Security Camera Motion Detector

TEAM 29

Name: Abdul Khan

NetID: akhan260

Email: akhan260@uic.edu

Name: Varang Patel

NetID: vpate63

Email: vpate63@uic.edu

Link to presentation

https://www.youtube.com/watch?v=rjSG1kCLm-Y&feature=youtu.be

Abstract:

This project will sensor any detection of objects near its range and will send an email to the owner. Two arduinos will be used in this project. The sensor and ldr will be the input and the notification and picture will be the main output. This project can be used to put it in a safe lock in case any one tries to open it or by doors incase any intruder might enter.

Detailed Project Ideas:

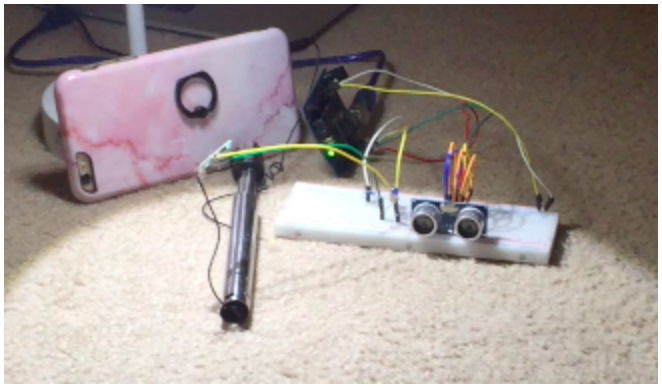
1. Description:

We want to create a project that provides security for people's homes. In this project, we will set up a home security camera that will take a picture whenever it detects any motion. From there it will send a notification/email, which will tell the user that motion was detected, and the picture was taken and saved into the camera. An alternate option would be using Python, that will notify the user via email.

2. Use of Multiple Arduinos:

For this project, we will be using 2 different, Arduino. The first Arduino will be connected to the camera, while the second one will detect any motion and send the email notification.

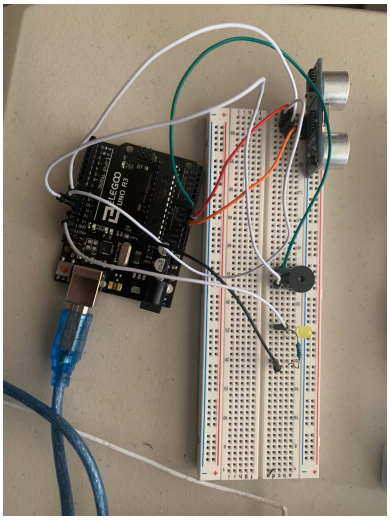
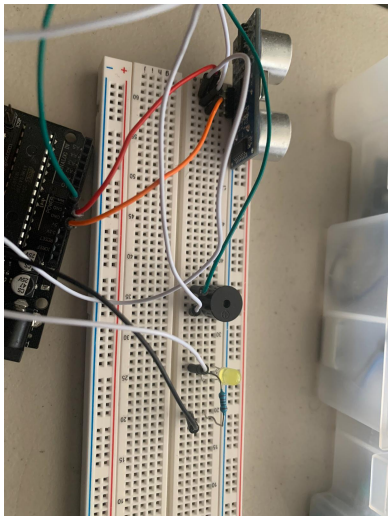
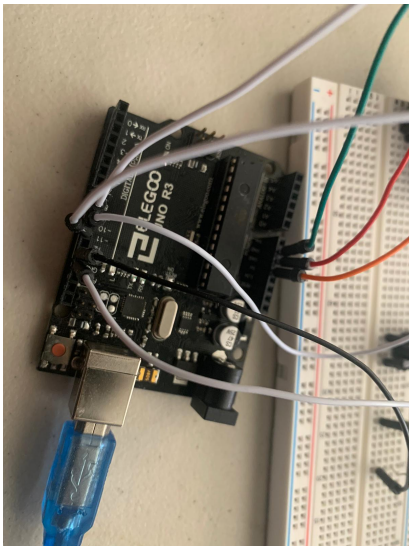
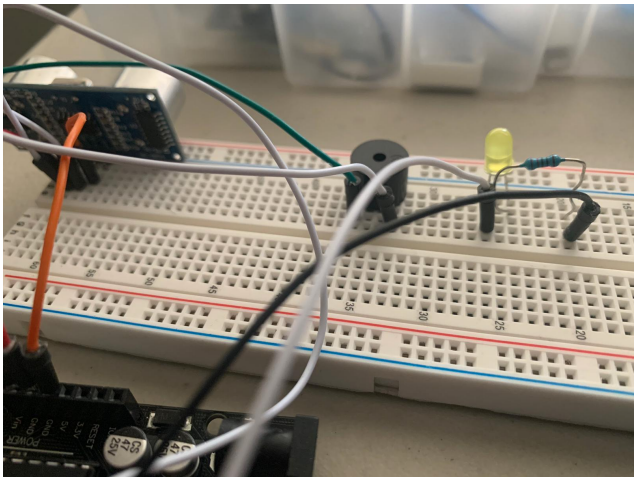
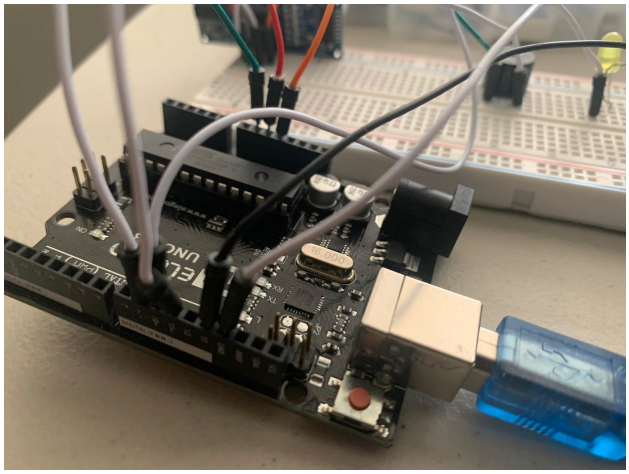
First Arduino:



