

LoRa based Smart Irrigation System

Ravi Kishore Kodali
Dept. of E.C.E.
National Institute of Technology,
Warangal
WARANGAL, INDIA
ravikkodali@gmail.com

Mohan Sai Kuthada
Dept. of E.C.E.
National Institute of Technology,
Andhra Pradesh
TADEPALLIGUDEM, INDIA

Yatish Krishna Yogi Borra
Dept. of E.C.E.
National Institute of Technology,
Andhra Pradesh
TADEPALLIGUDEM, INDIA

Abstract—World's total population has come to past 7.2 billion and the population rate is expanding step by step, in following 25 to 30 years there will be a major issue of food scarcity, so the improvement of farming is vital. Today, the farmers are experiencing the absence of downpours and shortage of water. The fundamental goal of this paper is to give a smart irrigation for saving time, cash and power of the farmer. The prototype uses Two ESP32TTGO LoRa boards in which one is placed in the field connected to sensors like temperature, soil moisture and water flow and the other is placed within the range of 5KM which is connected to the internet using the WiFi protocol of ESP 32 and thus the data is published onto the IBM Bluemix. The Data is analyzed to trigger the water pump automatically to pump the water into the fields. The trigger can also be given manually.

Keywords— *IoT, MQTT, Smart Irrigation, LoRa, ESP-32, Soil Moisture*

I. INTRODUCTION

This paper proposes a smart irrigation system based on ESP32 TTGO LoRa. The system monitors different environmental factors like temperature, moisture, and the volume of water required by the crops, utilizing sensors i.e temperature, soil moisture, and water flow. The information is gathered and given to the ESP 32 TTGO placed in the farm which is connected to another ESP32 TTGO placed within the range of 5KM (range can be improved if an antenna with high gain is used) through LoRa protocol. This module is connected to IBM cloud through the internet using the WiFi stack present on ESP32 which demonstrates the continuous qualities. This enables the farmer or IBM Bluemix to control irrigation pumps and sprinklers from distant places and to meet the standard qualities which would assist the farmer with yielding better quantity and quality of the crop Agriculture is a field where water is required in more amount. Wastage of water is a real issue in agriculture. During the cultivation more amount of water is given to the fields. There are numerous methods to spare or to control wastage of water in agriculture [1].

In the world, the majority of irrigation systems work manually. These outdated techniques are supplanted with semiautomatic and automatic procedures [8]. The accessible customary methods resemble drip irrigation, sprinkler system, ditch irrigation, terraced irrigation [2]. The worldwide irrigation situation is classified by expanded interest for higher agricultural efficiency, poor execution and diminished accessibility of water for agriculture [3]. These issues can be rectified if we make use of smart irrigation systems.

Through Internet of Things, agriculture products will have a fresh growth state, better storage preservation, and best quality. With the advancement of Internet of things, its innovation has been broadly connected to all the aspects of agricultural.

II. SMART IRRIGATION

"Smart irrigation is one such innovation which has pulled the attention of many researchers and is advancing " [4]. Smart irrigation is where water usage is limited and is more feasible economically, socially and conventionally. The thought and advancement of smart irrigation system is essentially centered around decreasing human efforts and lessens water usage and electricity utilization. The increasing demand for food because of population expansion put farmers to confront many issues with respect to the amount and nature of crops, which in fact brought another challenge on researchers to create and approach a keen irrigation system that would give farmers a smart instrument which helps them in yielding a quality crop. Even though smart irrigation has developed, yet so far no solution is acquired to measure the precise flow of water. Hence our prime move all through the project work has been to outline an irrigation system which furnishes all the above highlights alongside ordinary highlights accessible in a keen water system, for example, estimating dampness profile of the field, the goal to keep crops from waterlogging issues, temperature detecting because crops are sensitive to temperature too. The parameters are calculated using various sensors..

