-portal/data-hub/soil-properties/physical-properties/en/.

[31] FAO, "drip irrigation." https://edis.ifas.ufl.edu/publication/ae457.

[32] M. Gil, L. Rodríguez-Sinobas, R. Sánchez, and L. Juana, "Procedures for Determining Maximum Emitter Discharge in Subsurface Drip Irrigation," *J. Irrig. Drain. Eng.*, vol. 137, no. 5, pp. 287–294, 2011, doi: 10.1061/(asce)ir.1943-4774.0000299.

[33] D. L. Bjorneberg and R. E. Sojka, "IRRIGATION - Methods," *Encycl. Soils Environ.*, vol. 4, pp. 273–280, 2004, doi: 10.1016/B0-12-348530-4/00276-9.

Received 30/July/2021; Accepted 7/November/2021