

Project IOT 102

Lê Đức Anh - HE150197

**Project 1: Smart traffic light.**

-Mô tả: +)Giải quyết vấn đề xảy ra ở đường cao tốc: đường chính sẽ không chuyển đèn khi đường phụ không có người.

+)Sử dụng cảm biến siêu âm để xác định xem đường phụ có người tham gia giao thông hay không.

+)Trên mô phỏng nếu cảm biến quét được vật cản ở khoảng cách lớn hơn 10cm thì xác định có người ở đường phụ.

-Thành phần: +)Arduino

+)6 đèn led màu

+)6 điện trở 220 ôm

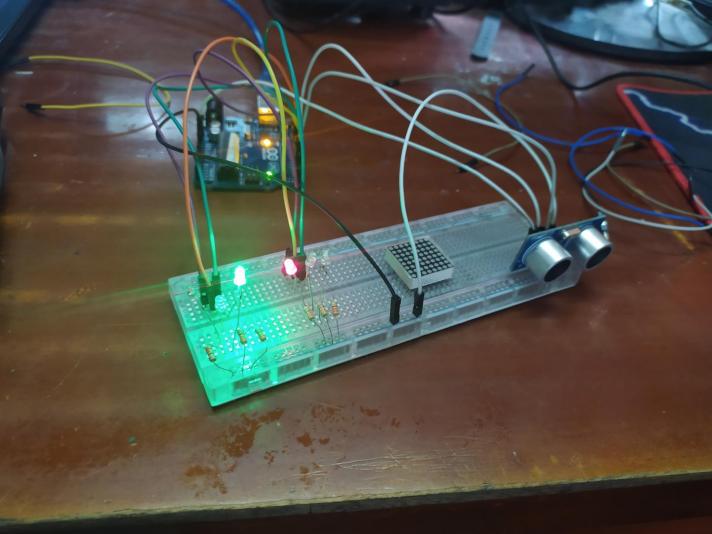
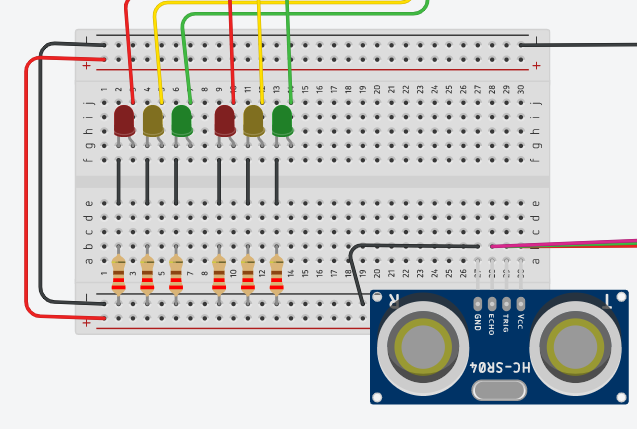
+)dây nối đực đực

+)Cảm biến âm thanh.

-Link: +)Tinkercad:

+)Video: https://drive.google.com/file/d/1CcENPDK0v\_fxpfIVLRc4IoYi3mHwlHjd/view?usp=sharing

-Ảnh minh họa:



-Code:

// C++ code

//

unsigned long t0=0, t;

unsigned long k = 0;

//

const unsigned int TRIG\_PIN=3;

const unsigned int ECHO\_PIN=2;

const unsigned int BAUD\_RATE=9600;

void setup() {

  //Cam Bien

  pinMode(TRIG\_PIN, OUTPUT);

  pinMode(ECHO\_PIN, INPUT);

  Serial.begin(BAUD\_RATE);

  //Cam Bien

  //TrafficLight

  pinMode(13, OUTPUT);

  pinMode(12, OUTPUT);

  pinMode(11, OUTPUT);

  pinMode(7, OUTPUT);

  pinMode(6, OUTPUT);

  pinMode(5, OUTPUT);

  //TrafficLight

}

void loop() {

  //CamBien

  digitalWrite(TRIG\_PIN, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN, LOW);

 const unsigned long duration= pulseIn(ECHO\_PIN, HIGH);

 int distance= duration/29/2;

 if(duration==0){

   Serial.println("Warning: no pulse from sensor");

   }

  else

{

      Serial.print("distance to nearest object:");

      Serial.print(distance);

      Serial.println(" cm");

  }

 delay(1000);

  //CamBien

  t=millis();

  if (t-t0 < 5000 &&1==1 ){

    one();}else if(t-t0 < 6000)

  {two();}else if(t-t0 < 11000 + k)

  {

    three();

  }else if(distance > 30 )//when no one on the non-highspeed road

    {

    delay(1000);

    k=k+1000;