For the first Arduino the signal of detection will trigger the Camera to take the picture. We used the Ground wire and Mic wire of the selfie stick and connected it to the breadboard with the photoresistor. The LED light will be the signal that the motion

has been detected. Once the LED lights up, the photoresistor's analog value will trigger the wire to capture the picture.

Second Arduino:

For the second Arduino, an ultrasonic sensor with a buzzer and led are connected. Here the ultrasonic sensor works as an input device and led and buzzer works as output devices . All arduinos are in serial connection, once the system is activated, the ultrasonic sensor starts working. Whenever a motion is detected the buzzer starts beeping and the led lights up indicating us with unwanted motion detection. And at the same time the Python program also takes the input from the ultrasonic sensor, and sends an email to the user saying unwanted motion was detected.

3. Communication of Multiple Arduinos:

Unfortunately due to the pandemic going on, we were unable to connect all of our arduinos together, But we worked around it by testing all of our arduinos individually, but let say it the pandemic would not have happened then, we would have connected our arduinos with serial connection, so both the arduinos could work in synchronize. If that would have happened, then the second arduino would have detected the motion, and trigger the Python software to send the email, and at the same time it would have triggered the first arduino as well, and the camera would have captured the picture.

4. Expected IO Devices:

First Arduino:

Input:
Smartphone Camera
Ultrasonic sensor
Photoresistor

Output:
Picture in Camera
Buzzer
LED light

Second Arduino

Input:
Ultrasonic sensor

Output:
Buzzer
LED light
Email notification

The inputs in this project are Smartphone Camera, Ultrasonic sensor, LED lights, and photoresistor. The output will be the buzzer that will set the alarm, notification of Email, light of LED, and picture from the Camera. The soundwave's will help the Ultrasonic sensor capture the motion and then buzzer and LED light will go on, and after that the photoresistor will capture the LED light and trigger to capture the picture. The email notification will be sent simultaneously when the LED light goes on.

5. Original Work:

For this project we implemented our work on sending the email notification without bluetooth. To capture a picture from the camera we had we had to reverse engineer our way to find a trigger so we used the wired-selfie stick and it's chip. We also set the time-frame for the notifications, motion sensor using Ultrasonic Sensor, and controlling the detection of any motion. Also, we added features for LED light and Buzzer which would be an alarm that goes off in case any intruder enters.

6. How to Build the Project:

The trig-Pin and Echo-Pin are placed on pin 9 and 10 respectively. The buzzer pin is placed on pin 11. The LED pin is placed on pin 13. The trigPin and Echo-Pin are connected to the ultrasonic sensor. Left side of the buzzer is connected to the GND and the right side is connected to pin 11. The LED will have a resistor attached and its wire will be connected to GND wire and the left side will be connected to pin 13. Place the photoresistor near the front of the led light. In Front of the LED light the two wires of the selfie-stick should be attached. We soldered the chip with the wire and connected it on a breadboard. Check the diagram to see which wire will go where.

