SOUTHEAST UNIVERSITY (SEU)

Department of Computer Science & Engineering

Introduction To Embedded System Lab

(CSE3028.1)

Lab Report

On

Touchless Hand Wash and Refill System

A dissertation submitted to the Southeast University in partial fulfillment of the requirements for the degree of B. Sc. in Computer Science & Engineering

Submitted by

Md. Rakibul Hassan Rony	2018000000012
Bishajit Chakraborty	2018000000010

Submitted to

Md. Ishan Arefin Hossain
Lecturer, Department of CSE
Southeast University

Touchless Handwash and Refill System

In order to create a successful Touchless Hand Wash and Refill System, the following criteria, such as human distance, liquid level of bottles must be regulated in such a way that anyone may wash their hands smoothly without the machine failing.

Project Schematic Diagram:

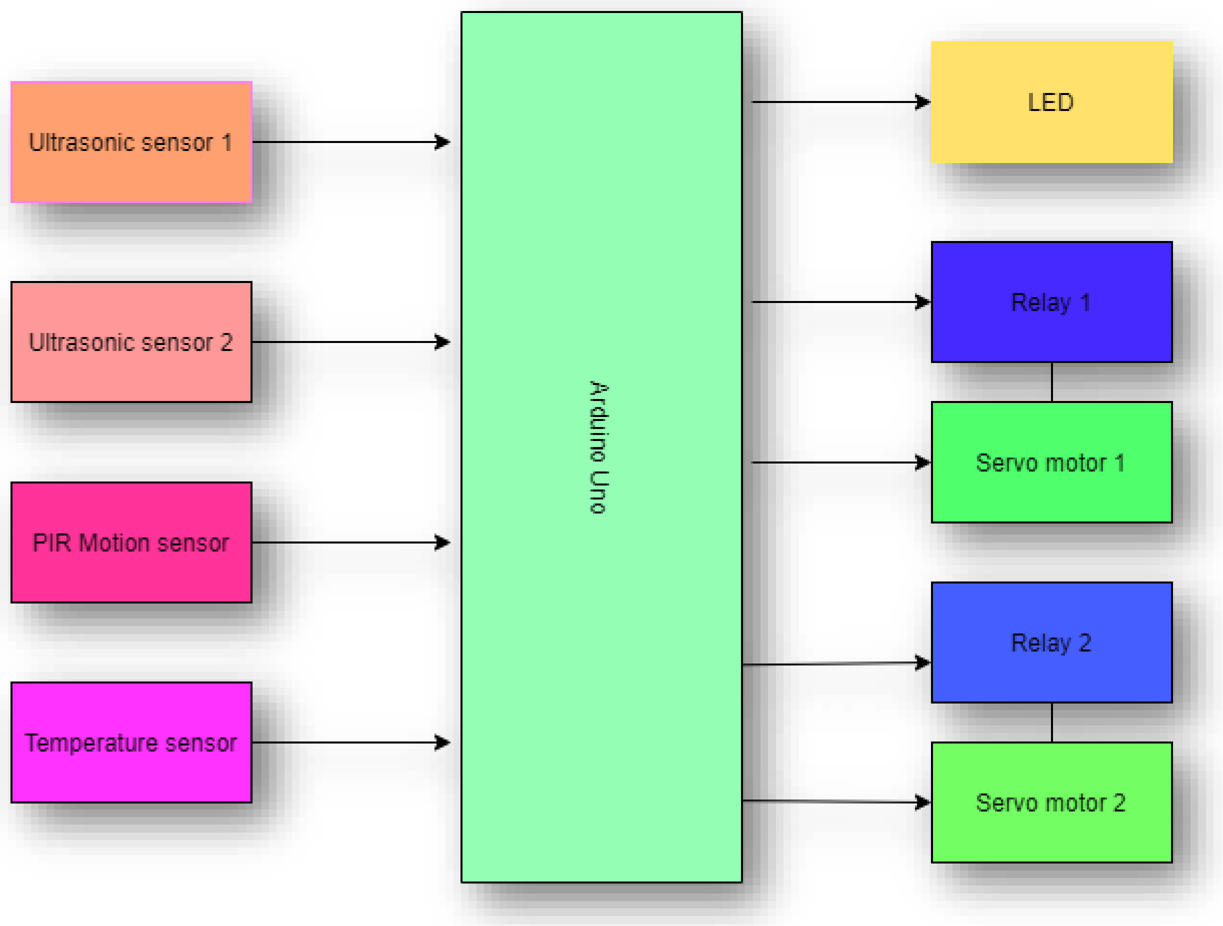

Figure: Project Schematic Diagram of Touchless Hand-wash and Refill System

Circuit Diagram

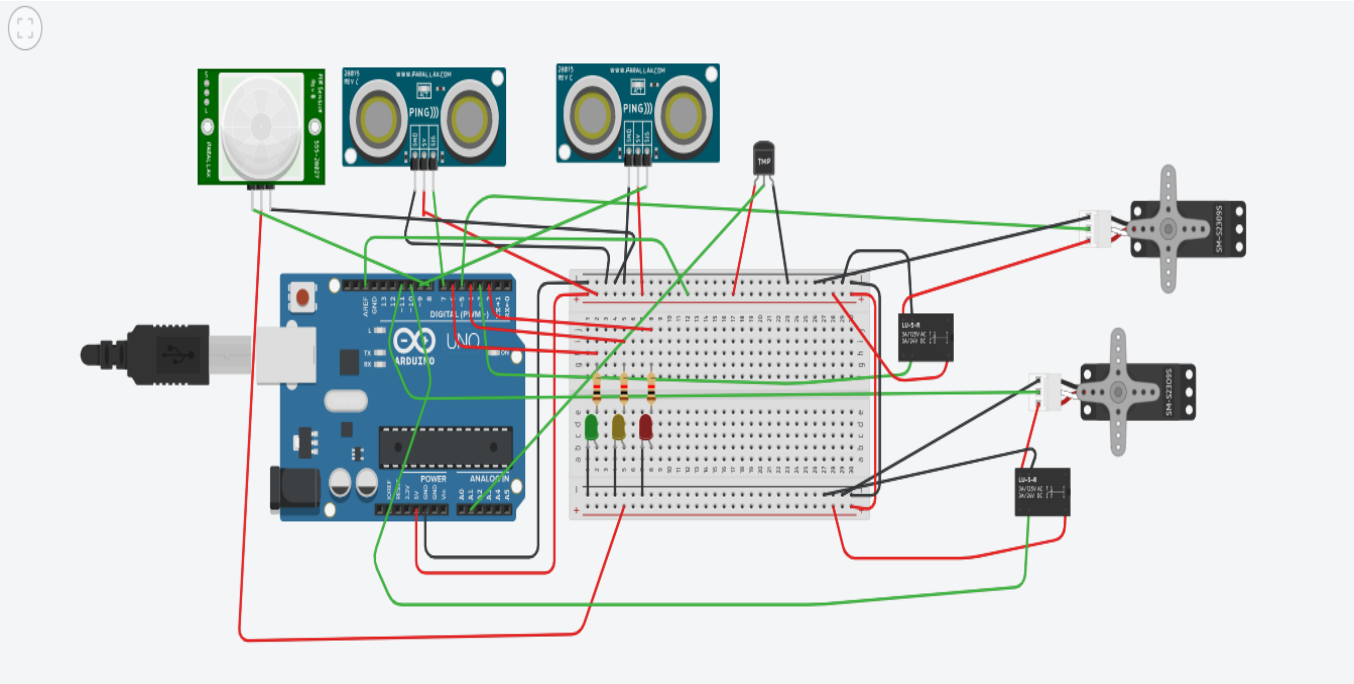

Fig: Circuit Diagram Drawn in Tinkercad

Necessary Hardware List

A. Arduino UNO R3

B. Sensors (4 sensors)

1. Ultrasonic sensor (2 pieces)
2. PIR Motion sensor (1 pieces)
3. Temperature sensor (1 pieces)

C. Actuators (3 actuators)

1. Servo motor (2 pieces)
2. LED light (3 pieces)

D. Relay (2 pieces)

E. Breadboard

A. Arduino UNO R3

In this project, a microcontroller board called an Arduino Uno R3 is used. There are 14 digital input/output pins (six of which can be used as PWM outputs), six analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button on the board. All of these modules are required to support the microcontroller, and to get started, simply connect it to a computer via USB cable or power it with an AC-to-DC adapter or battery. With the deployment of the internet of things, a circuit has been constructed in conjunction with this hardware (IoT).

