**Practical 1**

1) Write a 8051 C program to sending values 00-FF to port P1

#include <reg51.h>

void main(void)

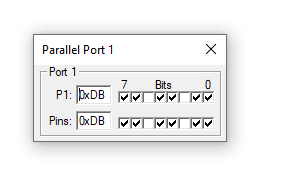
{

unsigned char z;

for(z=0;z<=255;z++)

P1=z;

}



**Practical 2**

2.Sending characters “YES” TO UART with baudrate 9600 using serial communication

#include<reg51.h>

void send(char x);

void main(void)

{

TMOD=0x20;

TH1=0xFD;

SCON=0x50;

TR1=1;

send('Y');

send('E');

send('S');

send('\r');

while(1);

}

void send(char x)

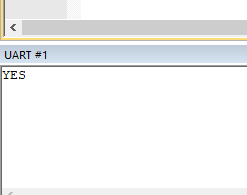
{

SBUF=x;

while(TI==0);

TI=0;

}

}

**Practical 3**

3. Configuring Timer Counter Register(TCON) of 8051 microcontroller and generate time delay to enable interrupt(Toggling two values(10101010 and 01010101) in port P1 with the given time delay)

#include<reg51.h>

void ToDelay(void);

void main(void)

{

while(1)

{

P1=0x55;

ToDelay();

P1=0xAA;

ToDelay();

}

}

void ToDelay()

{

TMOD=0x01;

TL0=0x00;

TH0=0x35;

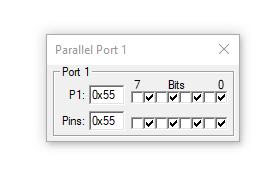
TR0=1;

while(TF0==0);

TR0=0;

TF0=0;

}



**Practical 4**

4. Generating Square wave In D/A converter(logic analyser window)

#include<reg51.h>

void delay();

void main()

{

while(1){

P1=0xff;

delay();

P1=0x00;

delay();

}

}

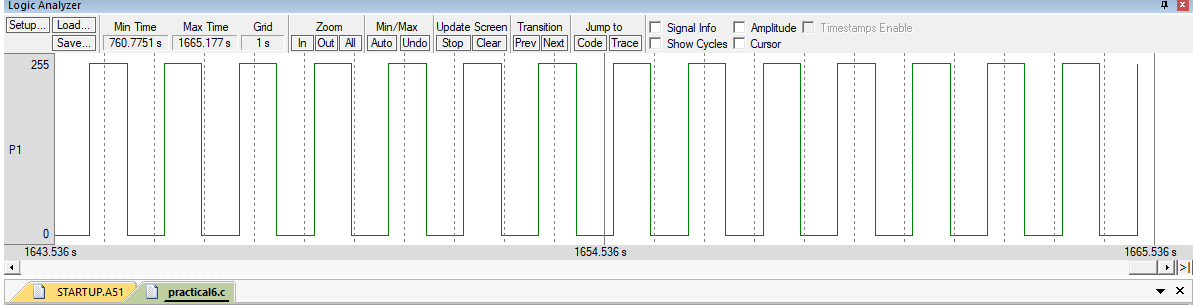
void delay(){

unsigned int i,j;

for(i=0;i<100;i++)

for(j=0;j<1275;j++);

}



**Practical 5**

5. Simulating 8 bit binary counter

include<reg51.h>

void delay(int time);

void main()

{

P1 = 00000000;

while(1)

{

P1++;

delay(100);

}

}

void delay(int time)

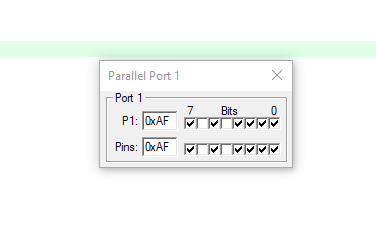
{

int i,j;

for(i=0;i<=time;i++)

for(j=0;j<=23;j++);

