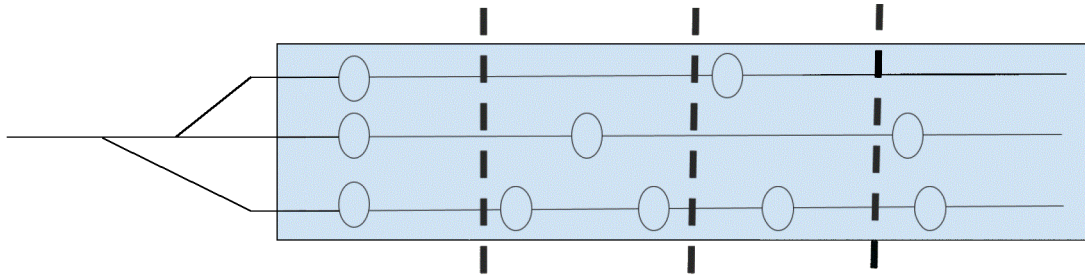


Arduino automatic fiddle yard shunt

James Wentworth

This is for a club layout fiddle yard that will span four baseboards.

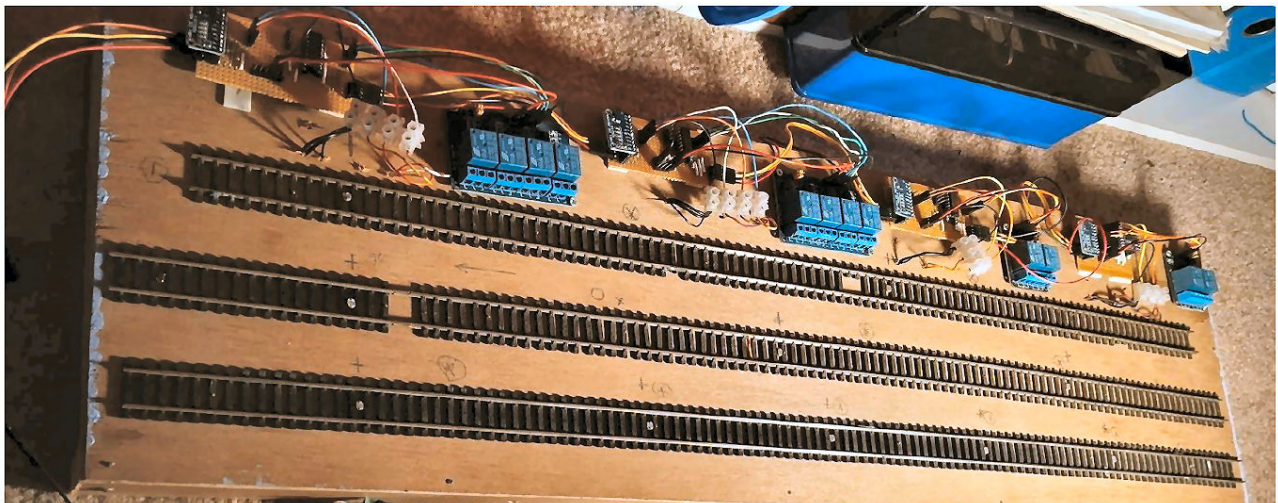


The circles are the LDRs that detect a train's presence.

Its control panel has LEDs and 'Exit' buttons for each of the lines.

When you press a button, the train comes out of that track and any others further back shuffle up.

This the test setup.



You can see the relays that control power to the track sections.

You can also see the LDRs between the sleepers and the ADS1115 boards that use them to detect each train's presence.

The aim was to reduce the amount of wires and cost.

