Arduino Programs on Tinker cad

Contents

[Arduino interfacing with servo motor 1](#_Toc100659662)

[Program 1](#_Toc100659663)

[Arduino interfacing with temperature sensor 3](#_Toc100659664)

[Program 3](#_Toc100659665)

[Photodiode (LDR) interfacing with Arduino 5](#_Toc100659666)

[Program 5](#_Toc100659667)

[7 segment display interfacing with Arduino 7](#_Toc100659668)

[Program 7](#_Toc100659669)

[IR sensor 9](#_Toc100659670)

[Program 9](#_Toc100659671)

[PIR sensor 11](#_Toc100659672)

[Program 11](#_Toc100659673)

[Home automation 13](#_Toc100659674)

[Program: 13](#_Toc100659675)

[Gas sensor, Led interfacing with Arduino 15](#_Toc100659676)

[Program 15](#_Toc100659677)

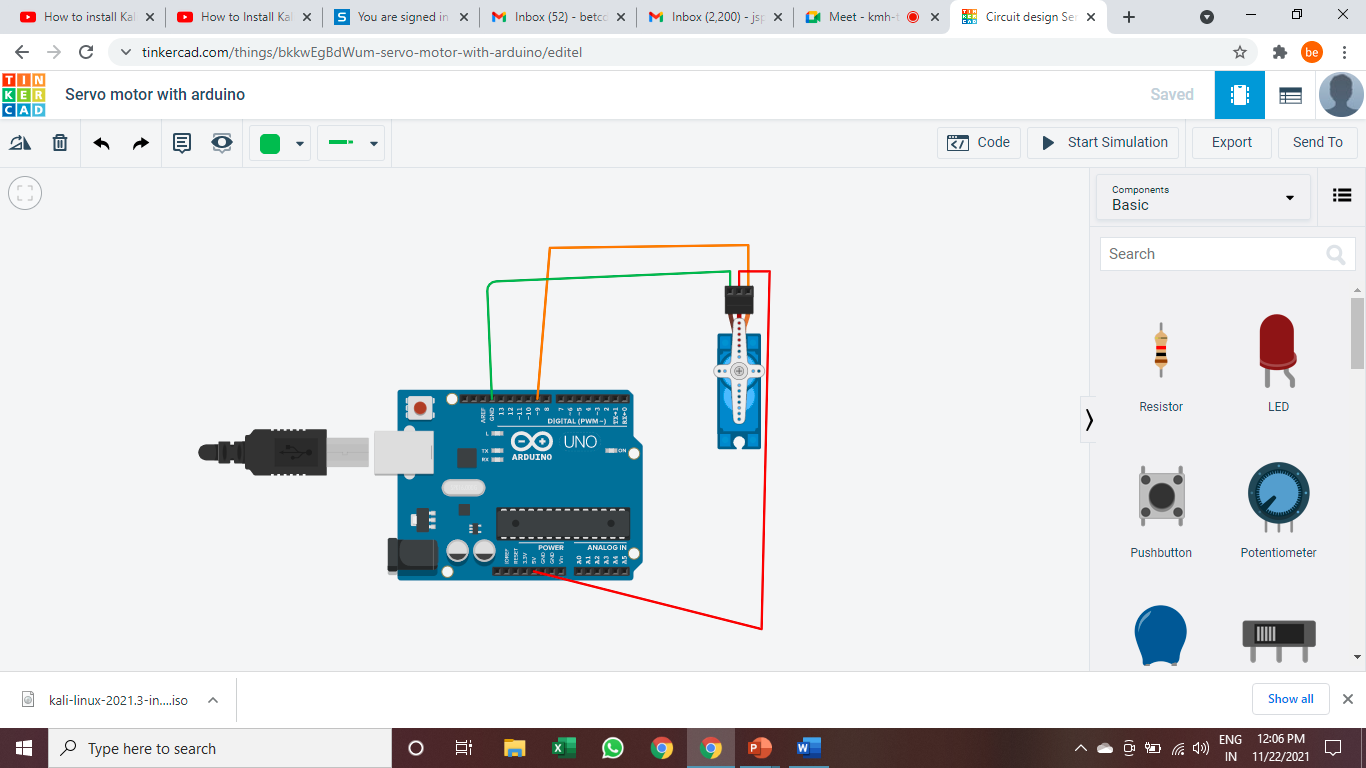
[Ultrasonic sensor with Arduino 17](#_Toc100659678)

[Program 17](#_Toc100659679)

[LCD interfacing with Arduino 19](#_Toc100659680)

[Program 19](#_Toc100659681)