}



**Practical 6**

6. Generating triangular wave in D/A converter for given frequency

#include<reg51.h>

void main()

{

P1=0x00;

while(1)

{

do

{

P1+=0x05;

}

while(P1<0xFF);

do

{

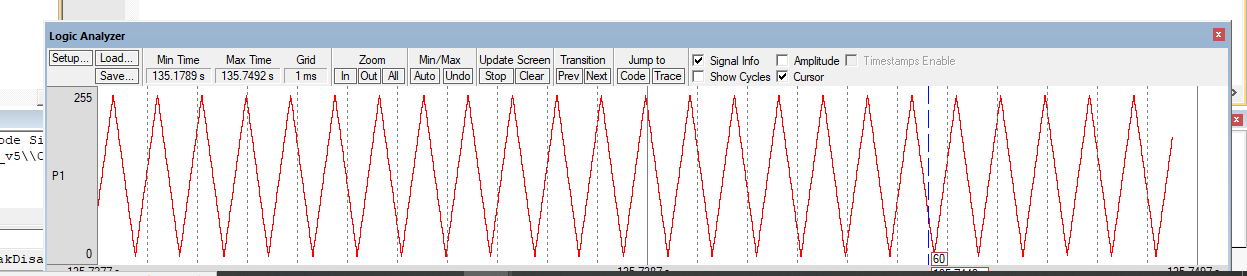
P1-=0x05;

}

while(P1>0x00);

}

}



**Tkinkercad**

**Generating pattern using three LED’S**

**CODE:**

void setup()

{

pinMode(13, OUTPUT);

pinMode(8, OUTPUT);

pinMode(7, OUTPUT);

}

void loop()

{

digitalWrite(13, HIGH);

delay(1000);

digitalWrite(13, LOW);

delay(1000);

digitalWrite(8, HIGH);

delay(500);

digitalWrite(8, LOW);

delay(500);

digitalWrite(8, HIGH);

delay(500);

digitalWrite(8, LOW);

delay(500);

for(int i = 0;i <3;i = i+1)

{

digitalWrite(7, HIGH);

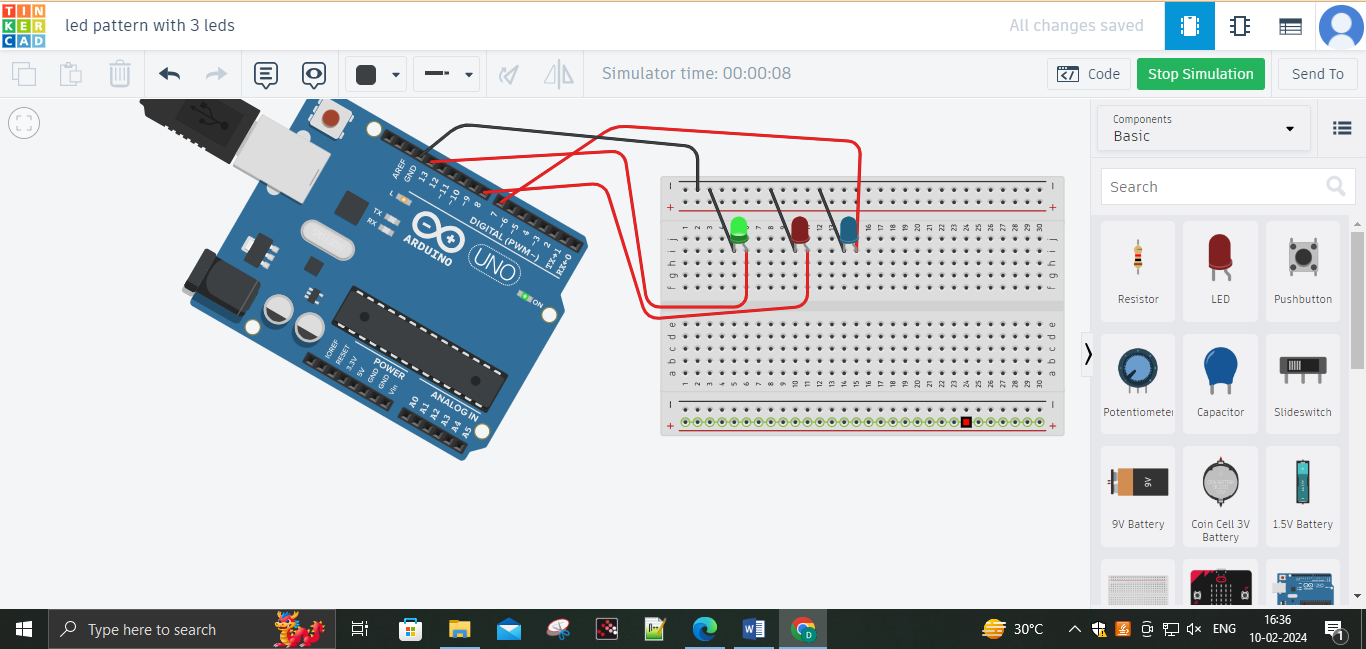
delay(500);

