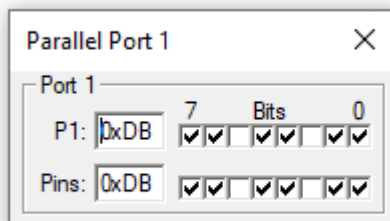


Practical 1

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

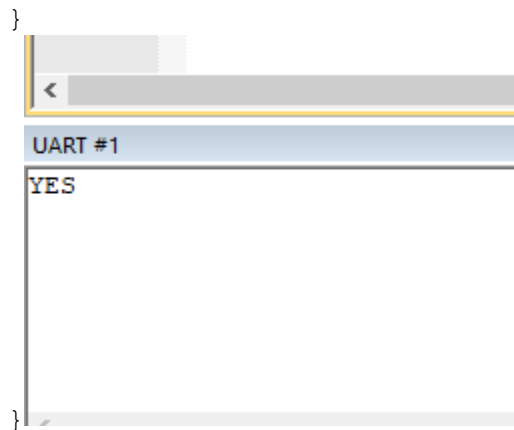
```
#include <reg51.h>
void main(void)
{
    unsigned char z;
    for(z=0; z<=255; z++)
        P1=z;
}
```



Practical 2

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

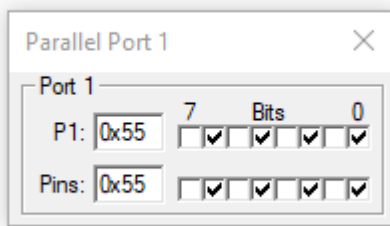
```
#include<reg51.h>
void send(char x);
void main(void)
{
    TMOD=0x20;
    TH1=0xFD;
    SCON=0x50;
    TR1=1;
    send('Y');
    send('E');
    send('S');
    send('\r');
    while(1);
}
void send(char x)
{
    SBUF=x;
    while(TI==0);
    TI=0;
}
```



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)

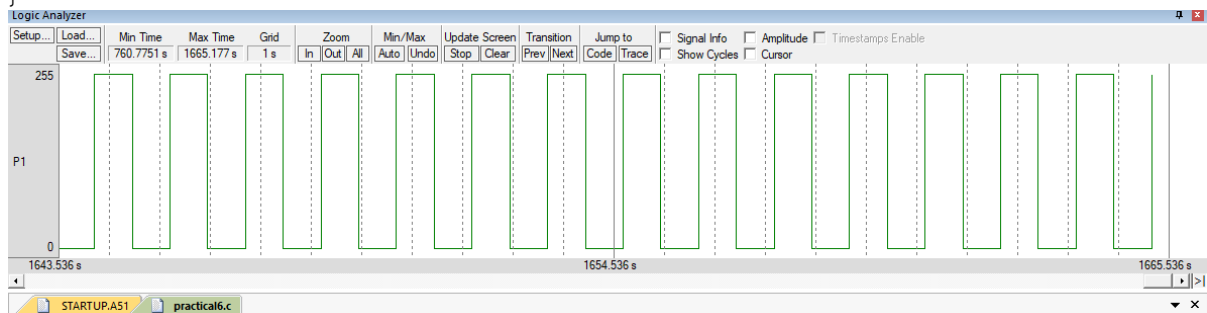
```
#include<reg51.h>
void ToDelay(void);
void main(void)
{
while(1)
{
P1=0x55;
ToDelay();
P1=0xAA;
ToDelay();
}
}
void ToDelay()
{
TMOD=0x01;
TL0=0x00;
TH0=0x35;
TR0=1;
while(TF0==0);
TR0=0;
TF0=0;
}
```



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>

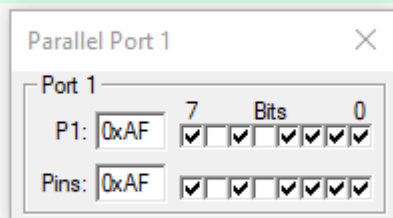
void delay(int time);
void main()
{
    P1 = 00000000;
    while(1)
    {
        P1++;
        delay(100);
    }
}

void delay(int time)
{
    int i,j;