# Arduino interfacing with servo motor



## Program

#include <Servo.h>

Servo myservo; // create servo object to control a servo

// twelve servo objects can be created on most boards

int pos = 0; // variable to store the servo position

void setup() {

myservo.attach(9); // attaches the servo on pin 9 to the servo object

}

void loop() {

for (pos = 0; pos <= 180; pos += 5) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

myservo.write(pos); // tell servo to go to position in variable 'pos'

delay(15); // waits 15ms for the servo to reach the position

}

for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees

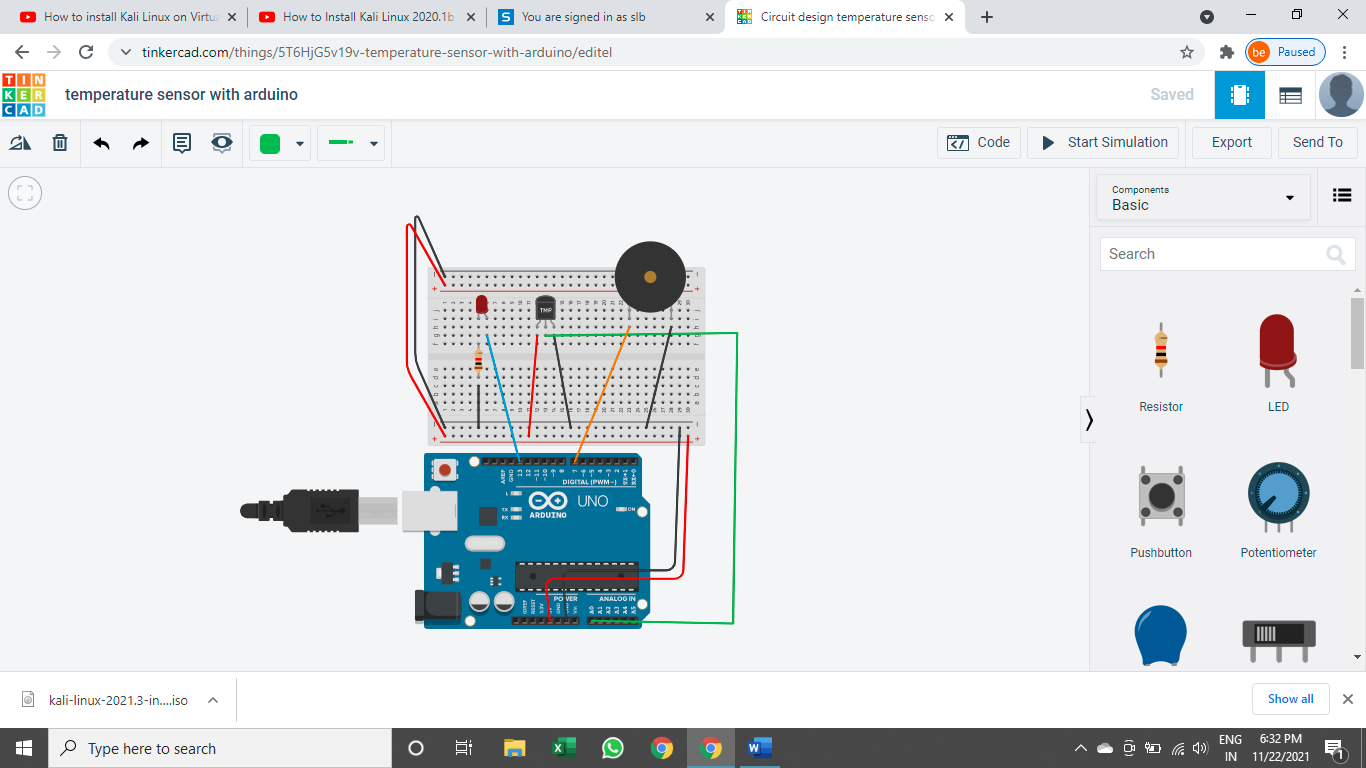
myservo.write(pos); // tell servo to go to position in variable 'pos'

delay(15); // waits 15ms for the servo to reach the position

}

}

# Arduino interfacing with temperature sensor



## Program

void setup()

{

pinMode(A0,INPUT);

Serial.begin(9600);

pinMode(7,OUTPUT);

pinMode(13,OUTPUT);

}

void loop()

{

float tempvalue = analogRead(A0);

float tempvaluemv = tempvalue\*5000/1024;

float tempvaluec = (tempvaluemv - 500)/10;

Serial.println(tempvaluec);

if(tempvaluec>=100)

{

tone(7,220,100);//pin number, frequency, time

delay(500);

}

else

if(tempvaluec>60 && tempvaluec<100)