digitalWrite(7,LOW);

delay(500);

}

}



**FADING A LIGHT**

**Code:**

int led = 9;

int brightness = 0;

int fadeAmount = 5;

void setup()

{

pinMode(led, OUTPUT);

}

void loop()

{

analogWrite(led, brightness);

brightness = brightness + fadeAmount;

if (brightness <= 0 || brightness >= 255)

{

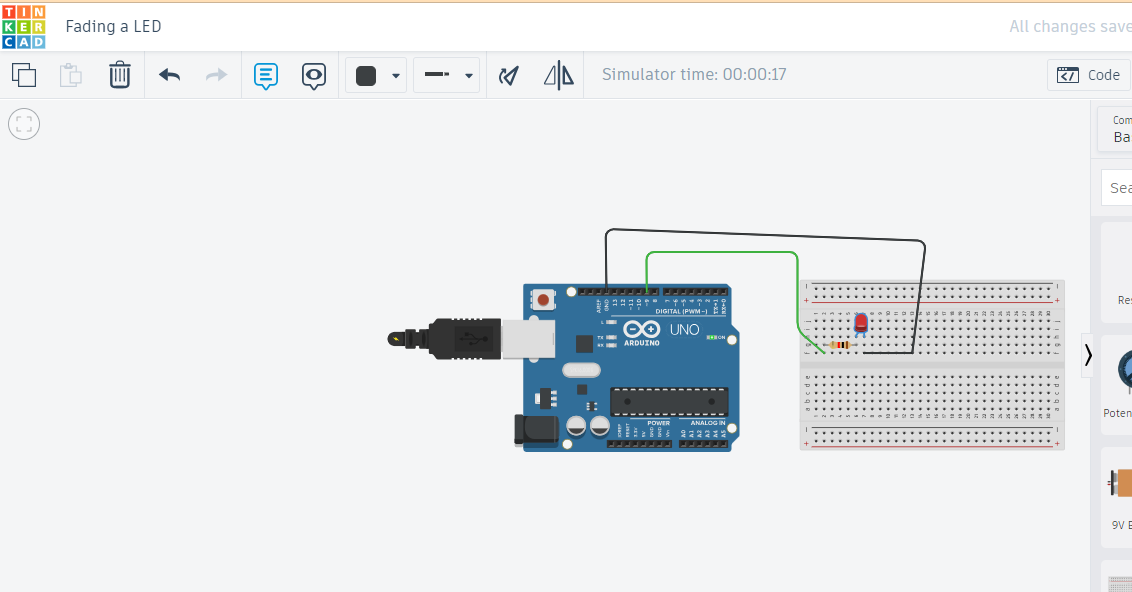
fadeAmount = -fadeAmount;

