

Automated Irrigation System Using Wireless Sensor Network (WSN)

Akshata. S. Jagatap, Asso. Prof: S. R. Dhotre

Abstract— Adopting an optimized Irrigation system has becomes necessity due to the lack of the world water resources. The system has a soil moisture sensor, temperature and relative humidity sensor. The project focuses on a smart irrigation system which is cost effective. The objective of this project is to control the water supply to each plant automatically depending on values of soil moisture, temperature and relative humidity. Mechanism is done in such a way that soil moisture sensors are dipped in to the soil and Temperature and Humidity sensor is kept open in the land. Automatic irrigation scheduling consistently has shown to be valuable in water use efficiency with respect to manual irrigation based on direct soil water measurements. The aim of the implementation is to demonstrate that the automatic irrigation can be used to reduce water use.

Index Terms— Automation, Internet, irrigation, measurement, water resources, wireless sensor networks (WSNs).

I. INTRODUCTION

Agriculture plays the major role in economics and survival of the peoples in India. India is basically the agriculture based country. Agriculture depends on fresh water resource it is important to manage and use the water. Automated irrigation system is implemented to optimize the water use in agriculture. Irrigation is the artificial application of water to the land. The Automated Irrigation System consists of a distributed wireless sensor network with sensors (client). Sensor networks are used for collecting the data, storing and sharing the sensed data with the Raspberry Pi by the internet. The Raspberry Pi act as a processing unit (server), it is considered as a heart of the system. By this automated irrigation system we can save 90% of water while comparing it with the other irrigation methods used in olden days. In Gardens, Green house and official premises it is much in use. It is cost effective, we need to purchase sensors, Raspberry and need to adjust our own internet in the land. We need to set different setup for different crops.

A. Objective

The objective of the project is to implement the automated irrigation system. Here the WLN (wireless LAN network) is used to control the system wirelessly while the PHP scripting programming language is used for automation purpose to the Raspberry Pi (RPi). Monitoring system is based on wireless communication technology that is developed to control remotely, which depends on the measurement of soil moisture content, temperature and relative humidity level in the land.

Manuscript received June 05, 2018

Akshata. S. Jagatap, Computer Science and Engineering, Belagavi, India

Asso.Prof. S. R. Dhotre, Computer Science and Engineering, Belagavi, India

II. SYSTEM ARCHITECTURE

System consists of a distributed architecture of Wireless Sensor Unit (WSU) and wireless information unit (WIU). The system block representation is shown in Fig.1. The sensing unit and the information units are connected to the Wireless Local Area network by the router. The router is connected to the internet and provides remote access to the system. The server is designed using a Raspberry Pi (RPi) microcomputer board which is smaller and energy efficient while delivering the computing performance and features like Linux OS and wireless connectivity.

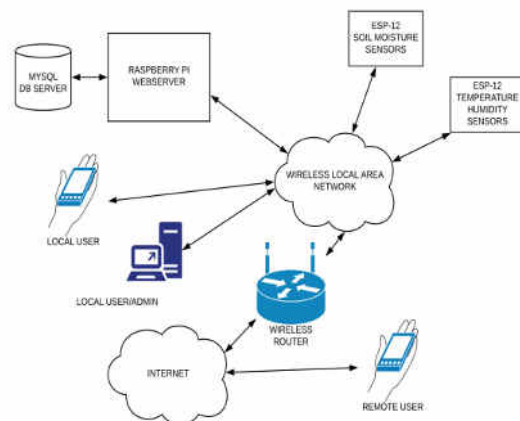


Fig.1 System block Representation

The sensors are designed with an ESP-12E Wi-Fi module which is connected to soil moisture, temperature and humidity sensors. The power is provided by the Lithium ion battery. In land we can power it by the solar panels. The server (Raspberry pi) is configured by the terminal and the program for the automation is written in to PHP scripting language. Python scripts are used for the ON/OFF control mechanism. Pump is connected to the server's (Raspberry Pi) GPIO (General purpose input and output) pins.

III. SYSTEM DESIGN

Automated irrigation system mainly consists of two components: Wireless Sensor Unit (WSU) and Wireless information Unit (WIU). The system design for the irrigation system is shown in fig.2. Wireless Sensor Unit (WSU) consists of two sensors soil moisture sensor, temperature and relative humidity sensor. The server is designed using a Raspberry Pi (RPi) microcomputer board. Here water pump is connected to the Raspberry pi through the RPi's GPIO pins. Here you need to configure the Raspberry Pi and this system needs different setup for different crops based on their nature of water needed for the crop.

Automated Irrigation System Using Wireless Sensor Network (WSN)

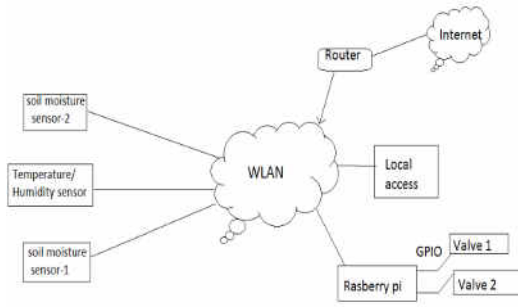


Fig.2 System design

IV. COMPONENTS USED TO IMPLEMENT

A. Soil moisture sensor

VH400 probe is used for detecting the soil moisture. It is of low power consumption (<7 mA) and low cost. The module consists of two probes one as a positive and other as negative probe and sensor board. The sensor will detect the soil moisture surrounding it. Sensor is shown in Fig.3.