{

digitalWrite(13,HIGH);//pin number, frequency, time

}

else

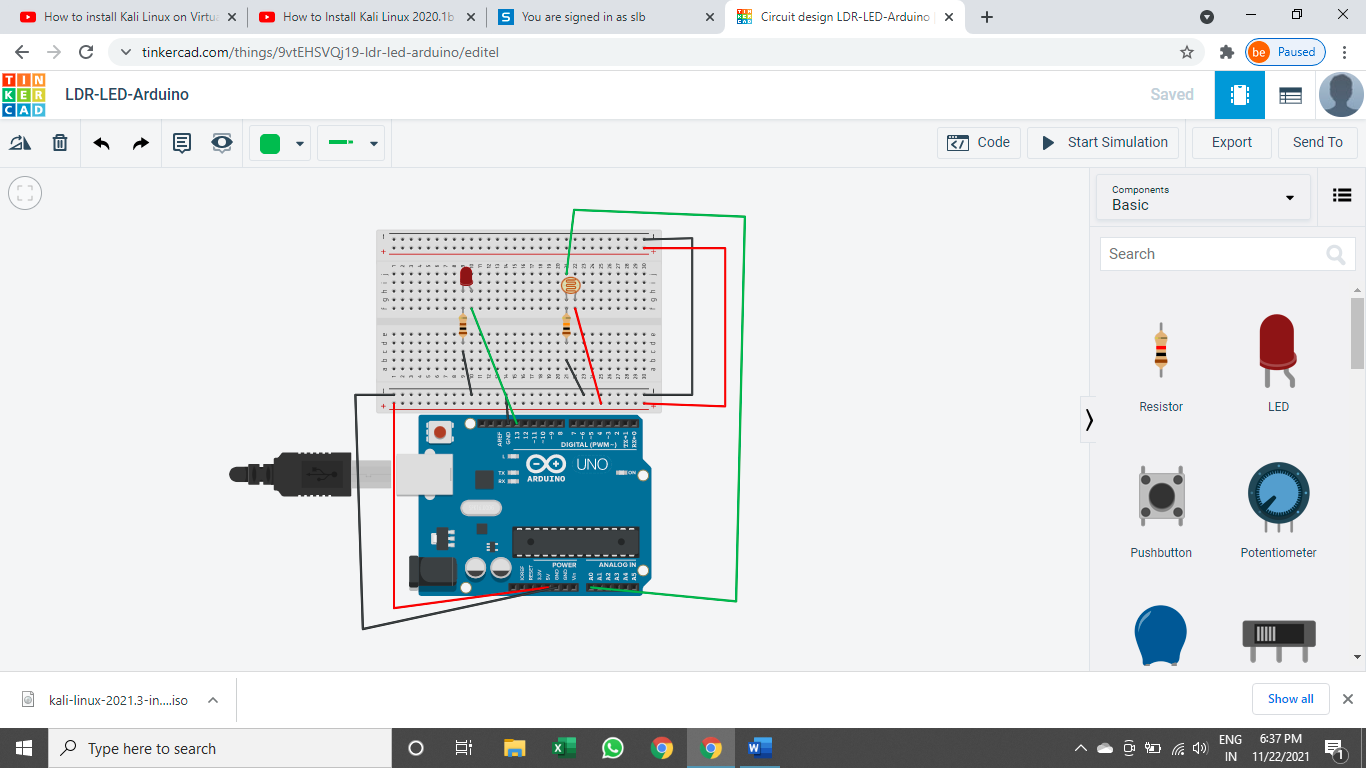
{

digitalWrite(13,LOW);

}

}

# Photodiode (LDR) interfacing with Arduino



## Program

int sensorPin = A0; // select the input pin for the potentiometer

int ledPin = 13; // select the pin for the LED

int sensorValue = 0; // variable to store the value coming from the sensor

void setup() {

// declare the ledPin as an OUTPUT:

Serial.begin(9600);

pinMode(ledPin, OUTPUT);

pinMode(sensorPin, INPUT);

}

void loop() {

sensorValue = analogRead(sensorPin);

if(sensorValue<300)

{

digitalWrite(ledPin, HIGH);

delay(1000);

Serial.println("LDR is dark,LED is on");

}

else

{

digitalWrite(ledPin, LOW);