7. How the Project Will be Used:

As of now, the project will have to first run in Arduino and after that run in IDLE. Once those two softwares are connected, the Project will be running. Any motion set in front of it's distance will set off the buzzer, and light the LED. This will be the indicator that the picture has been taken and the notification has been sent. The buzzer and LED light will go off for 4-5 seconds and one notification will be sent and a picture will be taken. After 8-12 seconds if the motion is still detected the cycle will continue again.

Required Supporting Materials:

1. Timeline of Development:

Week 5 : We formed the team and duchess about the project.

Week 6: Went in detail about the project, like Functionality and components.

Week 7: Discuss more about the project and finalized the project and the idea.

Week 8: Started getting all the required materials for the project, and formed Team 29.

Week 9: Started implementing the idea of ultrasonic sensor, Photoresistor, LED, and buzzer, and made sure it all worked together.

Week 10: Put the project on hold, as we didn't know what was going on or what would happen due to the virus.

Week 11: Got the update from the professor and re-discuss upon the project, and started to work on the project but by yourself and implemented the idea of the camera.

Week 12: Stared debugging the code, and worked on python software.

Week 13: Started testing and making sure that everything works properly.

Week 14: Made the demo, and our presentation for EXPO.

Week 15: Turn everything regarding the project in.

2. Final List of Materials Needed:

2 Arduino
2 Ultrasonic sensor
2 Photoresistor
2 LEDs
2 Buzzer
1 Cell phone
1 self stick
Multiple wires

3. Final List of References:

Email Communication
https://realpython.com/arduino-python/

Camera Trigger via wired selfie stick

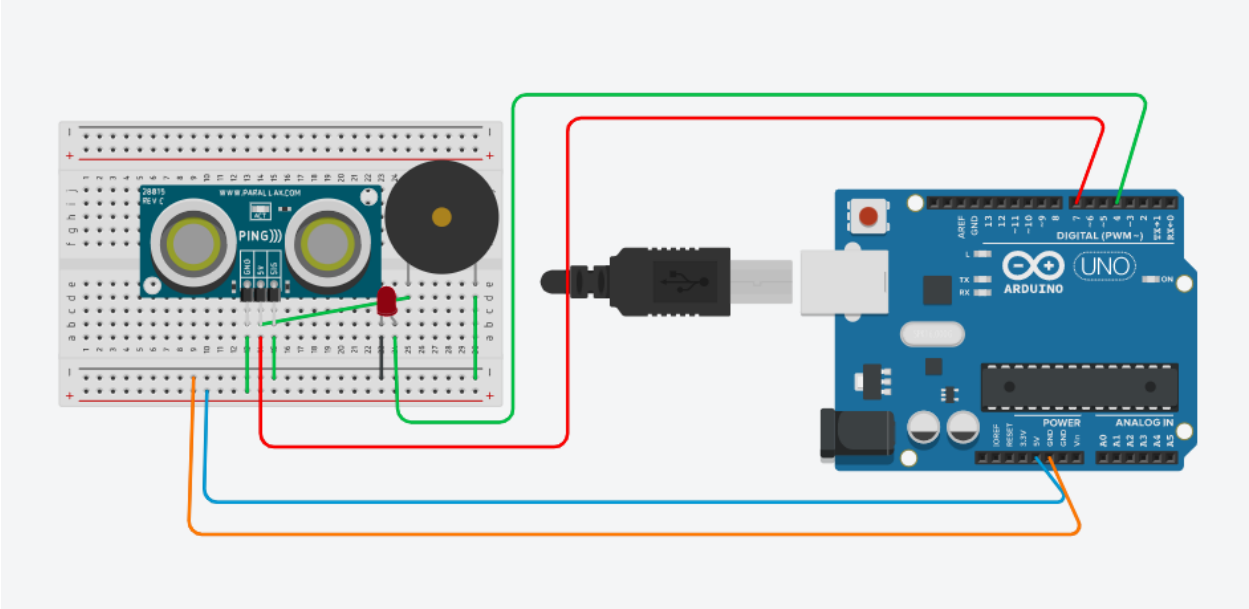
https://blog.medien.ifi.lmu.de/swh/2018/09/11/how-to-build-a-smartphone-ca mera-trigger/

