AUTONOMOUS SPRINKLER IRRIGATION SYSTEM

MINOR PROJECT REPORT

Submitted in partial
fulfillment of the
requirements for the award
of the degree of

BACHELOR OF TECHNOLOGY
IN
CIVIL ENGINEERING

NATIONAL INSTITUTE OF

TECHNOLOGY KURUKSHETRA

KURUKSHETRA (Haryana)-136119

INDIA

Submitted By
RAHUL CHAHEL
11610125
C8

Under Supervision of
DR.ARUN GOEL
PROFESSOR
CIVIL ENGINEERING

INDEX

CHAPTER 1
INTRODUCTION

1.1 ABSTRACT

This project propounds a design of automatic water supply via sprinklers in farmland using Raspberry pi, NodeMCU, Solenoid valves, Water Flowmeter and couple of sensors. The components used in the system ensures fecund, spirited and scalable implementation. Depending on the water content of farmland, future precipitation and type of plantation, the system can detect the appropriate water requirement and also can track the amount of water accumulated. The analog data of the water content from soil moisture sensors are transmitted by NodeMCU as digital signal to cloud network.

The system autonomously control the water supply with respect to the future precipitation, current state and current requirement of water to particular plant. All the parameters, sensor data, events and activities can be observed using User Interface provided by cloud system Cayenne.

The system can be applied in farmland varying from small scale to large and as well as small pot plants. Using this system, a promising outcome was observed in irrigation of land in more optimistic way.

1.2 PROJECT TOPIC AND IMPORTANCE

The project entitled “Autonomous Sprinkler irrigation system” consists of irrigating the field for specific plant form using recommended water requirement. Here the irrigation of field and data collection is done by using both software and hardware collaboration.

An attempt has been made via this project to reach an optimized approach to irrigate the field with limiting and controlling the input resources and fulfill the demand of the irrigation at it’s very extend.

The importance of the project lies in obtaining the data from various sensors and processing it with certain predefined parameters to save the wastage of resources and energy.

1.3 OBJECTIVES OF PROJECT

1. To optimize the water usage in field irrigation.
2. To maximize the crop output from limited land and controlled water usage.
3. To save water and other resources.
4. To minimize the Man power involved during irrigation.
5. To take step towards future farms and completely autonomous farms.

1.4 SCOPE OF PROJECT

This project covers the design of an autonomous sprinkler irrigation system. As in the modern world, the quantity of freshwater lessens and would be at severely low level as forecasted. There is a requirement of an optimized approach to it’s use, this project would be one of the main influence as currently agriculture accounts for 70 percent of global freshwater withdrawals and set to increase by 10 percent as per Food and agriculture organization USA.

CHAPTER 2

LITERATURE REVIEW

Numerous researchers have worked in the field of automatic irrigation system. They opted different approaches to attain the result with combination of various devices. They have also discussed and used different power sources for whole system and components individually. Besides, the techniques to create network between different components of the system and design of control system is different in each case.

The article on the automated water supply system shows that the system can effectively manage water resources for urban residential areas [4].The water distribution system was effective for the urban residential area where the control of the water is not that much frequent and the number of parameters to be kept in mind to decide the supply are not that much dynamic. As the system used doesn't contain potential to handle the complex and dynamic parameters it can't be used for designing the irrigation system efficiently. But in case of our system high processing power equipment are used which can handle various number of parameters efficiently and can control the whole irrigation system with minimum errors.

The paper discussed certain of the technology which can be used while implementation of the automated irrigation system. In this paper, a system is discussed to distribute water to agricultural land. The implementation of GPRS module and wireless sensor network in the system was mentioned [5].The system suggested uses GPRS module which is an old technology with very less speed and more repose time while the sensor network is not interconnected in hierarchy which makes the system to work with more difficulty while the system in our project work in hierarchy which makes it more stable and controllable. And our system is embedded with Wi-Fi network which is more efficient then the GPRS network.

Another author used ZigBee chips to setup the mesh network systems.[6]In the purposed system GSM shield was used for SMS based commutation system to strengthen the system's control and management. The module used for communication is GSM module which is only efficient in sending the SMS. The SMS type of communication is not eligible enough is built a completely automated irrigation system. Our system uses Wi-Fi based inter communication system which is completely efficient to build an autonomous irrigation system.

[7] Automated irrigation system was developed by Joaquín Gutiérrez et al in which photovoltaic panels was used as a power supply. As individually the photovoltaic cells can't guarantee uninterrupted power supply to the system so rechargeable batteries are provided as alternate power supply.

The author in [8] published the work on control of the irrigation system with the help of the wireless sensor network. They used GPS technology to locate the exact position of the sensors and the sprinklers in a large area and Bluetooth technology for wireless communication. Our system uses latest NodeMCU embedded with Wi-Fi transmitter and receiver and Raspberry Pi as main control center which will communicate more efficiently then the Bluetooth technology and cover larger area easily. Beside, our proposed system doesn't use GPS and the sensor's position will be decided by the system itself according to the size of field.

In [9], proposed review paper on irrigation system based on Radiofrequency module pointed out certain glitches in the existing system like security and slow communication. Our system uses Raspberry Pi microcomputer and NodeMCU which work on Wi-Fi technology with up to 80mbps and also provide WPA2 security to prevent any unauthorized access to the network. This makes our system's commination faster and more secured.

In the automatic irrigation system purposed in [10], the system uses temperature sensor, soil moisture sensor and humidity sensor which doesn't account for various parameters like weather forecast, amount of water content required etc. The system discussed doesn't include the area of land to be covered or large scale implementation.

Considering all the shortcomings, I propound a system which is suitable for any size of farmland and can irrigate the farm more efficiently while displaying all the parameters to any remote device. The weather forecast is one of the main attraction of the system as it can obtain the forecast of the area being irrigated of up to 5 days including amount of precipitation, maximum temperature, minimum temperature, humidity etc. The status of all the components can be obtained remotely without being in direct contact with the network.

CHAPTER 3
Project Visit

Name of Organization: Haryana Irrigation Research and Management Institute (HIRMI)
Registered Address: Canal Colony, Thanesar, Haryana 136119
Objective of visit: To obtain knowledge about working of sprinkler irrigation system.
Date: 10 Aug, 2019
Project:

Introduction:

The Sprinkler irrigation system is setup in working condition to obtain the knowledge and statistical data regarding the amount of water consumed and amount of crop produced with respect to conventional irrigation system.

Description:

Firstly conventional water supply via submersible is provided as the main source of water supply for whole field with conventional power source. The water supply from the submersible is divided into two, one for the sprinkler irrigation system and other for the rest of field which is supplied via open rectangular channels, additional water tank is provided in case of requirement during non-availability of main line or other alternative source of power supply. The supply to the sprinkler system is embedded with different components. Firstly, Flow meter Gauge is present to calculate the amount of water supplied to the storage tank present. A water pump is present in the storage tank with power source from solar power. The pump outlet leads to two components one is back to the tank which would be used in case of blockage of other path and prevent any damage to the pump; and the other lead to the clean master filter with additional inlet provide to add certain add-ons it is then connected to Super flow plus gold which is then connected to the field sprinklers and ARU C (for air release).The water supplied to the field is controlled and divided into two parts to irrigate two parts of the field. Two separate valves are provided to control the flow and supply to these blocks. Further the Parallel-Series network of sprinklers is provided with valve in each of the branch.