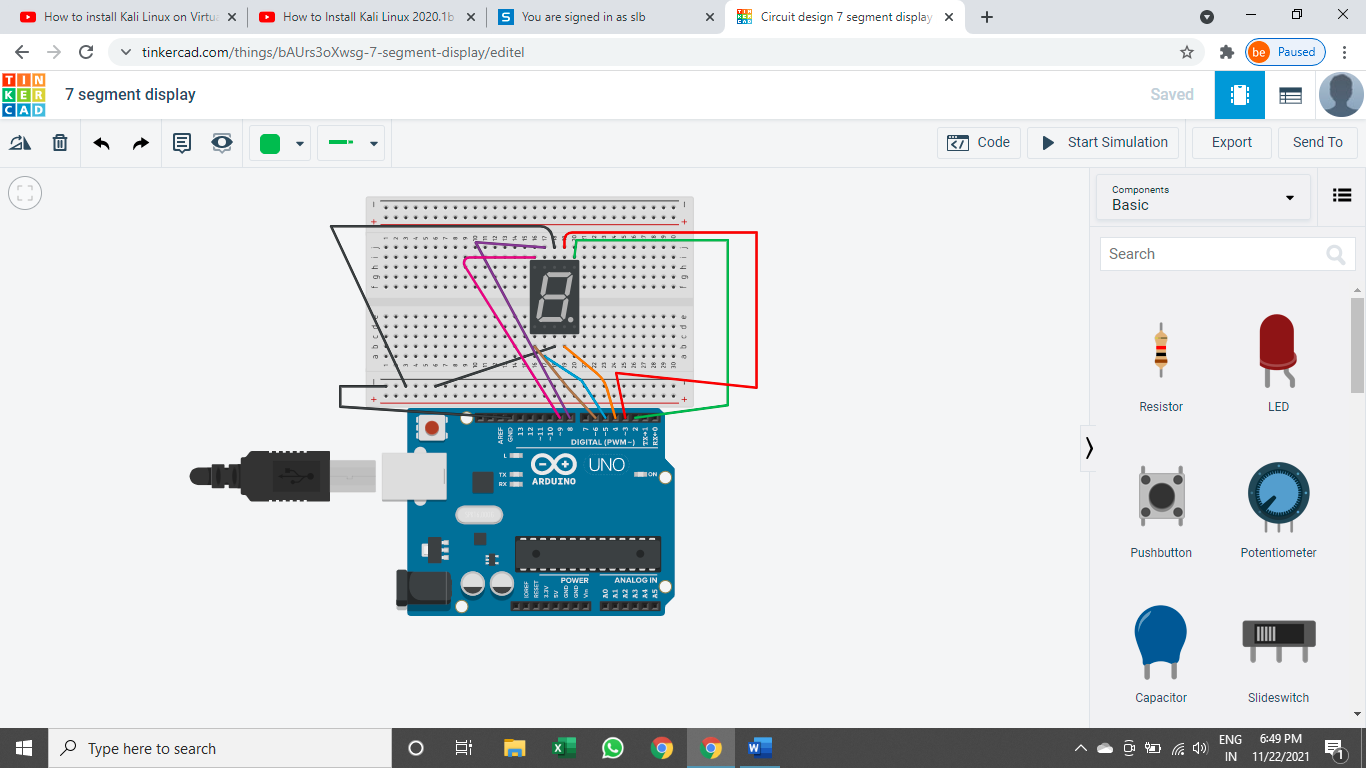
delay(1000);

Serial.println("LDR is bright,LED is off");

}

}

# 7 segment display interfacing with Arduino



## Program

#define a 3

#define b 2

#define c 4

#define d 5

#define e 6

#define f 8

#define g 9

void setup()

{

pinMode(a, OUTPUT);

pinMode(b, OUTPUT);

pinMode(c, OUTPUT);

pinMode(d, OUTPUT);

pinMode(e, OUTPUT);

pinMode(f, OUTPUT);

pinMode(g, OUTPUT);

}

void loop()

{

digitalWrite(a, LOW);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, LOW);

digitalWrite(e, LOW);

digitalWrite(f, LOW);

digitalWrite(g, LOW);

delay(1000); // Wait for 1000 millisecond(s)

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, LOW);

digitalWrite(d, HIGH);

digitalWrite(e, HIGH);

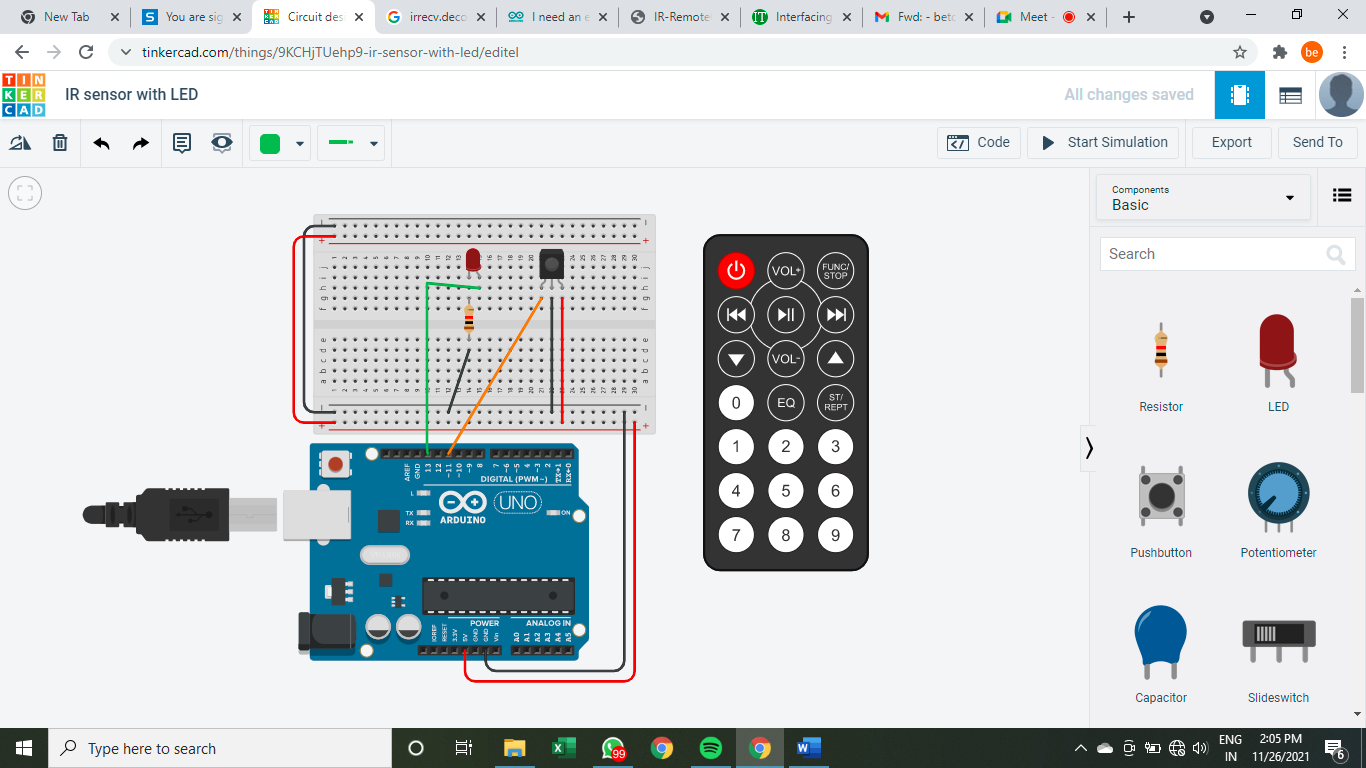
digitalWrite(f, LOW);