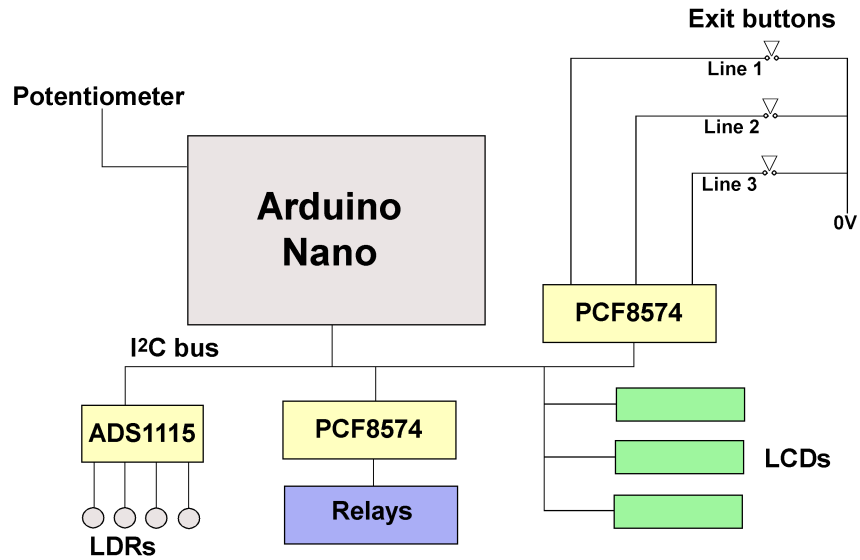
I2C is used for communications between all the devices.

This is an experiment to see if it can be done this way,

The main components are:

- Arduino Nano
- ADS1115 4 channel analog to digital converter
- PCF8574 Remote 8-Bit I/O Expander for I2C Bus
- LDRs Light dependent resistors
- LCDs Liquid Crystal Displays
- Relay boards

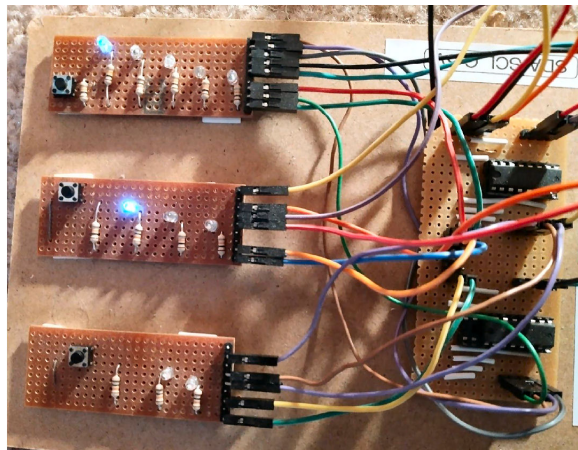
Here is the block diagram of the system.



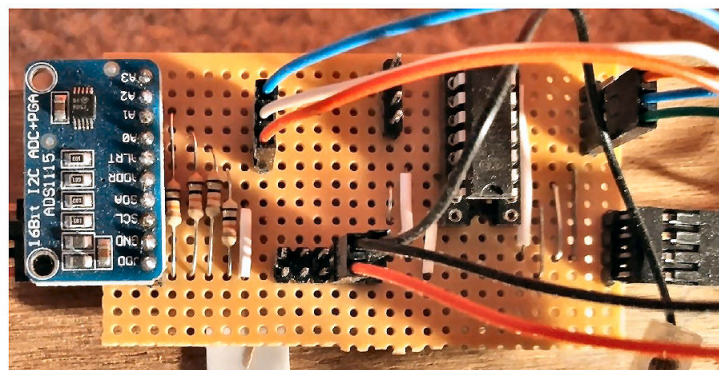
To simplify the illustration, only one set of relays and their control chip is shown.

Below are pictures of the test setup.

This is the set of buttons and LEDs connected to one of the PCF8574 chips.

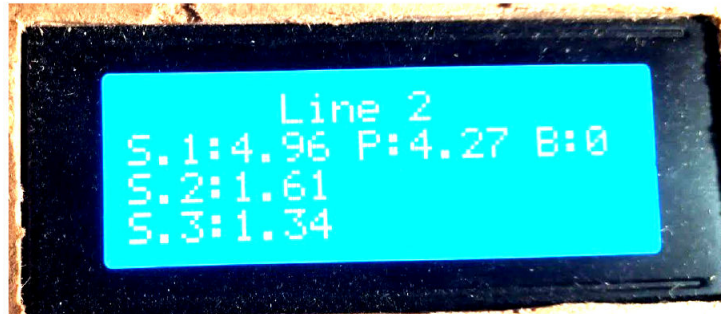


Here is the ADS1115 module. Each module checks the light level from four LDRs and sends the readings back to the Arduino Nano.



There is a potentiometer connected to the Nano, this is for setting the trigger for on and off for different light levels.