}

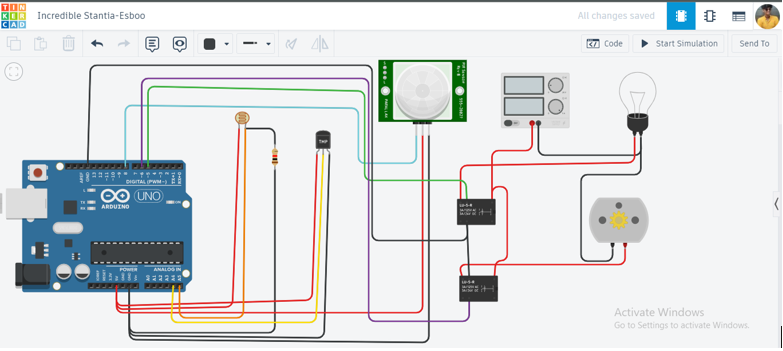
delay(30);

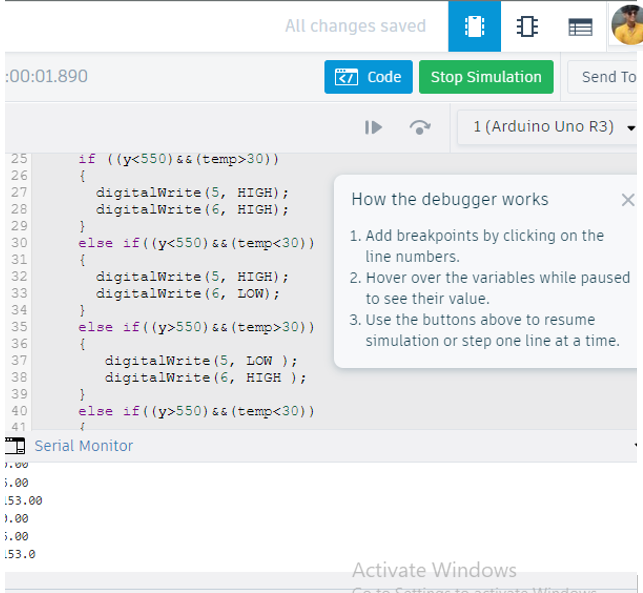
}

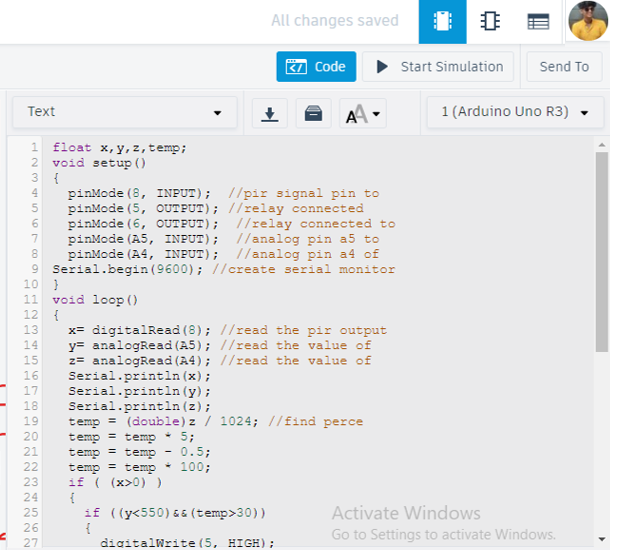
**Output:**

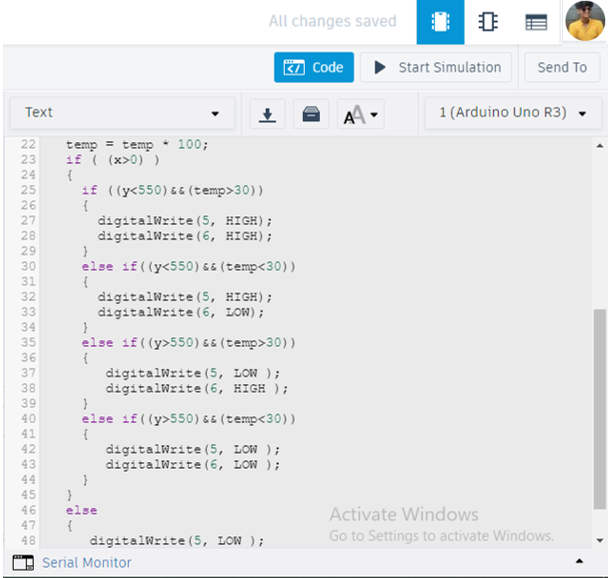
****

**Home Automation 1**

****

****

****