digitalWrite(g, HIGH);

delay(1000);

}

# IR sensor



## Program

#include <IRremote.h>

#define LEDPin 13

#define IRPin 11

#define BUTTON1 0XFD08F7

#define BUTTON2 0XFD8877

IRrecv irrecv(IRPin);

decode\_results results;

void setup()

{

pinMode(LEDPin, OUTPUT);

Serial.begin(9600);

irrecv.enableIRIn();

}

void loop()

{

if(irrecv.decode(&results))

{

Serial.println(results.value,HEX);

irrecv.resume();

}

if(results.value==BUTTON1)

{

digitalWrite(LEDPin, HIGH);

}

else

if(results.value==BUTTON2)

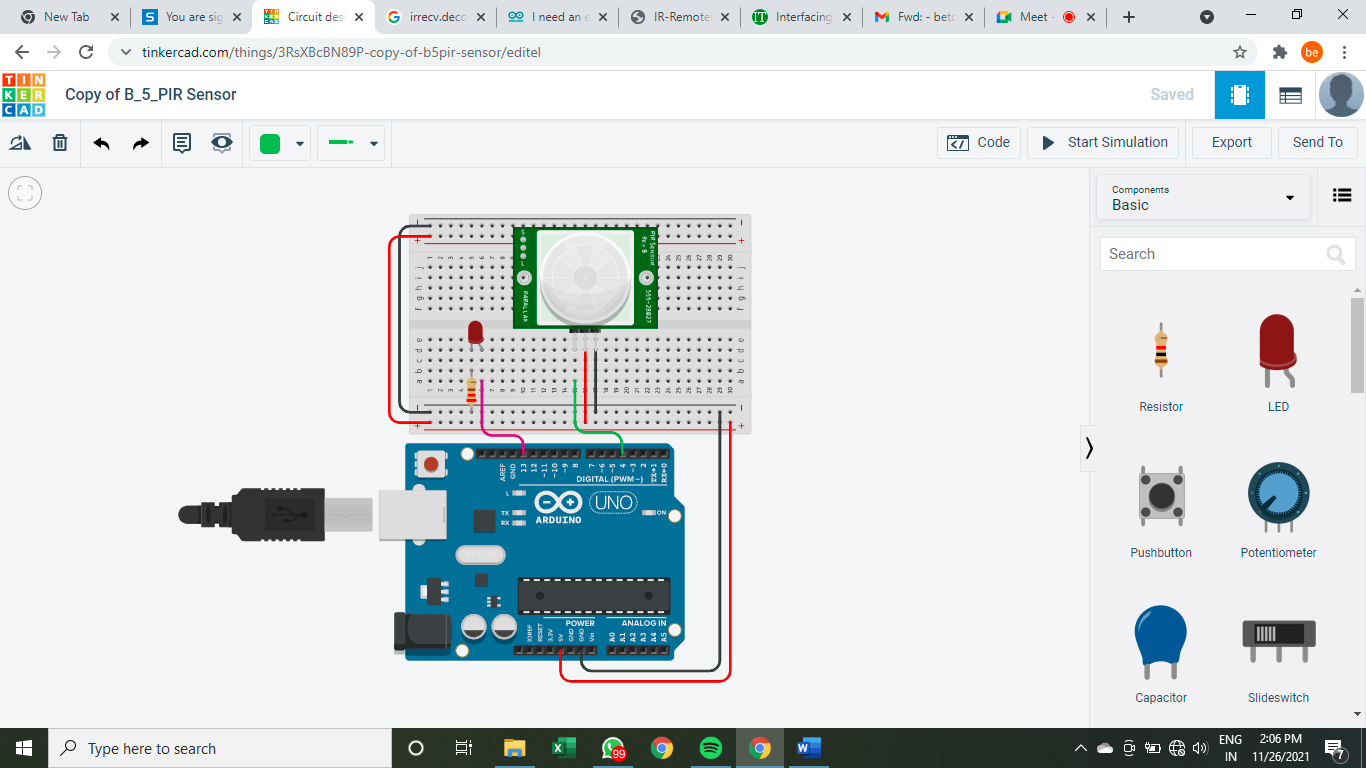
{

digitalWrite(LEDPin, LOW);

}

}

# PIR sensor



## Program

int ledPin=13;

int inputPin=4;

int val=0;

void setup() {

pinMode(ledPin,OUTPUT);

pinMode(inputPin, INPUT);

Serial.begin(9600);

}

void loop() {

val=digitalRead(inputPin); // check the pin status (High=1/Low=0)

if(val==HIGH)

{

Serial.print("Object Present\n");

digitalWrite(ledPin,HIGH);

}

else

{

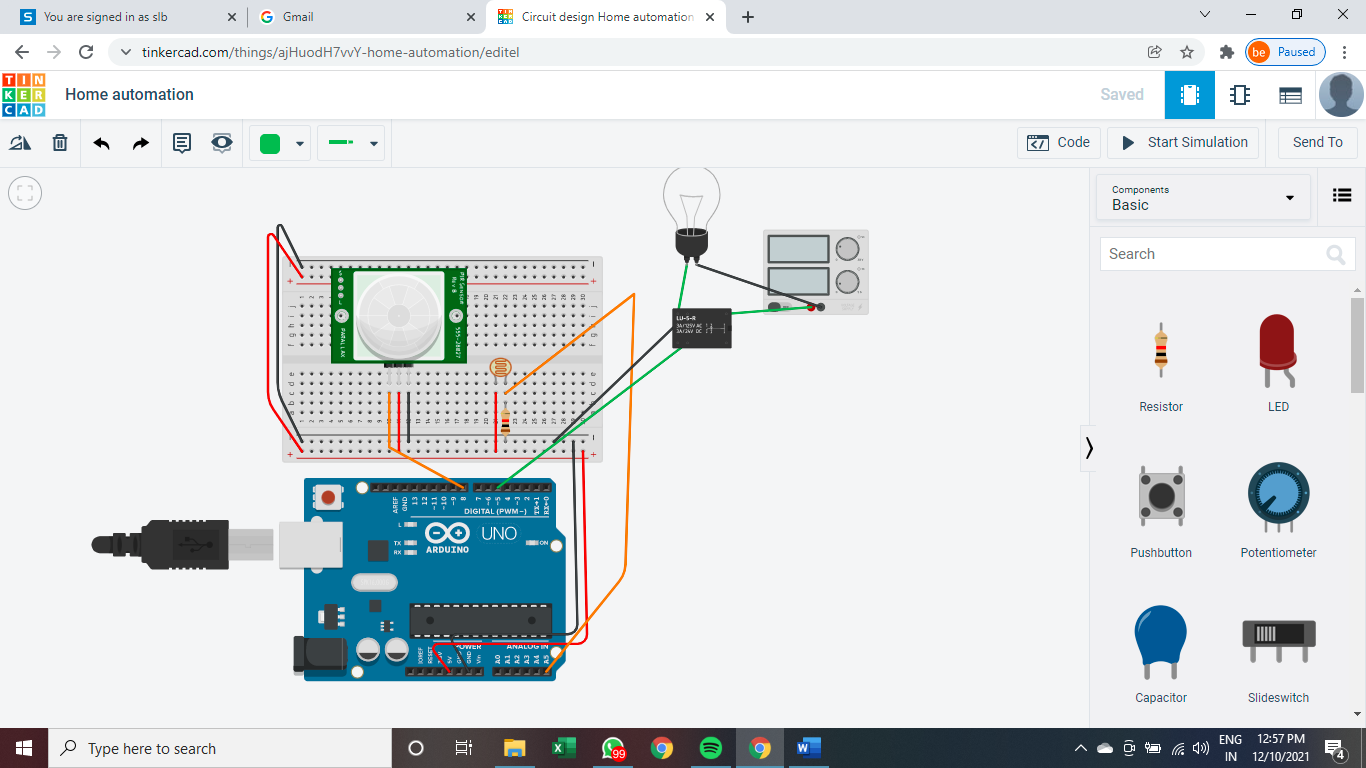
Serial.print("Object Absent\n");

digitalWrite(ledPin,LOW);

