



Always On Guard

Mechatronics Engineering Project

Prof. Dr. Ayman El Badawi

Group Members	ID Numbers	Group
Andrew Emile	28-3925	T-22
Karim Medhat	28-5468	T-22
Michel Sherif	28-13337	T-22

Submitted on 2/5/2015

Table of Contents

Design Summary 3

 Overview 3

 House Model..... 4

System Details 5

 Schematic 5

Design Evaluation 6

Partial Parts List 7

Appendix..... 8

 Code 8

 Wiring Details 14

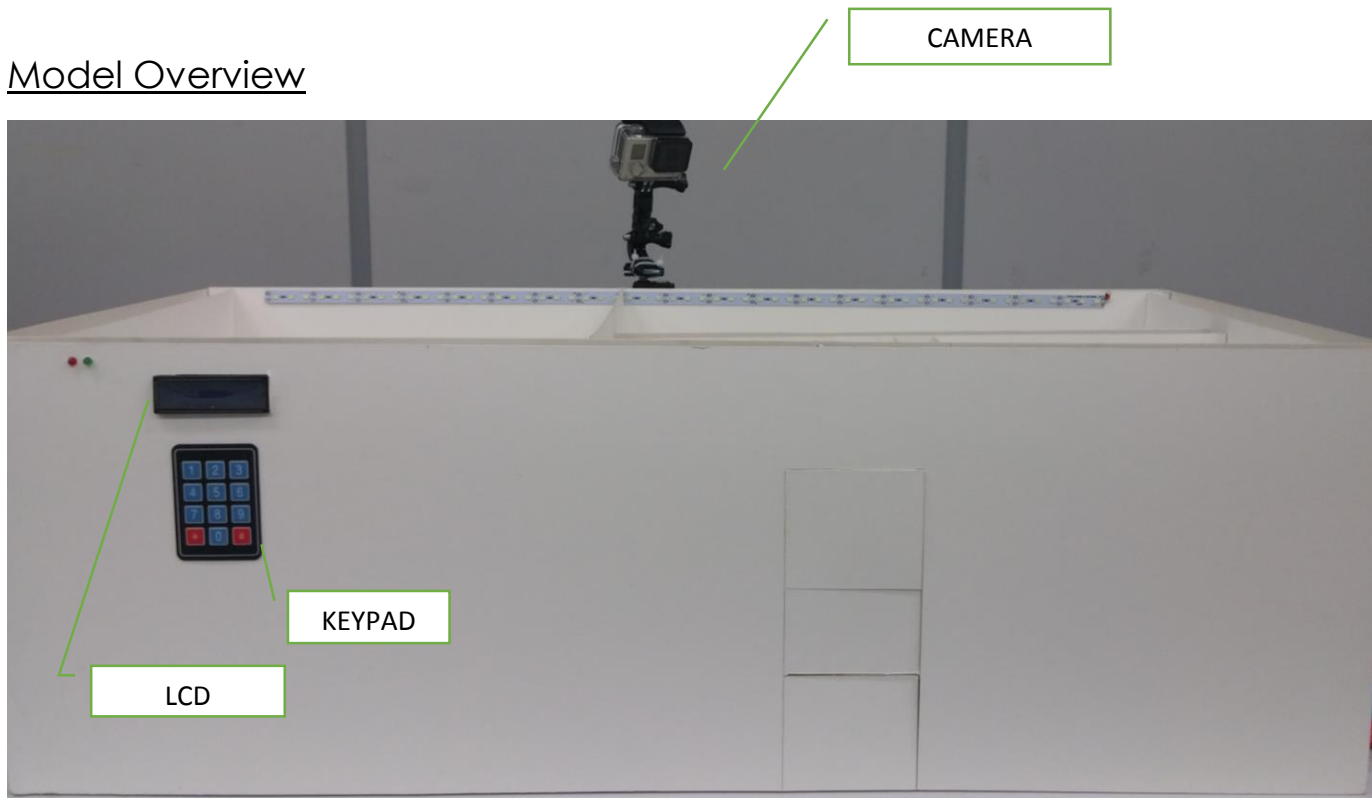
Overview

"Always On Guard" is a Smart Home Security System. A house of 3 Rooms and a Garage will be modeled where any motion will be sensed by a PIR "Passive Infrared Sensor" by detecting the changes in the amount of infrared impinging upon it or and doors opened through Infrared proximity Sensors.

