

# Smart irrigation system with an Android-based remote logging and control

Olugbenga Kayode Ogidan  
Department of Electrical and Electronics Engineering  
Elizade University  
Ilara Mokim, Ondo State, Nigeria  
olugbengaogidan@gmail.com

Kennedy Richmond Afia  
Department of Electrical and Electronics Engineering  
Elizade University  
Ilara Mokin, Ondo State, Nigeria  
kennedyafia1@gmail.com

**Abstract**— In order to eliminate the drudgery associated with irrigation practice, a smart irrigation system with an Android-based mobile application and control system is developed and tested under the local and cloud server architectures. The system uses four sensors (moisture content, temperature, humidity and ultrasonic) to acquire real-time data from the irrigation site which are fed to the central controller. Wi-Fi device is used to provide Internet connectivity to a local and cloud server. A mobile application is developed to display the four sensor measurements on a user-friendly graphically interface in real-time as received from the server. The system automatically starts/stop irrigation pumps based on measurements received from the soil moisture content sensor after attaining certain threshold values. Its remote logging capability makes it possible for users to view on their mobile phone the status of their farmland such as atmospheric temperature, humidity, level of water in the reservoir and soil moisture content from any location, provided Internet access is available. It also gives the user the flexibility of starting a pump on the farmland remotely using his mobile device. Experimental results reveal that the local server is more stable than the cloud server architecture.

**Keywords**—smart irrigation, Internet of Things, Arduino, Blynk, remote logging.

## I. INTRODUCTION

In many parts of Africa, irrigation is carried out using traditional procedures such as the use of watering cans and when electrical pumps are used for the irrigation, they are manually operated. Many times people saddled with the responsibility of operating the irrigation equipment reside in locations that are several kilometers away with very poor access roads. These conditions make irrigation practices to be inefficient and ineffective.

There is therefore the need to develop effective and efficient methods that optimises water usage during irrigation as well as reducing the drudgery associated with the operation of irrigation equipment. Several attempts have been made to automate irrigation systems.

In an attempt to control the pumps remotely, some of the available methods acquire soil and environmental parameters using SMS sent to their mobile phone or Google-mail while they issue control command to the irrigation pumps or valves using SMS [1]. SMS are also sent from time to time to tell the user the state of the system. This type of communication is not done in real-time and would increase cost of operation due to the cost of SMS. In [2], the focus on the water management was accomplished by using soil moisture content sensors to detect the wet zones and ensuring these zones are by-passed during irrigation. Authors have developed irrigation systems that

uses sensors to acquire environmental conditions of which are used to operate water pumps or valves without a remote control or logging of measured values in a mobile device [3, 4]. To monitor these devices, it is usually done via a Liquid Crystal Display which means the operator will have to be physically present on the site. The approach described here enables the user to control the pumps in the irrigation site from a mobile phone using an Android-based mobile application developed for this purpose. The mobile device connects to a cloud server through which instructions are issued to the controller unit. The controller controls irrigation pumps through a relays. The system also makes it possible for the user to view in real-time parameters such as the level of water in the reservoir, soil moisture content, temperature and humidity of the irrigation site. While the other methods use Programmable Logic Controller (PLC) [5, 6], 16F877 PIC micro-controller [5] and Rasbery Pi [7, 8] as controller, this approach uses the Atmeg 382 on the Arduino Uno as controllers in a master-slave configuration.

## II. METHODOLOGY

The system development can be divided into three namely: the data acquisition stage, control stage while the third is remote data logging and control that is mainly the application of the Internet of Things (IoT). The block diagram is illustrated in Fig. 1.

### A. Sensor and Data acquisition

The sensor used include the Digital Humidity and Temperature (DHT) sensor that acquires the humidity measurements and temperature measurements of the irrigation site, the moisture content sensor that acquires the soil moisture content and the ultrasonic sensor. The ultrasonic sensor is attached to the lid of the reservoir and measures water level as distance between the lid and the water surface in the reservoir. The data acquired from the irrigation site by these sensors are sent as information to the central controller as illustrated in Fig. 2.