}

}

# Home automation



## Program:

float x,y;

void setup()

{

pinMode(5, OUTPUT);

pinMode(8, INPUT);

pinMode(A5, INPUT);

Serial.begin(9600);

}

void loop()

{

x= digitalRead(8);

Serial.println(x);

y= analogRead(A5);

Serial.println(y);

if((x>0)&&(y<550))

{

digitalWrite(5, HIGH);

delay(5000);

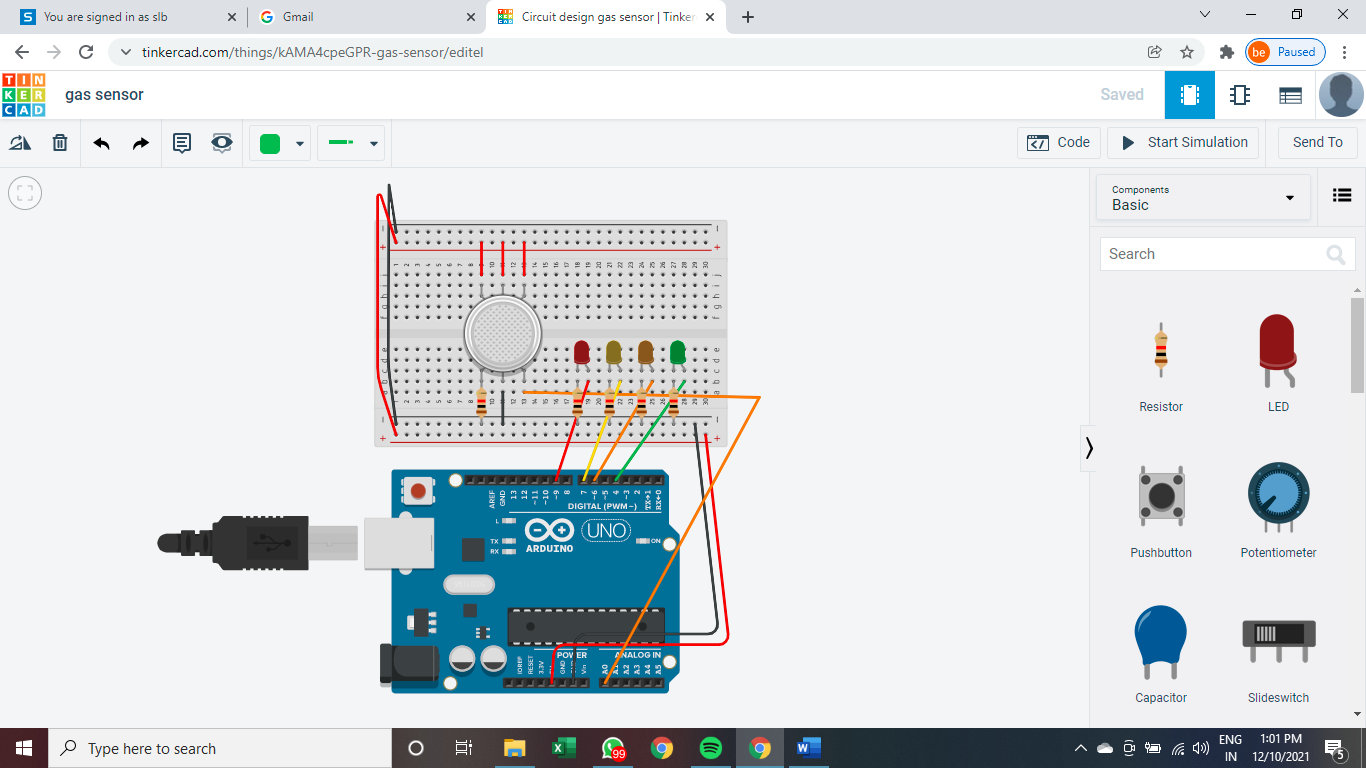
}

else

digitalWrite(5,LOW);

}

# Gas sensor, Led interfacing with Arduino



## Program

int const GasSensor = A1;

int Green = 4;

int Yellow = 7;

int Orange = 6;

int Red = 9;

void setup()

{

pinMode(Green, OUTPUT);

pinMode(Yellow,OUTPUT);

pinMode(Orange,OUTPUT);

pinMode(Red,OUTPUT);

}

void loop()

{

int val= analogRead(GasSensor);

val = map(val,300,750,0,100);

if(val<= 20){

digitalWrite(Green, HIGH);

digitalWrite(Yellow, LOW);

digitalWrite(Orange, LOW);

digitalWrite(Red, LOW);

}

if(val<= 40){

digitalWrite(Green, LOW);

digitalWrite(Yellow, LOW);

digitalWrite(Orange, HIGH);

digitalWrite(Red, LOW);