Fig.3 Soil moisture sensor

B. Temperature sensor

The temperature is measured by the temperature sensor (DHT12). DHT12 works at 3.3 volts is shown in fig.4. DHT12 is a Digital temperature and humidity sensor. It is of small size, low power consumption and it is suitable for variety of applications.



Fig.4 Temperature sensor (DHT12)

C. Wi-Fi Modem ESP-12E (ESP8266)

ESP8266 Wi-Fi Module is with an integrated TCP-IP (Transfer control protocol) protocol stack. It will access any microcontroller to your Wi-Fi network. The module is shown in fig.5 and in Fig.6 the ESP-12E development board is given.



Fig.5 ESP-12E Module

D. Raspberry Pi (RPi)

The Raspberry Pi (RPi) is low cost, credit-card sized microcomputer board that will connect to the computer

monitor or TV, and it uses the keyboard and mouse. The Raspberry Pi is shown in fig.6. If you are connecting it by the Ethernet cable, take a cable connect to it and to the monitor. The RPi screen will automatically show its screen on the monitor.



Fig.6 Raspberry Pi

E. Relay (5v)

The Relay switch it is electrically operable is shown in Fig.7. Here it is used for the ON/OFF mechanism of the pump. Relay control the electrical circuit by opening and closing the contacts in other circuit. Relays are used to switch the smaller currents in a control circuit. The pin description of Relay is shown in fig.8. And description is given in Table.1.



Fig.7 Relay (5v)

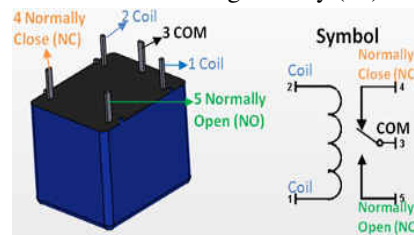


Fig.8 Relay pin description

Table.1. Pin description of Relay

Pin no.	Pin name	Description
1	Coil End 1	Used to trigger the relay (ON/OFF). Normally one end connected to 5v and the other end is connected to the ground.
2	Coil End 2	Used to trigger (ON/OFF) the relay. Normally one end it is connected to 5v and the other end are connected to the ground.
3	Normally Close (NC)	The other end of the load is either connected to Normally Open (NO) or Normally Close (NC). If connected to NC then the load remains connected before trigger.
4	Common (COM)	Common is connected to one end of the load and it is to be controlled.
5	Normally Open	Other end of the load is either connected to NO or NC. If

	(NO)	connected to NO the load remains disconnected before the trigger.
--	------	---

V. BLOCK DIAGRAMS

A. System hardware Architecture

The server (Raspberry Pi) is designed using a Raspberry Pi computer board. It requires 5v 3A DC current which is provided here by the lithium ion battery. The operating system (OS) and the application software's are placed in the micro SD card connected to the Raspberry Pi (Rpi) computer. The Raspberry Pi GPIO (General-purpose input/output) port is connected to the solid state relay driver board which is again connected to the water pump. The overall hardware architecture is shown in Fig.9.

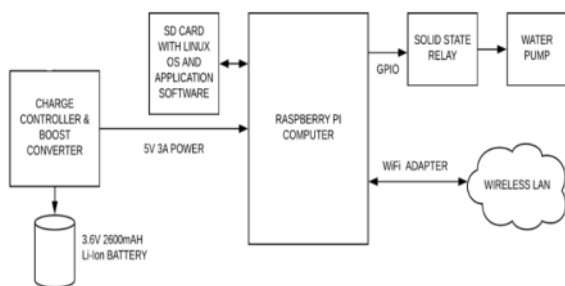


Fig.9 System hardware architecture

B. Soil moisture sensor

The soil moisture sensor is designed using the ESP-12E IOT (Internet of Things) development board. And it measures the soil-moisture by electrical resistance, it measures the electrical charge conducted by the soil and water conducts electricity. The sensor will measure the amount of electrical charge conducted by the soil. i.e. if the soil moisture is more the electrical charge conducted by the soil is more and vice-versa. The soil moisture sensor hardware architecture is shown in Fig.10.

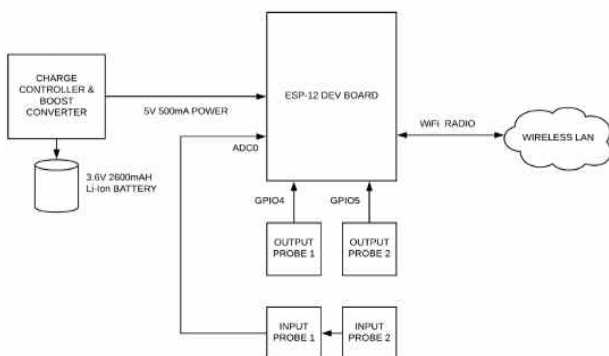


Fig.10 Soil Moisture sensor hardware architecture

The measured values from probe 1 & probe 2 are averaged and then converted in to the percentage value i.e. 0 to 100%. This value is posted to the server end script for the further display and processing. The modules are powered by the Lithium Ion battery at 3.6 V 2600 mA voltage and current. Once battery is fully charged the modules can work approximately around 36 hours.