Figure 1 HIRMI kurukshetra

Land Under Cultivation: 1 hectare or 10000m^2

Figure 2Land under Cultivation HIRMI

Various Components Installed:

Flowmeter Gauge:

Manufacturer: Sensus

Least count: 0.001m^3

Figure 3Flowmeter Gauge

Submersible:

Submersible was provided as the main source of water for the whole filed and to provide water to water tank dedicated for the sprinkler irrigation system with Flow meter gauge embedded in the connecting pipeline.

Input Source: Main Line

Power: 20HP

Depth: 300ft or 91.44m

Water Tank:

Concrete water tank provided to ensure the water availability to the sprinkler irrigation system at desired pressure.

Type: Open Top Concrete water tank

Capacity: 10,000L

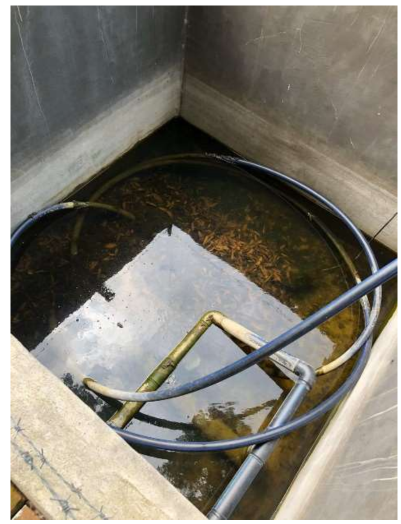


Figure 4Intermediate Water Tank

Pump for sprinkler system:

Water pump to supply water to sprinkler system from the storage tank

Manufacturer: Jain Irrigation Systems Ltd.

Input Source: Photovoltaic solar energy plant

Power: 3HP

Photovoltaic solar energy plant:

Manufacturer: Jain Irrigation Systems Ltd.

Number of Cells: 10

Output Power (per cell): 300W

Total Power Output: 3000W

Figure 5Photovoltaic Solar Energy Plant

Add-On Joint:

Add-on joint is provided to add various fertilizers or other substances to water supply to maximize the output or to eliminate unessential substances from the field.

Figure 6Add-On Joint

Clean Master (Filter 1):

Filter provided for filtration of water in micro irrigation systems to prevent clogging due to physical and biological impurities.

Manufacturer: Jain Irrigation Systems Ltd.

Features and Specification:

1. **Standard Pure Polyester / Epoxy coating for Protecting from Corrosion** Coated with more than 70 micron thick deep blue colored pure Polyester powder on outer surface & Epoxy coating from inner side for protection against corrosion and weather effects
2. **Unique Manifold Design** Unique design of manifold for single tank unit facilitates flushing with filtered water.
3. **High Quality Silica Sand as Media** Filtration media is crushed silica sand/quartz gravel of particle size 1 mm to 2 mm (0.039 to 0.078 inch).
4. **Innovative Candle Assembly** Innovative Candle assembly provided to pass filtered water in system.
5. **Various Connection Options Available** Threaded connection, Flanged connection or Easy Fix™ connection available
6. **Various Options for Backwash** Available in manual, semi-automatic or fully automatic backwash options

Figure 7Clean Master (Filter 1)

Aru c:

It is a Valve used for Air Release and to maintain the pressure in the pipe.

Manufacturer: Jain Irrigation Systems Ltd.

Features and Specification:

1. **Compact Design** Innovative compact design manufactured from high performance plastic
2. **Unique Orifice Design** Unique orifice design facilitates easy and quick expulsion of air
3. **Optional Multiple Air Release Valve Assembly** Multiple Air Release Valve assembly option can also be available for bigger size pipe lines
4. **Pressure Balanced Float** Perfect pressure balanced float dynamically removes even small air pockets
5. **Innovative Double Action Function** Continuous acting air release valve that releases air entrapped in the pipeline during start up and while in operation continuously
6. **Threaded Inlet Connection** Threaded inlet connection available for connection with system

Figure 8Aru c (L: Field), (R: Ref: Jain irrigation system)

Super Flow plus Gold (Filter 2):

It is a filter used for water filtration of drip & landscape irrigation system.

Manufacturer: Jain Irrigation Systems Ltd.

Features and Specification:

1. **Unique Smart Clean Element** Unique ‘Smart–Clean’ element, assures high performance and effective filtration. (Flow direction Out to In)
2. **Angular or Straight Outlet Option Available** Optional two outlet positions facilitates installation in angular or straight fashion (2" onwards)
3. **Various Connection Options Available** Threaded connection, Flanged connection or Easy Fix™ connection available.
4. **Easy for Maintenance** Strong and smooth opening, drip–tight SS clamp closure
5. **Vacuum Releasing Facility** Vacuum breaker provided on top of filter to release air / vacuum at start or end of system (from 2" onwards)
6. **Draining Facility Available** Additional Drain Valve provided to remove dirt from filter (1.5" onwards)

Figure 9 Super Flow plus Gold (Left: field, Right: Jain irrigation system)

Central Valves:

PVC Ball Valves are used to control the flow of water to two different blocks of the field

Figure 10 Central Valves

Sprinkler (5022_SD_U) Network:

The Parallel-Series network application was present with individual branch controllable via Valves.

Manufacturer: Jain Irrigation Systems Ltd.

Number of sprinkles per series/branch: 3

Number of parallel branches (per block): 4

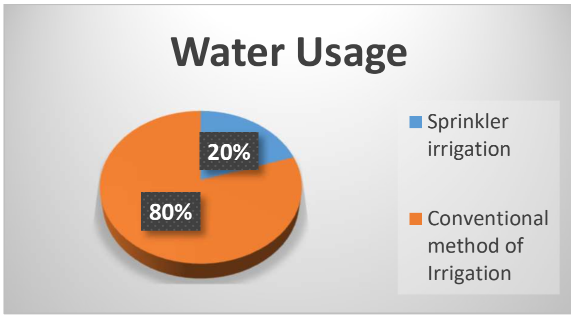
Number of parallel branches: 8

Discharge water (per sprinkler):520L/hr.

Figure 11 Sprinkler (5022_SD_U)

Previous Statistics:

Initially one block was irrigated by sprinklers and other by conventional method the usage of water was in the ratio of 1:4 and production was 12:11 respectively in case of cultivation of rice.