The LCDs show the readings from the LDRs, potentiometer and buttons. This provides an indication of the current situation on the tracks.



In this example, S.1, S.2, etc. refers to track section 1, 2 and so on.

The value of 4.96 next to S1 is the current voltage reading from the LDR in section 1.

The P value is the voltage level from the setting of the potentiometer.

The B value shows whether the button has been pressed.

The control panel houses the buttons, the LCDs and the Arduino.

On the baseboards there will be the LDRs (along with the ADS1115) and the relays (along with the PCF8574 chip that controls them).

The program

Adafruit and YouTube have been good sources of information for me.

There is a time delay of about 4 seconds, somewhere in the program. Just can't find it.

I have been working on it on and off for 2 years.

The code has loads of comments so it should be easy to follow.

```

#include <Wire.h>
#include <Adafruit_ADS1015.h>
#include <LiquidCrystal_I2C.h>
#include <PCF8574.h>

// LDR //
Adafruit_ADS1115 ADCboard1( 0x49); // 0x49 EXIT
Adafruit_ADS1115 ADCboard2( 0x48); // 0x48 board 2
Adafruit_ADS1115 ADCboard3( 0x4A); // 0x4A board 3
Adafruit_ADS1115 ADCboard4( 0x4B); // 0x4B board 4

// RELAY //
PCF8574 pcf8574_21(0x21);// 0x21 Exit Relay
PCF8574 pcf8574_23(0x23);// 0x23 Board 2 Relay
PCF8574 pcf8574_26(0x26);// 0x26 Board 3 Relay
PCF8574 pcf8574_22(0x22);// 0x22 Board 4 Relay

// LED and BUTTONS PCF8574A//
PCF8574 pcf8574_38(0x38);// 0x38 Board LEDS lines 1 and 2
PCF8574 pcf8574_39(0x39);// 0x39 Board LEDS line 3

// DISPLAY //
LiquidCrystal_I2C lcd1(0x27,16,2);
LiquidCrystal_I2C lcd2(0x25,20,4);
LiquidCrystal_I2C lcd3(0x20,20,4);

float scalefactor = 0.1875F;
float volts;
float potVolts;

int potentiometer = A6;
int potValue = 0;
int buttonLine1Exit = 0;
int buttonLine2Exit = 0;
int buttonLine3Exit = 0;

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

  ADCboard1.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV
0.1875mV (default)
  ADCboard2.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV
0.1875mV (default)
  ADCboard3.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV
0.1875mV (default)

  ADCboard1.begin();