C. Temperature and relative humidity sensor

The temperature and humidity sensor measures environmental temperature and relative humidity using the digital sensor (DHT12). The sensor is interfaced with a digital GPIO line of the ESP-12E module and Arduino DHT12 libraries are used to measure temperature and humidity.

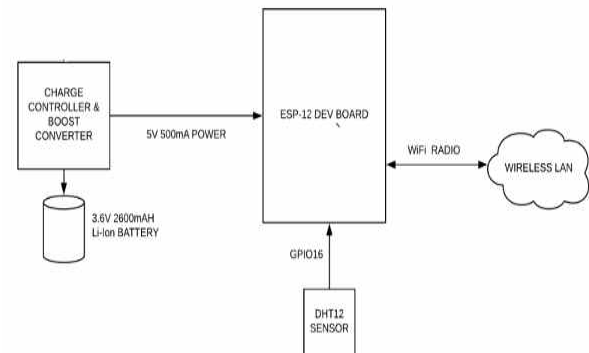


Fig.11 Temperature sensor hardware architecture

The complete hardware arrangement is shown in fig 11. The modules are powered by the same lithium Ion battery. Once fully charged it also works approximately around 36 hours.

VI. IMPLEMENTATION

A. System with WLN (Wi-Fi)

The system has wireless distributed architecture. It uses wireless network connectivity for its functionality. The sensors measure soil moisture, temperature and relative humidity data and transfer the same to the Raspberry Pi which stores these data into the database system. The working is shown in Fig.12. The user interface will display the most recent values received by the system.

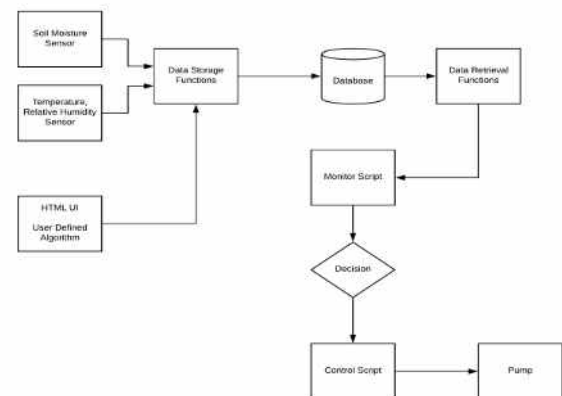


Fig.12 Working

Here the soil moisture sensor, temperature and relative humidity sensors should be programmed in C++ language. To upload the program for the sensors connect the sensor board to the laptop through the USB port and by the Arduino IDE software upload the code to the sensors.

B. Raspberry pi configuration

The sensors will sense data and they will share the same data with the Raspberry pi. To configure the Rspberry first we need to download the putty and the Linux operating system (OS) for the Raspberry and place the Operating system in to the micro SD card placed in the Raspberry Pi. The software development can be divided into the following parts and the

Raspberry Pi computer requires setting the operating system, configuration and installation of many software programs for the development and the execution of system user interface, monitoring and the controlling programs.

- a. Installation of Raspberry pi operating system
- b. Raspberry pi operating system configuration
- c. Installing and configuring the apache server
- d. Installing and configuring PHP5
- e. Installing and configuring MySQL Server
- f. Installing and configuring PhpMyAdmin
- g. Configure the apache to work with PhpMyAdmin
- h. Installing and configuring Pure-FTPd
- i. Configuring raspberry pi computer system to auto start the programs
- j. Installing and configuring FileZilla FTP Client on computer for uploading the web Pages to the raspberry pi computer
- k. Uploading programs to ESP-12E module using Arduino IDE
 - a. Raspberry Pi operating system(OS) installation and configuration

The Raspberry pi B+ to be loaded and configured, the installation and configuration steps involved to setup the board completely to run our application. The operating system for RPi is available in the format of zipped image. This image file must be written to the Class 8 /10 Micro SD card with the minimum size of at least 4GB, by the windows utility called “WIN32 Disk Imager” is shown in Fig.13.

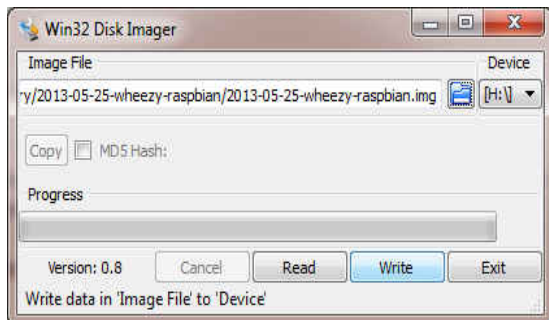


Fig.13 Win32 disk image

b. Raspberry Pi operating system Configuration

After installing the Raspberry Pi operating system you need to configure it. To do its configuration the following steps are needed. To do this Expand the file system, then change the Password, Boot to GUI (Graphical User Interface) is enabled, Under internationalisation options the language and Keyboard layout is set to U.S. English, and time zone is set to Kolkata GMT +5.30, System overclock to 1000MHz for the faster and optimized performance, Enabling the SSH for remote system management, Rebooting and exiting the system when prompted by “raspi-config”. It is shown in the Fig.14.