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

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

}
```



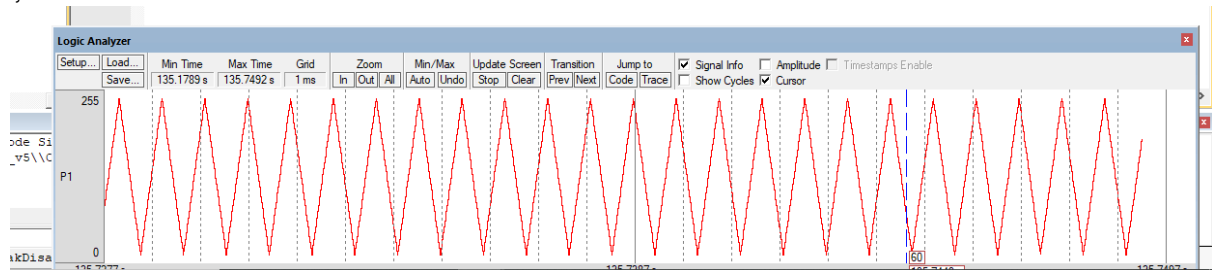
Practical 6

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);
}
}
```



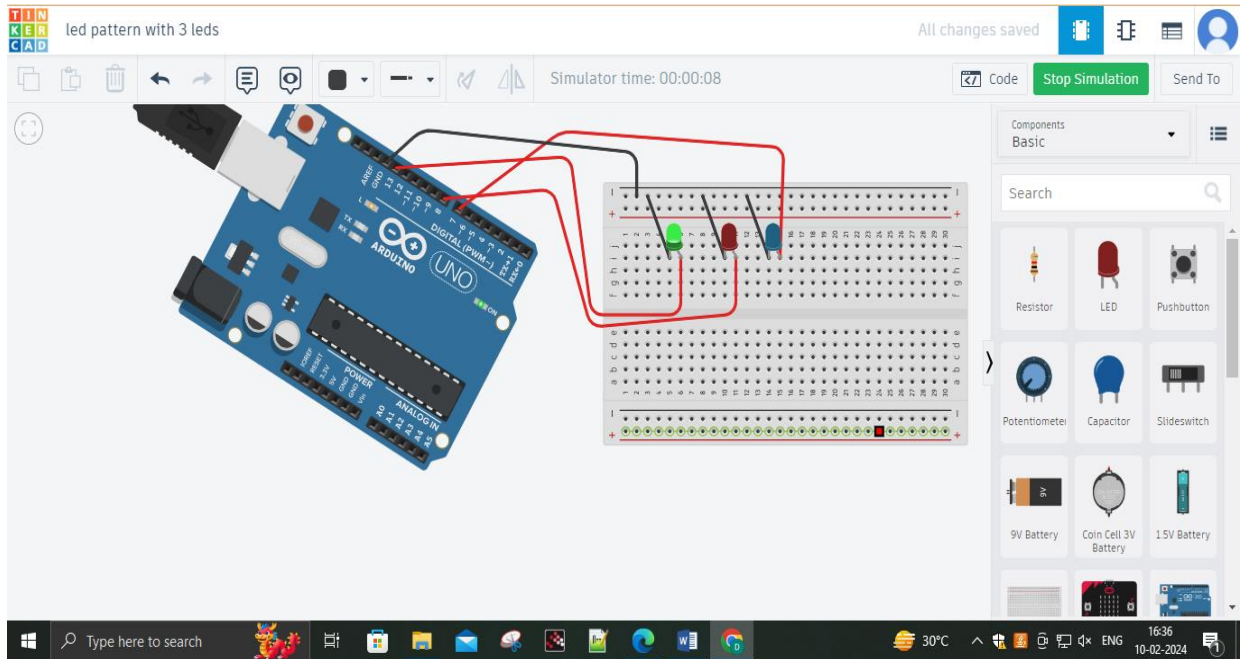
Tkintercad

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:

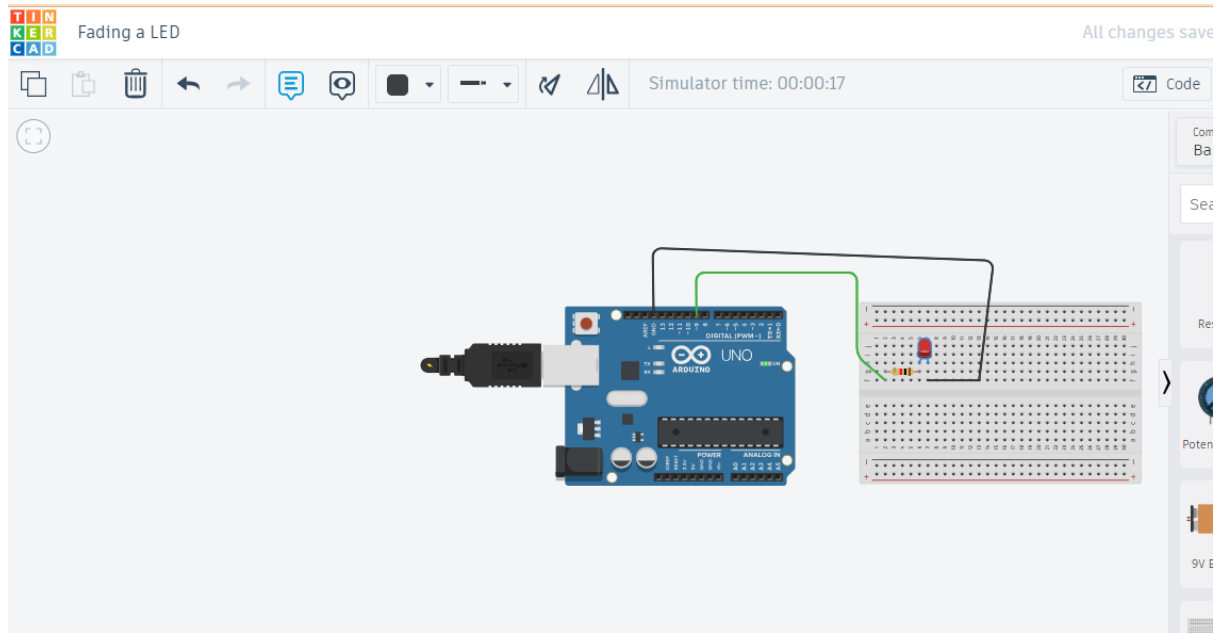
```
int led = 9;
int brightness = 0;
int fadeAmount = 5;
void setup()
{
  pinMode(led, OUTPUT);
}
void loop()
{
  analogWrite(led, brightness);
  brightness = brightness + fadeAmount;
  if (brightness <= 0 || brightness >= 255)
  {
```

```

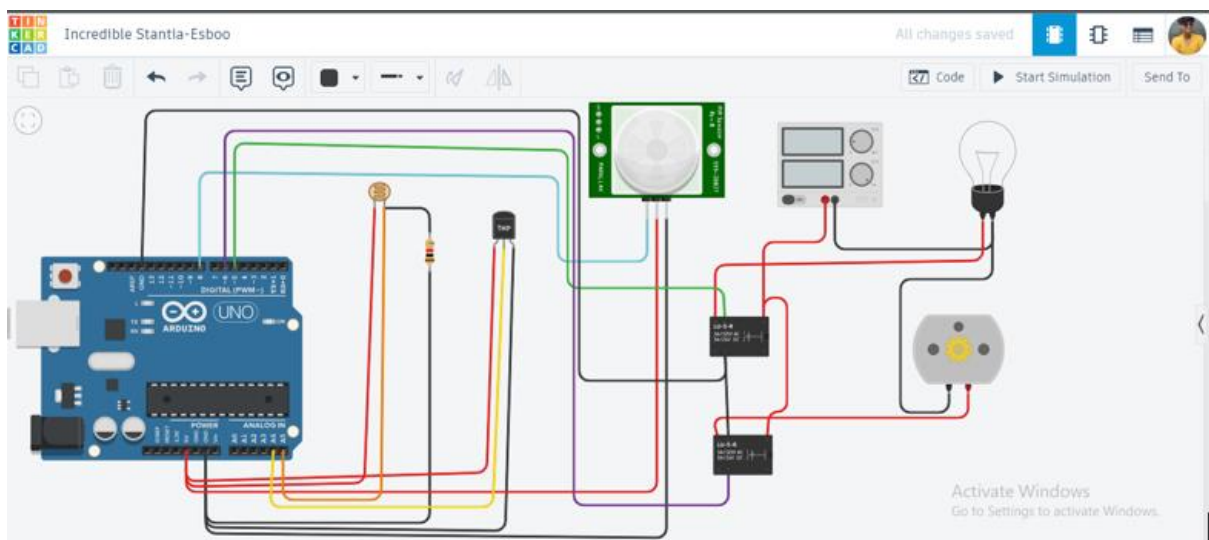
    fadeAmount = -fadeAmount;
  }
  delay(30);
}

```

Output:



Home Automation 1



All changes saved



00:01.890



Code

Stop Simulation

Send To



1 (Arduino Uno R3) ▾

```
25     if ((y<550)&&(temp>30))
26     {
27         digitalWrite(5, HIGH);
28         digitalWrite(6, HIGH);
29     }
30     else if((y<550)&&(temp<30))
31     {
32         digitalWrite(5, HIGH);
33         digitalWrite(6, LOW);
34     }
35     else if((y>550)&&(temp>30))
36     {
37         digitalWrite(5, LOW );
38         digitalWrite(6, HIGH );
39     }
40     else if((y>550)&&(temp<30))
41     {
```

How the debugger works



1. Add breakpoints by clicking on the line numbers.
2. Hover over the variables while paused to see their value.
3. Use the buttons above to resume simulation or step one line at a time.