LDR-WIRE Trigger

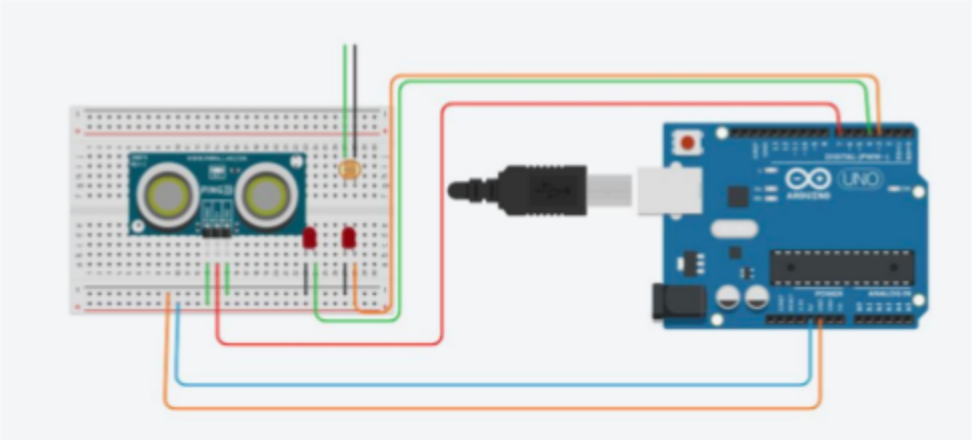
https://create.arduino.cc/projecthub/walid-mafuj/android-motion-detector-ca
mera-with-arduino-mcu-306789

4. Final Diagram:

Sending Email and turning on the Alarm



Capturing Picture from Camera



5 Code:

Code of First Arduino:

```
// defines pins numbers
const int trigPin = 9;
const int echoPin = 10;
const int buzzer = 11;
const int ledPin = 13;

// defines variables
long duration;
int distance;
int safetyDistance;


void setup() {
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as an Input
pinMode(buzzer, OUTPUT);
pinMode(ledPin, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}


void loop() {
// Clears the trigPin
digitalWrite(trigPin, LOW);
delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echoPin, HIGH);

// Calculating the distance
distance= duration*0.034/2;

safetyDistance = distance;
```

```
if (safetyDistance <= 30){
  digitalWrite(ledPin, HIGH);
  delay(3000);
}
else{
  digitalWrite(ledPin, LOW);
}

// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.println(distance);
}
```

Code of Second Arduino:

```
// defines pins numbers
const int trig_Pin = 9;
const int echo_Pin = 10;
const int buzzer = 11;
const int ledPin = 13;

// defines variables
long duration;
int distance;
int safetyDistance;


void setup() {
pinMode(trig_Pin, OUTPUT); // Sets the trig_Pin as an Output
pinMode(echo_Pin, INPUT); // Sets the echoPin as an Input
pinMode(buzzer, OUTPUT);
pinMode(ledPin, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}


void loop() {
// Clears the trigPin
digitalWrite(trig_Pin, LOW);
delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trig_Pin, HIGH);
delayMicroseconds(10);
digitalWrite(trig_Pin, LOW);
```

```
// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echo_Pin, HIGH);

// Calculating the distance
distance= duration*0.034/2;

safetyDistance = distance;
if (safetyDistance <= 5){
  digitalWrite(buzzer, HIGH);
  digitalWrite(ledPin, HIGH);
  Serial.println("Movement");
  delay (4000);
  digitalWrite(buzzer, LOW);
  delay (5000);
}
else{
  digitalWrite(buzzer, LOW);
  digitalWrite(ledPin, LOW);
}


}
```

Code of Python (Second Arduino):

```
import time
import serial
import smtplib
TO = 'testfor362@gmail.com'
GMAIL_USER = 'testsendemail362'
GMAIL_PASS = 'Password here'
SUBJECT = 'Alert'
TEXT = 'Your sensor detected movement, someone access your are; and the divice
has taken a pictuer!!!'


ser = serial.Serial('/dev/tty.usbmodem14101', 9600)


def send_email():
    print("Sending Email")
    smtpserver = smtplib.SMTP("smtp.gmail.com",587)
```

```
    smtpserver.ehlo()
    smtpserver.starttls()
    smtpserver.ehlo
    smtpserver.login(GMAIL_USER, GMAIL_PASS)
    header = 'To:' + TO + '\n' + 'From: ' + GMAIL_USER
    header = header + '\n' + 'Subject:' + SUBJECT + '\n'
    print (header)
    msg = header + '\n' + TEXT + ' \n\n'
    smtpserver.sendmail(GMAIL_USER, TO, msg)
    smtpserver.close()

while True:
    message = ser.readline()
    print(message)
    if message[0] == 'M' :
        send_email()
    time.sleep(0.5)
```