III. PREVIOUS WORK

Smart irrigation methods were proposed using various devices: ESP8266, Arduino [1], ZigBee [2], GSM, GPRS [5], Smartphone [7]. ESP8266 was used like an extension to Arduino to send the data over the WiFi protocol without connecting it to the internet. GPRS and GSM indulge more cost from the service provider. ZigBee is an outdated protocol. The LoRa protocol uses an unlicensed band which reduces the cost to transmit and receive the data. An array of TTGO boards can be connected to ESP32 TTGO which is connected to the Internet thus making the system to be controlled from any part of the world.

IV. ADVANTAGES

- Conservation of water and energy
- Reduced irrigation cost and improved quality of irrigation.
- Less manpower

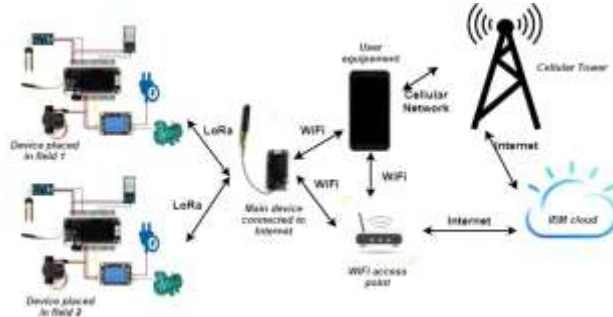


Fig. 1 SYSTEM SETUP

- Utilizes unlicensed band i.e. no service provider cost.
- Whole system is connected to Cloud through one device.

V. DESIGN AND IMPLEMENTAATION SETUP

In Smart irrigation system, we utilize different types of sensors to update the farmer about the field. Sensors utilized are water flow sensor which can measure the amount of water utilized, a soil moisture sensor which can compute the moisture of the field keeping the crops from waterlogging issues and a temperature sensor to check the temperature since crops are temperature sensitive as well. Thus system aware the farmer and IBM Watson, so that sprinklers or water pump is turned on to save the crop.

- The sensors are connected to the TTGO board and are placed in different fields. Each board is given a name (field 1, field 2.....)
- Sensor data along with the name of the board is sent to main TTGO board which is connected to the Cloud.
- The published sensor information on the cloud is analyzed using IBM Watson or by the user, by comparing the data with known optimized data available.
- Necessary triggers are sent back from the cloud to the main device along with device name and action to be carried out whether to turn 'on' or 'off' the water pump.
- The main node sends the information to the respective nodes and thus automates the water pump using the relay.

A. COMPONENTS REQUIRED

1. ESP 32 TTGO LoRa.
2. Soil Moisture Sensor.
3. Temperature Sensor.
4. Water flow sensor.
5. Smart Phone.
6. Relay
7. WiFi access point.

B. ESP-32 TTGO LoRa

The device has an SX1276 chip based on the ESP32 core. The device has an onboard OLED display, LoRa, WiFi and Bluetooth stacks. The device is operated at 865MHz with high transmission range and it is more reliable. It has a sensitivity over 148dBm and an output power of +20dBm. The ESP32 TTGO LoRa board is programmed using Arduino IDE. The required libraries can be downloaded from Github [12]. To connect the board to the Windows, Linux or OSX systems, CP2102 USB to UART bridge should be

TABLE I ESP32 LORA SPECIFICATIONS

Parameter	Specifications
Operating Voltage	+ 3.3 V to +7 V
Operating Temperature	-40 °C to 90 °C
WiFi	802.11 b/g/n 2.4 GHz
Bluetooth	V 4.2 (BLE)
SRAM/ ROM	520 KB/ 448 KB
Frequency (LoRa)	865 MHz
Receiver sensitivity	-98 dBm

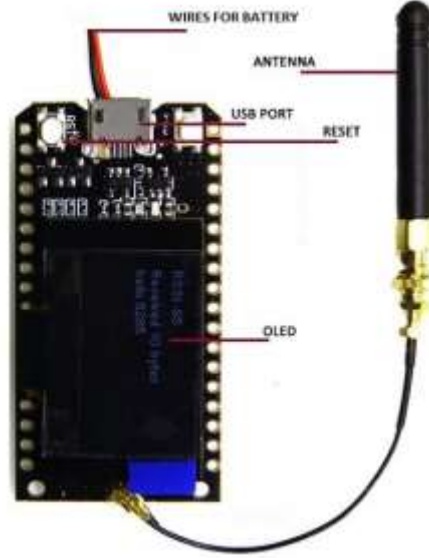


Fig. 2 ESP32 TTGO LoRa Module

installed on the system which is provided by the Silicon Labs [11].

