

The Trespassing Alarm Security System

By Sang Joon Lee

Description of the product:

Sometimes, I imagine a robber breaking into my house when everyone is asleep. I always double check the door is properly locked before I go to bed, but it will be more secure if there is another security system. For this reason, I designed the trespassing alarm system for my final project. If someone passes the front door, the ultrasonic sensor will detect this. After that, the mini loudspeaker will make emergency alarm sound, and red LED will turn on, and the servo motor will turn on the room light by clicking the switch. After several tests, I checked this system properly works, and it was quite useful. Now, I can sleep more comfortably than before thanks to the trespassing alarm security system.

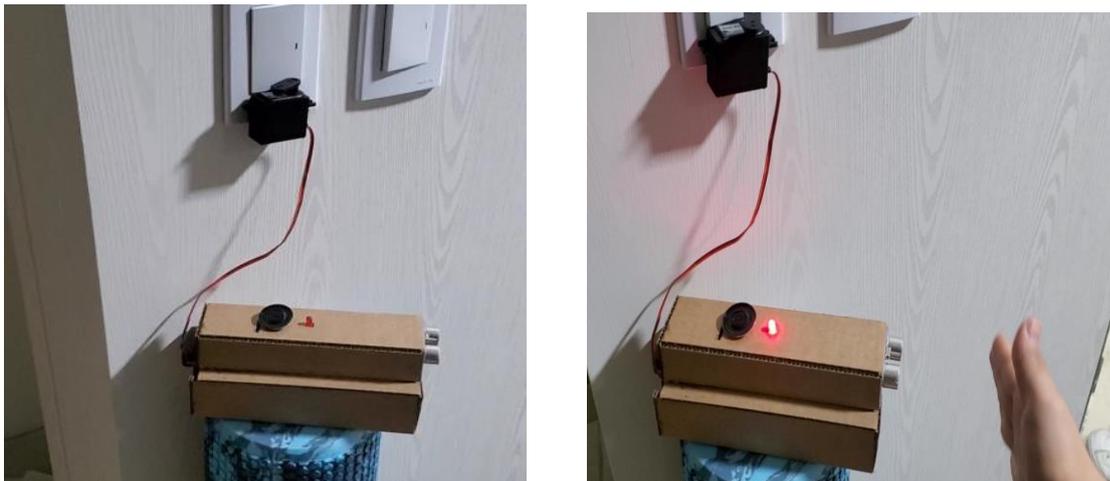


Figure 1: The trespassing alarm security system installed on the front door of my house. The servo motor is put on the light switch, and the ultrasonic sensor is pointing the entrance. The photos are the alarm OFF state (left) and the alarm ON state (right).



Figure 2: This is a photo of wide view of installed trespassing alarm security system (left). Also, it is important to measure the exact distance to the wall from the sensor to avoid the sensor detecting the wall as a person. So, I measured this distance for my house's front door and set it as a maximum distance in the arduino code.

Circuit:

Parts list:

Item	Description	Qty
Adafruit HUZZAH32 – ESP32 Feather Board	Black	1
Mini loudspeaker	Yootop 1W 8 Ω Round Internal Magnet	1
Ultrasonic sensor	HC-SR04, 4 pins	1
LED	Red, SLR-56VR	1
Li-ion battery	3.7V, 2200mAh	1
6V battery pack	6V, Including four AA batteries	1
Servo Motor	Black, Analog Servo-HD-3001HB	1
Motor hub	2 arms, Black	1
Switch	Black, connected to Li-ion battery	1
Resistor	100 Ω	1

(1) Microcontroller: The Li-ion battery is used for power the microcontroller instead of connecting USB for portability. It controls the ultrasonic sensor, loudspeaker, and LED by arduino code in it.

(2) Sensor: HC-SR04 Ultrasonic sensor is used for this machine. It can measure the distance to the enough size of object from the sensor. It works at about 2cm to 400cm away, but the recommended range of distance is 10cm-250cm. The power and logic voltage is about 5V. Therefore, 6V battery pack is used for power this sensor.