Serial Monitor

0.00
0.00
153.00
0.00
0.00
153.0

Activate Windows

All changes saved



Code

Start Simulation

Send To

Text

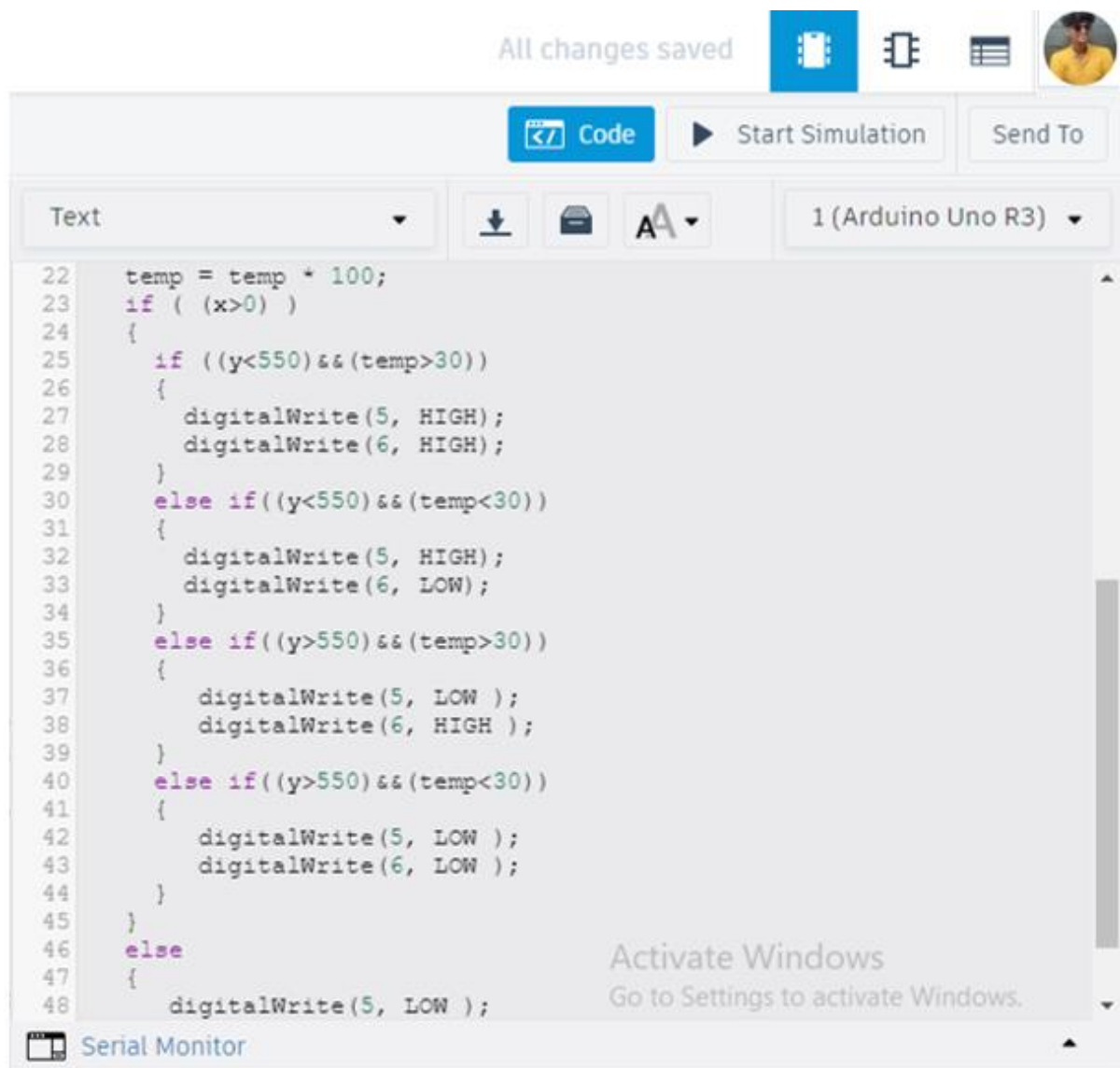


1 (Arduino Uno R3)

```
1 float x,y,z,temp;
2 void setup()
3 {
4   pinMode(8, INPUT); //pir signal pin to
5   pinMode(5, OUTPUT); //relay connected
6   pinMode(6, OUTPUT); //relay connected to
7   pinMode(A5, INPUT); //analog pin a5 to
8   pinMode(A4, INPUT); //analog pin a4 of
9   Serial.begin(9600); //create serial monitor
10 }
11 void loop()
12 {
13   x= digitalRead(8); //read the pir output
14   y= analogRead(A5); //read the value of
15   z= analogRead(A4); //read the value of
16   Serial.println(x);
17   Serial.println(y);
18   Serial.println(z);
19   temp = (double)z / 1024; //find perce
20   temp = temp * 5;
21   temp = temp - 0.5;
22   temp = temp * 100;
23   if ( (x>0) )
24   {
25     if ( (y<550)&&(temp>30))
26     {
27       digitalWrite(5, HIGH);
```

Activate Windows

Go to Settings to activate Windows.



Interfacing LCD (Output) with Arduino

//C++ Code

```
#include<LiquidCrystal.h>
```

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

```
void setup()
```

```
{
```