    Serial.print(k);

    }

  else

  {four();delay(3000);t0=t;k=0;}

 }

void one(){

digitalWrite(13, HIGH);

  digitalWrite(5, HIGH);

  digitalWrite(12,LOW);

  digitalWrite(7,LOW);

}

void two(){

//digitalWrite(13, HIGH);

  digitalWrite(6, HIGH);

  digitalWrite(5, LOW);

}

void three(){

  digitalWrite(13, LOW);

  digitalWrite(6, LOW);

  digitalWrite(11,HIGH);

  digitalWrite(7,HIGH);

}

void four(){

  digitalWrite(11,LOW);

 // digitalWrite(7,HIGH);

  digitalWrite(12,HIGH);

}

**Project 2: Ledmatrix**

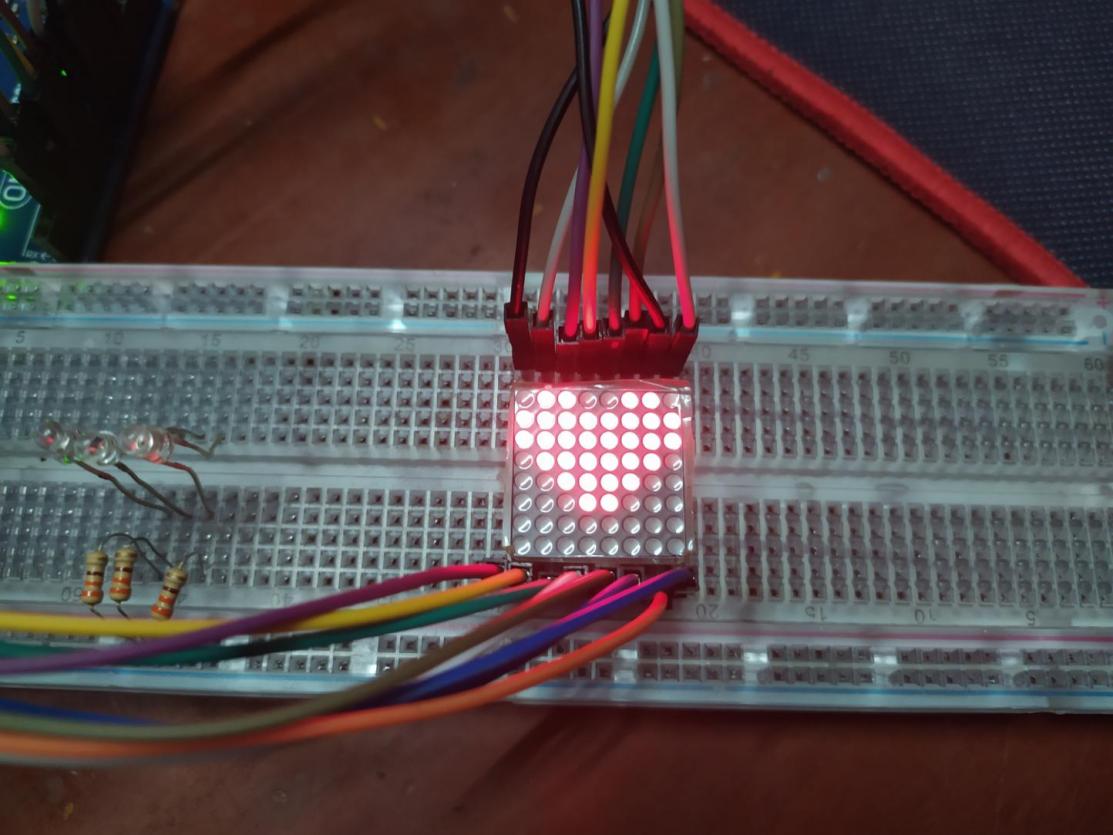
-Mô tả: +)Sử dụng đèn ma trận hiển thị hình ảnh động

+) Hiển thị chữ: “Se1508”

-Thành phần: +)Arduino

+)Ledmatrix

-Link: +)Video:



-Code:

// 2-dimensional array of row pin numbers:

const int row[8] = {

 2, 7, 19, 5, 13, 18, 12, 16

};

// 2-dimensional array of column pin numbers:

const int col[8] = {

 6, 11, 10, 3, 17, 4, 8, 9

};

// 2-dimensional array of pixels:

int pixels[8][8];

int count = 1000;

char str[] = "FABCDEDCBAGHIJKL";

int strLen = sizeof(str);

int ptrChar = 0;

typedef bool charMapType[8][8];

const charMapType charBlank = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType heart0 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType heart1 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType heart2 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 0, 0, 1, 1, 0},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {0, 1, 1, 1, 1, 1, 1, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0}