(3) Actuators: There are three actuators are used for this machine. First, the mini loudspeaker makes the emergency sound. The sound can be controlled by changing frequency in the arduino code. Second, the Red LED turns on when the security system is activated, and the 100 Ω resistor is connected to LED for the optimal amount of current. Both loudspeaker and LED are powered by microcontroller. For the last, the servomotor rotates 180 degrees after activation and come back when the alarm is off. It can be used various purpose, but I used it for turn on the light switch. It is powered by 6V battery pack, so it has 0.12sec/60° of speed and 4.4kg·cm of stall torque at 6V.

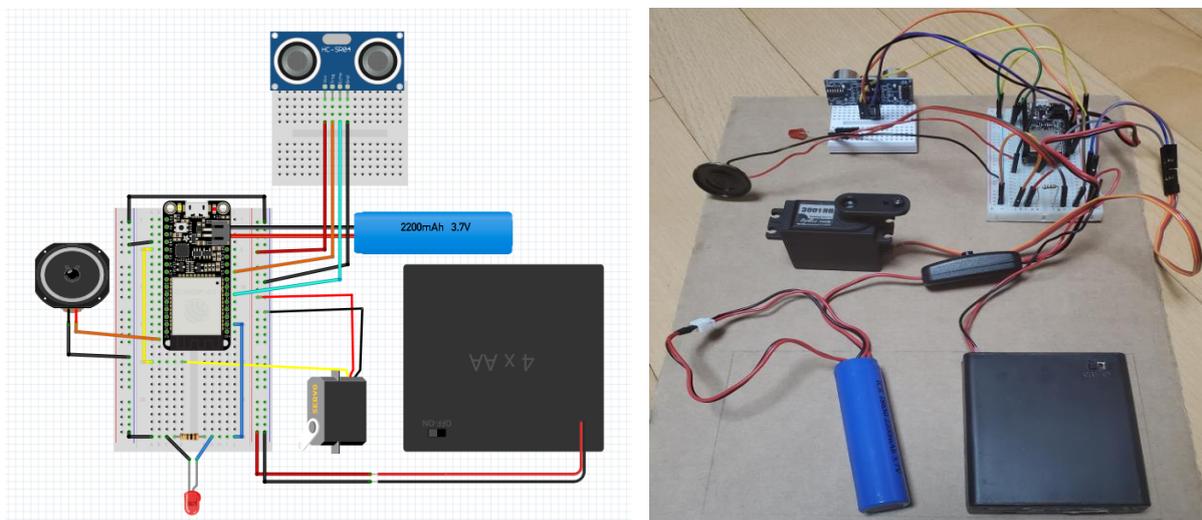


Figure 3: The circuit diagram for this machine (left) and the circuit components (right).

Electromechanical details:

The housing of the circuit is made of cardboard. The circuit components are exactly fit into the housing, so it prevents the parts from moving around. On top of the base housing, an additional cardboard structure is stacked up with the sensor and holes for LED and loudspeaker.

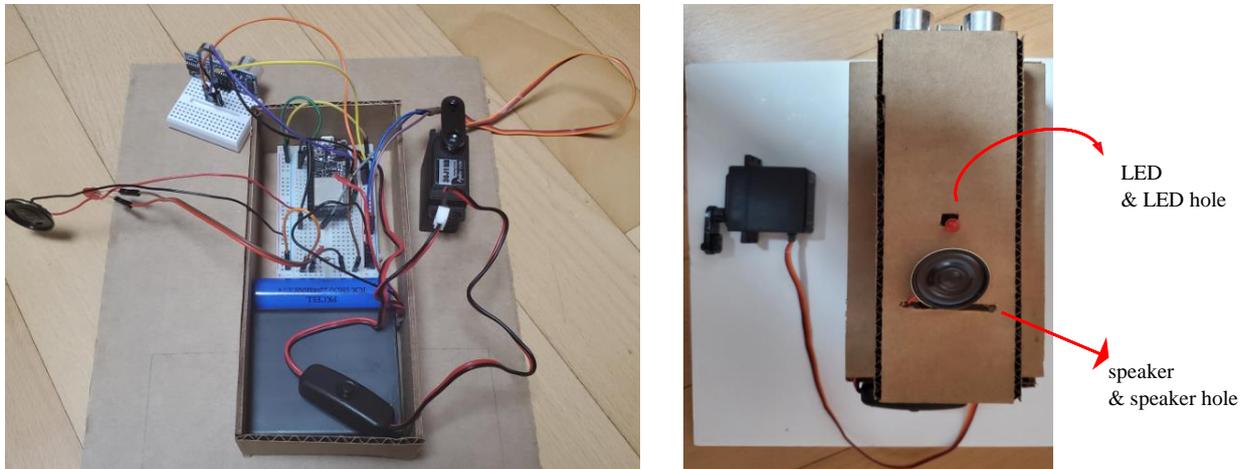


Figure 3: The cardboard box for the product (left) and the top view of final product (right). For the quality of the product, there are suitable sizes of holes for the actuator parts.

