

Design and Fabrication of Crop Production Irrigation System with Flow Analysis

Mr.S. Muralidharan.M.E¹, Karthickraja S², Karthikeyan M³,
Kavin R⁴ Kishorekumar R⁵

Assistant Professor¹, Muthayammal engineering college, Rasipuram-637408, Student^{2,3,4,5}, Muthayammal engineering college, Rasipuram-637408.

Submitted: 15-06-2022

Revised: 20-06-2022

Accepted: 25-06-2022

ABSTRACT:

This project on "Design and Fabrication of Crop Production Irrigation System with Flow Analysis" is intended to control overflow of water and prevent dry running of water pump by detecting the water moisture level through Soil moisture sensor. Measure Moisture level of water & Control forms an important part of any process in the industries as well as domestic purposes. It is generally observed that the measurement and control of liquid moisture level is done either with the help of various methods to control of pump starting and stopping is done for both for industrial & domestic purposes. Here in this particular work we propose an automatic crop irrigation moisture level monitoring & modified control strategy in the control algorithm which helps the pump not to operate when there is no moisture in the farm land thus preventing the dry run of the pump helping in conservation of energy, preventing water wastage as well as protecting the pump from operating under dry run condition by switching the pump off. Moreover in this project we have used to non-contact sensor to measure the reading of water flow passing through the flow meter and calculated. This project uses soil moisture sensor and flow meter to detect the moisture level present in the farm land and also calculate the water flow.

KEYWORDS: ArduinoEsp32Board, Logic Level Converter, Arduino Software (Ide), RelayModule.

I. INTRODUCTION

Water is one of the most important natural resource useful to mankind in many of the possible ways in domestic, agricultural, as well as industrial applications. Hence wastage of water is one of the major concern that mankind needs to look at under every circumstances. Its usage in all sphere of life from domestic to industrial applications highlights

the need of effective water management scheme in every area. We use pumps everywhere from our houses to industries. Major purpose of a pump is to pull liquid from one place to another place. Generally in our homes, we pull water from an underground reservoir tank or bore well to a terrace mounted overhead water tank. In this process it is kind of a hassle to keep the records of when the overhead tank is empty, when to turn on the pump, when the overhead tank is full, when to turn off the pump, weather if the underground reservoir tank has water or not and many more things. This full process of keeping an eye on all the things and timely turning on or off the pump is somewhat time consuming, irritating and a big hassle.

In pumps water acts as coolant to cool off the heat generated by the impeller that heats up other parts. But when the pump runs dry without water, air acts as the working fluid which doesn't act as a coolant. So, the pump is overheated, the impeller will melt along the shaft and impeller seizes to the shaft. Overall, the pump seizes. Then the pump has to be taken apart and it costs almost half the cost of the pump to get it refurbished.

A method of monitoring the flow of water flowing through the pipe by using flow sensor which sends data to the Micro-controller. When there is very low flow or no flow is recognized by the microcontroller, it then triggers the relay which then breaks the circuit of the pump. Thereby reducing the risk of severe damage. The microcontroller is turned on as the pump is turned on. The microcontroller has a program written in it which waits for 45 seconds to ensure that there is no flow before triggering the relay to break the motor circuit. If the pump is run dry for less than 45-60 seconds, the pump should not suffer any damage.

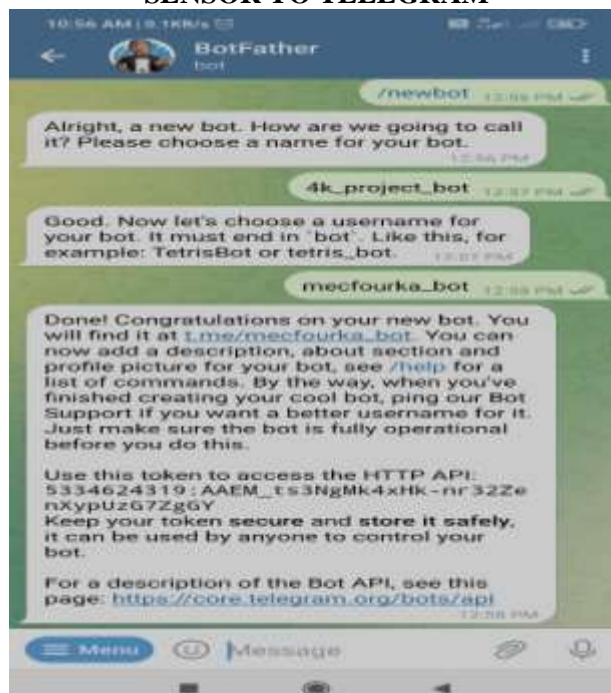
In this work we focusses on developing a discrete water level control system which serves for