};

const charMapType heart3 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 0, 0, 1, 1, 0},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {0, 1, 1, 1, 1, 1, 1, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType heart4 = {

 {0, 1, 1, 0, 0, 1, 1, 0},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {1, 1, 1, 1, 1, 1, 1, 1},

 {0, 1, 1, 1, 1, 1, 1, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 0, 0, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType S = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 1, 1, 1, 1, 0, 0},

 {0, 1, 0, 0, 0, 0, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 1, 0, 0},

 {0, 1, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType E = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 1, 1, 1, 0, 0},

 {0, 1, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 1, 0, 0, 0, 0},

 {0, 1, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 1, 1, 1, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType N1 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 1, 1, 0, 0, 0, 0},

 {0, 0, 0, 1, 0, 0, 0, 0},

 {0, 0, 0, 1, 0, 0, 0, 0},

 {0, 0, 0, 1, 0, 0, 0, 0},

 {0, 0, 1, 1,1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType N5 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 1, 1, 1, 0, 0},

 {0, 1, 0, 0, 0, 0, 0, 0},

 {0, 1, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 1, 0, 0},

 {0, 1, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType N0 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 1, 0, 0, 1, 1, 0, 0},

 {0, 1, 0, 1, 0, 1, 0, 0},

 {0, 1, 1, 0, 0, 1, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType N8 = {

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 1, 0, 0, 0, 1, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 1, 0, 0, 0, 1, 0, 0},

 {0, 0, 1, 1, 1, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0},

 {0, 0, 0, 0, 0, 0, 0, 0}

};

const charMapType \*charMap[13] = {&heart0, &heart1, &heart2, &heart3, &heart4, &charBlank, &S, &E, &N1, &N5, &N0, &N8};

void setup() {

 // initialize the I/O pins as outputs

 // iterate over the pins:

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

 // initialize the output pins:

 pinMode(col[thisPin], OUTPUT);

 pinMode(row[thisPin], OUTPUT);

 // take the col pins (i.e. the cathodes) high to ensure that

 // the LEDS are off:

 digitalWrite(col[thisPin], HIGH);

 }

 //setupScreen();

 setupChar();

}

void loop() {

 // draw the screen:

 refreshScreen();

 if(count-- == 0){

 count = 1000;

 setupChar();

 }

}

void setupChar(){

 char c = str[ptrChar];

 int offset = c - 'A';

 const charMapType \*cMap = charMap[offset];

 //charMapType \*cMap = &charDummy;

 for (int x = 0; x < 8; x++) {

 for (int y = 0; y < 8; y++) {

 bool v = (\*cMap)[x][y];

 if(v){

 pixels[x][y] = LOW;

 }else{

 pixels[x][y] = HIGH;

 }

 }

 }

 ptrChar++;

 if(ptrChar>=strLen-1){

 ptrChar = 0;

 }

}

void refreshScreen() {

 // iterate over the rows (anodes):

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

 // take the row pin (anode) high:

 digitalWrite(row[thisRow], HIGH);

 // iterate over the cols (cathodes):

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

 // get the state of the current pixel;

 int thisPixel = pixels[thisRow][thisCol];

 // when the row is HIGH and the col is LOW,

 // the LED where they meet turns on:

 digitalWrite(col[thisCol], thisPixel);

 // turn the pixel off:

 if (thisPixel == LOW) {

 digitalWrite(col[thisCol], HIGH);

 }

 }

 // take the row pin low to turn off the whole row:

 digitalWrite(row[thisRow], LOW);

 }

}

**Project 3: Automatic lights**

-Mô tả: Đèn tự động sáng khi phòng có người và thiếu ánh sáng.