C. LoRa Technology

Long Range (LoRa) brought by Semtech in 2012 [13], utilizes Unlicensed sub-gigahertz RF bands like 470 MHz (China), 865MHz(India), 868 MHz (Europe), and 915 MHz (North America). LoRa provides long-range transmissions (up to 10 KM) with low power consumption. This communication is introduced in two parts — LoRa, the physical layer and Long Range Wide Area Network (LoRaWAN), the upper layers. LoRa uses a star architecture, a central node to which all other nodes are connected, and a gateway serves as a bridge for communication between the central node and end devices in the backend. Gateways are network servers that are connected using standard IP whereas end devices use single hop communication to connect to one or many gateways. All the end-points have bidirectional communication. This helps for low power consumption and long-range communication.

LoRa is based on "Chirp spread spectrum modulation," organizations claim that it maintains low power qualities and essentially increases the ease of communication [3]. Spread spectrum modulation was utilized as a part of military and space organizations for a considerable time, LoRa is the first low power invention for commercial utilization. Messages between gateway and end-devices

spread out in different directions and data rates. The data rate is selected as an exchange between transmission range and message length. Spread spectrum creates a 'virtual' channels which increase the capacity of the gateway. LoRaWAN data rates extend from 0.3 kbps to 50 kbps.

ESP32 TTGO LoRa is driven by an SX1276 chip developed by Semtech which gives the necessary hardware for ESP32 to get furnished with LoRa protocol. It has four distinct modes. Spreading factor, preamble length, sync word, bandwidth, equality FEC (forward error correction), and so on., these parameters should be initialized and sent to registers of SX1276 [19]. These parameters are required to setup the LoRa for transmission and receiving. The above parameters should be set, before changing the state of the device from sleep mode to required mode.

Security:

- Network layer- Unique Network key (EUI64)
- End to end security on Application layer- Unique Application key (EUI64)
- Device specific key (EUI128)

VI. IBM BLUEMIX IoT PLATFORM

"IBM Bluemix is a cloud platform as a service (PaaS), it allows customers to develop, run and manage applications without the complexity of building and maintaining the whole infrastructure that is normally associated with application development" [15] First, create an IBM id using our email, followed by creating Internet of Things Service [9]. Later, we have to launch the Bluemix from the instance. The Bluemix dashboard tab appears. In the dashboard, go to devices and add the device by assigning the device type, device id and authentication token. These parameters are later placed in the code. To monitor the data, go to cards and create three new cards to monitor Temperature, Soil moisture and Water content. To analyze the data and to produce the necessary trigger an Application Programming Interface (API) has to be created.



