#include <EEPROM.h>

int rx = 0;

int tx = 1;

int buzzer = 5;

// detectores

int ultrasonido = 8;

int pirPin = 7;

int pirPin2 = 13;

// leds

int ledNaranja = 3;

int ledVerde = 12;

int ledPinRojoH1 = 9;

int ledPinRojoH2 = 10;

int ledPinRojoH3 = 11;

byte val; // value read on from the serial port

byte dataR; //it contains the byte read from EEPROM

enum State { NONE, FULL\_ALARM, HALF\_ALARM, ALARM\_SPLIT };

State estado;

long readUltrasonicDistance(int pin);

void loopAlarmaCompleta();

void enciendeMediaAlarma();

void loopMediaAlarma();

void setup()

{

// 112500

Serial.begin (112500);

delay(500);

pinMode(ledNaranja, OUTPUT);

pinMode(ledVerde, OUTPUT);

pinMode(buzzer, OUTPUT);

pinMode(ledPinRojoH1, OUTPUT);

pinMode(ledPinRojoH2, OUTPUT);

pinMode(ledPinRojoH3, OUTPUT);

pinMode(pirPin, INPUT\_PULLUP);

pinMode(pirPin2, INPUT\_PULLUP);

pinMode(ultrasonido, INPUT);

//initialize UART pins

pinMode (rx, OUTPUT);

pinMode (tx, INPUT);

estado = NONE;

}

void apagaBuzzer() {

noTone(buzzer);

}

void enciendeLedVerde() {

digitalWrite(ledVerde, HIGH);

}

void enciendeLedNaranja() {

digitalWrite(ledNaranja, HIGH);

}

void apagaLedVerde() {

digitalWrite(ledVerde, LOW);

}

void apagaLedNaranja() {

digitalWrite(ledNaranja, LOW);

}

void buzzer1() {

tone(buzzer,350);

}

void buzzerClaveCorrecta() {

delay(200); // tono de clave correcta

tone(buzzer,500);

delay(100);

noTone(buzzer);

tone(buzzer,600);

delay(100);

noTone(buzzer);

tone(buzzer,800);

delay(100);

noTone(buzzer);

}

void buzzerClaveIncorrecta() {

delay(200); // tono de clave incorrecta

tone(buzzer,300);

delay(100);

noTone(buzzer);

tone(buzzer,100);

delay(100);

noTone(buzzer);

}

void buzzerAlarma() {

tone(buzzer, 800); // play 400 Hz tone for 400 ms

delay(200);

tone(buzzer, 500); // play 800Hz tone for 400ms

delay(200);

noTone(buzzer);

}

bool intrusos = false;

bool alarmOn = false;

bool halfAlarmOn = false;

// los tres siguientes solo se activan cuando

// queremos encender cada hab por separado

bool hab1On = false;

bool hab2On = false;

bool hab3On = false;

void desactivaDeteccion() {

digitalWrite(ledPinRojoH1, LOW);

digitalWrite(ledPinRojoH2, LOW);

digitalWrite(ledPinRojoH3, LOW);

apagaBuzzer();

enciendeLedVerde();

apagaLedNaranja();

alarmOn = false;

halfAlarmOn = false;

}

void loop()

{

if(Serial.available() > 0)

{

val = Serial.read(); //read the next byte

/\*

0 - enciende led verde

1 - apaga led verde

2 - enciende led naranja

3 - apaga led naranja

4 - apaga buzzer

5 - enciende media alarma

6 - desactiva deteccion

7 - activa habitacion 1

8 - activa habitacion 2

9 - activa habitacion 3

10 - change pass (cambio de contraseña)

\*/

if (val == '0') {

desactivaDeteccion();

} else if (val == '1') {

apagaLedVerde();

} else if (val == '2') {

enciendeLedNaranja();

alarmOn = true;

} else if (val == '3') {

buzzerClaveIncorrecta();

} else if (val == '4') {

apagaBuzzer();

} else if (val == '5') {

enciendeMediaAlarma();

} else if (val == '6') {

desactivaDeteccion();

// tono de clave correcta?

buzzerClaveCorrecta();