B. Sensors

1. Ultrasonic Sensor (2 pieces)

An ultrasonic sensor is an electronic device that uses ultrasonic sound waves to detect the distance between a target item and converts the reflected sound into an electrical signal. The ultrasonic sensor has three pins that must be connected.

a. **VCC** is the ultrasonic sensor's power supply, which we attach to the Arduino's 5V pin..
b. **Trig** The ultrasonic sound pulses are triggered by the (Trigger) pin.
c. **Echo pin** : When the reflected signal is received, the echo pin generates a pulse. The pulse's duration is proportional to the time it took to detect the sent signal..
d. **GND** : it always connects to the ground pin of the Arduino.

<u>Use cases :</u>

1. One ultrasonic sensor was used to measure the distance of the human body (hand) from the handwash bottle. And when a human is close to the bottle of a certain amount then a signal is sent to the arduino and then the arduino will send a signal to a particular servo motor which will rotate to fill the bottle with liquid hand wash.

2. Another ultrasonic sensor was used to measure the liquid level of the handwash of the cylinder, which will send the current liquid level of the cylinder to the arduino. Arduino will send the signal to the particular Servo motor when the cylinder is required to fill with liquid hand wash.

2. PIR Motion Sensor (1 piece)

A passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often used in PIR-based motion detectors. We utilized this sensor to detect human movements near the handwash bottle, integrating the Ultrasonic and PIR sensor signals to ensure the presence of a human. Following the comparison of these signals, Arduino sends a signal to the Servo motor, which then pours liquid into the person's hand.

