Design of a Prototype Automatic Plant Irrigation Control System

Noel T. Florencondia

http://orcid.org/0000-0002-1848-9495

Nueva Ecija University of Science and Technology
Cabanatuan City, Nueva Ecija, Philippines

Abstract

Farmers have been using irrigation techniques through manual control. This manual process is time consuming and usually results in water wastage. The study is about the construction of a prototype automatic plant irrigation control system. The device is assembled using ARDUINO UNO and equipped with a soil moisture sensor to detect its condition. In this automated system, irrigation will automatically take place when there is an intense requirement of water, as suggested by its moisture content and will automatically turn off when the soil finally received the amount of water it needs. The testing of the device under different soil conditions was interpreted based on the sensor input value of the microcontroller. The device worked and functioned properly according to its program.

Keywords: ARDUINO; Sensor; Irrigation; Agriculture; Soil

Introduction

Agriculture plays a vital role in our economy and development. For decades, it has been the main source of food production and national income for most developing countries. Although most countries consider agriculture as the backbone of their economic growth, this sector of the economy is already facing problems that may threaten its future. One of the main problems most agricultural producing countries are facing today is the lack of water. The farmers use irrigation technique through the manual control, in which the farmers irrigate the land at regular intervals (Gopinath, et al., 2014). Water-saving agricultural practices and water management strategies are therefore required to ensure the viability of the farming industry in places that experience less rain or water shortage. Accordingly, if the plants get water at the proper time then it helps to increase the production from 25 to 30 % (Kumbhar, et al., 2013). Automation in plant irrigation system brings lesser work for the farmer. Through automatic plant irrigation control system, the presence of farmer in the field is not obligatory to do the irrigation process. Automated irrigation systems are developed to operate the water pump remotely (Hassan, et al., 2013). Intelligent automatic plant irrigation system concentrates watering plants regularly without human monitoring using a moisture sensor (Carpena, 2005). Naga et al. (2013) developed a micro controller based automatic plant irrigation system used for the plants. To have a better usage of water, there must be a proper schedule of irrigation. The study is to build a simple system using microcontroller for the automatic plant irrigation control system with lesser human interventions. According to Hussain, et al. (2013) ARDUINO is an open source computer hardware and software, project and user community that designs and manufactures Microcontroller-based tools for building digital
devices and interactive objects that can sense and control the real world.

**Objectives of the Study**

This study aims to develop an automatic plant irrigation control system. Specifically, it sought answers to the following:

1. Design an automatic plant irrigation control system using ARDUINO UNO microcontroller.
2. Develop a prototype of an automatic plant irrigation control system.
3. Describe the soil condition based from sensor input of the microcontroller.

**Methodology**

In conducting this study, the researcher employed experimental research and manipulated one or more variables to determine its effects on another variable. This design became an efficient procedure to obtain data to yield valid and objective conclusions.

Trial run and testing were conducted to collect data for the study. Through this, the researcher was able to observe the flow of the device’s activity and its functional capabilities.

**Design of the Prototype Device**

**Needed Materials in Constructing the Prototype Device of Automatic Plant Irrigation Control System**

Table 1 shows the list of materials used in making the prototype automatic plant irrigation control system while Figure 2 illustrates the components of the prototype of an automatic plant irrigation control system.

**Table 1. List of Materials.**

<table>
<thead>
<tr>
<th>MATERIAL</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO Board</td>
<td>1</td>
</tr>
<tr>
<td>Relay Module (1 Channel, 5V)</td>
<td>1</td>
</tr>
<tr>
<td>Soil Moisture Sensor</td>
<td>1</td>
</tr>
<tr>
<td>LED</td>
<td>3</td>
</tr>
<tr>
<td>Resistor (330 ohms)</td>
<td>3</td>
</tr>
<tr>
<td>Water Pump</td>
<td>1</td>
</tr>
<tr>
<td>Jumper Wires</td>
<td>13</td>
</tr>
<tr>
<td>PCB</td>
<td>1</td>
</tr>
<tr>
<td>Enclosure</td>
<td>1</td>
</tr>
</tbody>
</table>
Steps in the Design of the Prototype Device