namely three purposes firstly saving water at large by preventing the overflow, energy conservation by switching off the pump in certain appropriate conditions & last but not least the objective is to also prevent the dry run of the pump enhancing the

longevity of the operating pump. The automatic water pump controller also finds its application in agricultural farms, industries, households and all places where discrete monitoring & control of liquid level is performed.



II. CONNECTING SOIL MOISTURE SENSOR TO TELEGRAM



Connect the soil sensor to the Arduino ESP32. The soil moisture sensor is connected to the Arduino ESP32, it will sense the moisture level in the soil and the depending upon the moisture level further action would be taken. We have color coded the wires for easy understanding of how to connect the soil sensor with the Arduino ESP32.

Connect the Level Shifter to the Arduino ESP32 and Relay Module. The Level Shifter is connected to the Arduino ESP32 and Relay module. The logic level Shifter is convert the voltage from 3.3v to 5v. The Arduino ESP32 board is consumption output power from one side in 3.3v

III. EXPERIMENTATION

This project consists of two sections: the external sensor unit, and the inbuilt processing unit. In the external sensor unit, the basic requirement of sensing the moisture level, flow rate and provide input to the IC. The major advantage of the system derives from the fact that it is not limited. A lower limit level of the soil moisture level will be present in the program code for the flow rate below which

and another side is 5v. The level shifter is used for relay module. Because the relay module is reach the 5v to turn ON and OFF. The code to Be Upload to Arduino ESP32. The code required is uploaded to Arduino using IDESetup the Flow Meter. The point to note here is that there are 6 analog pins on Arduino that can be used to sense soil moisture. Flow the water through the pipe and to be calculated. Connect the Pump to the Flow Meter. The pump is connected to the flow meter main line which will pump water to the line which will supply the water to the plants for watering.

the pump is not allowed to turn on thus preventing the dry run. The controller will check the reading of that moisture sensor mounted in the farm land or garden and then decide the on the operating status of the pump.

There will be two present values for lower and upper level for consideration of the moisture level low or high status. If the reading of the moisture level is found to be below the lower preset

value and the pump is allowed to turn on via the relay module and it will continue to irrigate the water until the level of water reaches the moisture level of preset value. After the moisture level gets irrigation to the high upper level preset value the system will not trigger the pump until & unless the level of water in the farm land reaches the lower level preset value again.

The wiring is properly connected with sensors and power source. The soil moisture sensor and water flow sensor are connected with Arduino ESP32.

SOURCE CODE FOR THE MICROPROCESSOR:

```
#ifndef ESP32
#include <WiFi.h>
#else
#include <ESP8266WiFi.h>
#endif
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <ArduinoJson.h>
const char* ssid = "Redmi";
const char* password = "1234567890";
#define BOTtoken "5334624319:AAEM_ts3NgMk4xHk-nr32ZenXypUzG7ZgGY"
#define CHAT_ID "1696232899"
WiFiClientSecure client;
UniversalTelegramBot bot(BOTtoken, client);
int botRequestDelay = 1000;
unsigned long lastTimeBotRan;
volatile int flow_frequency;
unsigned char flowsensor = 4;
float vol = 0.0, l_minute;
unsigned long currentTime;
unsigned long loopTime;
const int moistureSensor = 34;
int moistureLevel;
int x = 0;
const int pump = 32;
bool mlow = false;
bool pumpState = HIGH;
void flow() {
  flow_frequency++;
}
String getReadings() {
  String message = "Rate: " + String(l_minute) + " l/min \n";
  message += "Consumption: " + String(vol) + " L \n";
  return message;
}
bool ledState = LOW;
void handleNewMessages (int numNewMessages) {
```

Telegram Bot is used to show the water flow rate and volume.

When the moisture level is low, the relay coil gets energized and turns on the motor. Similarly, if the moisture level is high the relay coil reenergized and turns off the motor. At the same time if the water flow rate is very flow it will turn off the motor.