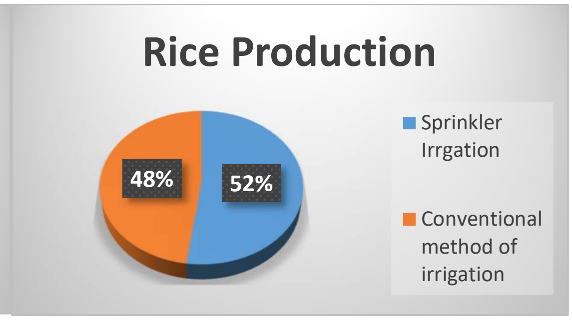


Figure 12 HIRMI current Rice crop

CHAPTER 4
PURPOSED DESIGN

The system designed contain various components majorly divided into five category:

1. Ground module
2. Parameter module
3. Connecting module
4. Control module
5. Cloud

The correct and efficient collaboration all these five modules is the backbone of the system. The ground collect data regarding the condition of soil i.e. the water content present in the soil while the parameter module collects the weather forecast of the area from certain weather API. Both of the modules upload the data to cloud via Connecting module, all the data received on the cloud is processed with certain predefined parameters and respective signals are sent to control module to control the sprinkler irrigation system and different components of the irrigation system.

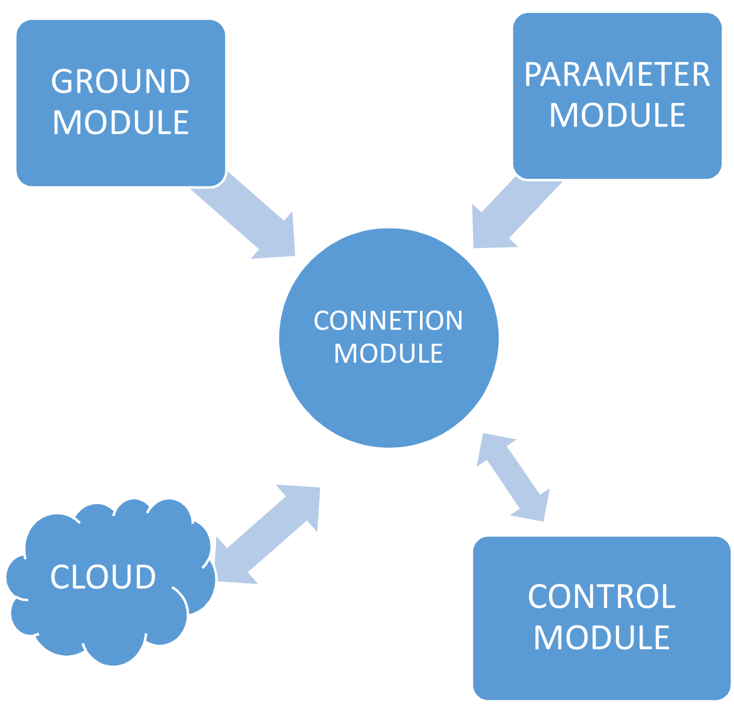


Figure 13 Flow of project

GROUND MODULE

Each of the ground module is assigned certain Id by the cloud platform so that all the modules can be identified individually and each of the ground module embedded with 4 components connected with each other using wires and soldering to fulfil the task.

Components of Ground module:

1. NodeMCU
2. Soil Moisture sensor
3. Battery
4. IC 7805

NODEMCU

NodeMCU or ESP8266XCD is a low cost Wi-Fi module embedded with on board microcontroller that can be configured using Arduino IDE to connect various electronic components which deals with Analog and digital signals. It is one of the most popular device used in Internet of Things(IoT).It can be used to control, monitor, analyze the Electrical ,Mechanical and Electronic components.

Number of Digital Pins: 8

Number of Analog Pins: 1

Input voltage: 4V to 6V

Digital Pin voltage/Output voltage: 3.3V

Type of communication: Wi-Fi

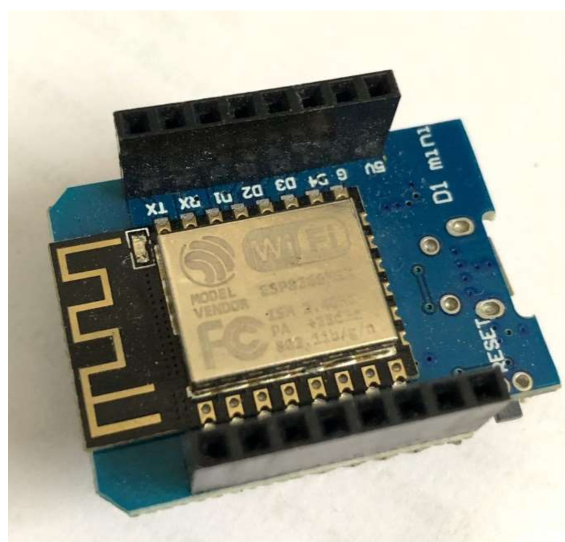


Figure 14 NodeMCU

Use in Ground Module:

It is acting as a brain of the ground module which decodes the analog signals from the soil moisture sensor and convert them into numerical values from 0 to 100.It transmit the data i.e. the water content to the connecting module then further to cloud.

Soil Moisture sensor

Soil moisture sensor or YL-69 is used to detect the water content of the soil. It contain two pieces the electronic circuit and probe with two pads. The sensor embedded with built-in potentiometer for sensitivity adjustment of the digital output pin. The basic principle of working of the soil moisture sensor is the increase in the conductivity of the soil as amount of water increase and number of ions increases to conduct the current between two pads of the probe.

Working voltage: 5V

Calibration Required: Yes

Digital pin: 1

Analog pin: 1

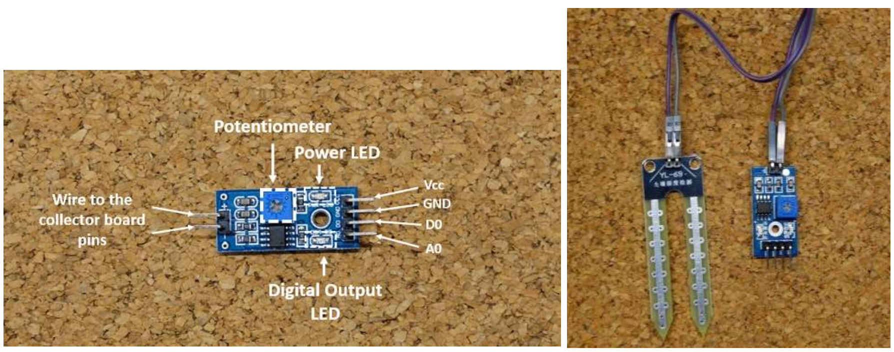


Figure 15 Soil Moisture sensor chip & module

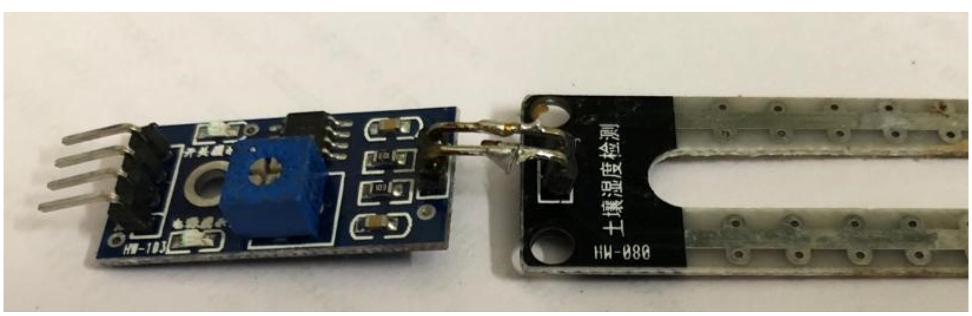


Figure 16 Soil moisture sensor

Use in Ground Module:

It act as medium between the physical work and the digital world as it senses the water content of the soil and transmit it in the form of the digital and analog signal to the microcontroller/NodeMCU.