****

**Interfacing LCD (Output) with Arduino**

//C++ Code

#include<LiquidCrystal.h>

LiquidCrystal lcd(12,11,5,4,3,2);

void setup()

{

lcd.begin(16,2);

}

void loop()

{

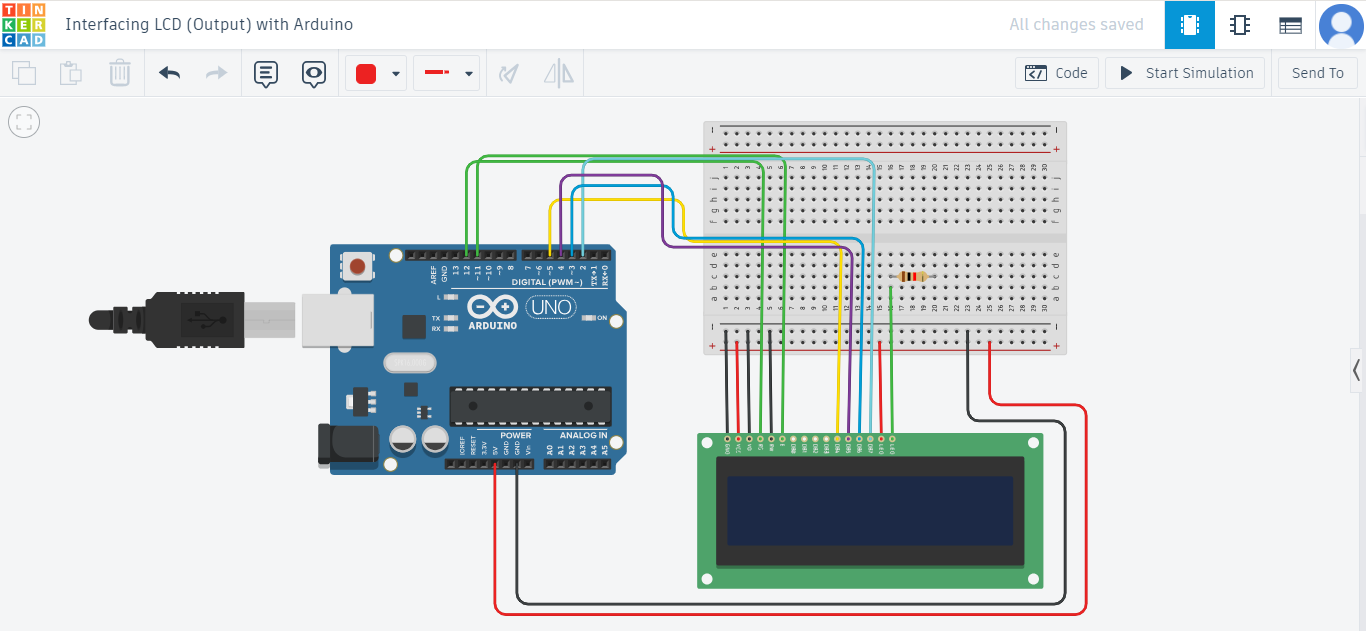
lcd.setCursor(0,0);