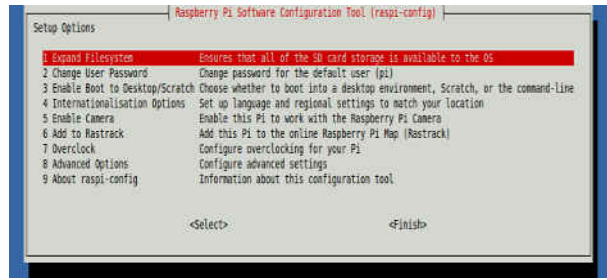


Fig.14 Raspberry Pi configuration

After restarting the system will boots up and runs the default window manager LXDE which is the Light weight X server Desktop Environment After this the system is rebooted and next it is performed with the update using following commands in the terminal. The system will take around 2 hours for performing the updates.

- sudo apt-get install upgrade
- sudo rpi-update

c. Installing and configuring apache server

Apache is a popular web application server and it can be installed on the Raspberry Pi computer board to serve the web pages. It can serve HTML (Hyper Text Markup Language) files over the HTTP (Hyper Text Transfer Protocol) protocol. And the additional modules will serve the dynamic web pages by using PHP scripting language. For installing the Apache 2 package the following command is entered in to the Terminal.

- sudo apt-get install apache2 -y

By default the Apache server puts a test HTML file in the web folder. This default web page is served when we browse the server by <http://192.168.1.2> (the RPi’s IP address). Browsing to the default web page either on the Raspberry Pi or from another computer on the network then the following page can be seen.



Fig.15 Apache server default page

This default web page is in HTML format it is in the file system at the location: /var/www/index.html

d. Installing and Configuring PHP5

Now we are running Apache Server on the system. Here we need to install the PHP5 for the server side scripting this is done by entering the following command in to the Terminal.

- sudo apt-get install php5 libapache2-mod-php5 -y

After installation we need to verify the PHP support is enabled for the Apache server or not? This is done by removing the default web page. To remove the web page enters the following command in to the terminal.

- sudo rm /var/www/index.html

And make it index.php file by opening the default text editor.

- sudo nano /var/www/index.php

The following context is added to the index.php file for testing the available php functionality

<?php phpinfo(); ?>

Now we can access the default page of RPi by typing its IP address (192.168.1.2) in the browser and the following page is visible, showing installed PHP5 scripting capabilities and the support, is shown in Fig.16.

e. Installing and configuring MySQL Server

The database engine MySQL server is installed by entering following commands in the terminal.

- sudo apt-getinstall mysql-server

During the installation a prompt will appear, it is asking for the user to enter a password for the MySQL root user and confirm the same, it is shown in Fig.17.

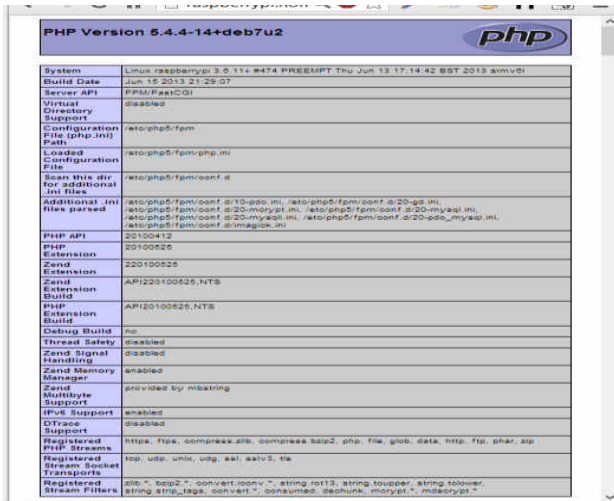


Fig.16 PHP-INFO testing PHP support

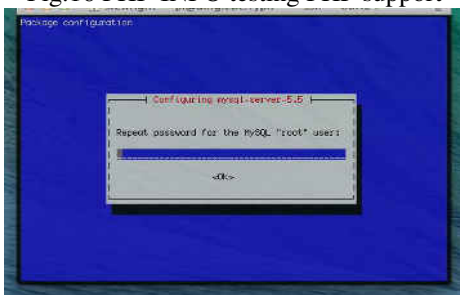


Fig.17 MySQL root password prompts

f. Installing and Configuring PhpMyAdmin

PhpMyAdmin is the web based MySQL Database management tool and it is installed by entering the following command in the terminal.

- sudo apt-get install phpmyadmin

While installing the PhpMyAdmin it will prompt for selecting the active server on the system. Here the active server on the system is Apache2 it is selected so that PhpMyAdmin can work with the Apache server, is shown in Fig.18.