-Thành phần: +)Arduino Uno R3

+)LCD 16 x 2

+)Ultrasonic Distance Sensor

+)10 kΩ Potentiometer

+)10 kΩ Potentiometer

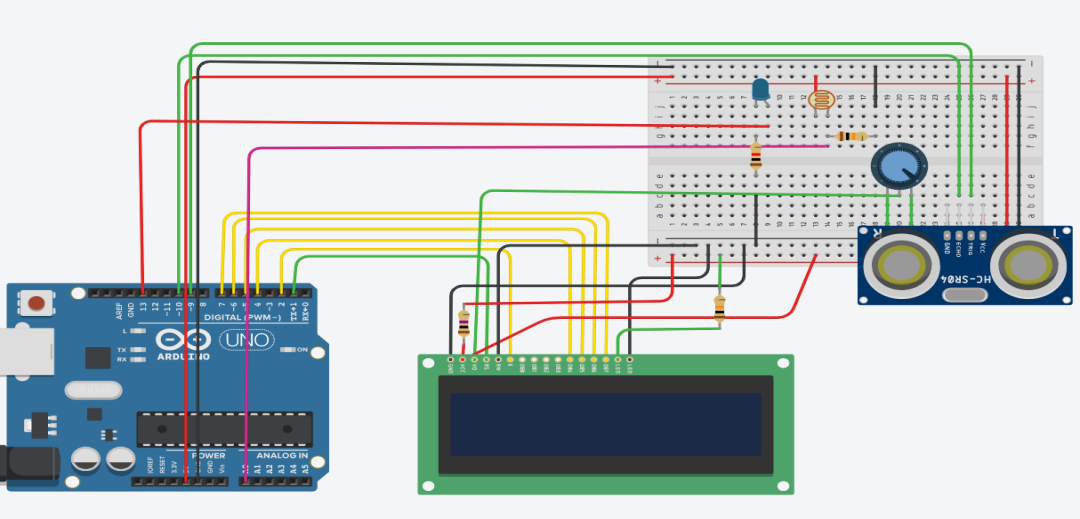
+)10 kΩ Potentiometer

+)1 Blue LED

+)1 kΩ Resistor

+)Photoresistor

-Link: +)tinkercad:



Code:

#include <LiquidCrystal.h> //Please replace the single quote characters ('') with the parenthesis character (<>)

LiquidCrystal lcd(1, 2, 4, 5, 6, 7); // Creates an LCD object. Parameters: (rs, enable, d4, d5, d6, d7)

int sensorPin = A0;

const int trigPin = 9;

const int echoPin = 10;

const int ledPin = 13;

int sensorValue = 0;

long duration;

int distanceCm;

void setup() {

lcd.begin(16,2); // Initializes the interface to the LCD screen, and specifies the dimensions (width and height) of the display

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(ledPin, OUTPUT);

}

void loop() {

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distanceCm= duration\*0.034/2;

lcd.setCursor(0,0); // Sets the location at which subsequent text written to the LCD will be displayed

lcd.print("Distance: "); // Prints string "Distance" on the LCD

lcd.print(distanceCm); // Prints the distance value from the sensor

lcd.print(" cm");

delay(10);

lcd.setCursor(0,1);

if(distanceCm < 100) {

  lcd.print("Lights on !");}else{

  lcd.print("Lights off !");}

delay(10);

if(distanceCm < 100) {

   // read the value from the sensor:

  sensorValue = analogRead(sensorPin);

  // turn the ledPin on

  if (sensorValue <= 300 ){

  digitalWrite(ledPin, HIGH);

  // stop the program for <sensorValue> milliseconds:

  //delay(sensorValue);

  // turn the ledPin off:

  }else{

  digitalWrite(ledPin, LOW);

  }

  // stop the program for for <sensorValue> milliseconds:

  //delay(sensorValue);

  }

  else {

    digitalWrite(ledPin, LOW);

  }

}

**Project 4:Remote control LED lights on and off**

-Mô tả: Sử dụng phím 0,1,2,3,4…,9 để điều khiển bật và tắt 10 cái đèn màu. Nhấn lần đầu đèn sáng, nhấn lần nữa thì đèn tắt.

-Thành phần: +)Arduino Uno R3

+)10 đèn led

+)10 điện trở

-Link:

-Code:

#include <IRremote.h>

int voltage[12];

unsigned int i = 0;

IRrecv irrecv(12);

decode\_results results;

unsigned long key\_value = 0;

//--------------------setup

void setup(){

  Serial.begin(9600);

  SetUpPins();

  irrecv.enableIRIn();

  irrecv.blink13(true);

}

//--------------------loop

void loop(){

  TranslateIR();

  //lcd.setCursor(0,0);

}

//--------------------functions

//-----ToggleLED-----

void ToggleLED(int pin) {

  if (voltage[pin]) {

    Serial.print("LED ");

    Serial.print(pin - 2);

    Serial.println(" was turned off");

    digitalWrite(pin, LOW);

    voltage[pin] = 0;

  } else {

    Serial.print("LED ");

    Serial.print(pin - 2);

    Serial.println(" was turned on");

    digitalWrite(pin, HIGH);

    voltage[pin] = 1;