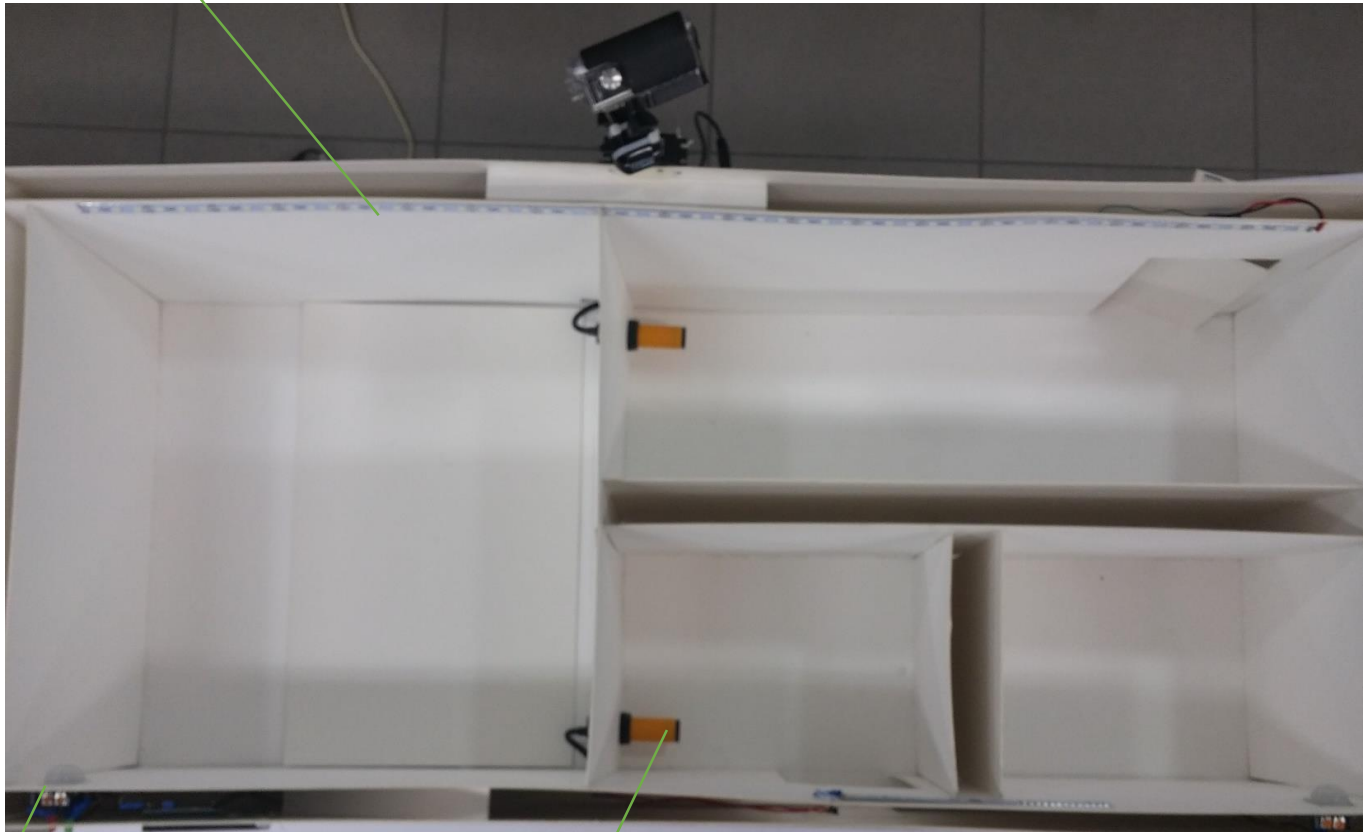
The Arduino will receive a signal from the PIR and the Proximity Sensor upon any change triggering both an alarm buzzer and LED Lights then a GoPro Camera will be activated and controlled by the servo motor to video the room where the thief is!

The "Always On Guard" Security System Can be Enabled and Disabled by a Security code that can be given by a Keypad at the Front Door!

Model Overview



LED Strip



PIR

Proximity Sensor

Beneath the Model



PIR Sensor



Flex LED



Buzzer Alarm

Overview function diagrams where slight modifications will be done to the real one.



Design Evaluation

The Home Security System Operated as desired such that if system is activated by the correct pin the program initializes the camera position and the rest of the variable ready to detect motion. Once motion is detected in any room piezo buzzer alarm is triggered then the camera goes to record the room where motion is detected. Until System is deactivated by reentering the Security password disabling the whole system ignoring any motion detection unless System is activated by entering the correct password is entered once again.