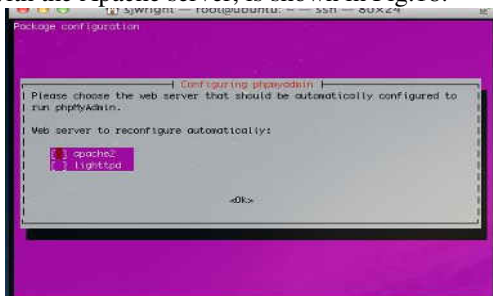


Fig.18 PhpMyAdmin configuration

Again PhpMyAdmin will prompt with the dbconfig screen to attach with the existing databases, is shown in Fig.19.

g. Configure Apache server to work with PhpMyAdmin

Here we need to alter the Apache server configuration which we have done earlier to access with the PhpMyAdmin. To configure this we need to enter the following command in the Terminal.

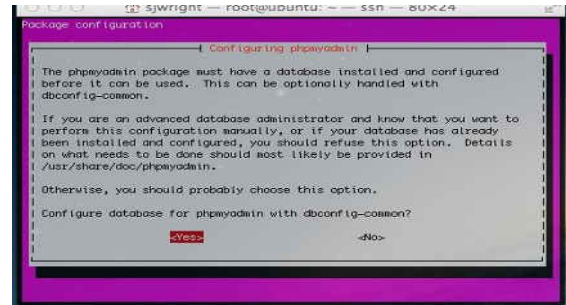


Fig.19 PhpMyAdmin dbconfig screen to enter the following command in the Terminal.

- sudo nano /etc/apache2/sites-enabled/000-default.conf

Here nano is used to open the configuration file, for this the following line is added at the end of file.

- include /etc/phpmyadmin/apache.conf

After this the Apache server need to be restarted before it starts working with the PhpMyAdmin and it is done by entering the following command in the Terminal.

- sudo service apache2 restart

After restarting Apache Server on the system the PhpMyAdmin can be accessed by typing the RPi's IP address followed by "phpmyadmin/" i.e. <http://192.168.1.2/phpmyadmin/>. This will result browser is to be loaded with the PhpMyadmin authentication page and for the first logon the user ID is: root and password is the same as configured in above installation steps. But here user id is: root and password is p@ssw0rd. The authentication page is shown in Fig.20.



Fig.20 PhpMyAdmin authentication page

After login to PhpMyAdmin we need to create a new user called "pi" password is of your choice and it grants the user with all the database operation privileges.

h. Installing and configuring Pure-FTPD

Pure FTPD is a FTP server it is for uploading and downloading files from the apache server and from the any remote machine authenticated using the login credentials. The pure FTPD server is installed by entering the following command in the Terminal.

- sudo apt-get install pure-ftpd

Here we need to create the new user group named as an ftpgroup and a new user named ftpuser for the FTP users, and make sure that this "user" has no home directory.

- groupadd ftpgroup

- `useradd ftpuser -g ftpgroup -s /bin/nologin -d /dev/null`

To provide access to the default site folder we need to enter following commands

- `sudo chown -R ftpuser: ftpgroup /var/www/-m`

Password of the virtual user is required after this command line is entered in to the Terminal. And next, set up a virtual user database by typing the following command in the Terminal.

- `sudo pure-pw mkdb`

Define the authentication method by making a link to the file `/etc/pure-ftpd/conf/PureDB`.

- `ln -s /etc/pure-ftpd/conf/PureDB /etc/pure-ftpd.auth/60puredb`

Next we restart the program by entering

- `sudo pure-ftpd restart`

After completion of these steps we have a FTP with user name as "upload" and password as configured in above steps (here it is p@ssw0rd). Further we can transfer file from a host computer by using a FTTP client like WSFTP by pointing to the RPi IP address i.e. "192.168.1.2" with username "upload and password" the FTP client will then show the contents of Apache Servers default site folder "`/var/www/html/`".

- i. Configuring Raspberry Pi computer system to auto start programs

Here after these steps you need to configure the Raspberry Pi to the auto start programs. The PHP monitor script for monitoring and control located at "`/var/www/html/vendor/php`", to achieve this we need Linux crontab program scheduler.

To add a new cronjob to crontab the following command to be entered in the terminal.

- `sudo crontab -e`

After execution of this command the crontab prompts for default text editor and nano is selected as default editor for adding the cronjob here. The cronjob can be added by entering following line.

`@reboot curl http://127.0.0.1/vendor/php/monitor.php`

This will result in monitor script `monitor.php` located at `/var/www/html/ vendor/php/monitor.php` are executed every minute. There are some basic configuration steps to set the steps for setting up a web server capable of handling PHP server side scripting and database connectivity, with execution of a PHP script called `monitor.php` which will monitor and control the watering pump. After this configuration we need to create web pages with HTML, PHP and JavaScript for user interface.

- j. Installing and configuring FileZilla FTP Client on computer for uploading web pages to the Raspberry Pi Computer

Filezilla is a FTP client which can be used on Windows, mac or Linux operating system to connect to the remote FTP server for upload and downloading the files. After installation on the target operating system (OS) the program can be executed by double clicking on its icon.

