



WATER CONSERVATION



VICTOR ACUNA, 8TH GRADE

EVELYN MIGUEL, 8TH GRADE

ROSELYN ALVAREZ, 8TH GRADE

ARMANDO RIVERA JR., 7TH GRADE

LYNWOOD MIDDLE SCHOOL

UC IRVINE MESA

MARCH 10, 2018
LYNWOOD MIDDLE SCHOOL

Problem Statement

Archie Arias is a very busy businessman. He travels from place to place, and hardly has time to water his plant. So, we are designing an automated plant monitor and watering device powered by Arduino to meet his needs.

First, it will test the dryness of a plant with a Soil Moisture Sensor. If the Soil Moisture Sensor detects that the soil is dry, then a LED will light up, a Buzzer will start to buzz at the same time as the LED, a LCD will show the readings of the sensor, and a Pump will start to water the plant until water isn't needed anymore.

Some limitations that our team encountered were that the LCD was not showing the readings of the moisture sensor, and the pump was delaying every time the buzzer and light paused. We looked at the code and it turned out that the code was written to show the readings on the serial monitor rather than the LCD, and that there was a delay function on the code for the pump, which we took out.

Design Process

- Project Overview/Abstract

Water is one of the building blocks of life, but in today's world there are many challenges that communities face in regard to water, especially here in the State of California. Because of severe drought in California over a couple of years, people have been asked to cut down on residential water usage. Conserving water has become a major issue to the extent that people who default in conserving water as required by law are being fined.

Up to 70 percent of water use is used on plants, so managing the amount of water on your plants is one of the easiest ways to save water around the home. That is why we tried to design a device

that will enable us to water your plants only when it is needed and/or in a way that conserves much needed water.

- Research

In the state of California there has been a drought, so our team members thought that we should create a prototype that could help us in this drought. People were asked to not water their plants that much, but some didn't know if they put too much or too little or just the right amount of water for those who put too much or too little their plant died. With this prototype the LCD, LED, buzzer and sensor will alert the pump if it would need water then the pump would water the plant. This prototype helps the clients needs when they are not home and if their plant needs water the pump will water the plant.

- Design Process/Testing

Our team was brainstorming then we decide that we should do create a prototype that would alert a person when their plant needs water. The major design choices we had to make was to add a sensor and LCD because that would help us determine if the plant needed water and if it was dry. Another major choice was that we included a LED and buzzer to tell the client when the plant needed water. The modifications were that we added a pump so that when the client was on business trips the LED and buzzer will alert the pump to water the plant.

- Prototype Development

The evolution of our prototype was that the first prototype we did had no pump but then later on we developed the prototype and add the pump. This helped the clients need when he went on business trips and couldn't water his plant.

- Discussion of Electronics Hardware Integration

Our challenges in this prototype was that the pump wouldn't work and the LCD would disfunction at times. We chose to use a small breadboard but then we changed it to fit a larger breadboard since the smaller breadboard caused the Arduino to get confused.

- Discussion of Software Development

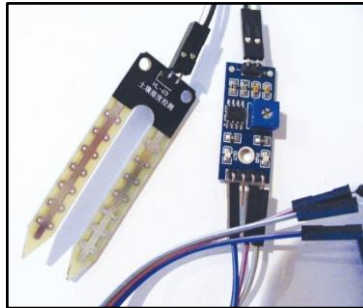
The sketch first defines soilReader so that it reads the moisture sensor value. It then defines Arduino pin 8 as output for the buzzer, and pin 13 as output for the LED. Use the Serial.println() function to send the reading from the sensor to the IDE, to see the value on the screen. Use lcd.print() to send the reading of the IDE to the LCD.

- Conclusion and Recommendations

Creating the Plant Monitor Project with Arduino was very scary when we started initially. But, it became easier as time passed by when we started getting some ideas from here and there. We used minimal equipment which were, the Elegoo Uno Board, the soil moisture testing kit (which detects when the plant has or doesn't have water), alarm buzzer (which buzzes when there is no water), an LED (Which lights up when a plant has no water), a water pump (which waters the plant when there is no water), an LCD (which prints out the data of the moisture sensor), and a potentiometer (Which makes the data on the LCD visible). In the end, we ended up creating a very low budget and cost-effective plant monitoring kit. A recommendation would be to add a Bluetooth module, so it could send you a message on your phone so that you can know when your plant has or doesn't have water. Although for us, we must continue to work on that, until we toughly understand how to make that be functional.

- Testing Procedures

Step 1: Connect the sensor's two pins to the + and – pins on the controller using the provided connecting wires, as shown below.



Step 2: Connect the three prongs from the controller to +5V, GND, and Arduino A0 directly on the Arduino, as shown in the following table. The DO pin is not used.