Partial Parts List

Part	Price
Bread Board	90
Relay	25
Arduino Mega 2560	160
Servo Motor 6kg.cm	100
3x 4 Matrix Array Keypad	20
PIR Motion Sensor	40
Flexible LED Strip	45
GoPro Camera	Already Available
LCD Arduino Module "I2C"	90
Proximity Sensor	70

- Breadboard functions as an external power source (5v) for sensors, actuators...etc
- Relay , to trigger the 12V LED strip by an arduino Signal
- 3X4 Keypad as Input to activate/deactivate the system
- Arduino ATMmega2560 16MHz 40mA Current as a microcontroller
- Camera for video Recording.

Appendix

Arduino Code:

```
#include <Wire.h>
#include <Password.h>
#include <Keypad.h>
#include <Servo.h>
#include <LiquidCrystal_I2C.h>

//////////SENSORS & LEDs //////////
int pirPin1 = 2; // Bedroom Pir
int pirPin2 = 34; // Garage Pir
int proxPin1 = 8; // Front Proximity
int proxPin2 = 9; // Back Proximity
int greenLED = 30;
int redLED = 32;
int LED_Relay = 14; //LED Strip Relay

//////////Password//////////
Password password = Password("2015");

//////////Servo Motor Intialisation//////////
Servo camservo; // Servo Object Creation
int pos = 100; // Intial Servo Position

//////////Other Variables//////////
int HomeZone = 0;
int pass_pos = 11; // Password Input Position
int buzzerPin = 22;
boolean systemActive = false;
boolean systemStatus = false;
int calibrationTime = 30; //the time we give the sensor to calibrate
boolean motionGarage = false;
boolean motionBack = false;
boolean motionFront = false;
boolean motionBed = false;
//////////Keypad Creation//////////
const byte ROWS = 4;
const byte COLS = 3;
char keys[ROWS][COLS] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
byte rowPins[ROWS] = {53, 51, 49, 47}; //Rows Pinouts
byte colPins[COLS] = {45, 43, 41}; //Columns Pinouts
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```


//////// Assignign arduino pins to LCD display module////////

LiquidCrystal_I2C lcd(0x27,16,2);

void setup() {

Serial.begin(9600); // Setup Baidu Rate

//Setup Sensors and Relay

pinMode(pirPin1, INPUT); //Bedroom

pinMode(pirPin2, INPUT); //Garage

pinMode(proxPin1, INPUT); //Front door

pinMode(proxPin2, INPUT); //Back door

pinMode(buzzerPin, OUTPUT);

pinMode(LED_Relay, OUTPUT); //LED Strip

digitalWrite(redLED, LOW);

digitalWrite(greenLED, HIGH);

digitalWrite(pirPin1, LOW);

digitalWrite(pirPin2,LOW);

lcd.init();

lcd.init();

lcd.backlight();

displayScreen();

// Default Display Screen

camservo.attach(12); // Attach Servo to Pin 9

keypad.addEventListener(keypadEvent);

camservo.write(pos);

}

void loop() {

// Get Pin//

keypad.getKey();

//Check For Motion//

if (systemActive == true){

if (digitalRead(proxPin1) == LOW){

delay(200);

if(digitalRead(proxPin1) == LOW && !motionFront)

{HomeZone = 0;

motionFront = true;

motionBed= false;

motionGarage = false;

motionBack = false;

MotionDetected();}

}

```

if (digitalRead(proxPin2) == LOW){
  delay(200);
  if(digitalRead(proxPin2) == LOW && !motionBack )
  { HomeZone = 2;
    motionBed= false;
    motionGarage = false;
    motionFront = false;
    motionBack = true;
    MotionDetected();}
  }
  if (digitalRead(pirPin1) == HIGH){
    delay(100);
    if (digitalRead(pirPin1) == HIGH && !motionGarage)
  {HomeZone = 1;
    motionBed= false;
    motionGarage = true;
    motionFront = false;
    motionBack = false;
    MotionDetected();}}
  if (digitalRead(pirPin2)== HIGH){
    delay(100);
    if (digitalRead(pirPin2) == HIGH && !motionBed)
    { HomeZone = 3;
      motionBed= true;
      motionGarage = false;
      motionFront = false;
      motionBack = false;
      MotionDetected();}
    }

  }

}

////////// Functions ////////////
////Motion Detected////
void MotionDetected(){
  int expected_pos; // Excpexcted Servo Position
  int rotateSteps; // Steps Of Rotation Till Specified Angle
  digitalWrite(LED_Relay, HIGH);
  digitalWrite(buzzerPin,HIGH);
  //tone(22,2000);
  password.reset();
  systemStatus = true;
  // LCD EDIT
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("BREACH..!!");
  lcd.setCursor(0,1);
  if (HomeZone == 0)
  {
    lcd.print("Frontdoor Open");
  }
}

```