The researcher conducted the following steps in constructing the device:

1. The researcher sought advice and help on programming Arduino UNO board.
2. He modified the device based on the relevant information from past studies.
3. He assembled the connections of soil moisture sensor, relay module and Arduino board on a breadboard.
4. The researcher connected the prototype to the water pump for the trial run.
5. When the device was finally functioning properly, he fixed it on a printed circuit board (PCB) and packed it in a hard plastic enclosure.

Actual Design

Figure 3 shows the prototype design. The main components of the device are the soil moisture sensor which measures its moisture content and the Arduino UNO board which is programmed to drive the relay module connected to the water pump and to turn on the LED indicators.

Results and Discussion

Design of a prototype Automatic Plant Irrigation Control System

The device was made possible with the Arduino UNO board, a brand of a programmable microcontroller. The Arduino Uno can be powered via an external power supply which can come either from an AC-to-DC adapter or battery and use USB connection (Mohite, et al., 2015). The device’s components are the Arduino UNO,
a soil sensor, a relay, 3 LEDs and a water pump. The sensor was programmed for three levels of soil moisture namely: dry, wet and slightly wet. The researcher used three LEDs as indicators for these three levels, and a relay to drive the pump. He submerged the water pump in the reservoir and connected it to a water hose or similar object. Figure 4 shows the interconnection and diagram of the prototype device.

**Description of the main connections:**

**LM 393 Driver to Arduino**
- VCC → +5V
- GND → GND
- A0 → A0

**LED to Arduino**
- Green LED → D11 (Dry soil, Pump is ON)
- Yellow LED → D12 (Damp soil, Pump is ON)
- Red LED → D13 (Wet soil, Pump is OFF)

**Relay to Arduino**
- VCC → +5V
- GND → GND

**Program of Arduino Board Code**

```java
int RLED = 13; // Wet Indicator at Digital PIN D13
int YLED = 12; // Damp Indicator at Digital PIN D12
int GLED = 11; // Dry Indicator at Digital PIN D11
int PUMP = 10; // relay input at Digital PIN 10
int SENSOR = A0; // Soil Sensor input at Analog PIN A0
const int numReadings = 20; //number of readings for smoothing
int readings[numReadings]; // the readings from the analog input
int index = 0; // the index of the current reading
int total = 0; // the running total
int avgvalue = 0; // average value
void setup() {
  Serial.begin(9600);
  pinMode(GLED, OUTPUT);
  // setup the Arduino board
  // begin the serial communication
  // set the pin mode
}
void loop() {
  // read the moisture sensor
  int reading = analogRead(SENSOR);
  readings[index] = reading;
  index = (index + 1) % numReadings;
  total += reading;
  avgvalue = total / numReadings;
  // turn on the LED
  if (reading > avgvalue) {
    digitalWrite(RLED, HIGH);
  } else {
    digitalWrite(RLED, LOW);
  }
  // turn on the pump
  if (reading < avgvalue) {
    digitalWrite(PUMP, HIGH);
  } else {
    digitalWrite(PUMP, LOW);
  }
  // wait for a second
  delay(1000);
```