SENSOR CONTROLER	ARDUINO
VCC	+5V
GND	GND
A0	A0
DO	Not used

Step 3: With a wire, connect one end to 5V on the Arduino Uno, and connect the other end to the positive side of the breadboard. With another wire, connect one end to GND on the Arduino Uno and connect the other end to the negative side of the breadboard.

ARDUINO UNO	BREABOARD
5V	Positive Side
GND	Negative Side

Step 4: Connect an LED directly to the breadboard. Make sure the shorter, negative leg is in GND and the longer, positive leg is in Arduino pin 13.

LED	ARDUINO

Positive Leg	Pin 13
Negative Leg	GND

Step 5: Connect the piezo buzzer's black wire to GND and its red wire to Arduino pin 11.

PIEZO BUZZER	ARDUINO
1 st Wire	Pin 8
2 nd Wire	GND

Step 6: Follow the chart to find the connections for the LCD, and potentiometer.

LCD PINS	CONNECTIONS
VSS	GND (On Breadboard)
VDD	5V (On Breadboard)
V0	Middle Pin of Potentiometer
RS	Pin 12 (On Arduino)
RW	GND (On Breadboard)
E	Pin 11 (On Arduino)
D0	Not Used
D1	Not Used
D2	Not Used
D3	Not Used
D4	Pin 5 (On Arduino)
D5	Pin 4 (On Arduino)
D6	Pin 3 (On Arduino)
D7	Pin 2 (On Arduino)
A	Goes Next to a Resistor Which Connects to 5V (On Breadboard)
K	Ground (On Breadboard)

POTENTIOMETER	CONNECTIONS
Pin on Left End	5V (On Breadboard)
Pin on Right End	Ground (On Breadboard)

Step 7: With the wire connector, insert the pump wires to one end of the wire connector (On wire on each hole), and insert 1 wire on each hole on the other side of the wire connector.

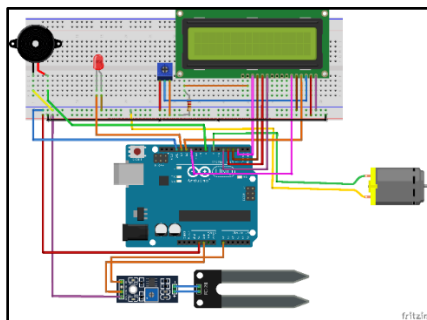
Then, you insert the PVC tube on the hole that is on the pump. Afterwards, follow the chart to find the connections of the pump.

PUMP	Connect
1 st Wire	Pin 7 (On Breadboard)
2 nd Wire	GND (On Breadboard)



Step 8: On the breadboard, connect one wire to one end of the breadboard in 5V and connect the other end of the wire to the other end of the breadboard in 5V.

Step 9: Check that your setup matches that below, and then upload the code in “Appendix” (see below).



Results

In conclusion, our prototype will be a viable solution for our client. The reason this prototype will be a solution for our client is because, it is automated (so it will be able to be functional by itself), and because it is cost effective. One of its strengths is it being able to water the plant itself. That way, you will have some way more time to do more important things.

Recommendations

A recommendation would be to add the Bluetooth module (As mentioned above). The reason is because with this Bluetooth module, you will be able to know if whether your plant has water or not, no matter where you are.

Data

Indoor Plants	Moisture Level of Plant
Peace Lily	Moist
Sprenger's Asparagus Fern	Dry
Orchids	Wet

We tested the Arduino device by going around every one of our houses and tested the plants that we have in common. We then put the sensor in the plant and saw the readings which shows that some were low, some were high, and some were ok. If the plant that was dry the numbers were above 900, so the LED and buzzer alerted the pump to water your plant. If the plant was wet, the number would be below 900, so the LED and buzzer will remain off.

Appendix

- Commented Arduino Code

```
//(Plant Monitor LCD)
#include <Wire.h>
#include <LiquidCrystal.h>
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```



```

const int soilReader = 0; //Analog pin 0
const int maxDryLevel = 900; // the value after the LED goes on
int buzzPin = 8; // Pin connected to the piezo buzzer
int ledPin = 13; // Pin connected to the LED
int pumpPin = 7; // Pin connected to Water Pump

