ME 102B Mini-project

Student: An Phan

Professors: Hannah Stuart and Homayoon Kazerooni

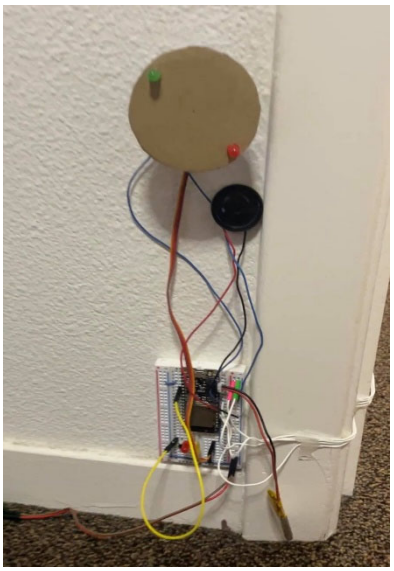
The Package Detector

Description of the product:

When the pandemic hit to our country, most of the time we stay home. As the result, all our activities move to virtual. Specially, we tend to order more products from online stores, so every house or apartment often have packages which are delivered into the front door every day. To be preventing the thieves from our packages, I want to create a device which can notify the packages when they are delivered.

This device will be placed at the main door of the house or apartment, where the packages will be delivered in front of them. When the packages are dropped in the front of the door, the device will blink on the LEDs system and the servo motor will rotate with 2 LEDs attaching on the top. The LEDs system keeps blinking until the packages are picked up and the flag comes down then.

The detector has two operation modes which are normal mode and silence mode. The normal mode plays the sound, flashes the LEDs, and rotates the servo motor backward and forward. The silence mode only flashes the LEDs



a

b

Figure 1: a) controller and actuator b) ultra-sonic sensor

Circuit:

There are 5 main elements in this project: (1) Ultra-sonic senor, (2) servo motor, (3) 2 LEDs, (4) ESP 32, and (5) speaker. All the parts, I have come from our lab kit except the servo motor which I take it from my other project.

(1) Ultra-sonic sensor (HC-HR04) which is used to measure the distance from the obstacle to the place the sensor attaching. The range of the measurement from 0.04 cm to 4 meters.
(2) Servo motor (SG90) which is used as an actuator for notifying the detection from the ultra-sonic sensor.
(3) 2 LEDs: are attached into the Servo motor and flash when the sensor detect a package.
(4) ESP 32 is the micro-controller.
(5) A mini speaker is an alarm playing a tone when a package is delivered.

I used Lithium 3.7V is the power supply for the ESP 32 and 5V power adapter to power the servo motor and the ultra-sonic sensor. Beside that, I use an switch to change the state of the device.

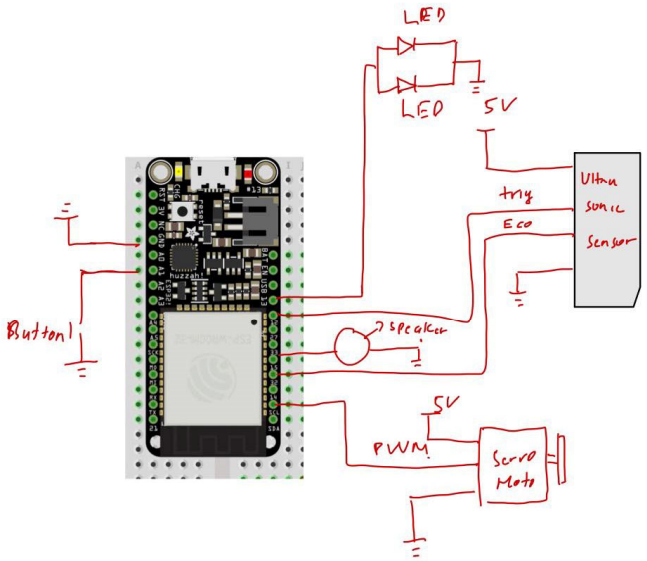


Figure 2: Wiring Diagram of the device

Finite Stage Diagram:

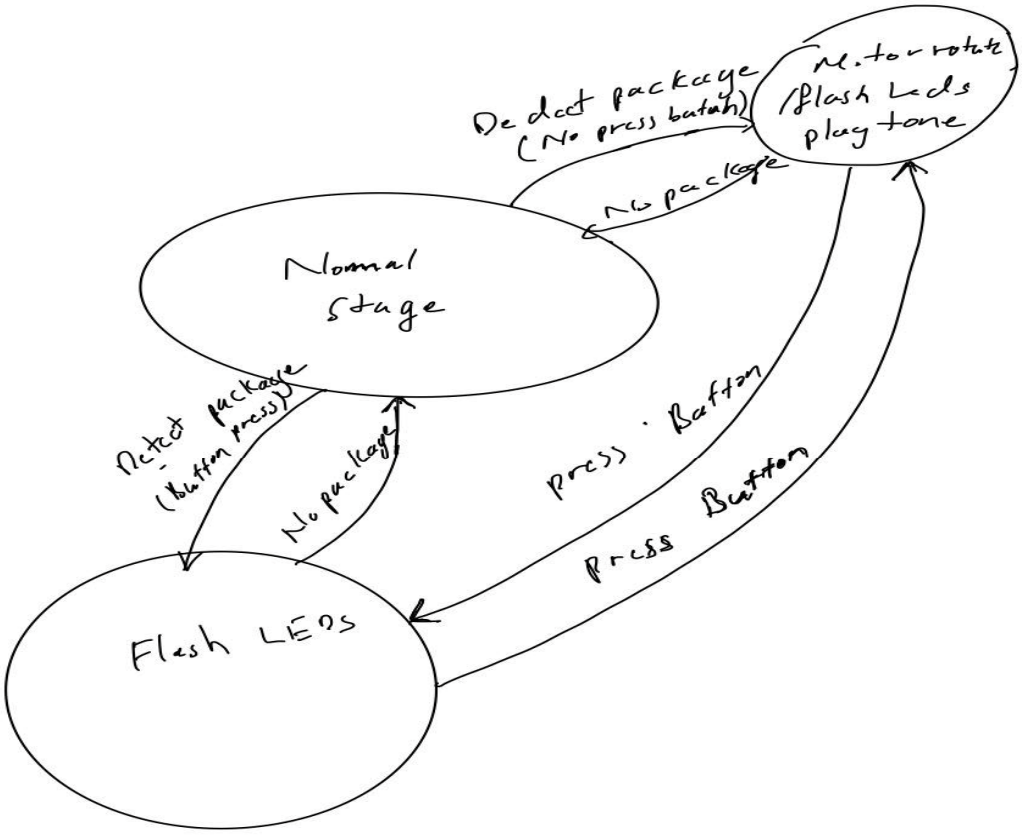


Figure 3: FSD of the device

Code

```
int trigPin = 12;            // HC-SR04 trigger pin
int echoPin = 15;            // HC-SR04 echo pin
float duration, distance;
long debouncing_time = 500; //Debouncing Time in Milliseconds
volatile unsigned long last_micros=0;
byte b=0;
int d=30;
#include <Servo_ESP32.h>
static const int servoPin = 14;
Servo_ESP32 servo1;
void debounce(){
   if((long)(millis() - last_micros) >= debouncing_time ) {
    isr();
    last_micros = millis();
  }
}
void  isr(){
 b=!b;
}
void setup()
{
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);// define trigger pin as output
  pinMode( 13, OUTPUT);
  servo1.attach(servoPin);
   ledcSetup(0,1E5,12);
  ledcAttachPin(33,0);
  pinMode(A1, INPUT_PULLUP);
  attachInterrupt(A1,debounce,FALLING);
}
void loop()
{
  digitalWrite(echoPin, LOW);   // set the echo pin LOW
  digitalWrite(trigPin, LOW);   // set the trigger pin LOW
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);  // set the trigger pin HIGH for 10µs
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);  // measure the echo time (µs)
  distance = (duration/2.0)*0.0343;
   Serial.print("Distance: ");
   Serial.print(distance, 1); Serial.println(" cm");
 // convert echo time to distance (cm)
  if (distance < d&&b==0){
    digitalWrite(13,HIGH);
    servo1.write(90);
    ledcWriteTone(0,800);
  delay(1000);
  uint8_t octave = 1;
  ledcWriteNote(0,NOTE_C,octave);
   digitalWrite(13,LOW);
  delay(1000);
  }
 if (distance < d&&b==1){
    digitalWrite(13,HIGH);
    servo1.write(90);

  delay(1000);


   digitalWrite(13,LOW);
  delay(1000);
  }
  if(distance>=d){
    digitalWrite(27,LOW);
    servo1.write(0);
  }
}
```