

# ECE 110 Honor Section Final Lab Report

## Virtual Door Safety System

David Zhao, Alan Cheng, Ben Pan

### I. Introduction

#### a. Problem Statement

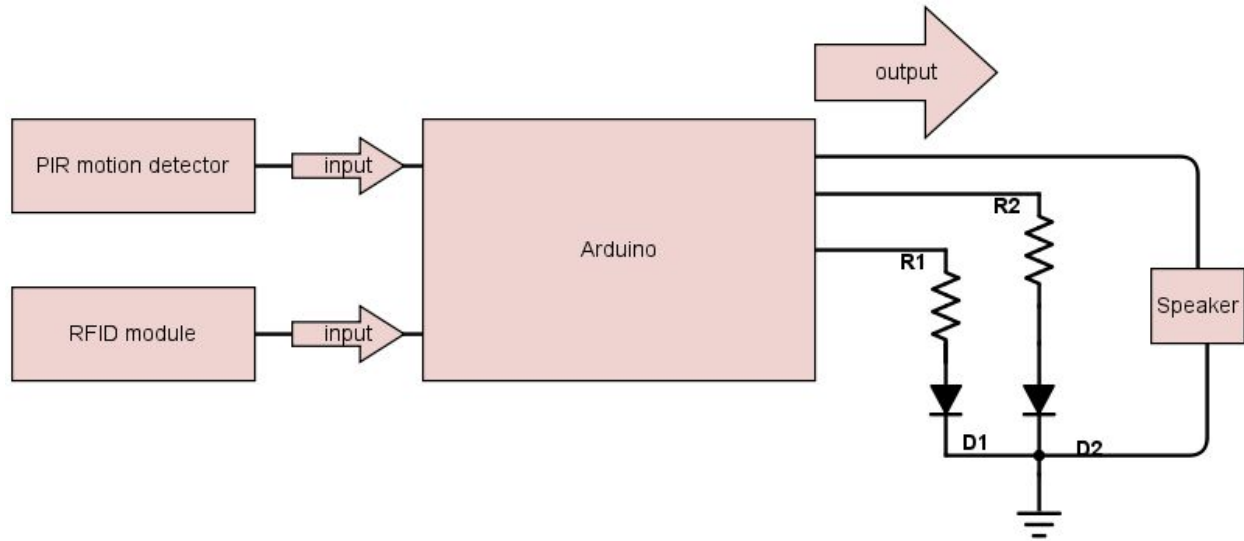
Many students complain about use traditional keys for door room such as easy to lose and not stylish. We observed that most students bring their i-cards every day, so we want to make it to unlock student's' room door. Besides, we want to improve the safety in residential hall by sending out alert when unauthorized person trespassed the room.

#### b. Overview of proposed statement

We will stick a small rfid tag onto the i-card. Then we will place a rfid detector on the door. Each rfid tag has a unique id. When the detector scan the tag with correct id, it will unlock the door. In our project, we did not include the door lock/unlock system. Thus, we would represent the authorized passing by lighting up a green LED light. If a rfid tag with incorrect id is detected, a red LED light will light up. We also implement a motion detector in our project. If it detects authorized passing from someone, a speaker will play a welcoming music. If it detects unauthorized passing, the speaking will beep to alert people.

### II. Design

#### a. Block Diagram



## b. Description of Blocks

### Input Side:

#### 1. The PIR motion detector

The PIR motion detector is an active sensor whose voltage value could change with motion detected. In this project, we used the voltage value of the motion detector as an input of arduino, and transmitted the motion signal to arduino to trigger the responses on LEDs and the speaker.

#### 2. The RFID module

We attached RFID module with arduino to detect the signal from two adhesive RFID tags, which we considered as keys to hotel doors in our project. With the RFID module, the arduino could detect the RFID signal when the tag is very close to the module, and the RFID signal will be put in the arduino through the module. Also, since different tag has a different serial number, so the RFID module could also distinguish every tags by their unique serial number. In our project, we

used this fact to differentiate the two tags, making one of them the right key and the other one the wrong key.

Output Side:

1. The LEDs

The two LEDs in our project are designed to indicate if the key people swiped are correct. That is, if people swiped the right key, the green LED would light up. Otherwise the red one would shine to indicate that people swiped a wrong key.