Configuring Connection:

The computer on which the Filezilla is installed and Raspberry Pi should be connected on the same Local Area

Network before beginning with the Connection configuration. It is shown in Fig.21.

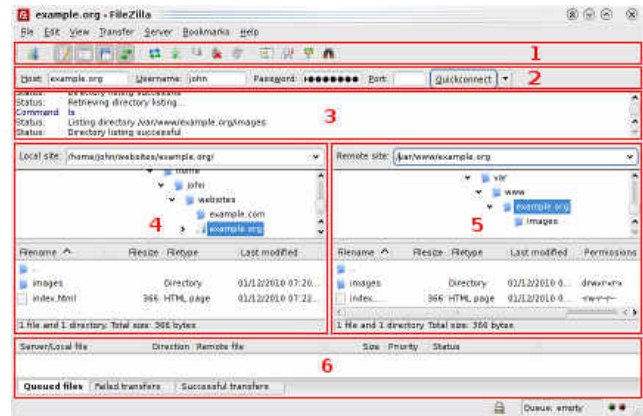


Fig.21 Filezilla program window

The program window consists of following sections.

Toolbar, Quick connect bar, Message log, Local Fils Pane, Remote Files Pane, transfer Queue

We will use Quickconnect bar for establishing the connection.

Fig.22 Filezilla Quick connect bar

Here enter the hostname (Raspberry Pi Computer IP Address)



into the quickconnect bar. Host field, enter the user name (here in this project username is upload and password is raspberry). The Port: field is left empty to the default port (21). Now by clicking on the Quickconnect will connect to the FTP (File transfer protocol) server (Raspberry), the status of the connection can be viewed in messafe log pane and successful connection will show the list of files on the Raspberry Pi Computer in remote files pane, further the local files pane is navigated to the project source codes directory and then the files can be uploaded by double clicking on item in local files pane, to the Raspberry Pi Computer. The transfer queue will display files being uploaded and also the status of upload.

- k. Uploading programs to ESP-12E module using Arduino IDE

Aurduino IDE is the simplest solution for programming the ESP-12E module. To install and configure Arduino IDE to develop and program for ESP-12E module the following steps are used.

1. Download the Aurduino IDE and install it.
2. Go to FILE then to Preferences and add the following link. http://arduino.esp8266.com/stable/package_esp8266com_in_dex.json in the Arduino boards manager URLS.
3. Go to tools ->Board manager.
4. Now we have the esp8266 as an option since we added it to additional board. Next we need to select it and install.
5. Wait for its installation.
6. Now, we have the configured ESP8266 board. We need choose the board from list to upload our code.
7. Finally we need to select COM port where the board is connected and then the board can be uploaded with the target code from the sketch menu using upload option.

VII. RESULTS

The following Fig.23 shows the complete experimental set up. Here it has the two sensors soil moisture sensor and temperature and relative humidity sensor. They are accessing with the Raspberry Pi wirelessly. And the server (Raspberry pi) is connected with the relay board and next to the water pump. Relay board is used for the ON/OFF mechanism for the pump.



Fig.23 Experimental setup

First start the raspberry (server) and sensors and do the internet connectivity to the server. Browse the server then you will get a login page as shown in Fig.24.

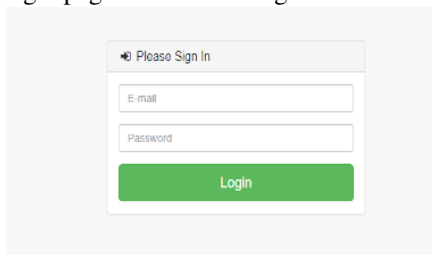


Fig.24 Login page

Login to the page by entering the mail ID as admin@irrigation.local and password as p@ssw0rd. After login you can get a dashboard with previous values as shown in figure.25.

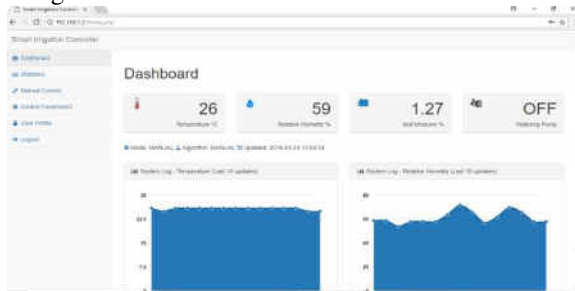


Fig.25 Dashboard

In the dashboard you have statistics, manual control, control parameter, User profile and logout. In statistics you can see the previous soil moisture, temperature and relative Humidity. The Following Fig.26 shows the previous temperature value 26 at time 10.50.