There is also a hole for the Arduino Micro USB port. It will be used to modify the Arduino code or charge the Li-ion battery in the future. There is a switch for the Li-ion battery, so it can be used as the ON/OFF switch for the whole machine. The On/Off switch of the 6V battery pack can be controlled through the hole indicated in the right below photo, and the cable of the servomotor is also used that hole to go out.

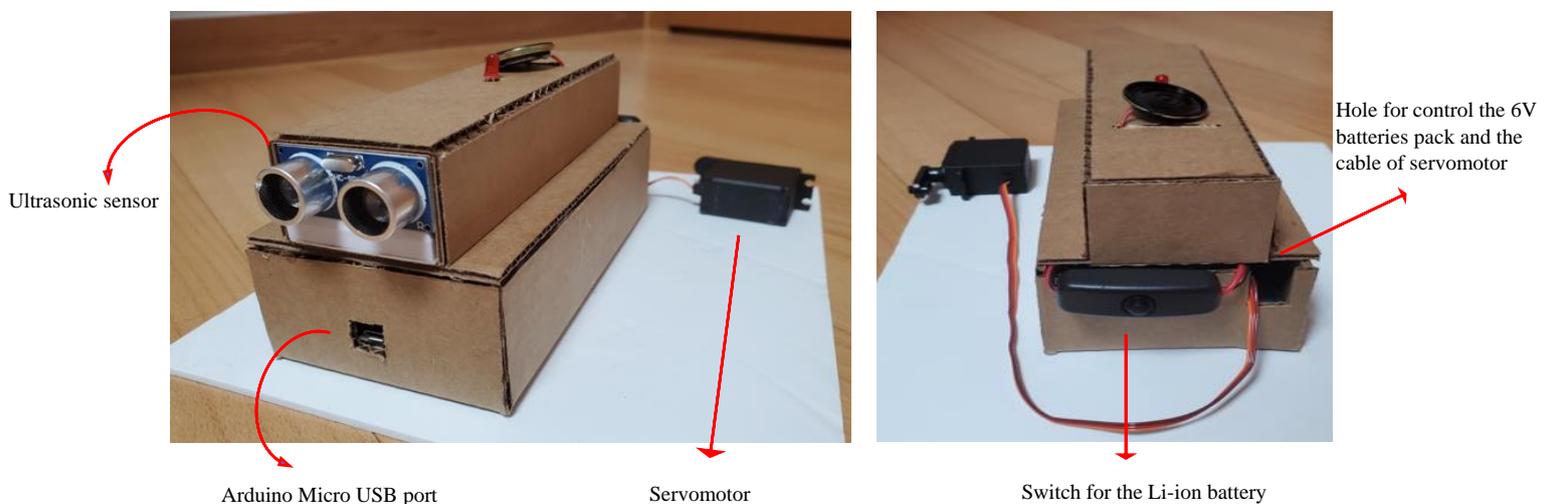
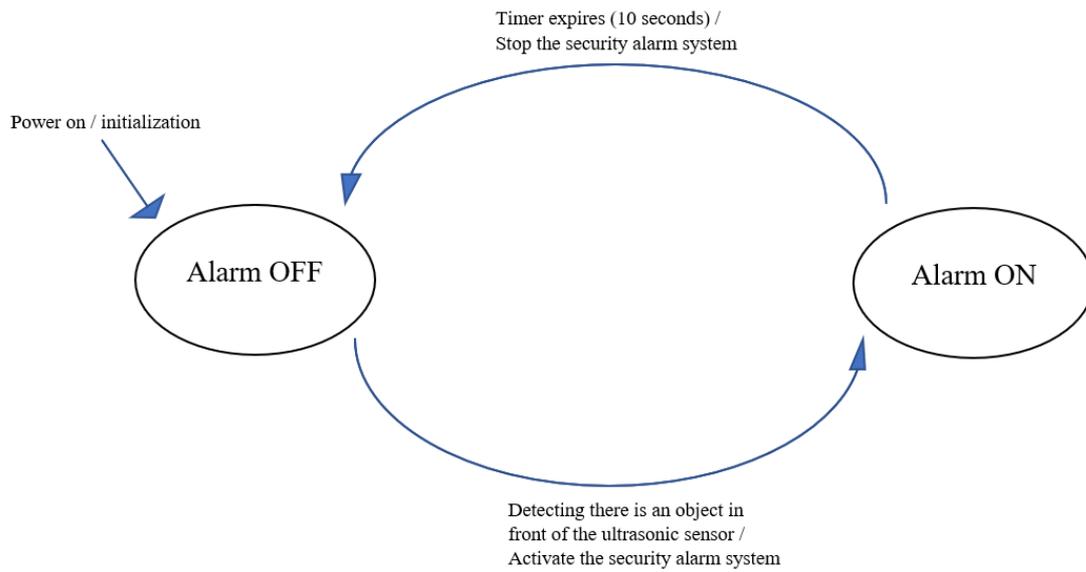