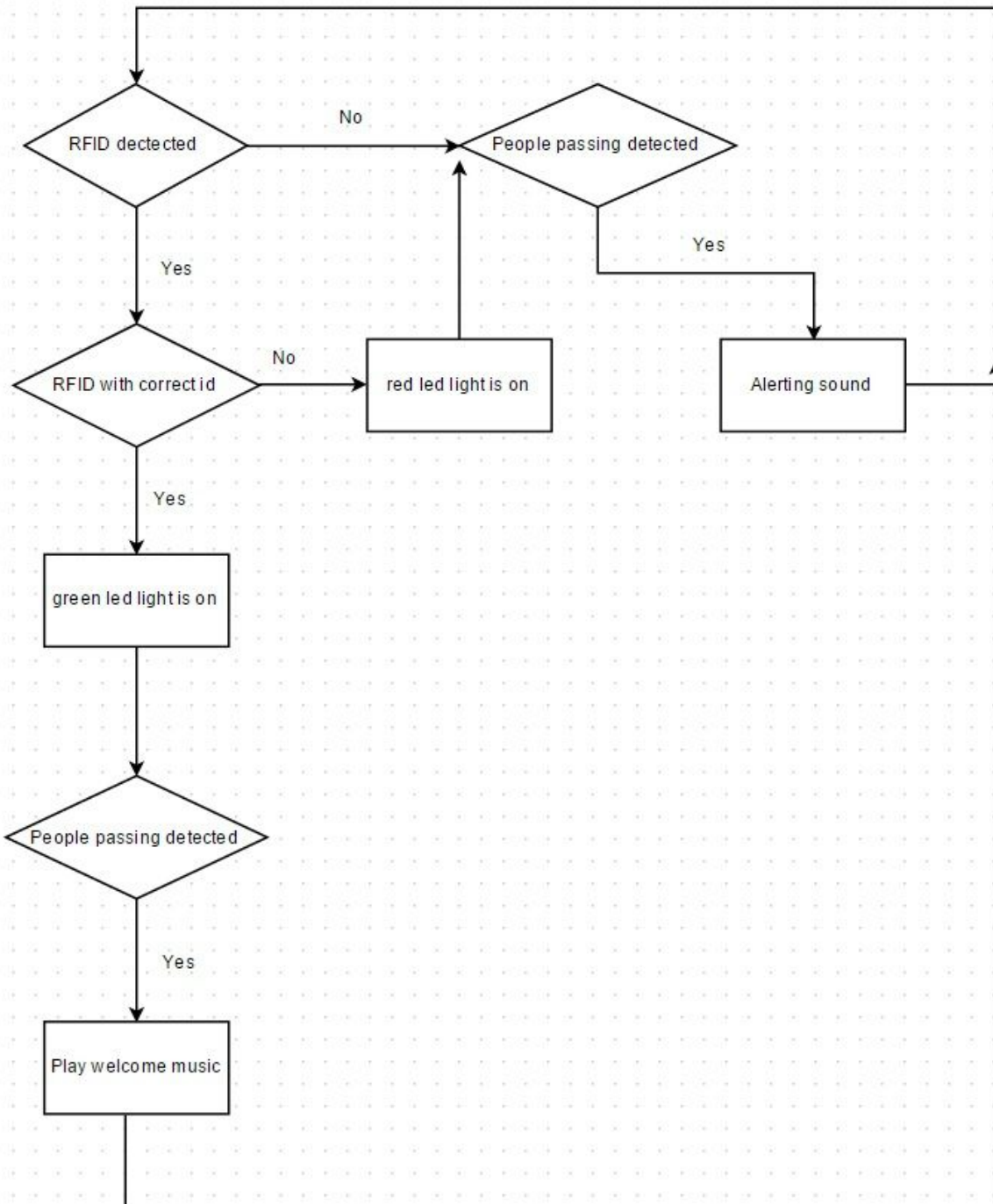
2. The speaker

We also added a speaker in our system to provide a welcome music or an alarm corresponding to different safety situations. We attached the speaker on a output pin on arduino and programed on it. So the welcome music will be played when people swipe the right key and enter the room, and the alarm will be triggered when people directly go into the room without swiping any key.

The Arduino:

We connected the motion detector to arduino's pin 2 and the RFID module to pin 7 & 8 as input pins. Also, the two LEDs were attached to pin 5 & 6, and the speaker was connected to pin 3 as output pins. In general, the arduino collects the input value of motion detector and RFID module, and based on the logic in our code, controls the LEDs and the speaker to function like a safety door lock system.