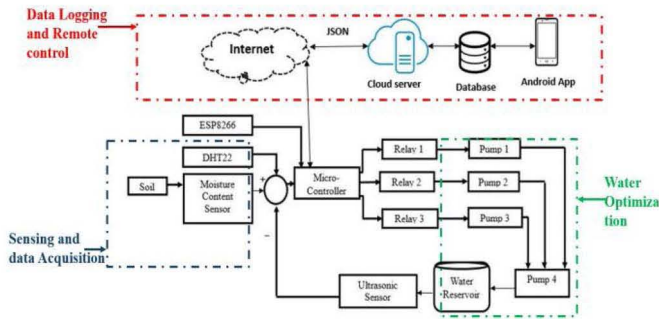


Fig. 1: Block diagram of the smart irrigation

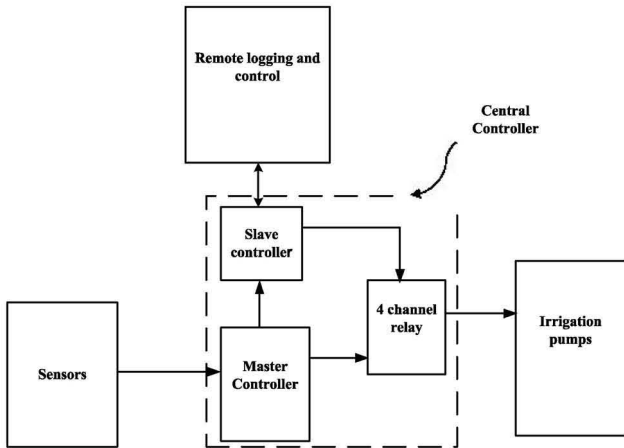


Fig. 2: Block diagram of the central controller

### B. Control (processing)

In this stage, the central controller does the processing work and comprises of the master controller, the slave controller and the 4 channel relay. The master and slave controllers in this work are Arduino Uno microcontrollers arranged in a master-slave configuration. The master controller receives sensor data from the sensors as which include the humidity and temperature of the irrigation site, the soil moisture content as well as the water level of the reservoir. The sensed data received are used in two ways which are the automatic and the manual modes. In the automatic mode, they are used by the master controller to derive the prioritization scheme for automatic control of the irrigation pumps. The master control by which the irrigation pumps operate. This is based on the level of water in the reservoir. For instance, pumps 1, 2 and 3 (feedforward pumps) are used to pump water via the sprinklers to the irrigation site comprising of farm A, farm B and farm C respectively. Pump 4 (feedback pump) is used to recycle excess water from farm A, farm B and farm C which represents the irrigation site back to the reservoir. This arrangement of the pumps in relation to irrigation site is illustrated in Fig. 3. The manual mode makes use of the slave controller. The sensor data being passed from sensors to the master controller are also shared by the slave controller. From the slave controller, they are sent to the cloud server for storage in the cloud data base (as shown in Fig. 1). In the manual mode, prioritization command to the irrigation pumps is by the user issuing prioritization commands as inputs on his mobile device. These command are sent to the slave controller which controls the irrigation

pumps via the 4 channel relay. The circuit diagram and the developed laboratory scale smart irrigation system are shown in Fig. 4 and 5 respectively.