Battery

6F22 i.e. 9V battery is used to power the ground module.

Output voltage: 9V

Ampere hour: 600mAh

Figure 17 Battery

IC7805

IC7805 is use to convert DC voltage from 6v to 24V into 5V DC.

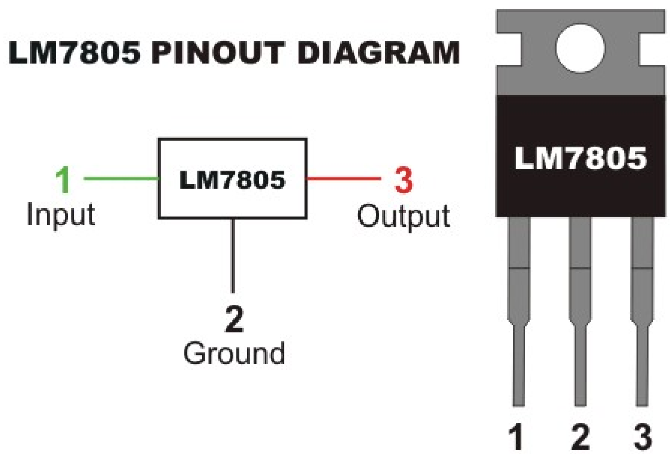


Figure 18 7805

Use in ground module:

As mentioned the NodeMCU and soil moisture sensor work on 5V so it act a voltage regulator between the battery and the ground module.

PARAMETER MODULE

Parameter module consisted of mainly one component i.e. Raspberry Pi which is a microcomputer the extract various parameters like weather forecast, humidity of air etc. from a remote server hosted by weather forecast firm and supply all the information to the cloud. A program is written in the Raspberry Pi I Python language to obtain and transmit the data.

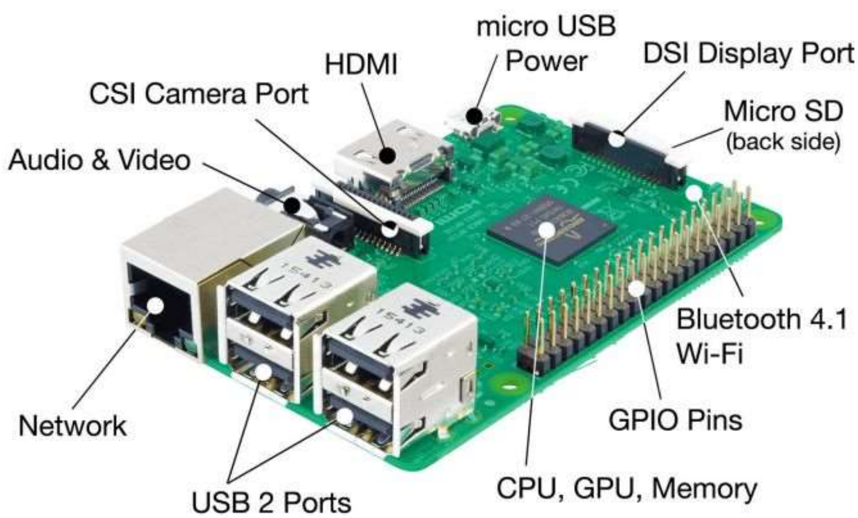


Figure 19 Raspberry Pi (Source: Silverline Electronics)

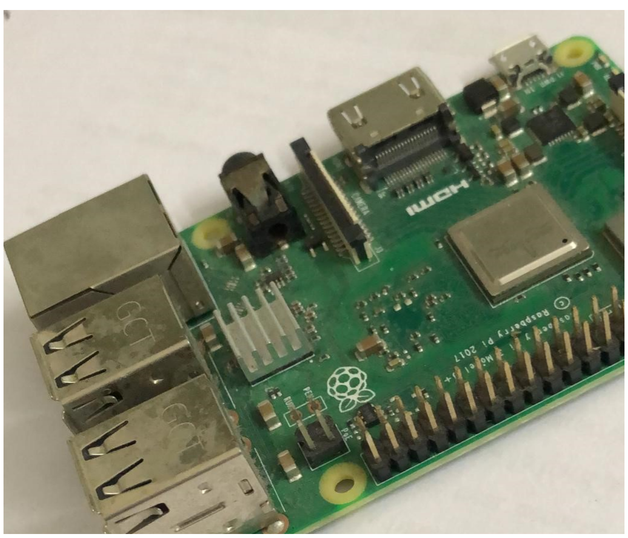


Figure 20 Raspberry Pi

CONTROL MODULE

Control Module is the module to control the sprinkler irrigation system physically. It contain 4 components

1. NodeMCU
2. Solenoid valve
3. Flowmeter
4. Motor driver L298

NodeMCU

Use in control module:

NodeMCU receive the data from the cloud to control the solenoid valves with the help of motor driver and obtain the discharge from the Flowmeter and calculate the amount of water used/passed then send the value to the cloud.

Solenoid Valve

Solenoid valve is a control unit in which when electricity energize it either shut off or allow the fluid to pass through it.

Working Voltage: 8V to 24V

Working Pressure: 0.02 to 0.8Mpa

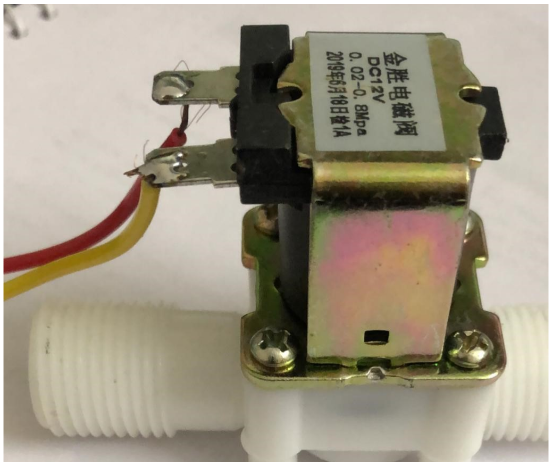


Figure 21 Solenoid Valve

Use in Control Module

It is being used to control physically the flow of water in the sprinkler system.

Flowmeter

Flowmeter or YF-S201 with 1 to 30L/min flow rate, It is connected in between the water supply line to measure the amount of water moved through .It contains a pinwheel sensor to measure the discharge. There is an integrated magnetic Hall Effect sensor which generates the electric pulse with every revolution. By counting the number of pulses from output of sensor, the discharge can be calculated.