void setup()
{
Serial.begin(9600); // Send Arduino reading to IDE
Serial.println("Soil Moisture...");
lcd.begin(16,2); // LCD screen is 16 characters by 2 lines.
lcd.clear();
lcd.setCursor(0,1);
lcd.print("Soil Moisture:");
pinMode(soilReader, INPUT); // Set pin as input
pinMode(ledPin, OUTPUT); // Set pin as output
pinMode(pumpPin, OUTPUT); // Set pin as output
pinMode(buzzPin, OUTPUT); // Set pin as output
}
void moistureStatus(int status)
{
digitalWrite(pumpPin, status);
}
void loop()
{
int moisture = analogRead(soilReader);
Serial.print("Moisture = ");
Serial.println(moisture);
lcd.clear();
lcd.print("Moisture = ");
lcd.print(moisture);
delay(1000);
if (moisture > maxDryLevel) { // if the reading is above 900
moistureStatus(HIGH);
lcd.clear();
lcd.print("Water Needed"); // the lcd will print that water is needed
digitalWrite(ledPin, HIGH); // the led will light up
digitalWrite(buzzPin, HIGH); // and the buzzer will sound
delay(1000); // wait for a second
digitalWrite(buzzPin, LOW);
digitalWrite(ledPin, LOW); // turn the LED off by making the voltage LOW
}
else {
moistureStatus(HIGH); // if the reading is below 900
digitalWrite(ledPin, LOW); // the led will stay off
digitalWrite(buzzPin, LOW); // and so will the buzzer
}
}

```

}
}

- Itemized Budget Sheet

MESA Center: UCI

MESA School: Lynwood Middle School

Level: Middle School

Advisors: Mr. Garcia and Mr. Kalu

Student Team: Evelyn Miguel -7th grade, Victor Acuna -7th grade, Steven Macias -7th grade

Part	Unit Dimension s	Retail Price \$	Price Per Unit \$	Quantity Used	Total Cost \$	Retail Source	Receipt
Elegoo UNO Board	3" x 2.1"	\$10.90	\$10.90/Board	1 Board	\$10.90	Amazon.com	
Breadboard 830 Point	2.2" x 7"	\$1.65	\$1.65/Breadboard	1 Bread board	\$1.65	Aliexpress.com	1
Soil Moisture Sensor	5 cm x 1.7 cm	\$0.44	\$0.44/Sensor	1 Sensor	\$0.44	Aliexpress.com	2
Active Buzzer	12 mm x 9.5 mm	\$0.88 for 10 Buzzers	\$0.09/Buzzer	1 Buzzer	\$0.09	Aliexpress.com	3
Battery Clip	6" x 0.5"	\$1.25 for 10 Battery Clips	\$0.13/Battery Clip	1 Battery Clip	\$0.13	Aliexpress.com	4
LCD Screen	7.5 cm x 3.5 cm	\$2.80	\$2.80	1 LCD	\$2.80	Aliexpress.com	5
Male to Male Jumper Wires	20 cm x 0.2 cm	\$0.96 For 65 Male to Male Wires	\$0.01/Wire	21 Wires	\$0.21	Aliexpress.com	6
Water Pump	17" x 2"	\$1.25	\$1.25	1 Water Pump	\$1.25	Aliexpress.com	7
5mm LED	5mm 20 LEDs	2.95	.14/LED	1 LED	0.14	Sparkfun.com	

Push Type Wire Connector	2 cm x 1.5 cm	\$2.98 for 50 Wire Connectors	\$0.05/Wire Connector	1 Push Type Wire Connector	\$0.05	Aliexpress.com	
9V Battery	4.5" x 2"	\$4.99	\$4.99	1 9V Battery	\$4.99	Amazon.com	
PVC Clear Vinyl Tube	20" x 3/8"	\$8.87	\$0.44/Inch	14 Inches	\$6.16	Homedepot.com	
Total Cost					\$28.81		

Bibliography

- How to build an Automatic Watering System For Plants Arduino Uno DIY TUTORIAL. (2016, June 23). Retrieved from <https://www.youtube.com/watch?v=Z2xLD0P83rw>
I cited this video as APA because it pertains to education.
- Arduino – HelloWorld. (n.d.). Retrieved from <https://www.arduino.cc/en/Tutorial/HelloWorld>
I cited this website as APA because it pertains to education.
- Arduino Reference. (n.d.). Retrieved from <https://www.arduino.cc/reference/en/language/functions/time/delay/>
I cited this website as APA because it shows you how to properly use the delay function.
- Arduino - DigitalPins. (n.d.). Retrieved from <https://www.arduino.cc/en/Tutorial/DigitalPins>
I cited this website as APA because it tells you what is an output and what is an input.
- Aliexpress. (n.d.). Retrieved from <https://www.aliexpress.com/>
I cited this website as APA because it shows us how to use the certain materials we used.
- Amazon. (n.d.). Retrieved from <https://www.amazon.com/>
I cited this website as APA because it shows us how to use the certain materials we used.

- The Home Depot. (n.d.). Retrieved from <https://www.homedepot.com/>

I cited this website as APA because it sells a lot of materials that pertain (In a way) to Arduino.

- SparkFun Electronics. (n.d.). Retrieved from <https://www.sparkfun.com/>

I cited this website as APA because it sells a lot of engineering based products and shows us how to properly use it.