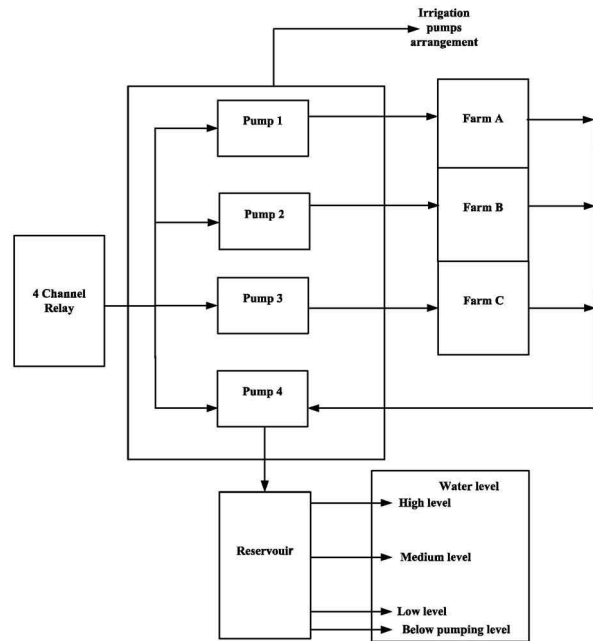


Fig. 3: Arrangement of irrigation pumps

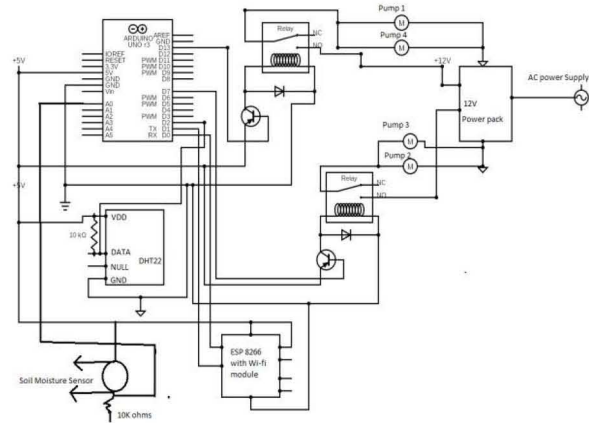


Fig. 4: Circuit diagram of the smart irrigation system



Fig. 5: Developed laboratory-scale smart irrigation system

### C. The Internet of Things (IoT)

This is the section concerned with remote logging and control of the smart irrigation system as shown in Fig. 1. A Wi-Fi enabled device is connected to the slave controller to

grant Internet access. This ensures that sensor information sent to slave controller reaches the cloud server and by extension, MySQL online data-base. From the online data-base, the user is able to have access to irrigation site information (temperature, humidity, moisture content and water level) in real-time from his mobile device via a user-friendly mobile App developed for this purpose as shown in Fig. 7. The users can manually start or stop irrigation pumps remotely by clicking on icons of the pumps they wish to start or stop. This way, the user can select the number of pumps to be operated on irrigation site and the order of their operation, thus prioritizing their operations. In order to actualize the IoT section, the Blynk, an open source platform which integrates (IoT) communication between the server and the hardware is used. A graphic interface is built with Blynk to connect Arduino using Android and iOS application. [9]. In this manner, an interfaces for controlling and monitoring smart irrigation hardware projects from the iOS and Android devices is achieved. Fig. 6a and 6b illustrate the Blynk cloud architectures and Blynk local architecture used in this study while Fig. 7 is the Android App interface for smart irrigation.

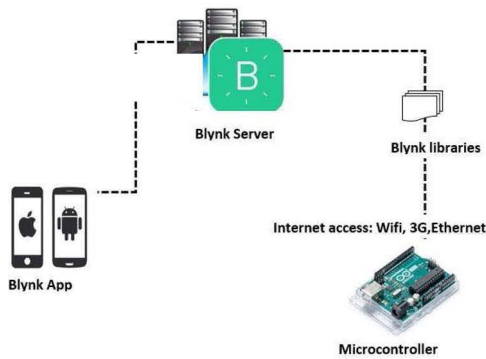


Fig. 6a: Blynk cloud server architecture [10]