```

```
ADCboard2.begin();
ADCboard3.begin();
```

```
//////////BOARD 1//////////
```

```
pcf8574_21.pinMode(P0, OUTPUT);// RELAY Line 1 exit
pcf8574_21.pinMode(P1, OUTPUT);// RELAY Line 2 exit
pcf8574_21.pinMode(P2, OUTPUT);// RELAY Line 3 exit
```

```
//////////BOARD 2//////////
```

```
pcf8574_23.pinMode(P0, OUTPUT);// RELAY Line 2 section 2
pcf8574_23.pinMode(P1, OUTPUT);// RELAY Line 3 section 2
pcf8574_23.pinMode(P2, OUTPUT);// RELAY Line 3 section 3
```

```
//////////BOARD 3//////////
```

```
pcf8574_26.pinMode(P0, OUTPUT);// RELAY Line 1 section 2
pcf8574_26.pinMode(P1, OUTPUT);// RELAY Line 3 section 4
```

```
//////////BOARD 4//////////
```

```
pcf8574_22.pinMode(P0, OUTPUT);// RELAY Line 2 section 3
pcf8574_22.pinMode(P1, OUTPUT);// RELAY Line 3 section 5
```

```
pcf8574_38.pinMode(P0, INPUT);// LINE 1 BUTTON
pcf8574_38.pinMode(P1, OUTPUT);// LED LINE 1 EXIT
pcf8574_38.pinMode(P2, OUTPUT);// LED LINE 1 SECTION 2
pcf8574_38.pinMode(P3, INPUT);// LINE 2 BUTTON
pcf8574_38.pinMode(P4, OUTPUT);// LED LINE 2 EXIT
pcf8574_38.pinMode(P5, OUTPUT);// LED LINE 2 SECTION 2
pcf8574_38.pinMode(P6, OUTPUT);// LED LINE 2 SECTION 3
```

```
pcf8574_39.pinMode(P0, INPUT);// LINE 3 BUTTON
pcf8574_39.pinMode(P1, OUTPUT);// LED LINE 3 EXIT
pcf8574_39.pinMode(P2, OUTPUT);// LED LINE 3 SECTION 2
pcf8574_39.pinMode(P3, OUTPUT);// LED LINE 3 SECTION 3
pcf8574_39.pinMode(P4, OUTPUT);// LED LINE 3 SECTION 4
pcf8574_39.pinMode(P5, OUTPUT);// LED LINE 3 SECTION 4
```

```
pcf8574_38.begin();
pcf8574_39.begin();
```

```
}
```

```
void loop(void)
```

```
{
  buttonLine1Exit = pcf8574_38.digitalRead(P0);
  buttonLine2Exit = pcf8574_38.digitalRead(P3);
```

```
buttonLine3Exit = pcf8574_39.digitalRead(P0);
```

```
potValue = analogRead(potentiometer);
```

```
int Line1EXIT, Line2EXIT, Line3EXIT, board1spare;  
int Line2SECTION2, Line3SECTION2, Line3SECTION3, board2spare;  
int Line1SECTION2, Line3SECTION4, board3spare1, board3spare;  
int Line2SECTION3, Line3SECTION5, board4spare1, board4spare;
```

```
float Line1EXITVolts, Line2EXITVolts, Line3EXITVolts;//, board1spare;  
float Line1SECTION2Volts, Line2SECTION2Volts, Line3SECTION2Volts;  
float Line2SECTION3Volts, Line3SECTION3Volts,  
Line3SECTION4Volts,Line3SECTION5Volts;
```

```
Line1EXIT = ADCboard1.readADC_SingleEnded(0);  
Line2EXIT = ADCboard1.readADC_SingleEnded(1);  
Line3EXIT = ADCboard1.readADC_SingleEnded(2);  
board1spare = ADCboard1.readADC_SingleEnded(3);
```

```
Line2SECTION2 = ADCboard2.readADC_SingleEnded(0);  
Line3SECTION2 = ADCboard2.readADC_SingleEnded(1);  
Line3SECTION3 = ADCboard2.readADC_SingleEnded(2);  
board2spare = ADCboard2.readADC_SingleEnded(3);
```

```
Line1SECTION2 = ADCboard3.readADC_SingleEnded(0);  
Line3SECTION4 = ADCboard3.readADC_SingleEnded(1);  
board3spare1 = ADCboard3.readADC_SingleEnded(2);  
board3spare = ADCboard4.readADC_SingleEnded(3);
```

```
Line2SECTION3 = ADCboard4.readADC_SingleEnded(0);  
Line3SECTION5 = ADCboard4.readADC_SingleEnded(1);  
board4spare1 = ADCboard4.readADC_SingleEnded(2);  
board4spare = ADCboard4.readADC_SingleEnded(3);
```

```
Line1EXITVolts = (Line1EXIT * scalefactor)/1000.0;  
Line1SECTION2Volts = (Line1SECTION2 * scalefactor)/1000.0;
```

```
Line2EXITVolts = (Line2EXIT * scalefactor)/1000.0;  
Line2SECTION2Volts = (Line2SECTION2 * scalefactor)/1000.0;  
Line2SECTION3Volts = (Line2SECTION3 * scalefactor)/1000.0;
```

```
Line3EXITVolts = (Line3EXIT * scalefactor)/1000.0;  
Line3SECTION2Volts = (Line3SECTION2 * scalefactor)/1000.0;  
Line3SECTION3Volts = (Line3SECTION3 * scalefactor)/1000.0;  
Line3SECTION4Volts = (Line3SECTION4 * scalefactor)/1000.0;  
Line3SECTION5Volts = (Line3SECTION5 * scalefactor)/1000.0;
```

```
potVolts = potValue *(5.0 / 1023.0);
```