**Figure 3.** Prototype Design
Development of a Prototype Automatic Plant Irrigation Control System

The researcher presented the actual procedure in Figure 1 on how to develop the prototype Automatic Plant Irrigation Control System. They are the following:

```c
pinMode(YLED, OUTPUT);
pinMode(RLED, OUTPUT);
pinMode(PUMP, OUTPUT);
// initialize all the readings to 0:
for (int thisReading = 0; thisReading < numReadings; thisReading++)
readings[thisReading] = 0;
}
void loop() {
  // subtract the last reading:
total= total - readings[index];
  // read from the sensor:
readings[index] = analogRead(SENSOR);
  // add the reading to the total:
total= total + readings[index];
  // advance to the next position in the array:
index = index + 1;
  // if we’re at the end of the array...
if (index >= numReadings)
  // ...wrap around to the beginning:
index = 0;
  avgvalue = total / numReadings; // calculate the average
Serial.print("sensor=");
Serial.println(avgvalue);
//SENSOR IS NOT CONNECTED
if(avgvalue>=1000)
  { digitalWrite(RLED,LOW);
digitalWrite(YLED,LOW);
digitalWrite(GLED,LOW);
digitalWrite(PUMP,HIGH);
  }
//DRY
if(avgvalue<1000 && avgvalue>=600)
  { digitalWrite(GLED,HIGH);
digitalWrite(YLED,LOW);
digitalWrite(RLED,LOW);
digitalWrite(PUMP,LOW);
  }  //DAMP
if(avgvalue<600 && avgvalue>=370)
  { digitalWrite(GLED,LOW);
digitalWrite(YLED,HIGH);
digitalWrite(RLED,LOW);
digitalWrite(PUMP,LOW);
  }  //WET
if(avgvalue<370)
  { digitalWrite(GLED,LOW);
digitalWrite(YLED,LOW);
digitalWrite(RLED,HIGH);
digitalWrite(PUMP,HIGH);
  }  
delay(50);
}```
Table 2. Description of the Soil Condition Using the Automatic Irrigation System.

<table>
<thead>
<tr>
<th>Soil Condition</th>
<th>Moisture Content</th>
<th>Green</th>
<th>Yellow</th>
<th>Red</th>
<th>Relay Status</th>
<th>Water Pump Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dry</td>
<td>&lt; 1000</td>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td></td>
<td>≥ 600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Damp</td>
<td>≥ 600</td>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>Wet</td>
<td>&lt; 370</td>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
</tr>
</tbody>
</table>

- **Construction of the device.** The researcher assembled the prototype after the needed materials were gathered.

- **Trial Run.** The device underwent two trials before the final input values of the program of the microcontroller were achieved. When the device worked according to the three soil conditions, the researcher moved to the next step.

- **Testing of the device.** The researcher tested and observed the device in a small plant box under different input values each specified for three soil conditions programmed in the microcontroller.

- **Evaluation of the device.** The device was assessed on the basis of its operation during the testing. The device functioned properly according to the input values programmed in the microcontroller for each type of soil conditions.

**Description of the Soil Condition**

The condition of the soil was described based on the result of the input values programmed in the microcontroller in relation to the soil moisture sensor input. When the classification of soil is dry or has an integer value of less than 1000 but more than or equal to 600, the green LED turns “ON” while the relay driver and the pump turn “ON”. For the classification of slightly wet soil which has integer values of less than 600 but more than or equal to 370, the yellow LED turns “ON”. The relay driver and the pump are still “ON”. Moreover, for the classification of wet soil or has an integer value of less than 370, the red LED turns on, while the relay driver and the pump turn off.

In Table 2, the findings revealed the final input values for each type of soil conditions that was programmed to the microcontroller. The prototype device of automatic plant irrigation control system worked correctly. It is operating according to the codes the researcher programmed in it. All the components of the prototype functioned properly.

**Findings and Conclusions of the Study**

The prototype automatic plant irrigation control system was designed and developed...
with the use of ARDUINO microcontroller and soil moisture sensor. It was tested with assumption of three soil conditions and the results showed that it functioned properly. The basis for the description of the soil was from the input values programmed in the microcontroller. The dry soil classification has an integer values of less than 1000 but more than or equal to 600, the slightly wet soil classification has integer values of less than 600 but more than or equal to 370 and the wet soil classification has an integer values of less than 370. The ARDUINO UNO is used for easy connection to a jumper cable that can be directly connected to the microcontroller. The prototype automatic plant irrigation control system is a valuable tool for the farmers in their plant irrigation activity.

References