- c. Flow chart of software



d. Code

```
int melody[] = {
```

```
NOTE_C4, NOTE_G3, NOTE_G3, NOTE_A3, NOTE_G3, 0, NOTE_B3, NOTE_C4
```

```
};
```

```
// note durations: 4 = quarter note, 8 = eighth note, etc.:
```

```
int noteDurations[] = {
```

```
  4, 8, 8, 4, 4, 4, 4, 4
```

```
};
```

```
/*
```

```
RFID Eval 13.56MHz Shield example sketch v10
```

```
Aaron Weiss, aaron at sparkfun dot com
```

```
OSHW license: http://freedomdefined.org/OSHW
```

```
works with 13.56MHz MiFare 1k tags
```

```
Based on hardware v13:
```

```
D7 -> RFID RX
```

```
D8 -> RFID TX
```

```
D9 -> XBee TX
```

```
D10 -> XBee RX
```

```
Note: RFID Reset attached to D13 (aka status LED)
```

```
Note: be sure include the SoftwareSerial lib, http://arduiniiana.org/libraries/newsoftserial/
```

```
Usage: Sketch prints 'Start' and waits for a tag. When a tag is in range, the shield reads the tag,
```

```
blinks the 'Found' LED and prints the serial number of the tag to the serial port
```

```
and the XBee port.
```

06/04/2013 - Modified for compatibility with Arudino 1.0. Seb Madgwick.

\*/

String pw = "180117238158";

/\*Start

B475EE9E

B475EE9E

B475EE4E

B475EE4E

\*/

#include <SoftwareSerial.h>

SoftwareSerial rfid(7, 8);

SoftwareSerial xbee(10, 9);

//Prototypes

void check\_for\_notag(void);

void halt(void);

void parse(void);

void print\_serial(void);

void read\_serial(void);

void seek(void);

void set\_flag(void);

//Global var

int flag = 0;

```
int Str1[11];

//INIT

//green 6, red 5

int pirPin = A0;

void setup()

{

    pinMode(5, OUTPUT);

    pinMode(6, OUTPUT);

    pinMode(pirPin, INPUT);

    Serial.begin(9600);

    Serial.println("Start");

    // set the data rate for the SoftwareSerial ports

    xbee.begin(9600);

    rfid.begin(19200);

    delay(10);

    halt();

}

//MAIN

void loop()

{

    delay(1000);
```

```
//motion detector

digitalWrite(5,LOW);

digitalWrite(6,LOW);

int pirVal = analogRead(pirPin);

/*
if(pirVal < 1000){ //was motion detected
    Serial.println("Motion Detected");
    delay(10 );
}
else
{Serial.println("n");
    delay(10 );
}
*/

Serial.println(pirVal);

read_serial();

delay(500);

String sb = "";

for(int i = 8; i>= 5;i --)

sb+= Str1[i];

Serial.println("lalalala " + sb);

if(pw.equals(sb) && pirVal <1000)
```



```

{
for (int thisNote = 0; thisNote < 8; thisNote++) {

    // to calculate the note duration, take one second
    // divided by the note type.
    //e.g. quarter note = 1000 / 4, eighth note = 1000/8, etc.
    int noteDuration = 1000 / noteDurations[thisNote];
    tone(3, melody[thisNote], noteDuration);

    // to distinguish the notes, set a minimum time between them.
    // the note's duration + 30% seems to work well:
    int pauseBetweenNotes = noteDuration * 1.30;
    delay(pauseBetweenNotes);
    // stop the tone playing:
    noTone(3);
}
}

//green light
if(pw.equals(sb)) {
digitalWrite(6,HIGH);
delay(500);
digitalWrite(6, LOW);
}