  }

}

//-----SetUpPins-----

void SetUpPins() {

  pinMode(10, OUTPUT);

  pinMode(11, OUTPUT);

  pinMode(2, OUTPUT);

  pinMode(3, OUTPUT);

  pinMode(4, OUTPUT);

  pinMode(5, OUTPUT);

  pinMode(6, OUTPUT);

  pinMode(7, OUTPUT);

  pinMode(8, OUTPUT);

  pinMode(9, OUTPUT);

  digitalWrite(10, LOW);

  digitalWrite(11, LOW);

  digitalWrite(2, LOW);

  digitalWrite(3, LOW);

  digitalWrite(4, LOW);

  digitalWrite(5, LOW);

  digitalWrite(6, LOW);

  digitalWrite(7, LOW);

  digitalWrite(8, LOW);

  digitalWrite(9, LOW);

}

//-----TranslateIR-----

void TranslateIR() {

  if (irrecv.decode(&results)){

        if (results.value == 0XFFFFFFFF)

          results.value = key\_value;

        switch(results.value){

          case 0xFD00FF://power

          break;

          case 0xFD807F://vol+

          break;

          case 0xFD40BF://func/stop

          break;

          case 0xFD20DF://|<<

          break;

          case 0xFDA05F://>||

          break ;

          case 0xFD609F://>>|

          break ;

          case 0xFD10EF://down arrow

          break ;

          case 0xFD906F://vol-

          break ;

          case 0xFD50AF://up arrow

          break ;

          case 0xFD30CF://0

          ToggleLED(2);

          break ;

          case 0xFDB04F://eq

          break ;

          case 0xFD708F://st/rept

          break ;

          case 0xFD08F7://1

          ToggleLED(3);

          break ;

          case 0xFD8877://2

          ToggleLED(4);

          break ;

          case 0xFD48B7://3

          ToggleLED(5);

          break ;

          case 0xFD28D7://4

          ToggleLED(6);

          break ;

          case 0xFDA857://5

          ToggleLED(7);

          break ;

          case 0xFD6897://6

          ToggleLED(8);

          break ;

          case 0xFD18E7://7

          ToggleLED(9);

          break ;

          case 0xFD9867://8

          ToggleLED(10);

          break ;

          case 0xFD58A7://9

          ToggleLED(11);

          break ;

        }

        key\_value = results.value;

        irrecv.resume();

  }

}

**Project 5: Điều khiển led 3 màu bằng biến trở.**

-Mô tả: Sử dụng 3 biến trở để thay đổi màu sắc của đèn led 3 màu.

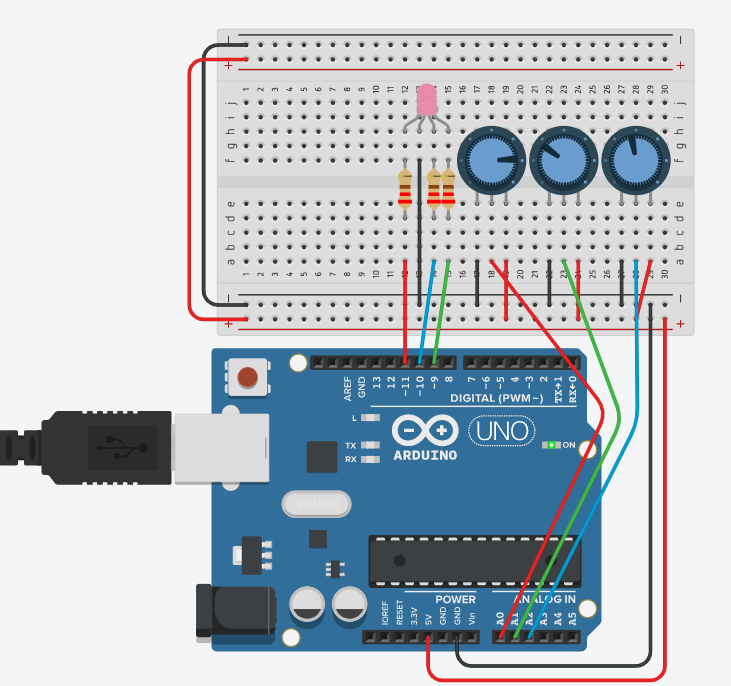
-Link:

-Thành phần: +)Arduino Uno R3

+)3 x 250 kΩ Potentiometer

+)Led RGB

+) 220 Ω Resistor



Code:

// C++ code

//

int g=9, b=10, r=11;

int red,green,blue;

void setup()

{

  pinMode(r, OUTPUT);

  pinMode(g, OUTPUT);

  pinMode(b, OUTPUT);

}

void loop()

{

  red=analogRead(A0);

  green=analogRead(A1);

  blue=analogRead(A2);

  analogWrite(r,red/4);

  analogWrite(g,green/4);

  analogWrite(b,blue/4);

}