Working voltage: 5-18V

Max current drawn: 15Ma@5V

Working flow rate: 1-30L/min

Figure 22Flow meter YF-S201

Motor Driver

Motor driver or L298 is the motor driver which help in controlling the motors that are operating on higher voltage then the voltage that microcontroller can handle and withdraw more amount of current.

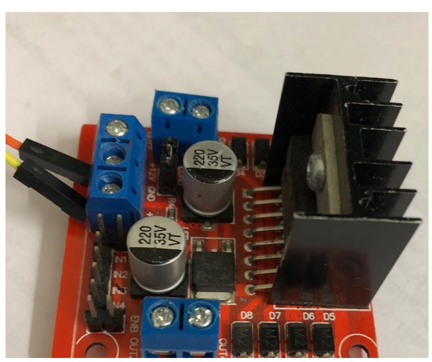


Figure 23 L298

Use in Control Module:

As NodeMCU can't deliver large amount of current that is required for operation of the solenoid valve so Motor driver is required to operate the valve and act as a medium of operation between the NodeMCU and Solenoid Valve.

CONNECTING MODULE

Connection module is mainly the Wi-Fi intermediator router or modem which help all the devices or components to connect on a common network and provide internet to the network so the all devices can access the cloud.

CLOUD

Cloud is an online platform to upload and share the data with other users and process the data. The cloud service that we are using is myDevices Cayenne which provide different tools and user-friendly UI. It IoT project builder that help in development, designing prototypes and share the project with other users so it can be brought to production.

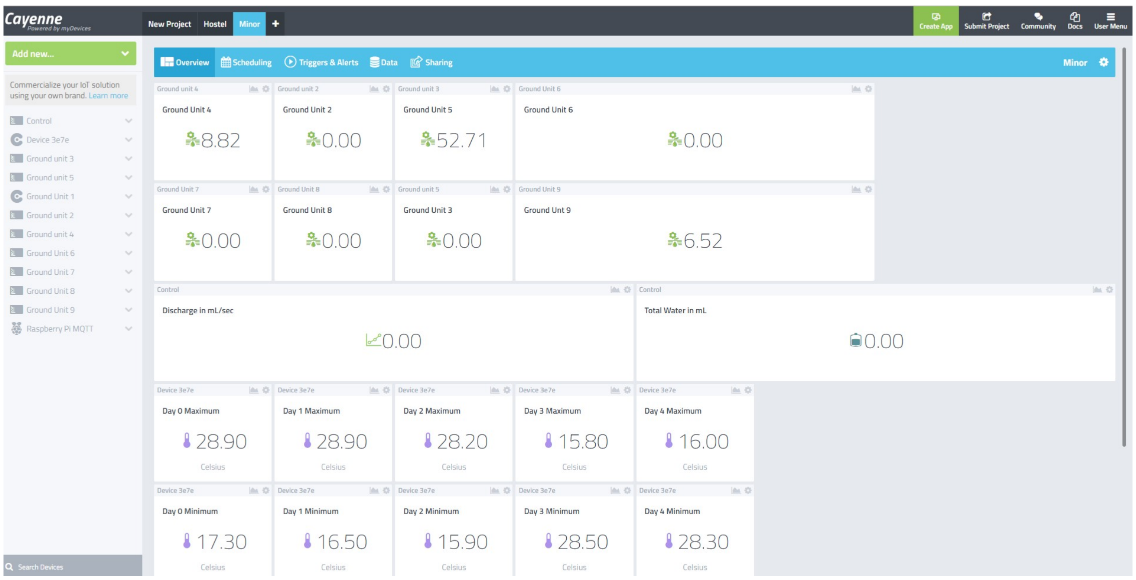


Figure 24 Cayenne User Interface

CHAPTER 5
METHODOLOGY

This project provides a new insight into the field of agriculture and how the opportunities given to us by technological advancement be used to amply align with them to optimize the throughput and more importantly use the crucial resources in the best optimal way. The aim is to fight the water scarcity challenge by automating the irrigation process with the side benefit of more yields.

Understanding the fact that moisture level in the soil directly affects the yields and keeping in mind the step to avoid the curve of over-irrigation, the field's water content is readily recorded at a pre-assigned interval. This measured by the Soil moisture sensor or YL-69 and the data received is in the raw format and is further processed using NodeMCU microcontroller and is used to upload this data on a remote repository or cloud. Thus a more thorough inspection of the field is possible by placing an array of such devices in the field. Whenever the moisture level in soil at a particular area drops lower than a threshold value, an automated sprinkler irrigation system is activated to provide water over that area. Also, not only the present value of field moisture level is the deciding factor in the need for irrigation, but also the prediction in weather is taken into account. The tools and devices were used such as to minimize the size and cost for the model and increase the processing and also understanding of the project. The details can be displayed over mobile devices, computers as well using the Cayenne application thus making it easier to evaluate the whole process and make amendments wherever necessary.

An array of ground modules are present over the entire area of the field. Each module consists of 2 parts, one is the Soil moisture sensor or YL-69 and NodeMCU Microcontroller. The sensor gives data in raw format and that is processed and pushed over to a remote database. This is possible through the NodeMCU IC which comes with an in-built Wi-Fi module that connects the ground module with the internet, or more precisely the database. Each ground module is connected to a 500 mAh battery and the device uses 18mA of power and thus the battery does not need to be changed very frequently. Once the data is sent over to the cloud, a number of other factors are included to process those readings such as, predicted values for the weather, humidity in the area and after that a threshold is calculated for the soil and if any of the sensors give measurements lower than them, that particular area is irrigated using sprinkler irrigation system. This whole system, excluding the battery changing part is automated and needs no constant monitoring. Specific alarms can be set to be triggered in unwanted conditions. These ground modules since will always be present inside the ground, are waterproof so as to sustain rainy weathers as well. The sprinkler irrigation system is controlled by solenoid valves which in turn is connected to and controlled by NodeMCU microcontroller using motor driver, which in turn makes its decision by values processed in the cloud. The whole system is thus wireless and does not need much expertise in setting up. The whole system can be overlooked and controlled by making an UI or using any open source available UI such as Cayenne. With the help of this, the data and the status of every module in the ground and of the irrigation system be displayed and observed and it also allows us to set a trigger to check the readings any particular time of the day and also alarms for any pre-defined conditions.