//////////LCD Line 1 //////////

```
lcd1.init();  
lcd1.backlight();  
lcd1.clear();  
lcd1.setCursor(0,0);  
  
lcd1.print("S.1:");  
lcd1.print(Line1EXITVolts);  
lcd1.print(" P:");  
lcd1.print(potVolts);  
lcd1.setCursor(0,1);  
lcd1.print("S.2:");  
lcd1.print(Line1SECTION2Volts);  
  
lcd1.print(" B:");  
lcd1.print(buttonLine1Exit );
```

//////////LCD Line 2//////////

```
lcd2.init();  
lcd2.backlight();  
lcd2.clear();  
lcd2.setCursor(6,0);  
lcd2.print("Line 2");  
lcd2.setCursor(0,1);  
lcd2.print("S.1:");  
lcd2.print(Line2EXITVolts);  
lcd2.print(" P:");  
lcd2.print(potVolts);  
lcd2.print(" B:");  
lcd2.print(buttonLine2Exit);  
lcd2.setCursor(0,2);  
lcd2.print("S.2:");  
lcd2.print(Line2SECTION2Volts);  
lcd2.setCursor(0,3);  
lcd2.print("S.3:");  
lcd2.print(Line2SECTION3Volts);
```

//////////LCD Line 2//////////

```
lcd3.init();  
lcd3.backlight();  
lcd3.clear();  
lcd3.setCursor(6,0);  
lcd3.print("Line 3");  
lcd3.setCursor(0,1);
```

```
lcd3.print("S.1:");
lcd3.print(Line3EXITVolts);
lcd3.print(" P:");
lcd3.print(potVolts);
lcd3.print(" B:");
lcd3.print(buttonLine3Exit);
lcd3.setCursor(0,2);
lcd3.print("S.2:");
lcd3.print( Line3SECTION2Volts);
lcd3.print(" S.3:");
lcd3.print( Line3SECTION3Volts);
lcd3.setCursor(0,3);
lcd3.print("S.4:");
lcd3.print( Line3SECTION4Volts);
lcd3.print(" S.5:");
lcd3.print( Line3SECTION5Volts);
```

```
/////Exit Section Line 1////////////////////////////////////
```

```
if (Line1EXITVolts >= potVolts && buttonLine1Exit == HIGH)//release
```

```
{
  pcf8574_38.digitalWrite(P1, LOW);//LED
  pcf8574_21.digitalWrite(P0, HIGH);//RELAY
}
```

```
if (Line1EXITVolts >= potVolts && buttonLine1Exit == LOW)//full
```

```
{
  pcf8574_38.digitalWrite(P1, HIGH);//LED
  pcf8574_21.digitalWrite(P0, LOW);//RELAY
}
```

```
if (Line1EXITVolts <= potVolts && buttonLine1Exit == HIGH)
```

```
{
  pcf8574_38.digitalWrite(P1, LOW);//LED
  pcf8574_21.digitalWrite(P0, HIGH);//RELAY
}
```



```

}

if (Line1EXITVolts <= potVolts && buttonLine1Exit == LOW)//empty
{
    pcf8574_38.digitalWrite(P1, LOW);//LED

    pcf8574_21.digitalWrite(P0, HIGH);//RELAY
}

/////Line 1 Section 2////////////////////////////////////
if (Line1SECTION2Volts >= potVolts && Line1EXITVolts <= potVolts)
{
    pcf8574_38.digitalWrite(P2, LOW);//LED

    pcf8574_26.digitalWrite(P0, HIGH);//RELAY
}

if (Line1SECTION2Volts >= potVolts && Line1EXITVolts >= potVolts)
{
    pcf8574_38.digitalWrite(P2, HIGH);//LED

    pcf8574_26.digitalWrite(P0, LOW);//RELAY
}

if (Line1SECTION2Volts <= potVolts && Line1EXITVolts <=potVolts)
{
    pcf8574_38.digitalWrite(P2, LOW);//LED

    pcf8574_26.digitalWrite(P0, HIGH);//RELAY
}

if (Line1SECTION2Volts <= potVolts && Line1EXITVolts >=potVolts)
{
    // pcf8574_38.digitalWrite(P2, LOW);//LED

    pcf8574_26.digitalWrite(P0, HIGH);//RELAY
}