Figure 4: This is the photos of final product - 3/4view (left), back view (right).

Finite state machine:



: There are two states – Alarm OFF and Alarm ON. At the Alarm OFF state, there is nothing to detect on the sensor, and none of the actuator is working. If someone is detected by the sensor, the security alarm system is activated, and the state transitions to the Alarm ON state. At this state, the LED turns on, and the loudspeaker makes emergency sound, and the servomotor rotates 180deg. After 10 seconds, the state transitions back to the Alarm OFF state.

Appendix I: Arduino Code

```
// %%% ULTRASONIC SENSOR %%%
#include <NewPing.h>

#define Trigger_Pin 13
#define Echo_Pin 27
#define Max_Distance 200 //setting max distance

NewPing sonar (Trigger_Pin, Echo_Pin, Max_Distance);

int distance = 0;

// %%% LED %%%
#define LED_Pin 14

// %%% Servo %%%
#include <ESP32Servo.h>
Servo myservo;
int pos = 0;
#define Servo_Pin 26

// %%% Timer %%%
bool printToggle = true;
hw_timer_t * timer = NULL;

void IRAM_ATTR isrTimer(){
printToggle = true ;
}

void setup() {
  Serial.begin(9600);

  //Speaker set up
  ledcSetup(0,100000,12);
  ledcAttachPin(21,0);

  //LED set up
  pinMode(LED_Pin,OUTPUT);

  //Servo setup
  ESP32PWM::allocateTimer(3);
  myservo.setPeriodHertz(50); // standard 50 hz servo
  myservo.attach(Servo_Pin, 600, 2400); // Controls sweep angle: min/max of
600us and 2400us

  //Timer setup
  timer = timerBegin(0,80,true);
  timerAttachInterrupt(timer,&isrTimer, true);
  timerAlarmWrite(timer, 100000, true); //repeat every 0.1 seconds
  timerAlarmEnable(timer);
}

void loop() {

  switch(printToggle){
    case true:
      distance = sonar.ping_cm(); // detect the distance from the
sensor in cm scale
```

```

        Serial.print("The Distance is: ");
        Serial.println(distance); //For the purpose of checking the
result in serial monitor
        printToggle = false;
        break;

        default:
        break;
    }

    //Detecting trespassing
    if(distance > 0){
        Serial.println("Trespassing detected!"); //For the purpose of checking
the result in serial monitor

        myservo.write(0);
        myservo.write(180); // Rotate the servo motor for 180deg

        digitalWrite(LED_Pin, HIGH); //Turn on the LED

        for(int i = 0; i<5; i++){ //Making emergency alarm sound for
10seconds
            ledcWriteTone(0,800);
            delay(500);
            ledcWriteTone(0,1600);
            delay(500);
        }

        ledcWriteTone(0,0); //Turn off the alarm sound
        myservo.write(180);
        myservo.write(0); // Rotate back to original position
        digitalWrite(LED_Pin, LOW); //Turn off the LED
    }
}

```