} else if (val == '7') {

// activa hab 1

hab1On = true;

} else if (val == '8') {

// activa hab 2

hab2On = true;

} else if (val == '9') {

// activa hab 3

hab3On = true;

}

} else {

//buzzerClaveCorrecta();

}

if (alarmOn) {

loopAlarmaCompleta();

} else if (halfAlarmOn) {

loopMediaAlarma();

}

delay(200);

}

long readUltrasonicDistance(int pin)

{

pinMode(pin, OUTPUT); // Clear the trigger

digitalWrite(pin, LOW);

delayMicroseconds(2);

// Sets the pin on HIGH state for 10 micro seconds

digitalWrite(pin, HIGH);

delayMicroseconds(10);

digitalWrite(pin, LOW);

pinMode(pin, INPUT);

// Reads the pin, and returns the sound wave travel time in microseconds

return pulseIn(pin, HIGH);

}

void loopAlarmaCompleta()

{

bool desactivando = false; // utilizamos el valor desactivando como valor intermedio

// hasta que no nos informen los dos sensores de que realmente no estan detectando

// a nadie no podemos encender el led verde como que todo está en orden. Tenemos

// que esperar a que ambos comprueben que no hay nadie para encender el verde

int proximity = digitalRead(pirPin);

delay(100);

int proximity2 = digitalRead(pirPin2);

delay(100);

int cm = 0.01723 \* readUltrasonicDistance(ultrasonido);

/\*Serial.begin (9600);

delay(300);

Serial.print("cm: ");

Serial.println(cm);

delay(300);

Serial.begin (112500);

\*/

if (proximity == HIGH) // If the sensor's output goes low, motion is detected

{

digitalWrite(ledPinRojoH1, HIGH);

intrusos = true;

}

else

{

digitalWrite(ledPinRojoH1, LOW);

desactivando = true;

}

if (proximity2 == HIGH) // If the sensor's output goes low, motion is detected

{

digitalWrite(ledPinRojoH3, HIGH);

intrusos = true;

desactivando = false;

}

else

{

digitalWrite(ledPinRojoH3, LOW);

}

if (cm < 200) {

digitalWrite(ledPinRojoH2, HIGH);

intrusos = true;

}

else

{

digitalWrite(ledPinRojoH2, LOW);

if (desactivando == true) {

intrusos = false;

}

}

if (intrusos) {

buzzerAlarma();

}

}

void enciendeMediaAlarma() {

enciendeLedNaranja();

apagaLedVerde();

halfAlarmOn = true;

}

// loop media alarma

// equivale al plan nocturno, donde van a estar activados dos de los tres sensores

// no queremos que el de la habitación nos detecte

void loopMediaAlarma()

{

bool desactivando = false; // utilizamos el valor desactivando como valor intermedio

// hasta que no nos informen los dos sensores de que realmente no estan detectando

// a nadie no podemos encender el led verde como que todo está en orden. Tenemos

// que esperar a que ambos comprueben que no hay nadie para encender el verde

// leemos el pir pin 1

int proximity = digitalRead(pirPin);

delay(100);

// leemos el ultrasonido

int cm = 0.01723 \* readUltrasonicDistance(ultrasonido);

/\*Serial.begin (9600);

delay(300);

Serial.print("cm: ");

Serial.println(cm);

delay(300);

Serial.begin (112500);

\*/

if (proximity == HIGH) // If the sensor's output goes low, motion is detected

{

//tone(buzzer, 350, 200);

digitalWrite(ledPinRojoH1, HIGH);

intrusos = true;

}

else

{

digitalWrite(ledPinRojoH1, LOW);

desactivando = true;

}

// comprobamos la distancia al ultrasonidos

// si es menor de 200 cm, encendemos el led rojo de la

// habitacion 2

if (cm < 200) {

// tone(buzzer, 350, 200);

digitalWrite(ledPinRojoH2, HIGH);

intrusos = true;

}

else

{

digitalWrite(ledPinRojoH2, LOW);

if (desactivando == true) {

intrusos = false;

}

}

if (intrusos) {

buzzerAlarma();

}

}