```

/////Exit Section Line 2////////////////////////////////////

if (Line2EXITVolts >= potVolts && buttonLine2Exit == HIGH)//release

```
{  
  pcf8574_38.digitalWrite(P4, HIGH);//LED  
  pcf8574_21.digitalWrite(P1, HIGH);//RELAY  
}
```

if (Line2EXITVolts >= potVolts && buttonLine2Exit == LOW)//full

```
{  
  pcf8574_38.digitalWrite(P4, HIGH);//LED  
  pcf8574_21.digitalWrite(P1, LOW);//RELAY  
}
```

if (Line2EXITVolts <= potVolts && buttonLine2Exit == HIGH)

```
{  
  pcf8574_38.digitalWrite(P4, LOW);//LED  
  pcf8574_21.digitalWrite(P1, HIGH);//RELAY  
}
```

if (Line2EXITVolts <= potVolts && buttonLine2Exit == LOW)//empty

```
{  
  pcf8574_38.digitalWrite(P4, LOW);//LED  
  pcf8574_21.digitalWrite(P1, HIGH);//RELAY  
}
```

/////Line 2 Section 2////////////////////////////////////

if (Line2SECTION2Volts >= potVolts && Line2EXITVolts <= potVolts)

```
{  
  pcf8574_38.digitalWrite(P5, LOW);//LED  
  pcf8574_23.digitalWrite(P0, HIGH);//RELAY  
}
```

if (Line2SECTION2Volts >= potVolts && Line2EXITVolts >= potVolts)

```

{
pcf8574_38.digitalWrite(P5, HIGH);//LED
pcf8574_23.digitalWrite(P0, LOW);//RELAY
}

if (Line2SECTION2Volts <= potVolts && Line2EXITVolts <=potVolts)
{
pcf8574_38.digitalWrite(P5, LOW);//LED
pcf8574_23.digitalWrite(P0, HIGH);//RELAY
}

if (Line2SECTION2Volts <= potVolts && Line2EXITVolts >=potVolts)
{
pcf8574_38.digitalWrite(P5, LOW);//LED
pcf8574_23.digitalWrite(P0, HIGH);//RELAY
}

/////Line 2 Section 3////////////////////////////////////

if (Line2SECTION3Volts >= potVolts && Line2SECTION2Volts <= potVolts)
{
pcf8574_38.digitalWrite(P6, LOW);//LED
pcf8574_22.digitalWrite(P2, HIGH);//RELAY
}

if (Line2SECTION3Volts >= potVolts && Line2SECTION2Volts >= potVolts)
{
pcf8574_38.digitalWrite(P6, HIGH);//LED
pcf8574_22.digitalWrite(P0, LOW);//RELAY
}

if (Line2SECTION3Volts <= potVolts && Line2SECTION2Volts <=potVolts)
{
pcf8574_38.digitalWrite(P6, LOW);//LED
pcf8574_22.digitalWrite(P0, HIGH);//RELAY
}

```

```
}  
if (Line2SECTION3Volts <= potVolts && Line2SECTION2Volts >=potVolts)  
{  
    pcf8574_38.digitalWrite(P6, LOW);//LED  
    pcf8574_22.digitalWrite(P0, HIGH);//RELAY  
}
```

```
/////Exit Section Line 3////////////////////////////////////
```

```
if (Line3EXITVolts >= potVolts && buttonLine3Exit == HIGH)//release  
{  
    pcf8574_39.digitalWrite(P1, HIGH);//LED  
    pcf8574_21.digitalWrite(P2, HIGH);//RELAY  
}
```

```
if (Line3EXITVolts >= potVolts && buttonLine3Exit == LOW)//full  
{  
    pcf8574_39.digitalWrite(P1, HIGH);//LED  
    pcf8574_21.digitalWrite(P2, LOW);//RELAY  
}
```

```
if (Line3EXITVolts <= potVolts && buttonLine3Exit == HIGH)  
{  
    pcf8574_39.digitalWrite(P1, LOW);//LED  
    pcf8574_21.digitalWrite(P2, HIGH);//RELAY  
}
```

```
if (Line3EXITVolts <= potVolts && buttonLine3Exit == LOW)//empty  
{  
    pcf8574_39.digitalWrite(P1, LOW);//LED  
    pcf8574_21.digitalWrite(P2, HIGH);//RELAY
```

```

}

/////Line 3 Section 2////////////////////////////////////
if (Line3SECTION2Volts >= potVolts && Line3EXITVolts <= potVolts)
{

    pcf8574_39.digitalWrite(P2, LOW);//LED

    pcf8574_23.digitalWrite(P1, HIGH);//RELAY

}

if (Line3SECTION2Volts >= potVolts && Line3EXITVolts >= potVolts)
{

    pcf8574_39.digitalWrite(P2, HIGH);//LED

    pcf8574_23.digitalWrite(P1, LOW);//RELAY

}

if (Line3SECTION2Volts <= potVolts && Line3EXITVolts <=potVolts)
{

    pcf8574_39.digitalWrite(P2, LOW);//LED

    pcf8574_23.digitalWrite(P1, HIGH);//RELAY

}

if (Line3SECTION2Volts <= potVolts && Line3EXITVolts >=potVolts)
{

    pcf8574_39.digitalWrite(P2, LOW);//LED

    pcf8574_23.digitalWrite(P1, HIGH);//RELAY

}

/////Line 3 Section 3////////////////////////////////////

if (Line3SECTION3Volts >= potVolts && Line3SECTION2Volts <= potVolts)
{

    pcf8574_39.digitalWrite(P3, LOW);//LED

    pcf8574_23.digitalWrite(P2, HIGH);//RELAY

}

if (Line3SECTION3Volts >= potVolts && Line3SECTION2Volts >= potVolts)
{