Use case:

PIR motion sensor was used to over-sure a human presence near the hand-wash bottle. So that it can give a 100% accuracy and no wastage of the handwashes.

3. Temperature Sensor (1 piece)

The TMP36 temperature sensor is an easy way to measure temperature using an Arduino! The sensor can measure a fairly wide range of temperature (-50°C to 125°C), is fairly precise (0.1°C resolution). We used this sensor to measure the person’s body temperature and give a signal on the led (Light emitting diode) based on the temperature.

Use case:

Temperature sensor was used to measure the temperature of the human who comes near of the machine and gives a signal through the led lights about the temperature level of the human body.

C. Actuators

1. Servo Motor (2 pieces)

A servo motor is a type of motor that may have its position controlled by a microcontroller such as an Arduino. A closed loop control system is referred to as a servo. To accomplish the intended outcome, a closed loop system uses the feedback signal to alter the motor's speed and direction. By delivering a series of pulses to the signal line, we can control the servo motor. A traditional analog servo motor anticipates receiving a pulse every 20 milliseconds (i.e. signal should be 50Hz).

Servo motors typically have **three** connections and are as follows:

- **GND** is a common ground for both the motor and logic.
- **5V** is a positive voltage that powers the servo.
- **Control** is input for the control system.

The Servo library is included with the Arduino IDE. Simple commands are included so that we may rapidly order the servo to turn to a specific angle.