And using the telegram bot to control the flow of water automatically and manually through your mobile phone.

```
Serial.println ("handleNewMessages");
Serial.println (String(numNewMessages));
for (int i=0; i<numNewMessages; i++)
{
  String chat_id = String (bot.messages[i].chat_id);
  if (chat_id != CHAT_ID)
  {
    bot.sendMessage (chat_id, "Unauthorized user", "");
    continue;
  }
  String text = bot.messages[i].text;
  Serial.println(text);
  String from_name = bot.messages[i].from_name;
  if (text == "/start")
  {
    String welcome = "Welcome, " + from_name + ".\n";
    welcome += "Use the following commands to control your outputs.\n\n";
    welcome += "/moisture_Level to analysis moisture level \n";
    bot.sendMessage (chat_id, welcome, "");
  }
  if (text == "/moisture_Level") {
    if (x == 0)
    {
      bot.sendMessage (chat_id, "Moisture Level is High", "");
    }
    else
    {
      bot.sendMessage (chat_id, "Moisture Level is Low you can turn on Pump by the /Manually or /Automatically", "");
    }
  }
  if (text == "/Automatically")
  {
    mlow = true;
    bot.sendMessage (chat_id, "System is gone for the automatic state pump will be automatically off when the moisture level reaches the maximum level To check the flow rate and volume click /readings, "");
  }
}
```

```
    if (text == "/Manually")
    {
bot.sendMessage (chat_id, "To turn on and of pump
click /turnOnPump and /turnOffPump", "");
    }
    if (text == "/turnOnPump")
    {
bot.sendMessage (chat_id, "Pump is turned on To
check the flow rate and volume click /readings", "");
pumpState = LOW;
digitalWrite (pump,pumpState);
    }

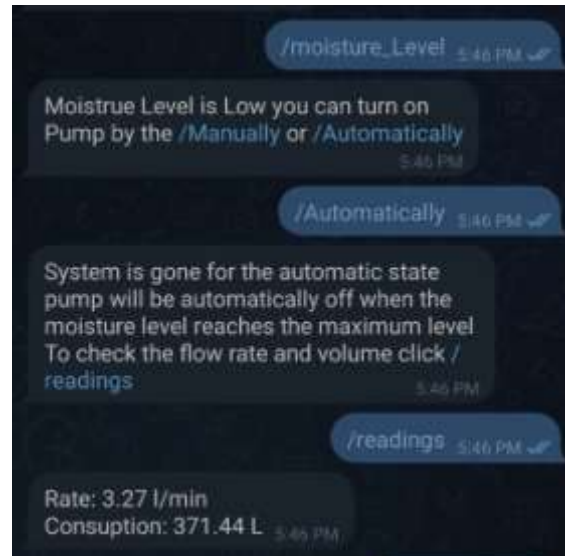
    if (text == "/turnOffPump")
    {
bot.sendMessage (chat_id, "Pump is turned off", "");
pumpState=HIGH;
digitalWrite (pump,pumpState);
    }
    if (text == "/readings")
    {
        String readings = getReadings();
bot.sendMessage (chat_id, readings, "");
    }
    }
}

void setup()
{
pinMode (flowsensor, INPUT);
digitalWrite (flowsensor, HIGH);
attachInterrupt (digitalPinToInterrupt(flowsensor),
flow, RISING);
pinMode (moistureSensor, INPUT);
pinMode (pump,OUTPUT);
digitalWrite (pump,pumpState);
Serial.begin (115200);
#ifdef ESP8266
client.setInsecure();
#endif
WiFi.mode (WIFI_STA);
WiFi.begin (ssid, password);
client.setCACert
(TELEGRAM_CERTIFICATE_ROOT);
while (WiFi.status() != WL_CONNECTED)
{
    delay (1000);
Serial.println ("Connecting to WiFi..");
}
Serial.println (WiFi.localIP());

bot.sendMessage (CHAT_ID, "Bot started up", "");
}

void loop()
{
    if (millis() >lastTimeBotRan + botRequestDelay)
    {
        int numNewMessages = bot.getUpdates
(bot.last_message_received + 1);
        if (flow_frequency !=0)
        {
            l_minute = (flow_frequency / 7.5);
Serial.println ("Rate: ");
Serial.println (l_minute);
Serial.println (" L/M");
l_minute = l_minute/60;
            vol = vol +l_minute;
Serial.println ("Vol:");
Serial.println (vol);
Serial.print (" L");
            flow_frequency = 0;
        }
        x = digitalRead (moistureSensor);
Serial.println ("moisture level");
Serial.print (x);
        if (mlow == true)
        {
            pumpState = LOW;
digitalWrite (pump,pumpState);
Serial.println ("ON");
            if (x == 0)
            {
                pumpState = HIGH;
digitalWrite (pump,pumpState);
mlow = false;
Serial.println ("OFF");
            }
        }
        while (numNewMessages)
        {
            Serial.println ("got response");
handleNewMessages (numNewMessages);
numNewMessages = bot.getUpdates
(bot.last_message_received + 1);
        }
        lastTimeBotRan = millis();
    }
}
```