CHAPTER 7
VISUALIZATION OF RESULTS

GROUND MODULE

Figure 25 Individual reading of one of the ground module

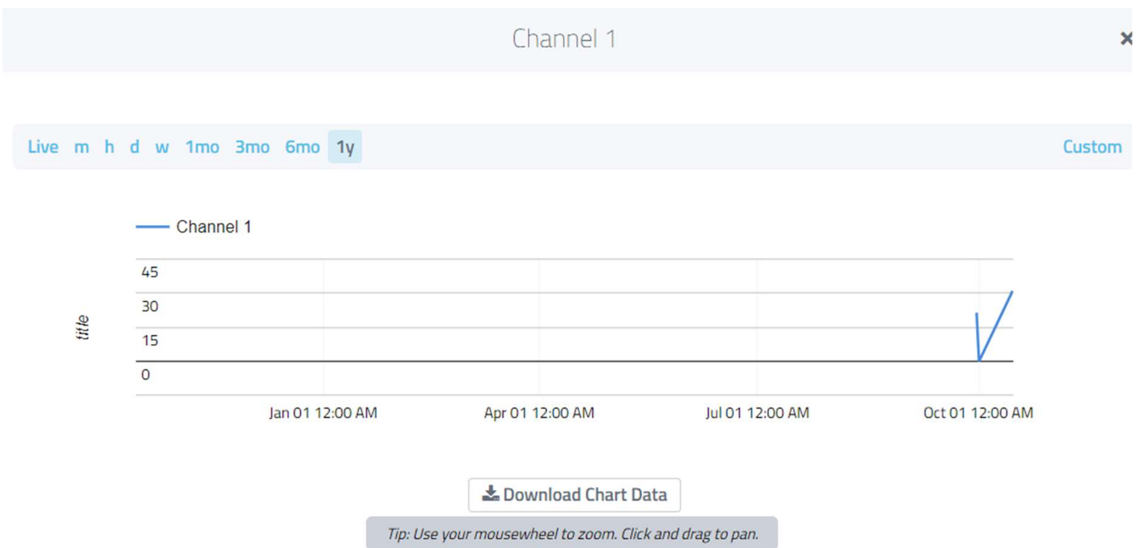


Figure 26Individual reading of one of the ground module in graph with history of values

RASPBERRY PI

Figure 27 Display of current state of Microcomputer

PARAMETER MODULE

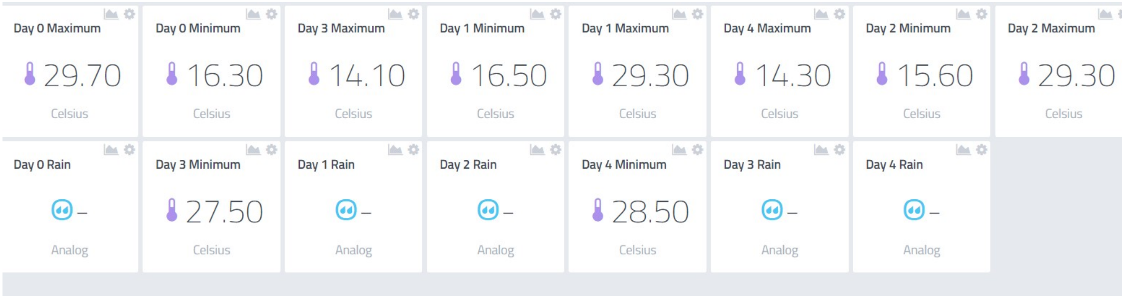


Figure 28 Display of all the perimeters delivered by Parameter module

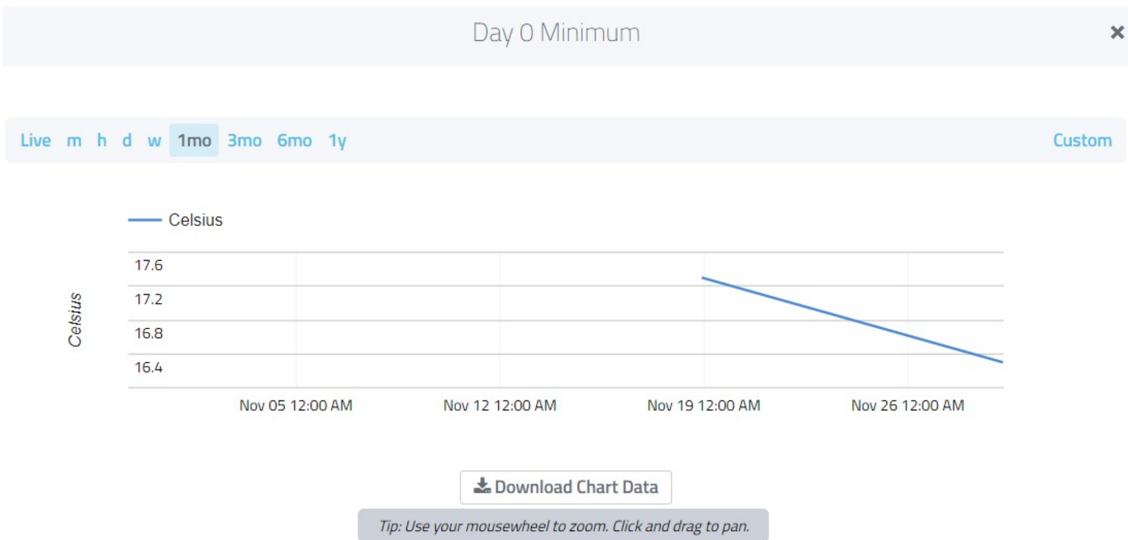


Figure 29 Graphical History of parameters

CONTROL MODULE

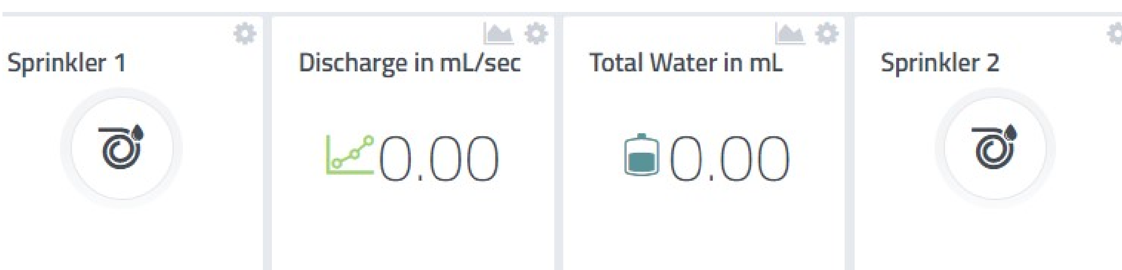


Figure 30 Display of Control Module INPUT AND OUTPUT parameters

CHAPTER 7
CALIBRATION

GROUND MODULE

To calibrate the ground module water content test was done with 3 different samples of soil as below

Water Content	Sample 1			Sample 2			Sample 3			Average Calibration Constant
Sensor No	Lab	Sensor	Calibration Constant	Lab	Sensor	Calibration Constant	Lab	Sensor	Calibration Constant	
1	23%	21.50%	1.0698	30%	29.30%	1.0239	45%	44.23%	1.0174	1.0370
2	23%	21.78%	1.0560	30%	29.60%	1.0135	45%	44.80%	1.0045	1.0247
3	23%	22.30%	1.0314	30%	29.43%	1.0194	45%	43.56%	1.0331	1.0279
4	23%	21.90%	1.0502	30%	30.23%	0.9924	45%	44.65%	1.0078	1.0168
5	23%	22.30%	1.0314	30%	29.12%	1.0302	45%	44.32%	1.0155	1.0257
6	23%	22.47%	1.0235	30%	29.62%	1.0129	45%	44.24%	1.0172	1.0179
7	23%	22.64%	1.0157	30%	29.64%	1.0120	45%	44.67%	1.0074	1.0117
8	23%	22.82%	1.0081	30%	29.67%	1.0111	45%	44.64%	1.0081	1.0091
9	23%	22.99%	1.0005	30%	29.70%	1.0102	45%	44.32%	1.0153	1.0087