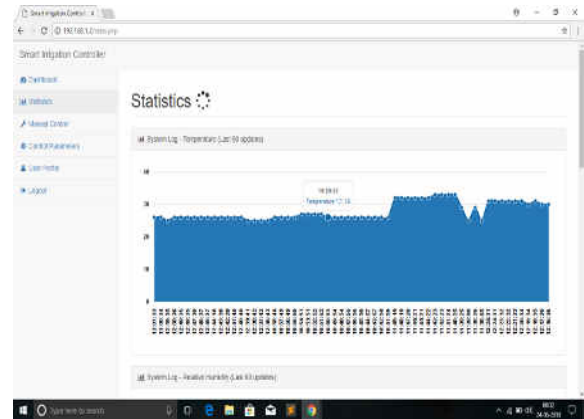


Fig.26 Temperature

Like this we can see the soil moisture and Humidity.

And the following Fig.27 shows the pump status ON. Here in this figure 1 indicates the pump is ON at 11:5. And in the same way it can show the off status of the pump by indicating it with zero (0).

Next part is the manual control. Here by this you can start/stop the pump. Herein the Fig.28 it has two states by red and green colour. Green shows pump ON and red is for OFF. It is shown in Fig.28.

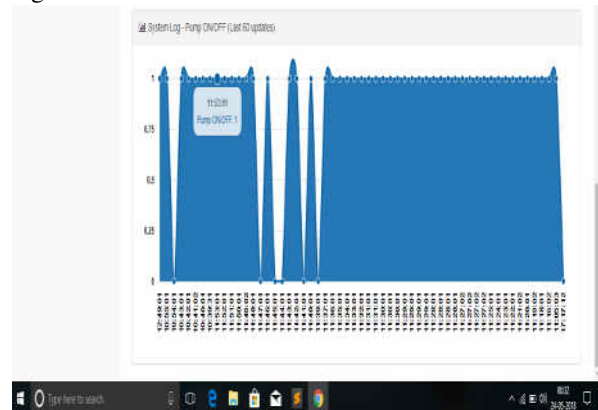


Fig.27 Pump status ON (1)

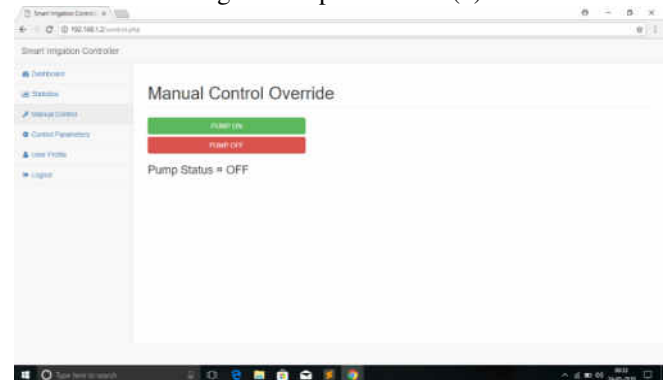


Fig.28 manual control

Following Fig.29 shows the status of pump is ON. Also we can see the off status here.

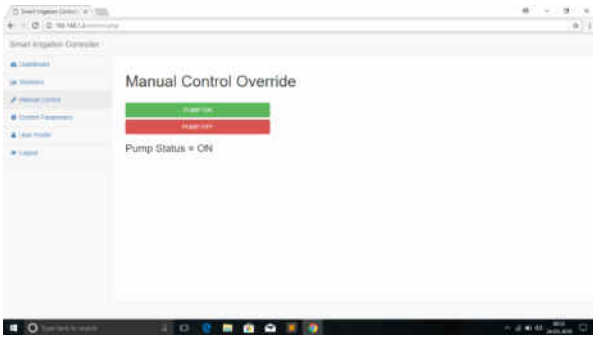


Fig.29 Pump ON

After this go to the control parameters, here you can automatically irrigate. Here some threshold values are set to the soil moisture, temperature and relative humidity. Pump works automatically based on the parameter values. If the value of these parameters exceeds the threshold values then the pump will OFF automatically. In the operation mode set to the auto mode. And set the algorithms for tests, then based on the values of parameters it works. The Fig.30 shows the Window to set the method and the test. Set it to manual then you can operate it manually. If you set auto then by choosing the tests in algorithm you can irrigate the field based on the threshold values.

The Fig.31 shows the window after setting the test. Also you can set it to Auto, here you can automatically Irrigate depending on the parameter values.

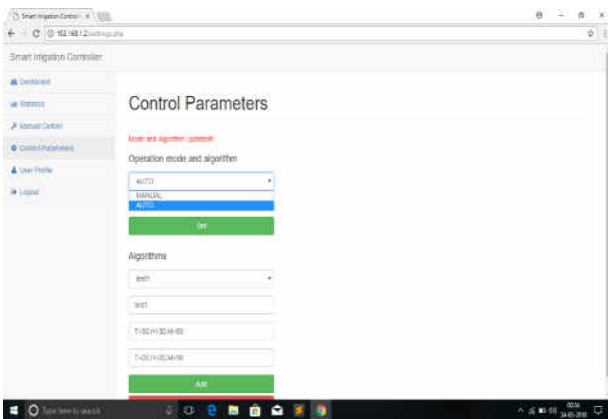


Fig.30 Control parameters

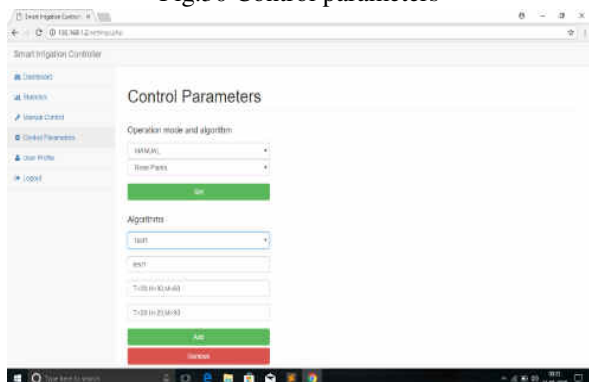


Fig.31 Window after setting the test

You may also change your password is shown in Fig.32. And in logout option you can logout.

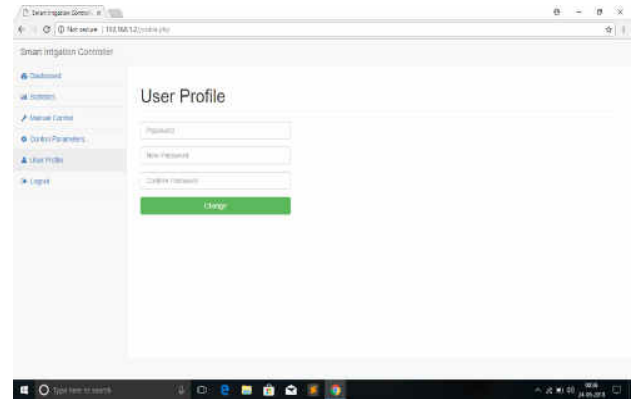


Fig.32 User profile

VIII. ADVANTAGES, DISADVANTAGES AND FUTURE SCOPE

A. Advantages

1. Remotely accessible
2. Reduce the man power and saves the time
3. Save water
4. Much land will be irrigated

B. Disadvantages

1. Works only through Internet
2. It is cost effective

C. Future scope

1. The performance of the system can be further improved in terms of speed and capacity.
2. In future, the automated irrigation system may extend not only for irrigating the field with water but also for deciding to spray the appropriate chemicals for proper growth.

IX. CONCLUSION

Automated Irrigation method is much better than the traditional Irrigation methods. The system should work automatically which reduces the man power, wastage of water. We can irrigate much land and we will get best product in less water. The irrigation system can be adjusted separately for different crop for the purpose of their specific crop needs and requires minimum maintenance cost. In green hose method and gardens it is much in use. Implementation done by this wireless LAN is faster than the previous Zigbee technique.

REFERENCES

1. Joaquín Gutiérrez, Juan Francisco Villa-Medina, Alejandra Nieto-Garibay, and Miguel Angel Porta- Gandara, "Automated Irrigation System Using a Wireless Sensor Network and GPRS Module," IEEE Transactions on Instrumentation and Measurement, vol. 63, no. 1, January 2014.
2. X. Wang, W. Yang, A. Wheaton, N. Cooley, and B. Moran, "Efficient registration of optical and IR images for automatic plant water stress assessment," Comput. Electron. Agricult., vol. 74, no. 2, pp. 230–237, Nov. 2010.
3. Y. Erdem, L. Arin, T. Erdem, S. Polat, M. Deveci, H. Okursoy, and H. T. Gültas, "Crop water stress index for assessing irrigation scheduling of drip irrigated broccoli

- (*Brassica oleracea* L. var. *italica*),” *Agricult. Water Manag.*, vol. 98, no. 1, pp. 148–156, Dec. 2010.
4. G. Yuan, Y. Luo, X. Sun, and D. Tang, “Evaluation of a crop water stress index for detecting water stress in winter wheat in the North China Plain,” *Agricult. Water Manag.*, vol. 64, no. 1, pp. 29–40, Jan. 2004.
 5. S. A. O’Shaughnessy and S. R. Evett, “Canopy temperature based system effectively schedules and controls center pivot irrigation of cotton,” *Agricult. Water Manag.*, vol. 97, no. 9, pp. 1310–1316, Apr. 2010.
 6. P. Corke, T. Wark, R. Jurdak, H. Wen, P. Valencia, and D. Moore, “Environmental wireless sensor networks,” *Proc. IEEE*, vol. 98, no. 11, pp. 1903–1917, Nov. 2010.
 7. R. G. Allen, L. S. Pereira, D. Raes, and M. Smith, *Crop Evapotranspiration-Guidelines for Computing Crop Water Requirements—FAO Irrigation and Drainage Paper 56*. Rome, Italy: FAO, 1998.
 8. J. S. Lee, Y. W. Su, and C. C. Shen, “A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi,” in *Proc. IEEE 33rd Annu. Conf. IECON*, Nov. 2007, pp. 46–51.

Authors:

Akshata. S. Jagatap, currently pursuing M.Tech IV sem in Gogte Institute of technology Udyambag Belagavi, Dept of Computer Science and Engineering.

Asso.Prof: S.R. Dhotre, Currently an associate professor in Computer Science and Engineering Department, Gogte Institute of Technology Udyambag, Belagavi.