IV. OBESERVATIONS OF RESULT FROM TELEGRAM



TELEGRAM SCREEN SHOT IMAGES

V. CONCLUSION

Thus the "Crop Production Irrigation System with Flow Analysis", has been designed and tested successfully. It has been developed by integrating all the features of all the hardware components used. A system to monitor moisture levels in the soil and calculate the flow rate of water flow was designed. The system was used to switch on/off the watering system/pump according to set soil moisture levels. The moisture content of the soil is continuously measured by the sensor. Its value and the status of motor i.e. ON or OFF condition of motor is displayed in the android telegram bot. If there is an enough moisture in the soil i.e. there is no need to irrigate the field then the

motor is not switched ON but if the moisture content is very less i.e. there is a need of irrigation then the motor is switched on automatically and display the flow rate of water passing through the flow meter then the flow of water rate and volume can be calculated. After the field attains the required moisture content, then the motor is switched off automatically.

To switch between the control and the irrigation systems a relay switching circuit was used. The soil moisture sensor to measure the water moisture in the farm land. The primary applications for this project are for farmers and gardeners who do not have enough time to water their crops/plant. It also covers those farmers who are wasteful of

water during irrigation. As water supplies become scarce and polluted there is a need to irrigate more efficiently in order to minimize water use and chemical leaching. Recent advances in soil water sensing make the commercial use of this technology possible to automate irrigation management for vegetable production.

However, research indicates that different sensors type perform under all conditions with no negative impact on crop yields with reductions in water use range as high as 70% compared to traditional practices. And we find out the more advantages from this project due to the all the people's from the world. In future water is most important one, because now the day's ground water is decreased. So our project used to maintain the water in most important purpose only and its usage is continued by a future.

SOME OF THE ADVANTAGES FROM THE ABOVE RESULTS

- Efficient utilization of electrical energy.
- Device /Pump Protection by preventing the dry Run of pump.
- Non wastage of water.
- And used telegram bot to control the pump ON and OFF & to calculate the reading of water flow.

REFERENCES

- [1]. About Water flow Meter
<https://www.seeedstudio.com/blog/2020/05/11/how-to-use-waterflow-sensor-with-arduino/>
- [2]. <https://randomnerdtutorials.com/guide-for-relay-module-witharduino/>
- [3]. <http://ageconsearch.umn.edu/bitstream/279744/1/02010009.pdf>
- [4]. <https://osoyoo.com/2017/07/27/arduino-lesson-pir-motion-sensor/>
- [5]. <https://www.arduino.cc/en/Tutorial/LibraryExamples>
- [6]. <https://diyusthad.com/2021/10/telegram-bot-with-esp32.html>
<https://forum.arduino.cc/t/how-to-split-a-string-with-space-and-store-the-items-in-array/888813>
- [7]. <https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/>
- [8]. <https://iotdesignpro.com/projects/telegram-bot-with-esp32-control-gpio-pins-through-telegram-chat>
<https://randomnerdtutorials.com/telegram-control-esp32-esp8266-nodemcu-outputs/>
- [9]. <http://randomnerdtutorials.com/telegram-esp32-motion-detection-arduino/>
- [10]. <https://randomnerdtutorials.com/telegram-control-esp32-esp8266-nodemcu-outputs/>
- [11]. <https://www.hackster.io/g4x/avocado-watering-monitor-with-telegram-bot-6d56d7>
- [12]. <https://iotdesignpro.com/projects/telegram-bot-with-esp32-control-gpio-pins-through-telegram-chat>
<https://randomnerdtutorials.com/telegram-esp32-motion-detection-arduino/>
- [13]. <https://randomnerdtutorials.com/telegram-control-esp32-esp8266-nodemcu-outputs/>