```

```

pcf8574_39.digitalWrite(P3, HIGH);//LED
pcf8574_23.digitalWrite(P2, LOW);//RELAY
}
if (Line3SECTION3Volts <= potVolts && Line3SECTION2Volts <=potVolts)
{
pcf8574_39.digitalWrite(P3, LOW);//LED
pcf8574_23.digitalWrite(P2, HIGH);//RELAY
}
if (Line3SECTION3Volts <= potVolts && Line3SECTION2Volts >=potVolts)
{
pcf8574_39.digitalWrite(P3, LOW);//LED
pcf8574_23.digitalWrite(P2, HIGH);//RELAY
}
/////Line 3 Section 4////////////////////////////////////
if (Line3SECTION4Volts >= potVolts && Line3SECTION3Volts <= potVolts)
{
pcf8574_39.digitalWrite(P6, LOW);//LED
pcf8574_26.digitalWrite(P1, HIGH);//RELAY
}
if (Line3SECTION4Volts >= potVolts && Line3SECTION3Volts >= potVolts)
{
pcf8574_39.digitalWrite(P6, HIGH);//LED
pcf8574_26.digitalWrite(P1, LOW);//RELAY
Serial.print("gggggggggggggg");
}
if (Line3SECTION4Volts <= potVolts && Line3SECTION3Volts <=potVolts)
{
pcf8574_39.digitalWrite(P6, LOW);//LED
pcf8574_26.digitalWrite(P1, HIGH);//RELAY
}

```

```

}

if (Line3SECTION4Volts <= potVolts && Line3SECTION3Volts >=potVolts)
{

    pcf8574_39.digitalWrite(P6, LOW);//LED

    pcf8574_26.digitalWrite(P1, HIGH);//RELAY

}

/////Line 3 Section 5////////////////////////////////////

if (Line3SECTION5Volts >= potVolts && Line3SECTION4Volts <= potVolts)
{

    pcf8574_39.digitalWrite(P5, LOW);//LED

    pcf8574_22.digitalWrite(P1, HIGH);//RELAY

}

if (Line3SECTION5Volts >= potVolts && Line3SECTION4Volts >= potVolts)
{

    pcf8574_39.digitalWrite(P5, HIGH);//LED

    pcf8574_22.digitalWrite(P1, LOW);//RELAY

}

if (Line3SECTION5Volts <= potVolts && Line3SECTION4Volts <=potVolts)
{

    pcf8574_39.digitalWrite(P5, LOW);//LED

    pcf8574_22.digitalWrite(P1, HIGH);//RELAY

}

if (Line3SECTION5Volts <= potVolts && Line3SECTION4Volts >=potVolts)
{

    pcf8574_39.digitalWrite(P5, LOW);//LED

    pcf8574_22.digitalWrite(P1, HIGH);//RELAY

}

}

```