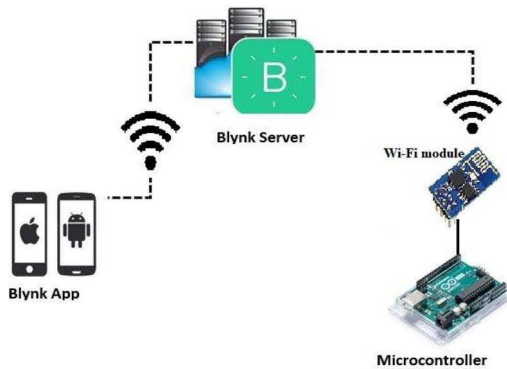


Fig. 6b: Blynk local server architecture [10]

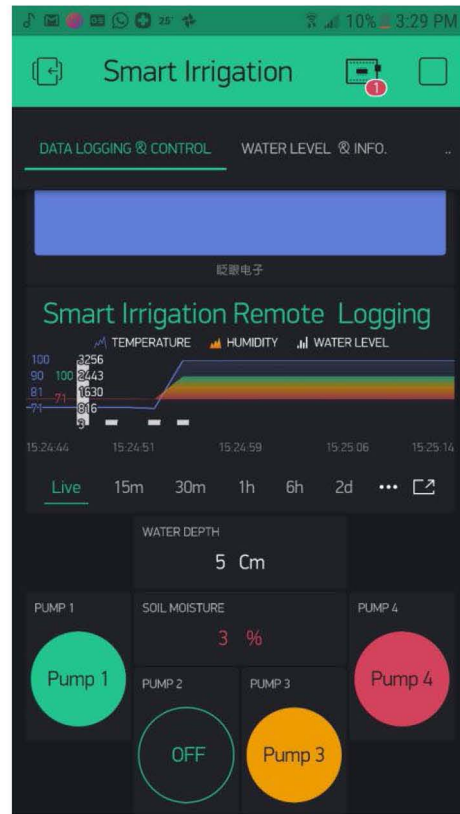


Fig. 7: Developed Android App interface for smart irrigation

### III. TESTING AND RESULT

The remote logging and control of the developed system was done by varying the distance between the Android Apps and the laboratory scale smart irrigation system while the Android App was being used to log sensor values and to control the pumps. This test was done using both the cloud server and the local server architectures. The stability of the system for each case are recorded in tables 1 and 2.

TABLE 1. REMOTE CONTROL OF PUMPS WITH VARYING DISTANCE BETWEEN THE ANDROID MOBILE APP AND THE DEVELOPED SMART IRRIGATION SERVER USING BLYNK CLOUD SERVER

S/ N	Parameter	Varying Wi-Fi ranges (meters)	Remote control of irrigation pumps	Logging of sensor measurements	Network connectivity Status
1	Distance from model	< 10 (within the laboratory)	possible	In real-time	Stable
2	Distance from model	$\geq 10$ & $< 20$ (within the Engineering building)	possible	In real-time	Stable
3	Distance from model	$\geq 20$ & $\leq 40$ (outside the Engineering building)	possible	In real-time	Fluctuating
4	Distance from model	$> 40$	possible	Not in real-time	Weak and Fluctuating

TABLE 2. REMOTE CONTROL OF PUMPS WITH VARYING DISTANCE BETWEEN THE ANDROID AND THE DEVELOPED SMART IRRIGATION SYSTEM USING BLYNK LOCAL SERVER

S/N	Parameter	Varying Wi-Fi ranges (meters)	Remote control of irrigation pumps	Logging of sensor measurements	Connectivity Status
1	Distance from model	< 10 (within the laboratory)	possible	In real-time	Excellent and Stable
2	Distance from model	$\geq 10$ & $< 20$ (within the Engineering Faculty building)	possible	In real-time	Excellent and Stable
3	Distance from model	$\geq 20$ & $\leq 40$ (within the university)	possible	In real-time	Excellent and Stable
4	Distance from model	> 40	possible	In real-time	Excellent and Stable