In this project, I used two servo motors which are connected to an Arduino. A temperature sensor is also connected with the Arduino. When the temperature sensor senses the temperature greater than 30 degree, it passes an input signal to the Arduino. Then Arduino instructs the servo to turn to a 90 degree angle {servo.write (90)}. When temperature is less than 30 degree, the angle of servo will remain in 0 degree.

Use cases:

1. **One servo** motor was used to flow the liquid from a big cylinder where most of the hand-washes were kept to a small bottle.

2. **Another servo** motor was used to flow the hand-washes to the outside of the bottle.

2. Led Lights (3 pieces)

LED is the abbreviation for Light Emitting Diode. A diode is a device that only allows current to flow in one direction. Light is produced by current passing through an LED.

Use case:

In this project three led was used. One red, one yellow and one green was used. So, when the temperature sensor gives the temperature level of the human then the led is switched on according the temperature level. Here if the temperature is between 98 to 100 degree then the green led is on. If the temperature is between 100 to 1002 degree then the yellow led is on. And lastly when the temperature level is above 102 degree then the red led is switched on.

D. Relay

A relay is a type of electromagnetic switch that operates in the same way that a light switch does. It can use only a very low voltage signal to control a high voltage circuit. A fixed coil, a moveable armature, a spring, and contacts make up the basic components of an electromechanical relay. When a low voltage control signal is sent, the coil generates a magnetic field.

The relay usually has 5 pins. Three of them (NC, COM, and NO) are high-voltage terminals that connect to the device we want to operate. A coil that functions as an electromagnet is located between the remaining two pins (coil1 and coil2). When current passes through the coil, the electromagnet charges up and moves the switch's internal contacts. At that time the normally open (NO) terminal connects to the common (COM), and the normally closed (NC) terminal becomes disconnected.

Normally Open (NO) and Normally Closed (NC) electromechanical relays are the two varieties that exist (NC). When a current is passed via a NO relay, the contacts are closed. A NC relay's contacts are also only open when a current is sent through it. When the regulated circuit will be off the majority of the time, a NO relay is preferable, and when the circuit will be on the majority of the time, an NC relay is preferred.

<u>Use cases:</u>

In our project we used 2 relays both in **NO (Normally Open) mode.**

1. The first relay was connected with the servo motor that enables liquid form the bottle to the outside of the bottle through a pipe. Whenever the Arduino gives signal to the relay to flow current then the servo starts roatting.

2. The second relay was connected to provide electric current to another servo motor which works for supplying liquid hand washes from a big cylinder to the small bottle. And lastly from the small bottle hand washes flows outside to the human hand.