CONTROL MODULE

To calibrate the flow meter present in the control module three time water sample is passed through it of known quantity

Sample no.	Actual Volume	Displayed Volume	Calibration Constant
1	400mL	412mL	0.9709
2	300mL	309mL	0.9709
3	500mL	505mL	0.9901

Calibration constant = (0.9709+0.9709+0.9901)/3=0.9773

CHAPTER 8
CONCLUSION

The autonomous irrigation system described in the report can ensure a systematic and scientific approach in irrigating and taking care of the crop which can dramatically improve the productivity. Such can be extended easily and it is very cost-effective. With the improvement of technology, the system will become more efficient and optimized. For example, more accurate weather forecast can help getting better results. If soil nutrition can be further determined then another parameter will increase which increase the delivery of the project. There are multiple areas of improvement that can also be explored further like hybridization of the system in case of control of the components i.e. components can be controlled and optimally use offline using machine learning and AI.

Certain kind of self-learning system can be designed which increase the crop yield by varying different parameters on it own using artificial intelligence and neural network.

To conclude, the system will help in optimization of the utilization of water resource and minimize the human effort in the field for irrigation which will consequently decrease the damage to crops due to human error.

APPENDIX I: PROGRAMMING

GROUND MODULE

Platform: Arduino IDE
General Code:

```
#define CAYENNE_PRINT Serial
#include <CayenneMQTTESP8266.h>
char ssid[] = "WiFi SSID";
char wifiPassword[] = "WiFi Password";
#define VIRTUAL_CHANNEL 1
char username[] = "Username assigned to device by cloud";
char password[] = " Password assigned to device by cloud ";
char clientID[] = " Client ID assigned to device by cloud ";
const int sensor_pin = A0;
void setup() {
  Serial.begin(9600);
  Cayenne.begin(username, password, clientID, ssid, wifiPassword);
}

void loop() {
  float moisture_percentage;
  float calibration_constant=1.9607;
  moisture_percentage =( 100.00 - ( (analogRead(sensor_pin)/1023.00) * 100.00
))*calibration_constant;
  if (moisture_percentage>=100)
   moisture_percentage=100;
  if (moisture_percentage<=0)
  moisture_percentage=0;
  Serial.print("Soil Moisture(in Percentage) = ");
  Serial.print(moisture_percentage);
  Serial.println("%");
 Cayenne.virtualWrite(VIRTUAL_CHANNEL, moisture_percentage);
  delay(1000);}
```

CONTROL MODULE

Platform: Arduino IDE
General Code:

```
#define CAYENNE_DEBUG
#define CAYENNE_PRINT Serial
#include <CayenneMQTTESP8266.h>
char* username = "58aa9740-ce3a-11e8-810f-075d38a26cc9";
char* password = "77a7519a3352c852702fbdfd405b38dad4e9a3ea";
char* clientID = "5dc6b1f0-0629-11ea-8221-599f77add412";
char ssid[] = "TP-LINK_D730";
char wifiPassword[] = "29410781";
#define PULSE_PIN D2
#define LED_PIN D7
#define VIRTUAL_CHANNEL 2
#define VIRTUAL_CHANNEL1 9
volatile long pulseCount=0;
float calibrationFactor = 4.5;
float flowRate;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;
unsigned long oldTime;
#define ACTUATOR_PIN 2
#define ACTUATOR_PIN1 14
void ICACHE_RAM_ATTR pulseCounter()
{
  pulseCount++;
}
void setup()
{
  Serial.begin(9600);
  pinMode(ACTUATOR_PIN, OUTPUT);
 Cayenne.begin(username, password, clientID, ssid, wifiPassword);
pulseCount        = 0;
  flowRate          = 0.0;
```

```
flowMilliLitres   = 0;
totalMilliLitres  = 0;
oldTime           = 0;
pinMode(PULSE_PIN, INPUT);
attachInterrupt(PULSE_PIN, pulseCounter, FALLING);
}

void loop()
{
  if((millis() - oldTime) > 1000)    // Only process counters once per second
  {
    detachInterrupt(PULSE_PIN);
    flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount) / calibrationFactor;
    oldTime = millis();
    flowMilliLitres = (flowRate / 60) * 1000;
    totalMilliLitres += flowMilliLitres;
    unsigned int frac;
    Serial.print("Flow rate: ");
    Serial.print(int(flowRate));  // Print the integer part of the variable
    Serial.print(".");             // Print the decimal point
    frac = (flowRate - int(flowRate)) * 10;
    Serial.print(frac, DEC) ;      // Print the fractional part of the variable
    Serial.print("L/min");
    Serial.print("  Current Liquid Flowing: ");             // Output separator
    Serial.print(flowMilliLitres);
    Serial.print("mL/Sec");
    Serial.print("  Output Liquid Quantity: ");             // Output separator
    Serial.print(totalMilliLitres);
    Serial.println("mL");
    pulseCount = 0;
    attachInterrupt(PULSE_PIN, pulseCounter, FALLING);
  }
  Cayenne.loop();
Cayenne.virtualWrite(5,totalMilliLitres);
```

```
Cayenne.virtualWrite(6,flowRate);
}
// This function is called when data is sent from Cayenne.
CAYENNE_IN(VIRTUAL_CHANNEL)
{
  int value = getValue.asInt();
  CAYENNE_LOG("Channel %d, pin %d, value %d", VIRTUAL_CHANNEL, ACTUATOR_PIN, value);
  // Write the value received to the digital pin.
  digitalWrite(ACTUATOR_PIN, value);
  Serial.println(value);
}
CAYENNE_IN(VIRTUAL_CHANNEL1)
{
  int value = getValue.asInt();
  CAYENNE_LOG("Channel %d, pin %d, value %d", VIRTUAL_CHANNEL1, ACTUATOR_PIN1, value);
  // Write the value received to the digital pin.
  digitalWrite(ACTUATOR_PIN1, value);
  Serial.println(value);
}
```