#### IV. DISCUSSION OF RESULTS

It could be seen from Fig. 8 that the system is able to present in graphical form the parameters of the irrigation site such as temperature, humidity, water level in reservoir and soil moisture content in real-time on an Android mobile App. The developed system also made possible the remote control of irrigation pumps using mobile App as shown from tables 1 and 2. However, the stability of real-time logging and remote control of irrigation pumps reduces as the Android device is moved away from the laboratory-scale smart irrigation system due to attenuation of communication signal. For example, the real-time logging of sensor values and remote pump control in the cloud server architecture produced stable result until a distance of  $\geq 10$  and  $< 20$  meters from the lab-scale irrigation. When the distance is greater than 20 meters, it begins to fluctuate as shown in table 1. When the distance is  $> 40$  meters, the data could no longer be logged in real-time but remote control was still possible. In the case of local server architecture shown in table 2, sensor measurements were logged in real-time and the remote pump control was accomplished with stable connectivity for all the cases tested.

#### V. CONCLUSION

In this work, the control (Start/Stop) of an irrigation system is achieved using an Android-based mobile

application. The system presents environmental parameters acquired from irrigation site such as temperature, humidity, water level in reservoir and soil moisture content in graphical form on an Android mobile App. This is for the purpose of remote control of irrigation pumps in an attempt to remove the drudgery associated with traditional irrigation system. The test carried out using laboratory-scale smart irrigation system with the Blynk cloud and Blynk local server architectures reveal a more stable system performance of the irrigation system with the local server. Future work would include boosting the communication signal for enhanced performance and implementing the concept in a farm setting. The work will go a long way to improve the efficiency in the practice of irrigation to ensure food security.

#### REFERENCES

- [1] G. Prasanna, S. Parvatham, and S. Krishna, "Web Based Automatic Irrigation System using Raspberry Pi Processor On Embedded Linux International Journal & Magazine of Engineering, Technology, Management and Research, vol 3, no 10, pp. 1036-1039, 2016.
- [2] A. Rajpal, S. Jain, N. Khare, and A. K. Shukla, "Microcontroller-Based Automatic Irrigation System with Moisture Sensors," Proceedings of the International Conference on Science and Engineering, pp. 94-96, 2011.
- [3] A. Tyagi, N. Gupta, J. P. Navani, M.R. Tiwari, and M.A. Gupta, "Smart Irrigation System, International Journal for Innovative Research in Science & Technology, vol 3, no 10, pp. 9-12, 2017.
- [4] S. R. Kumbhar, and A. P. Ghatule, "Microcontroller Based Controlled Irrigation System For Pina Proceedings of the International Multi Conference of Engineers and Computer Scientists, vol. 2, 2013.
- [5] R. Subalakshmi, A. A. Amal, S. Arthireena, "GSM Based Automated Irrigation Using Snn International Journal of Trend in Research and Development, ISSN: 2394-9333, 4-6, 2016
- [6] D. K. Roy, and M. H. Ansari, "Smart Irrigation Control International Journal of Environmental Research and Development, vol. 4, no 4, pp. 371-374, 2014.
- [7] V. L. Akubattin, A. P. Bansode, T. Ambre, A. Kachroo, and P. SaiPrasad, "Smart Irrigation System," In International Journal of Scientific Research in Science and Technology, vol. 2, no 5, pp. 343-345, 2016.
- [8] K. R. Kakade, A. R. Pisal, A. V. Chavanss, and S. B. Khedkar, "Smart Irrigation and Crop Suggestion Using Raspberry-Pi, International Journal of Scientific Research in Science and Technology, vol. 4, no 4, pp. 235-241, 2017.
- [9] M. M. Kamal, N.A.Z.M. Noar, and A.M. Sabri, "Development of Detection and Flood Monitoring via Blynk Apps, Indonesian Journal of Electrical Engineering and Computer Science, vol. 10, no 1, pp. 361-370, 2018.