Implementation Code

The code of the circuit diagram of my project used in the Tinkercad is given below:

```
#include <Servo.h>

Servo myservo;

Servo bigServo;
```

```
const int sonar1Big = 9;

const int sonar2Small = 7;

const int inputPir = 8;

const int relay1Small = 3;

int relay2 = 10;

int tempPin = 1;

int ledRed = 2;

int ledYellow = 4;

int ledGreen = 6;

//int servo2 = ;


void setup() {

  Serial.begin(9600);

  myservo.attach(5);

  bigServo.attach(11);

  myservo.write(0);

  bigServo.write(0);

  pinMode(inputPir,INPUT);

  pinMode(sonar1Big,OUTPUT);

  pinMode(sonar2Small,OUTPUT);

  pinMode(relay1Small,OUTPUT);

  pinMode(relay2,OUTPUT);

  analogReference(DEFAULT);

  //pinMode(tempPin,INPUT);

  pinMode(ledRed,OUTPUT);

  pinMode(ledYellow,OUTPUT);

  pinMode(ledGreen,OUTPUT);

}
```

```
void loop() {
  ledOff();
  int tempVal = analogRead(tempPin);

  int celsius = map(((tempVal - 20) * 3.04), 0, 1023, -40, 125);
  // convert to Fahrenheit

  int fahrenheit = ((celsius * 9) / 5 + 32);
  Serial.println("----------");
  Serial.println(tempVal);
  Serial.print(celsius);
  Serial.print(" C, ");
  Serial.print(fahrenheit);
  Serial.println(" F");
  Serial.println("----------");

  long cm = 0;
  cm = checkForAvailableSmall();
  int motion = digitalRead(inputPir);
  if(cm<50 && motion==HIGH){
    digitalWrite(relay1Small,HIGH);
    delay(500);
    myservo.write(90);
    delay(500);
    myservo.write(0);
    delay(500);
    Serial.print("above 50 ");
```

```
    Serial.print("Distance: ");
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();

      if(fahrenheit<=98){
       digitalWrite(ledRed,LOW);
       digitalWrite(ledYellow,LOW);
       digitalWrite(ledGreen,HIGH);
       delay(3000);
       ledOff();
     }
     else if(fahrenheit<=100){
       digitalWrite(ledRed,LOW);
       digitalWrite(ledYellow,HIGH);
       digitalWrite(ledGreen,LOW);
       delay(3000);
       ledOff();
     }
     else if(fahrenheit>100){
       digitalWrite(ledRed,HIGH);
       digitalWrite(ledYellow,LOW);
       digitalWrite(ledGreen,LOW);
       delay(3000);
       ledOff();
     }
    }
  else {
```

```
  digitalWrite(relay1Small,LOW);
  // servo1.write(i);
    // Print the distance
 // Serial.print("below 50 ");
//  Serial.print("Distance: ")
//  Serial.print(cm);
//  Serial.print("cm");
//  Serial.println();


}
long depthLiquid = checkForAvailableBig();
if(depthLiquid>50){
  digitalWrite(relay2,HIGH);
  for(int j=0;j<=10;j++){
            bigServo.write(0);
    delay(1000);
    bigServo.write(180);
    delay(1000);
  }
  Serial.print("liquid far from 50 ");
  Serial.print("Distance: ");
  Serial.print(cm);
  Serial.print("cm");
  Serial.println();
  bigServo.write(0);
} else{
  digitalWrite(relay2,LOW);
  bigServo.write(90);
```

```
    Serial.print("liquid in range 50 ");
    Serial.print("Distance: ");
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();
    bigServo.write(0);
  }
}

long checkForAvailableSmall(){
  pinMode(sonar2Small, OUTPUT);
  digitalWrite(sonar2Small, LOW);
  delayMicroseconds(2);
  digitalWrite(sonar2Small, HIGH);
  delayMicroseconds(5);
  digitalWrite(sonar2Small, LOW);

  // The same pin is used to read the signal from the PING))): a HIGH
  // pulse whose duration is the time (in microseconds) from the sending
  // of the ping to the reception of its echo off of an object.
  pinMode(sonar2Small, INPUT);
  long duration = pulseIn(sonar2Small, HIGH);

  // convert the time into a distance
  long cm = microsecondToCentimeter(duration);
  return cm;
}
long checkForAvailableBig(){
```

```
  pinMode(sonar1Big, OUTPUT);
  digitalWrite(sonar1Big, LOW);
  delayMicroseconds(2);
  digitalWrite(sonar1Big, HIGH);
  delayMicroseconds(5);
  digitalWrite(sonar1Big, LOW);

  // The same pin is used to read the signal from the PING))): a HIGH
  // pulse whose duration is the time (in microseconds) from the sending
  // of the ping to the reception of its echo off of an object.
  pinMode(sonar1Big, INPUT);
  long duration = pulseIn(sonar1Big, HIGH);

  // convert the time into a distance
  long cm = microsecondToCentimeter(duration);
  return cm;
}
long microsecondToCentimeter(long micro){
  return micro / 29 / 2;
}

void ledOff(){
  digitalWrite(ledRed,LOW);
  digitalWrite(ledYellow,LOW);
  digitalWrite(ledGreen,LOW);
}
```