}

if(val<= 60){

digitalWrite(Green, LOW);

digitalWrite(Yellow, HIGH);

digitalWrite(Orange, LOW);

digitalWrite(Red, LOW);

}

else{

digitalWrite(Green, LOW);

digitalWrite(Yellow, LOW);

digitalWrite(Orange, LOW);

digitalWrite(Red, HIGH);

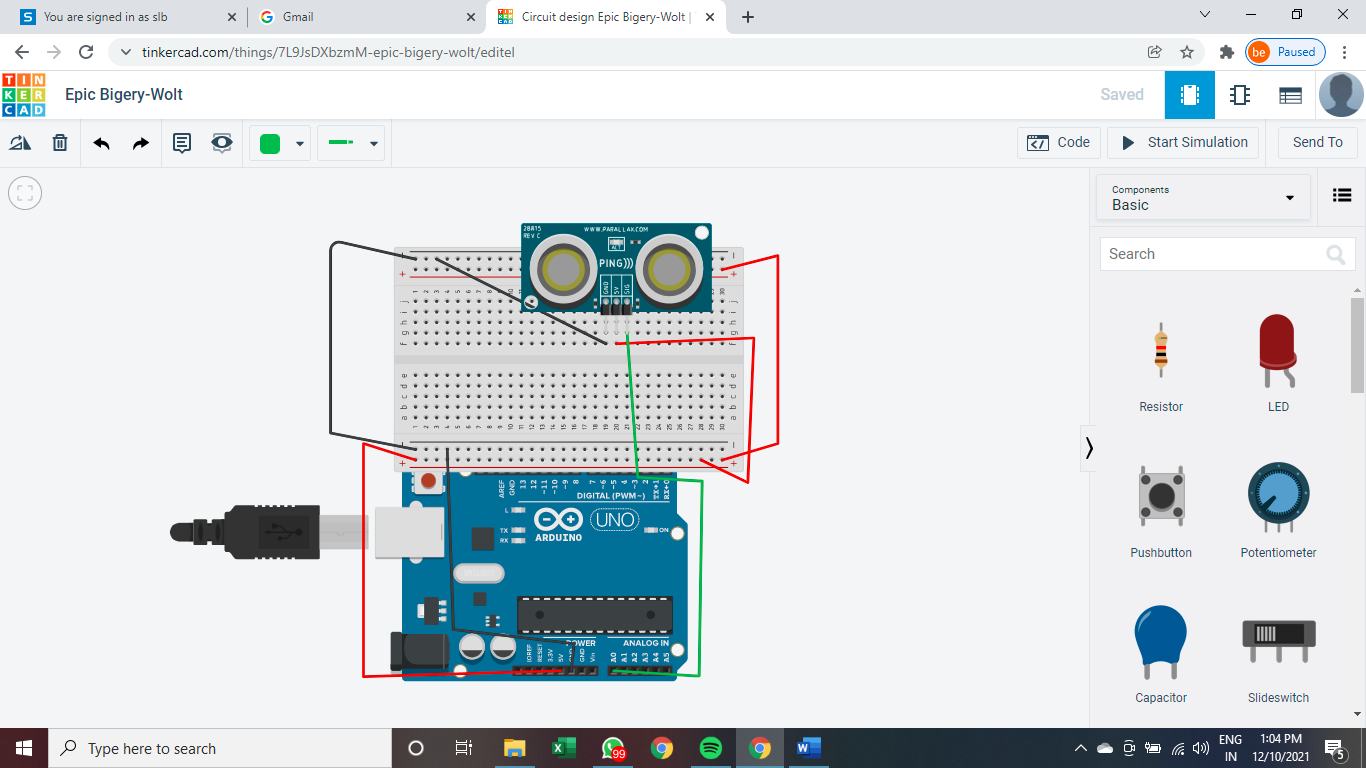
}

Serial.println(val);

delay(500);

}

# Ultrasonic sensor with Arduino



## Program

long readUltrasonicDistance(int triggerPin, int echoPin)

{

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

digitalWrite(triggerPin, LOW);

delayMicroseconds(2);

// Sets the trigger pin to HIGH state for 10 microseconds

digitalWrite(triggerPin, HIGH);

delayMicroseconds(10);

digitalWrite(triggerPin, LOW);

pinMode(echoPin, INPUT);

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

return pulseIn(echoPin, HIGH);

}

void setup()

{

Serial.begin(9600);

}

void loop()

{

Serial.println(0.01723 \* readUltrasonicDistance(A0, A0));

delay(10); // Delay a little bit to improve simulation performance

}

# LCD interfacing with Arduino

## Program

#include <LiquidCrystal.h>

LiquidCrystal lcd(8,9,4,5,6,7);

void setup()

{

lcd.begin(16,2);

lcd.print("Welcome to RSCOE");

}

void loop()

{

lcd.setCursor(0,1);

lcd.print(millis()/1000);