```

```
if(flag != 0 && !pw.equals(sb))
```

```
{
```

```
    digitalWrite(5,HIGH);
```

```
    delay(500);
```

```
}
```

```
if(flag == 0) {
```

```
    if(pirVal <1000)
```

```
        tone(3,NOTE_C6,1000);
```

```
        delay(300);
```

```
        noTone(3);
```

```
}
```

```
}
```

```
void check_for_notag()
```

```
{
```

```
    seek();
```

```
    delay(10);
```

```
    parse();
```

```
    set_flag();
```

```
    if(flag = 1){
```

```
        seek();
```

```
        delay(10);
```

```
        parse();
```

```
}
```

```
}  
  
void halt()  
  
{  
  //Halt tag  
  
  rfid.write((uint8_t)255);  
  
  rfid.write((uint8_t)0);  
  
  rfid.write((uint8_t)1);  
  
  rfid.write((uint8_t)147);  
  
  rfid.write((uint8_t)148);  
  
}  
  
void parse()  
  
{  
  while(rfid.available()){  
    if(rfid.read() == 255){  
      for(int i=1;i<11;i++){  
        Str1[i]= rfid.read();  
      }  
    }  
  }  
}  
  
void print_serial()  
  
{  
  if(flag == 1){  
    //print to serial port
```

```
Serial.print(Str1[8], HEX);  
Serial.print(Str1[7], HEX);  
Serial.print(Str1[6], HEX);  
Serial.print(Str1[5], HEX);  
Serial.println();  
  
//print to XBee module  
xbee.print(Str1[8], HEX);  
xbee.print(Str1[7], HEX);  
xbee.print(Str1[6], HEX);  
xbee.print(Str1[5], HEX);  
xbee.println();  
  
delay(100);  
  
//check_for_notag();  
  
}  
  
}  
  
void read_serial()  
{  
    seek();  
    delay(10);  
    parse();  
    set_flag();  
    print_serial();  
    delay(100);  
}
```

```
void seek()
{
    //search for RFID tag
    rfid.write((uint8_t)255);
    rfid.write((uint8_t)0);
    rfid.write((uint8_t)1);
    rfid.write((uint8_t)130);
    rfid.write((uint8_t)131);
    delay(10);
}

void set_flag()
{
    if(Str1[2] == 6){
        flag++;
    }
    if(Str1[2] == 2){
        flag = 0;
    }
}
```

### III. Results

#### a. Presented Results

The result was what we expected the system should do. There were two indistinguishable tags we used to test. Each tag had different hex number read by RFID receiver. So when correct tag moved to the RFID receiver with my hand passing by the motion detector, an entering music came up with green LED light up afterward. When the wrong tag moved to RFID receiver, the red LED lighted up, meaning you have the wrong key. When there was no tags but motion detector detected something was moving without any tags checked in, the speaker beeped, warning an uninformed break in.

#### c. Qualitative Analysis of results

Most parts of the work were fine except that the motion detector was too sensitive so that whenever some spectators moved, changing the reflection of light or making the table shake, the speaker kept beeping.

#### d. Quantitative Analysis of results

There are some data we collected from Arduino:

For two tags, the hex number for the door key is: B475EE9E and the hex number for the wrong key is: B475EE4E.

Except for that, the data we read from motion detector is when there is no motion, the number we read is 1022~1023. When there is motion, the number becomes 1. So when there is no motion, voltage going through is about 5V and 0V when something is moving.

### IV. Future Work

Since it is our first time experiencing with rfid, we only tried the adhesive rfid tag. We found out its signal decayed over time. By the end of the project, we need to scan the rfid on the detector to make it work. If we have time, we would try other rfid tags and choose one with stronger signal. If we have more time, we would make our robot send out alert text message instead of beeping to make our project more secure. Another side we would work on is to make our motion

detector more accurate. Currently, it is very sensitive. It would detect passing even if someone is hanging out in front of the door. Although, we tried to put it inside of a box to reduce surrounding interruption, it is still too responsive.

## V. Conclusion

### e. What Worked?

We successfully read the id of each rfid tag, and we can get feedback from all the sensors in our circuit. The green led will light up when the rfid tag with correct id is detected, and the red led will light up when the wrong one is detected. It can detect pass by playing beeping sound, and play welcome music when someone pass the door after unlocking the door with correct rfid tag.

### f. What Didn't Work?

When someone scans his/her rfid tag to unlock the door and wait for a few seconds, the alert will occur when he/she passed the door. The cause is that the program has executed a new loop and in that loop, rfid is not detected. So in our demonstration, we need to make the motion and rfid scanning simultaneously to make it work. We are planning to learn how to recode the program so that the loop will wait for motion detector signals in a certain period in the future.

### g. What did you learn?

We learned how to add modules to the arduino board so it can have additional features. In our case, we added the rfid and xbee module to arduino uno. We learned how to connect motion detector and speaker to our circuit. We also learned to how to communicate between hardware and software such as getting the feedback from our sensors and then processing them to command different led lights to light on. We improved our critical thinking skills by spotting the

problems and solving them throughout the project such as how to read the unique id for each rfid tag. Each member has a better understanding of team responsibility and can communicate more effectively by the end of the project.