Fig. 3 Creating IBM Watson account



Fig. 4 Creating IoT Service



Fig. 5 Adding a Device

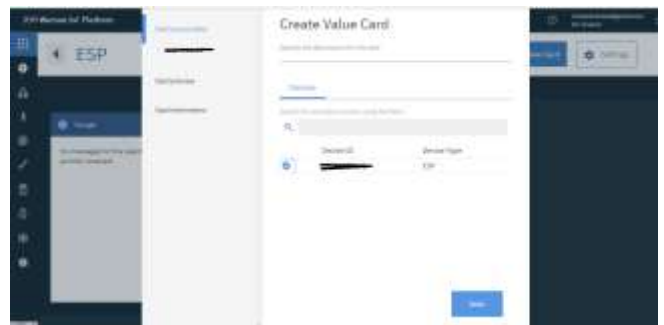


Fig. 6 Creating a Card



Fig. 7 Creating App



Fig. 8 Application Dashbaord

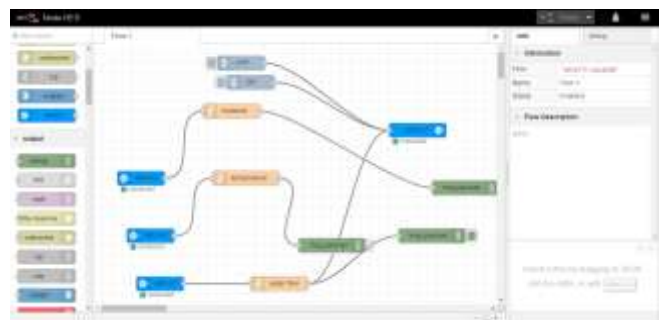


Fig. 9 A Flow in Node-RED

S.No.	Crop	Moisture Level(in %)	Water Requirement(in mm)	Optimum Temperature(in F)
1	Rice	20%	900-2500	77
2	Wheat	12%	450-850	72
3	Maize	18-24%	500-800	77
4	Soybean	13%	450-700	57
5	Sunflower	9.5%	350-500	80
6	Corn	15.5%	500-750	95
7	Tomato	10%	600-800	85
8	Potato	12.3%	500-700	60
9	Onion	11%	350-550	75
10	Bean	10%	300-500	75

Fig. 10 Optimized Values for various crops

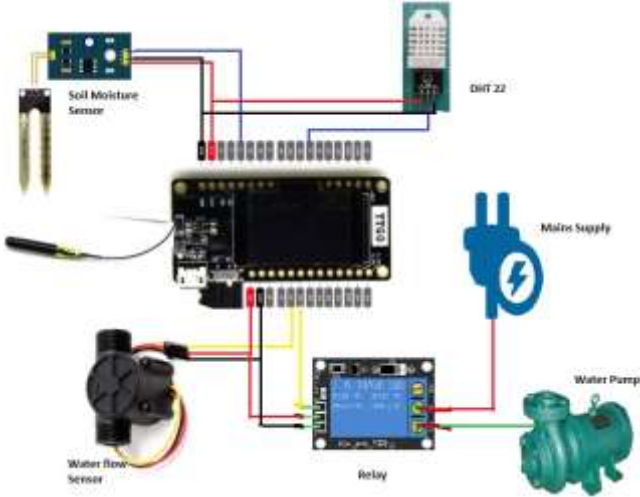


Fig. 11 System Setup

There are lots of programming services that support IBM but in this project we used NodeRED. Select the Node-RED service and create the IBM cloud foundry App. After creating the App, the created App appears in the main dashboard and now, make sure that the status of the App is "Running". Open the App, in the Overview tab, the URL of the App appears. Open the URL to create the API. The URL leads to the Node-Red flow editor. Register and login to the Node-RED. Nodes are the building blocks of the App. Place the Input and output node in the flow that produces the required trigger for our application. After creating the flow click on deploy to run the API. The user can control the device manually by injecting the ON and OFF nodes in the flow editor and can open the flow editor in any device by using the URL of the App.

For this prototype, we have used the optimum values of Rice crop to create the flows. The optimized values of temperature, soil moisture, and water for various crops are given in Fig.10 [4].

VII. HARDWARE

The Vcc, ground, A0 pins of Soil moisture sensor are connected to 3.3V, GPIO and gnd of ESP32 respectively. The Vcc, ground, Data pins of DHT are connected to 3.3v, gnd, and GPIO of ESP32 respectively. The Vcc, ground, output pins of Water flow sensor are connected to 5V, gnd, and GPIO of ESP32 respectively. The Vcc, ground, signal pins of Relay are connected to 5V, gnd, and GPIO of ESP32 respectively. C pin is connected to mains and NO is connected to Water Pump. The LoRa is connected to the IPEX interface present on the ESP-32 module. The module can be powered up either by a Lithium battery or by a USB power bank. Fig. 11 shows the system setup.

VIII. SOFTWARE DESCRIPTION

The ESP32 TTGO boards can be programmed using Arduino IDE which can be installed Windows, Linux or OSX Operating Systems. The module should be included in the boards of Arduino IDE through board manager. Install the libraries of DHT, Soil Moisture, Water Flow sensor and LoRa from the Github [12] through library manager of Arduino IDE. Select the ESP32 dev module as the board and select the COM port to which the module is connected. Add the required libraries to the original source code, replace the WiFi SSID, and PASSWORD. The ORG, DEVICE TYPE, DEVICE ID, and TOKEN should be replaced with the credentials of Internet of Things services in the IBM Bluemix.