```

    expected_pos = 120;
    delay(500);
}
if(HomeZone == 1){
    expected_pos = 125;
    lcd.print("Motion @ Bedroom");
    delay(500);
}
else if(HomeZone == 2){
    expected_pos = 150;
    lcd.print("Backdoor Open");
    delay(500);
}
else if(HomeZone == 3){
    expected_pos = 65;
    lcd.print("Motion @ Garage");
    delay(500);
}

if (expected_pos > pos) {rotateSteps = 2;}
else {rotateSteps = -1;}

for (pos = pos; pos != expected_pos; pos += rotateSteps) {
    camservo.write(pos);           // tell servo to go to position in variable 'pos'
    delay(5);                      // waits 5ms for the servo to reach the position
}

}

//KEYPAD EVENTS //
void keypadEvent(KeypadEvent eKey) {
    switch (keypad.getState()) {
        case PRESSED:
            if (pass_pos - 11 >= 5) {
                return ;
            }
            lcd.setCursor((pass_pos++), 0);
            switch (eKey) {
                case '#':           // # is to validate password
                    pass_pos = 11;
                    checkPassword();
                    break;
                case '*':           // * is to reset password attempt
                    password.reset();
                    pass_pos = 11;
                    break;
                default:
                    password.append(eKey);
                    lcd.print("*");
            }
    }
}

```

```

}
void checkPassword() {
  if (password.evaluate()) {
    if (systemActive == false && systemStatus == false){
      activate();
    }

    else if (systemActive == true || systemStatus == true) {
      deactivate();
    }

  }
  else {
    invalidCode();
  }
}

}

void activate(){

  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("SECURITY SYSTEM");
  lcd.setCursor(4,1);
  lcd.print("ACTIVATED");
  digitalWrite(redLED,HIGH);
  digitalWrite(greenLED,LOW);
  systemActive = true;
  password.reset();
  delay(1000);

}

void deactivate() {

  systemStatus = false;
  systemActive = false;
  digitalWrite(redLED, LOW);
  digitalWrite(greenLED, HIGH);
  digitalWrite(buzzerPin,LOW);
  motionFront = false;
  motionBack = false;
  motionBed = false;
  motionGarage = false;
  lcd.clear();
  lcd.print("SECURITY SYSTEM");
  lcd.setCursor(2,1);
  lcd.print("DEACTIVATED");
  password.reset();
  delay(2000);
  displayScreen();
}

```

```

}
void invalidCode(){
  if (systemActive == false){
    lcd.clear();
    lcd.setCursor(1,0);
    lcd.print("INVALID CODE!");
    lcd.setCursor(0,1);
    lcd.print("PLEASE TRY AGAIN");
    systemActive = false;
    digitalWrite(redLED,LOW);
    digitalWrite(greenLED,HIGH);
    password.reset();
    delay(1000);
    displayScreen();
  }
  else {

    lcd.clear();
    lcd.setCursor(1,0);
    lcd.print("INVALID!");
    lcd.setCursor(0,1);
    lcd.print("PLEASE TRY AGAIN");
    motionGarage = false;
    motionBack = false;
    motionFront = false;
    motionBed = false;
    password.reset();
    delay(1000);
    loop();
  }
}
void displayScreen() // Displaying start screen for users to enter PIN
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Enter PIN:");
  lcd.setCursor(0,1);
  lcd.print("ALWAYS ON GUARD!");
}
void calibrationScreen(){
  lcd.clear();
  lcd.setCursor(4,0);
  lcd.print("WELCOME!");
  lcd.setCursor(0,1);
  lcd.print("Please Wait...");
  for(int i = 0; i < calibrationTime; i++){
    delay(1000);
  }
  displayScreen();
}
}

```

Wiring Details

Component	Pin
Bedroom PIR	2
Garage PIR	34
Front Proximity Sensor	8
Back Proximity Sensor	9
Servo Motor	12
Red LED	32
Green LED	30
Keypad	41,43,45,47,49,51,53

PIR Connections	
Red	Vcc 5v
Brown	GND
Yellow	Pin (2 for Bedroom, 34 for Garage)

Proximity Connections	
Brown	Vcc 5v
Blue	GND
Black	Pin (8 for front, 9 for back)

Servo Connections	
Red	Vcc 5v
Brown	GND
Orange	Pin 12 (PWM)

LCD Connections	
Red	Vcc 5v
Black	GND
Green	Pin 20 (SCL)
Yellow	Pin 21 (SDA)