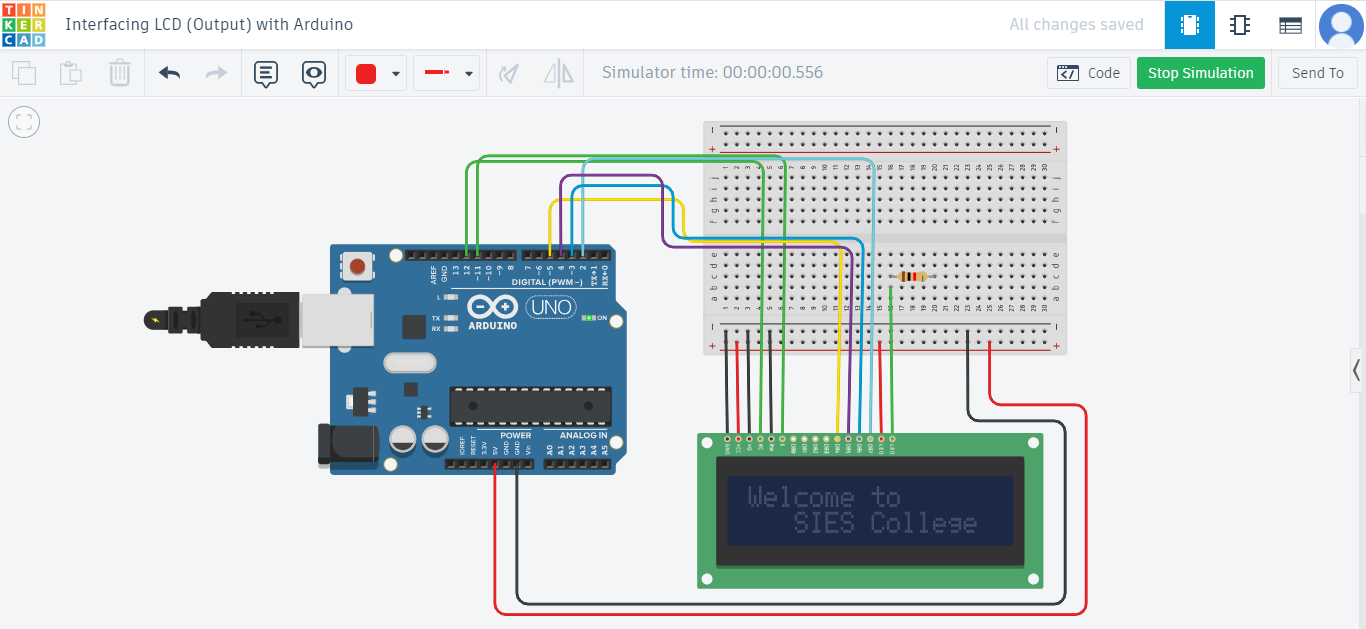
lcd.print("Welcome to");

lcd.setCursor(2,1);

lcd.print(" SIES College");