Fig. 12 Data published – IBM BlueMix

IX. RESULTS AND CONCLUSIONS



Fig. 13 Message sent from Node-RED to Motor event

After compiling and uploading the code to the modules. The device placed in field sends the data to the main device via LoRa which publishes the data to the IBM Bluemix. The Node-Red flow analyses the data and sends the message to the motor event in the IBM Bluemix. The main device subscribes to the message and sends it to the device placed in the field via LoRa whether to turn "ON" the motor or to turn it "OFF". The user can manually control the operation of the motor by injecting the ON and OFF nodes in the Node-Red flow which can be opened on any device using the URL of the App. The system helps in providing a better yield of the crop that results in more profit to the user and an increase in the food production.

The LoRa-based communication has been formulated considering the ease of use, low maintenance and cost. The device is completely automatic and IBM cloud makes it more secure. With the help of IBM Bluemix, the user can read the data from the sensors and control the device manually from anywhere in the world in a blink of an eye which saves the time of the user.

REFERENCES

- [1] C. M. Devika, K. Bose and S. Vijayalekshmy, "Automatic plant irrigation system using Arduino," 2017 IEEE International Conference on Circuits and Systems (ICCS), Thiruvananthapuram, 2017, pp. 384-387.
- [2] Gunturi, Venkata Naga Rohit. "Micro controller based automatic plant irrigation system." International Journal of Advancements in Research & Technology 2.4 (2013): 194-198.
- [3] Ramon Sanchez-Iborra; Jesus Sanchez-Gomez; Juan Ballesta-Viñas; Maria-Dolores Cano; Antonio F. Skarmeta (2018). "Performance Evaluation of LoRa Considering Scenario Conditions". Sensors.
- [4] P. Singh and S. Saikia, "Arduino-based smart irrigation using water flow sensor, soil moisture sensor, temperature sensor and ESP8266 WiFi module," 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Agra, 2016, pp. 1-4
- [5] J. Gutierrez, J. F. Villa-Medina, A. Nieto-Garibay and M. A. Portagándara, "Automated Irrigation System Using a Wireless Sensor Network and GPRS Module," in IEEE Transactions on Instrumentation and Measurement, vol. 63, no. 1, pp. 166-176, Jan. 2014.
- [6] Z. Rasin, H. Hamzah and M. S. Mohd Aras, "Application and evaluation of high power Zigbee based wireless sensor network in water irrigation control monitoring system," 2009 IEEE Symposium on Industrial Electronics & Applications, Kuala Lumpur, 2009, pp.548-551.
- [7] N. Kaewmard and S. Saiyod, "Sensor data collection and irrigation control on vegetable crop using smart phone and wireless sensor networks for smart farm," 2014 IEEE Conference on Wireless Sensors (ICWiSE), Subang, 2014, pp. 106-112.
- [8] Kansara, Karan, et al. "Sensor based Automated Irrigation System with IOT: A Technical Review." International Journal of Computer Science and Information Technologies 6.6 (2015).
- [9] <https://console.bluemix.net/docs/services/IoT/index.html>
- [10] <https://github.com/sandeepmistry/arduino-LoRa>
- [11] <https://www.silabs.com/products/development-tools/software/usb-touart-bridge-vcp-drivers>
- [12] <https://github.com>
- [13] <https://en.wikipedia.org/wiki/LoRa>
- [14] <https://circuitglobe.com/relay.html>
- [15] <https://tuts.codingo.me/connect-esp8266-to-ibm-bluemix>
- [16] <https://maker.pro/arduino/projects/arduino-soil-moisture-sensor>
- [17] <https://www.electroschematics.com/12145/working-with-water-flowsensors-arduino/>
- [18] <https://howtomechatronics.com/tutorials/arduino/dht11-dht22-sensor-temperature-and-humidity-tutorial-using-arduino/>
- [19] <https://www.wemos.cc>
- [20] <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>