PARMETER MODULE

Language: Python

General code:

```
import paho.mqtt.client as mqtt
import cayenne.client
import time
import logging
import sys
import json
import time
import urllib.request
num=0
username="58aa9740-ce3a-11e8-810f-075d38a26cc9"
password="77a7519a3352c852702fbdfd405b38dad4e9a3ea"
client_id="6083b140-0638-11ea-a4a3-7d841ff78abf"
client = cayenne.client.CayenneMQTTClient()
client.begin(username,password,client_id)
API="3EelORMiAwM9PduoJiqdM2DCRA7qfITi"
CountryCode="IN"
city=input("CityName")
key=""
def getLocation(CountryCode,city):

search_address="http://dataservice.accuweather.com/locations/v1/cities/
"+CountryCode+"search?apikey=3EelORMiAwM9PduoJiqdM2DCRA7qfITi&q="+city+
"&details=true"
    with urllib.request.urlopen(search_address) as search_address:
        data=json.loads(search_address.read().decode())
    location_key=data[0]['Key']
    return (location_key)
def getForcast(location_key):
    num=1

daily_forcastUrl="http://dataservice.accuweather.com/forecasts/v1/daily
/5day/"+location_key+"?apikey=3EelORMiAwM9PduoJiqdM2DCRA7qfITi&details=
true&metric=true"
    with urllib.request.urlopen(daily_forcastUrl) as daily_forcastUrl:
        data=json.loads(daily_forcastUrl.read().decode())
    for key1 in data["DailyForecasts"]:
        print("Weather Forcast for"+key1['Date'])
        temp1= "temp =" + str(key1['Temperature']['Minimum']['Value'])
        temp2= str(key1['Temperature']['Maximum']['Value'])
        print(temp1+" to "+temp2)
        rainprobD=str(key1['Day']['RainProbability'])
        rainD=str(key1['Day']['TotalLiquid'])
        print("Probablity of rain during day="+rainprobD+"%")
        print("Amount of rain="+rainD)
        rainprobN=str(key1['Night']['RainProbability'])
        rainN=str(key1['Night']['TotalLiquid'])
        print("Probablity of rain during night="+rainprobN+"%")
        print("Amount of rain="+rainN)

        client.loop()
        if num==1:

client.celsiusWrite(1,key1['Temperature']['Minimum']['Value'])

client.celsiusWrite(2,key1['Temperature']['Maximum']['Value'])
```

```
            client.virtualWrite(3,key1['Day']['TotalLiquid'],"tl",
"null")
        if num==2:

client.celsiusWrite(4,key1['Temperature']['Minimum']['Value'])

client.celsiusWrite(5,key1['Temperature']['Maximum']['Value'])
            client.virtualWrite(6,key1['Day']['TotalLiquid'],"tl",
"null")
        if num==3:

client.celsiusWrite(7,key1['Temperature']['Minimum']['Value'])

client.celsiusWrite(8,key1['Temperature']['Maximum']['Value'])
            client.virtualWrite(9,key1['Day']['TotalLiquid'],"tl",
"null")
        if num==4:

client.celsiusWrite(10,key1['Temperature']['Minimum']['Value'])

client.celsiusWrite(11,key1['Temperature']['Maximum']['Value'])
            client.virtualWrite(12,key1['Day']['TotalLiquid'],"tl",
"null")
        if num==5:

client.celsiusWrite(13,key1['Temperature']['Minimum']['Value'])

client.celsiusWrite(14,key1['Temperature']['Maximum']['Value'])
            client.virtualWrite(15,key1['Day']['TotalLiquid'],"tl",
"null")

        num=num+1

key=getLocation(CountryCode,city)
getForcast(key)
```

Sample output:

Python 3.7.3 (/usr/bin/python3)

>>> %Run weatherminor.py

Connecting to mqtt.mydevices.com:1883

CityNamethanesar

Weather Forcast for2019-11-29T07:00:00+05:30

temp =16.3 to 29.7

Probablity of rain during day=0%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Probablity of rain during night=0%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Connected with result code 0

SUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/cmd/+

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/sys/model
Python

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/sys/version
1.1.0

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/1
temp,c=16.3

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/2
temp,c=29.7

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/3
tl,null={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Weather Forcast for2019-11-30T07:00:00+05:30
temp =16.5 to 29.3
Probablity of rain during day=0%
Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}
Probablity of rain during night=0%
Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}
PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/4
temp,c=16.5

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-

7d841ff78abf/data/5

temp,c=29.3

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/6

tl,null={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Weather Forcast for2019-12-01T07:00:00+05:30

temp =15.6 to 29.3

Probablity of rain during day=0%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Probablity of rain during night=0%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/7

temp,c=15.6

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/8

temp,c=29.3

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/9

tl,null={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Weather Forcast for2019-12-02T07:00:00+05:30

temp =14.3 to 28.5

Probablity of rain during day=3%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Probablity of rain during night=0%

Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/10

temp,c=14.3

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/11
temp,c=28.5

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/12
tl,null={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

Weather Forcast for2019-12-03T07:00:00+05:30
temp =14.1 to 27.5
Probablity of rain during day=0%
Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}
Probablity of rain during night=0%
Amount of rain={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}
PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/13
temp,c=14.1

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/14
temp,c=27.5

PUB v1/58aa9740-ce3a-11e8-810f-075d38a26cc9/things/6083b140-0638-11ea-a4a3-7d841ff78abf/data/15
tl,null={'Value': 0.0, 'Unit': 'mm', 'UnitType': 3}

REFRENCES

[1] IoT based Autonomous Percipient Irrigation System using Raspberry Pi by Ahmed Imteaj, Tanveer Rahman, and Muhammad Kamrul Hossainand Saika Zaman

[2] Autonomous Irrigation System by M. Murali, Akshay Kumar, Mohit Khare International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7 Issue-6, March 2019

[3] Autonomous Sprinkler System with Internet of Things by Bishal Saha Chowdhury, Nadimpalli Raghukiran International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 16 (2017) pp. 5430-5432

[4] N.B. Bhawarkar, D.P. Pande, R.S. Sonone, Mohd. Aaquib , P.A. Pandit, and P. D. Patil, "Literature Review for Automated Water Supply with Monitoring the Performance System", International Journal of Current Engineering and Technology, Vol. 4, No. 5, Oct 2014.

[5] Suraj S.Avatade, Prof.S. P. Dhanure, "Irrigation System Using a Wireless Sensor Network and GPRS", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 5, May 2015.

[6] Ahmed Imteaj, Tanveer Rahman, Muhammad Kamrul Hossain "IoT based Autonomous Percipient Irrigation System using Raspberry Pi"

[7] Gutiérrez, Joaquín, et al. "Automated irrigation system using a wireless sensor network and GPRS module." IEEE transactions on instrumentation and measurement 63.1 (2014): 166-176.

[8] Kim, Yunseop, Robert G. Evans, and William M. Iversen. "Remote sensing and control of an irrigation system using a distributed wireless sensor network." IEEE Transactions on Instrumentation and Measurement 57.7 (2008): 1379-1387.

[9] Kim, Yunseop, Robert G. Evans, and William M. Iversen. "Remote sensing and control of an irrigation system using a distributed wireless sensor network." IEEE Transactions on Instrumentation and Measurement 57.7 (2008): 1379-1387.

[10] R. Vagulabranan, M.Karthikeyan, V. Sasikala, "Automatic IrrigatioSystem on Sensing Soil Moisture Content", IRJET, 2016