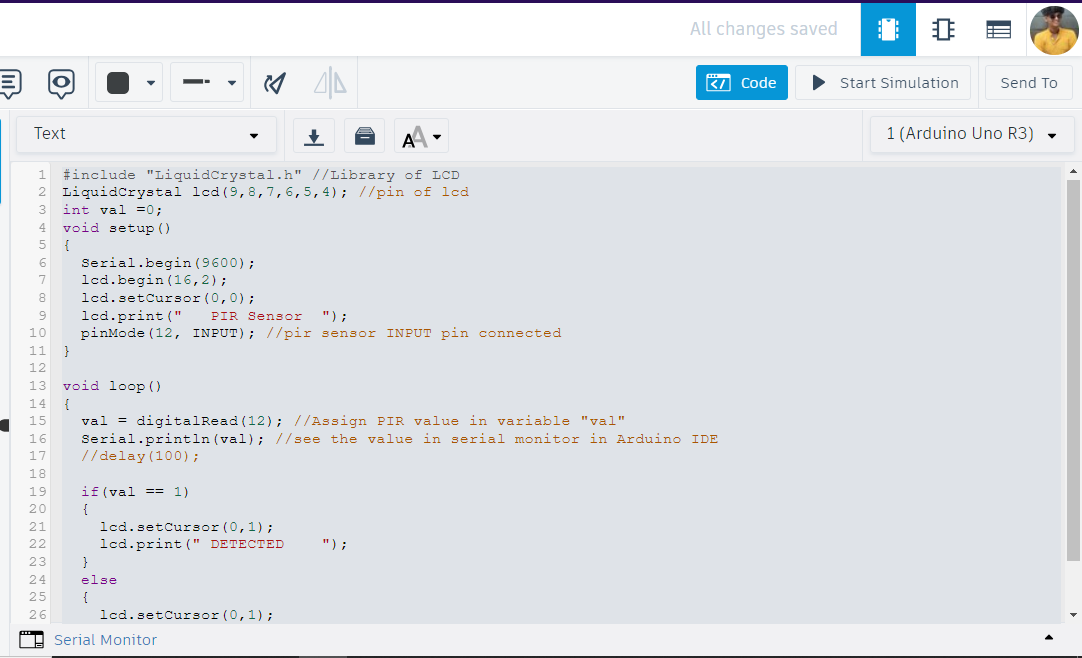
}

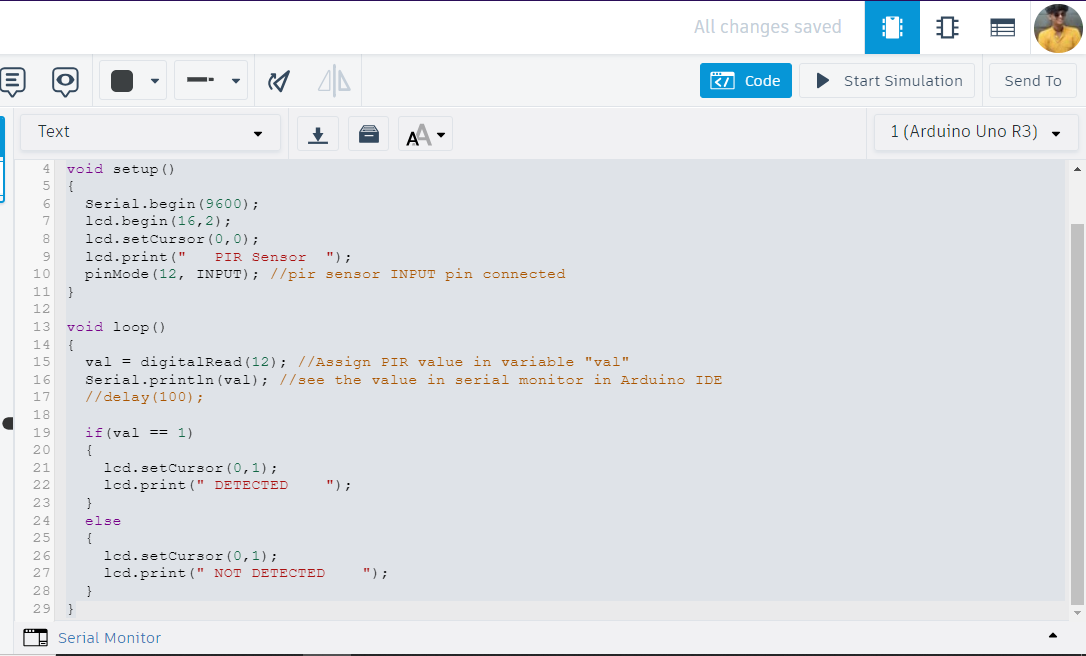




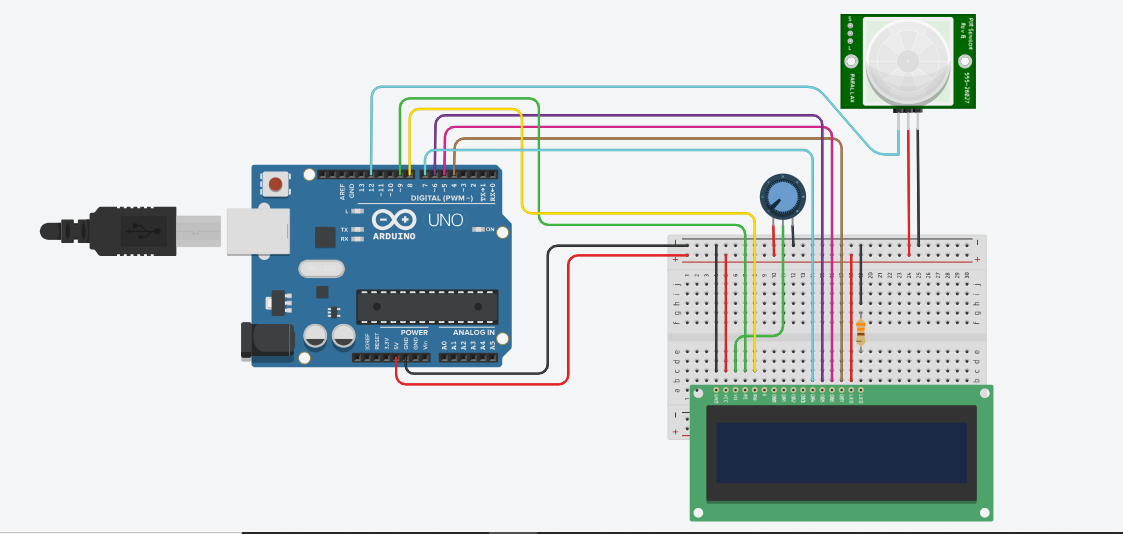
**PIR Sensor with Arduino**

CODE:





OUTPUT:



**Interfacing Ultrasonic sensor with Arduino**

CODE:

const int TRIG\_PIN **=** 13**;** //Arduino pin connected to Ultrasonic Sensor's TRIG pin

const int ECHO\_PIN **=** 12**;** //Arduino pin connected to Ultrasonic Sensor's TRIG pin

const int LED\_PIN **=** 11**;** //Arduino pin Connected to Ultrasonic LED's pin

const int DISTANCE\_THRESHOLD **=** 50**;** //centimeters

//variables wil change:

float duration\_us**,** distance\_cm**;**

void setup**()** **{**

Serial**.**begin**(**9600**);** //initialize serial port 9600 - Baud rate

pinMode**(**TRIG\_PIN**,** OUTPUT**);** //set arduino pin to output mode

pinMode**(**ECHO\_PIN**,** INPUT**);** //set arduino pin to output mode

pinMode**(**LED\_PIN**,** OUTPUT**);** //set arduino pin to output mode

**}**

void loop**()**

**{**

//generate 10-microsecound pulse to TRIG pin

digitalWrite**(**TRIG\_PIN**,**HIGH**);**

delayMicroseconds**(**10**);**

digitalWrite**(**TRIG\_PIN**,**LOW**);**

//measure duration of pulse from ECHO pin

duration\_us **=** pulseIn**(**ECHO\_PIN**,** HIGH**);** //capture the lenth of the pulse and if no pulse capture 0

distance\_cm **=** 0.017 **\*** duration\_us**;**

**if(**distance\_cm **<** DISTANCE\_THRESHOLD**)**

digitalWrite**(**LED\_PIN**,** HIGH**);** // turn on LED

**else**

digitalWrite**(**LED\_PIN**,** LOW**);** // turn off LED

//print the value of serial monitor

Serial**.**print**(**"distance: "**);**

Serial**.**print**(**distance\_cm**);**

Serial**.**println**(**" cm"**);**

delay**(**500**);**

**}**

OUTPUT:

