Smart Drip Irrigation System using Raspberry pi and Arduino

Nikhil Agrawal
Engineeing Manager, Siemens, Noida
nikhil.pa@gmail.com

Smita Singhal
ASET, Amity University, Noida
singhal.smita@gmail.com

***Abstract*—This paper proposes a design for home automation system using ready-to-use, cost effective and energy efficient devices including raspberry pi, arduino microcontrollers, xbee modules and relay boards. Use of these components results in overall cost effective, scalable and robust implementation of system. The commands from the user are processed at raspberry pi using python programming language. Arduino microcontrollers are used to receive the *on/off* commands from the rasperry pi using zigbee protocol. Star zigbee topology serves as backbone for the communication between raspberry pi and end devices. Raspberry pi acts a central coordinator and end devices act as various routers. Low-cost and energy efficient drip irrigation system serves as a proof of concept. The design can be used in big agriculture fields as well as in small gardens via just sending an email to the system to water plants. The use of ultrasound sensors and solenoid valves make a smart drip irrigation system. The paper explains the complete installation of the system including hardware and software aspects. Experimental set-up is also tested and explained for an automatic drip irrigation system to water 50 pots.**

***Index Terms*—Raspberry pi, Arduino, Xbee, Zigbee, automatic drip irrigation system**

I. INTRODUCTION

The requirement of building an automation system for an office or home is increasing day-by-day. Industrialist and researchers are working to build efficient and economic automatic systems to control different machines like lights, fans, air conditioners based on the requirement. Automation makes an efficient use of the electricity and water and reduces much of the wastage.

Drip irrigation system makes the efficient use of water and fertilizer. Water is slowly dripped to the roots of the plants through narrow tubes and valves. Water is fed directly to the base of the plants which is a perfect way to water plants. There should be proper drainage in the fields or pot plants to avoid any water logging which in case may affect the productivity [1].

There already exist automatic drip irrigation systems which water plants based on soil humidity, pH value of soil, temperature and light. These parameters are required in big agricultural fields where productivity of the crop matters. In small areas like office premises, buildings, house gardens etc. where watering plants at regular interval matters, our proposed irrigation system will be very efficient.

This paper presents an smart drip irrigation system to water plants with the use of devices like raspberry pi, Arduino microcontrollers. Xbee is used to control the system wirelessly while Python programming language is used for automation purpose. This paper contributes an efficient and fairly cheap automation irrigation system. System once installed has no maintenance cost and is easy to use.

Organization of the paper: In section II the components of raspberry pi, which is the control block in this automatic irrigation system, is described. The proposed automatic irrigation system with implementation details are mentioned in section III. Results and conclusion are included in section IV and section V respectively.

II. CONTROL BLOCK- RASPBERRY PI

Raspberry pi is a pocket personal computer with Linux operating system installed on it. This is super cheap to encourage young people for learning, programming, experimenting and innovation. Resembling like motherboard, raspberry pi has all the components to connect inputs, outputs and storage. Its various components include [2]:

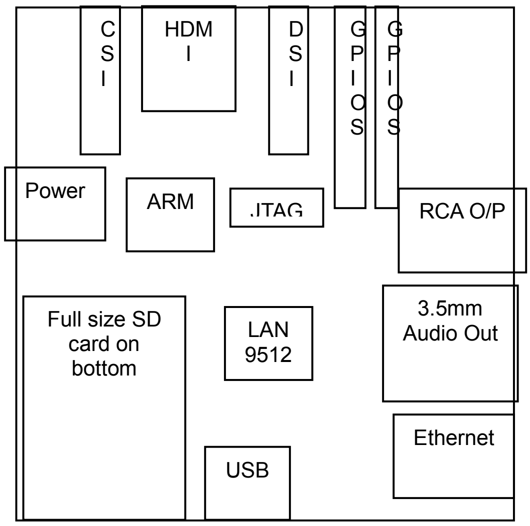

Fig. 1. Block Diagram of Raspberry Pi

A. ARM CPU/GPU: This is a Broadcom BCM2835 System on a Chip (SoC) that's made up of an ARM central processing unit (CPU) and a Videocore 4 graphics processing unit (GPU).

B. GPIO: These are general purpose input/output connection points.

C. RCA: This allows connection with analog TV or other similar points.

D. Audio Out: This point provides connection with audio out devices like speakers or headphones.

E. LED: This is used for indicator lights.

F. USB: Common connection port for peripheral devices like mouse, keyboard etc.

G. HDMI: This allows connection with compatible devices like HD television with the use of HDMI cable.

H. Power: This is a 5V USB micro connector for power supply.

I. *SD card slot: Full-sized SD card slot to hold the LINUX operating system SD card and is required for booting.*

J. *Ethernet:* This is used for wired network.

Other features can be added with the help of USB port or the USB hub if required.

A simple block diagram of raspberry pi is shown in Fig. 1.

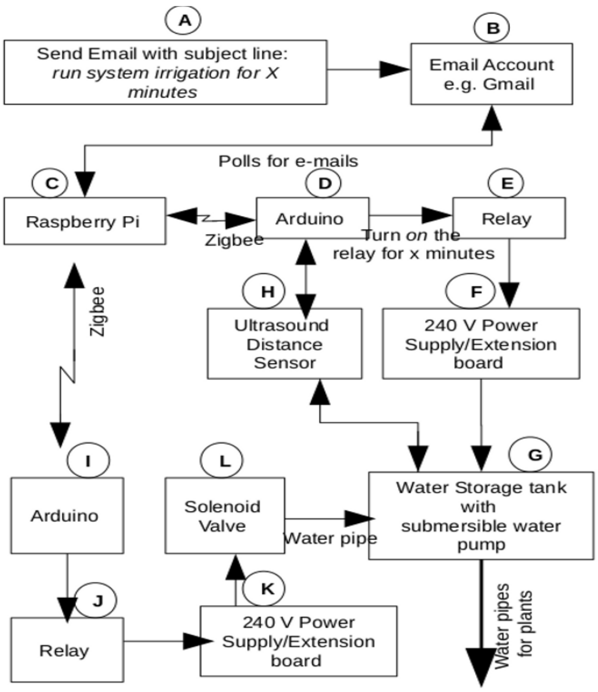

Fig. 2. Proposed System

III. PROPOSED DESIGN WITH IMPLEMENTATION

The block diagram of the proposed automatic irrigation system is shown in Fig. 2. The meaning and functionality of each block with respect to the labels mentioned in the figure is as follows:

A. *Send Email:* To start the drip irrigation system an email is sent to a defined account having subject line : “run irrigation system for *X* minutes”. For example in Fig. 2, to run irrigation system for *two* minutes, an email with the subject line is sent “run irrigation system for *two* minutes”

B. *Email Account:* Raspberry pi will poll for emails in this email account. Google email account is used in this paper.

C. *Raspberry Pi:* Model B of Raspberry is used in this paper. As soon as the email is received, one of the GPIO is turned *high.* A program written in Python programming language has been used to receive email and turning a GPIO pin *high* for the requested duration. The same program also sends the status updates to user's email address. Software libraries used in Python programming are:

1) SMTP library to send status email
2) IMAP library for email polling
3) BCM for GPIO control in raspberry pi.

The methods used in the code (pseudo) is as follows:

```
def gpio_setup() {
//setup the GPIO pins of raspberry pi
}
def check_for_pump_start() {
 // This function uses the IMAP library to check the mail
    from email account with the particular subject line
}
def send_email(to, subject, body) {
//uses SMTP server of email account to send an email for
    the successful run of the pump
}
main method {
//calls GPIO setup
gpio_setup() {
while (true)
    //check for the email
    check_for_pump_start()
    if (there is desired email)
```

//check for the number of minutes pump will
be run

//make GPIO pin of pi as *high*

//wait for X minutes

//send status mail for successful run

//make GPIO pin of pi *low*

}

Pi also sends and receives commands from and to the arduino microcontrollers using the zigbee modules. Programming is done for both pi and arduino using python and arduino core functions and libraries respectively for proper communication via zigbee. Figure 3 explains the communication circuitry between the raspberry pi and zigbee module. It is shown that pi uses USB to TTL converter to talk to the zigbee module.

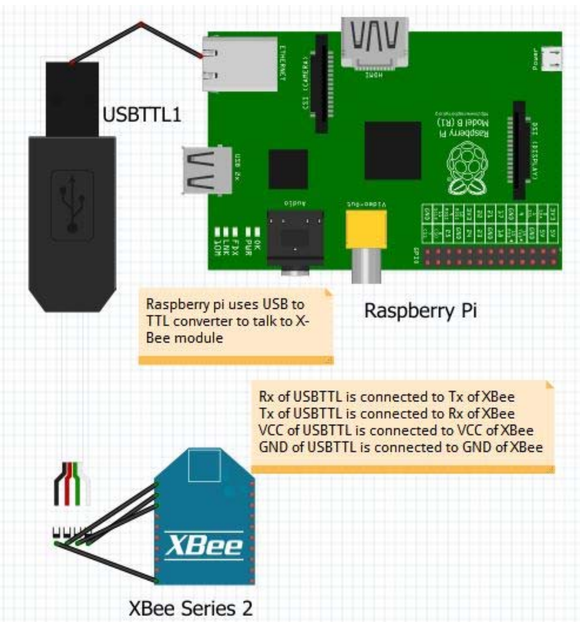

Fig. 3. Pi and Zigbee Communication Circuit

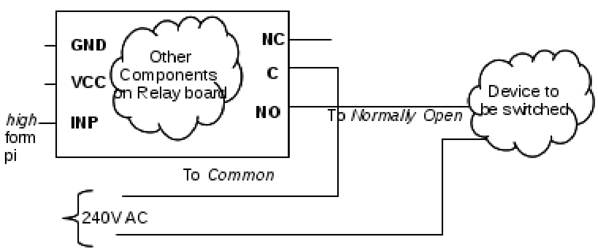

Fig. 4. Connection of 1-Channel Relay Board

D. *Arduino:* It is an open source micrcontroller which is used to control relay and ultrasound distance sensor. Freeduino flavor of arduino is used in this design. Arudino libraries and functions are used in the progam (not shown). In case a low water level is detected by the sensor (part H) then a signal will be sent from the microcontroller to pi. Pi will replay the same signal to arduino (part I)and solenoid valve (part L) will be turned *on* via relay. Figure 4 explains the communication circuitry between arduino and zigbee module. Figure 4 also explains the connection between the arduino and sensor (part H) as well as arduino and relay.

E. *Relay Board:* One-channel relay board which operates on 5-6V is used here. The circuit is used to control one 240V power appliance directly from microcontrollers or low voltage circuits. The connections to one-channel relay board is shown in Fig. 5 [3]. There are three pins on the relay board namely *normally open (NO), normally closed (NC)* and *common (C).* The *common* pin is connected to *NC* pin when the relay is *off* and to the *NO* pin when the relay is *on.* The input pin “INP” receives logic *high* from raspberry pi and in turn switches *on* the relay, thus *common* is connected to NO which turns the device *on* till the relay is *on.* The “VCC” and “GND” pins of the relay are connected to 5V supply and ground respectively.

F. *Power Supply*: The device to be switched, here, is an electrical water pump which runs on 240V supply. Its one end is connected to the 240 V AC supply and the other end is connected to *NO* pin of the relay board as shown in Fig. 5.

G. *Water Storage Tank and water pump:* Two 30 liters water storage tanks are used for testing purpose. Each tank has submersible water pumps with the rating of 220V/50Hz. It draws the current of 0.23A and power of 18W. The main water pipe is fed back to the water tank to avoid any water wastage. Water tank has ultrasound distance sensor which keeps a track of water depth in the tank. As soon as the water level falls below a threshold level, a signal is sent to microcontroller to open solenoid valve which is attached to the water tap and thus the water can be refilled into the water tank.

H. *Ultrasound Distance Sensor:* This sensor is used to measure the water level in the tank. The *on/off* signal is continously sent to the solenoid valve and thus the water level in tank does not drop below or above a threshold to avoids any damage in the water pump and also to avoid overflow of water from the water tank. The communication from sensor till solenoid valve is as follows:

a) Ultrosound sensor send signal to mirocontroller(D)

b) Arduino (D) communicate the signal to pi

c) Pi again send the signal to arduino (I) via zigbee

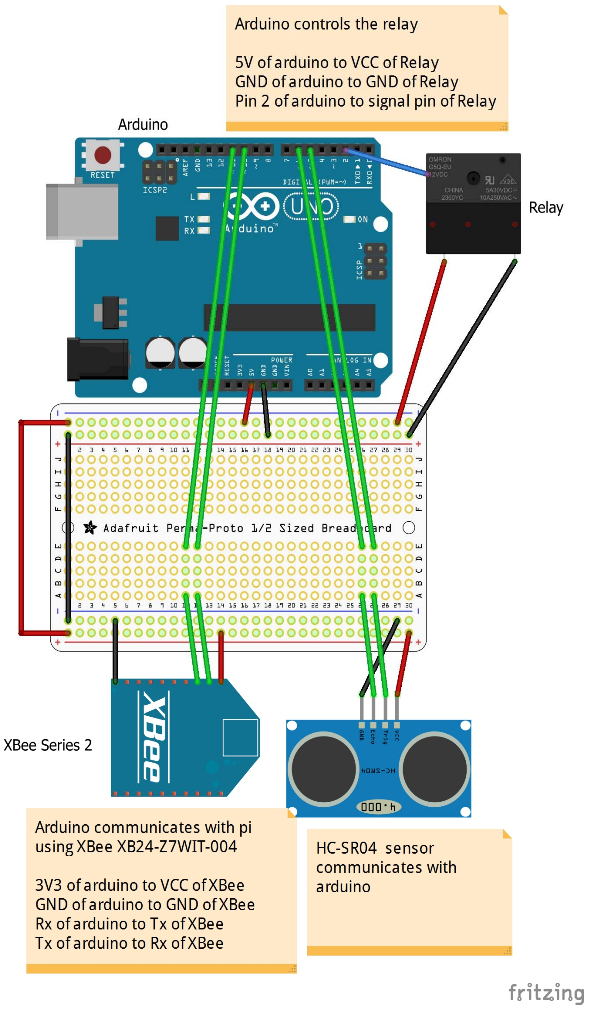

Fig. 5. Communication between arduino, zigbee, sensor and relay

Relay (J) is on/off according to the signal received from pi thus opening or closing the solenoid valve.

I. *Arduino:* The behavior of the microcontroller is same as explained in point D.

J. *Relay Board:* The behavior of the relay is same as explained in point E.

K. *Power Supply:* The behavior of the supply is same as explained in point F.

L. *Solenoid Valve:* A two-port, *normally close*, 0.5-10bar, 230V-50Hz, rotex solenoid valve is used in this design [5]. Here, the valve receives the signal from microcontroller and thus act according.

IV. RESULTS

The installation having raspberry pi, relay and water pump is shown in Fig. 7. Fifty plant pots are used in experiment. A drip kit [4] is used in the experiment which consists of main pipe with 16mm diameter, feeder pipes with 4mm diameter, drip hole punch and emitter valves. Figure 6 shows the the water tanks with water pumps and the main and feeder pipes.

The experiment was run for watering plants once a day till one month to check the reliability of the system. It is found that the system works properly and the water is passed to the plants as and when required. An email is sent to run the system for two minutes. After two minutes, an email is received for acknowledging the successful run of the system.

V. CONCLUSION

This smart drip irrigation system proves to be a useful system as it automates and regulates the watering without any manual intervention. Sending the emails to the system can be automated but manual sending of the emails has control over the system regarding whether or not to run the system depending upon the weather conditions. Using this system, solenoid valves and relay board can be controlled remotely which opens the opportunities to control the water flow as well as the electrical flow.

The limitation of this design is that the failure of any particular part or device is not informed and has to be tested manually.

Fig. 6. Irrigation System Showing Installation of Pipes

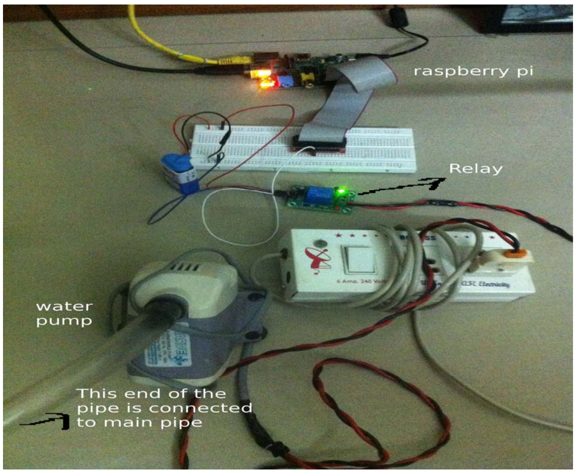

Fig. 7. Irrigation System Installation

REFERENCES

[1] R.Hussain, J.Sehgal, A.Gangwar, M.Riyag " Control of irrigation automatically by using wireless sensor network" International journal of soft computing and engineering, vol.3, issue 1, march 2013, pp.48t 324-328

[2] B.Johnson, "How the Raspberrypi works" Internet: computer.howstuffworks.com/raspberry-pi1.htm

[3] Probots, "1 Channel Relay Board" Internet: www.probots.co.in

[4] Pepper Agro, "M-Drip Kit" Internet: www.pepperagro.i/mdripkitmanual.html

[5] Rotex,"2-Port Solenoid Valve" Internet: www.rotexindia.com/products/2_Port_Solenoid_Valve/2_PORT_DIAPHRAGM_OPERATED,_NORMALLY_CLOSED__OPEN_SOLENOID_VALVE.htm