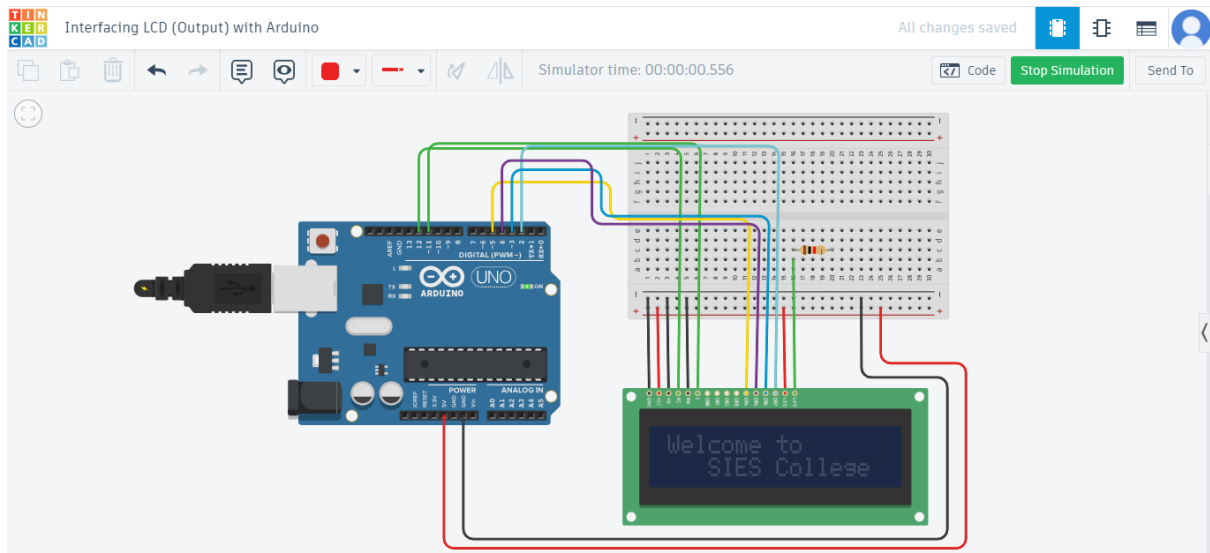
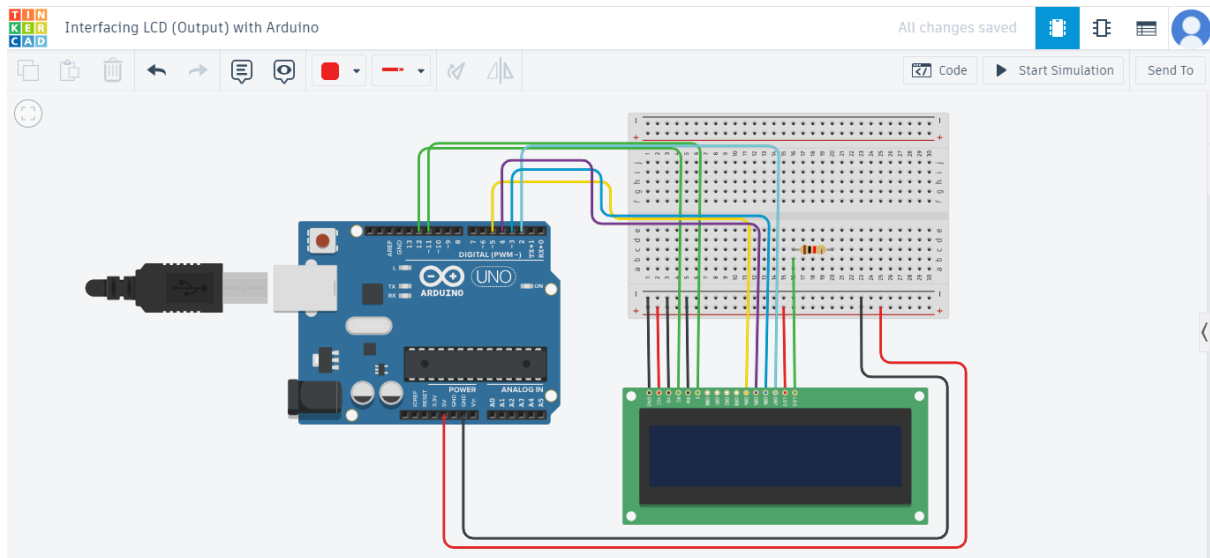
```
  lcd.begin(16,2);
```

```
}
```

```

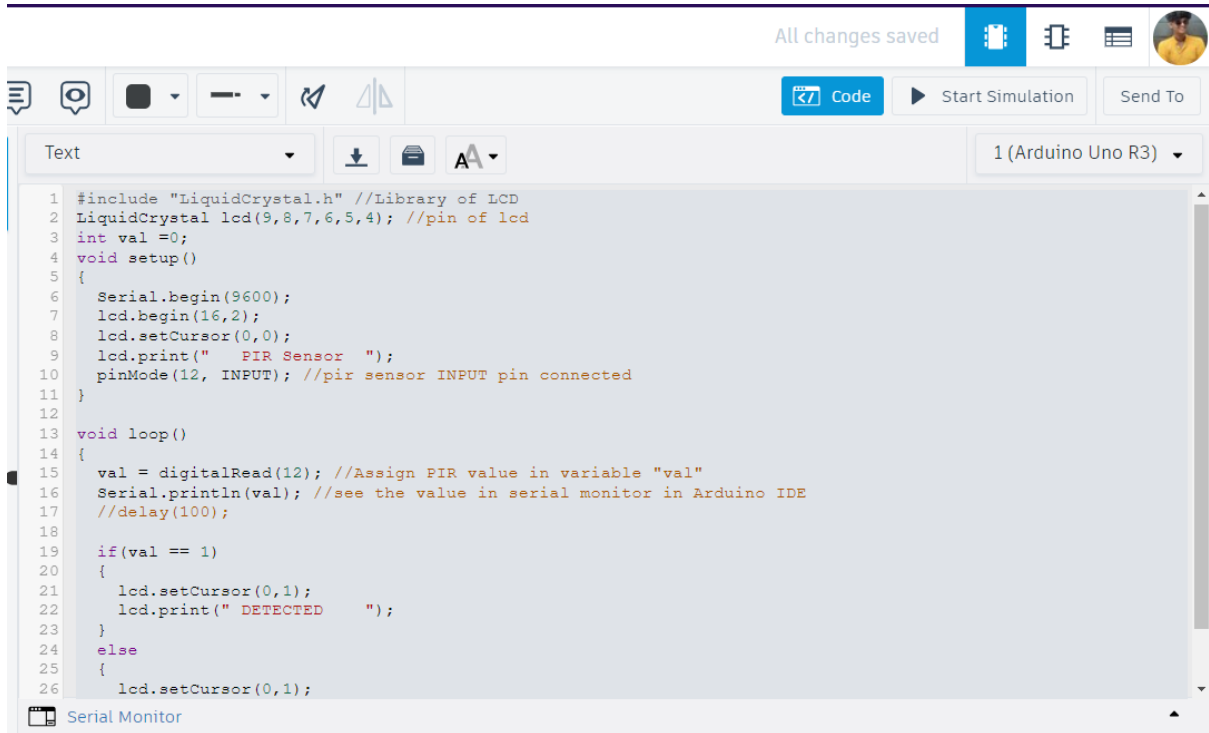
void loop()
{
    lcd.setCursor(0,0);
    lcd.print("Welcome to");
    lcd.setCursor(2,1);
    lcd.print(" SIES College");
}

```

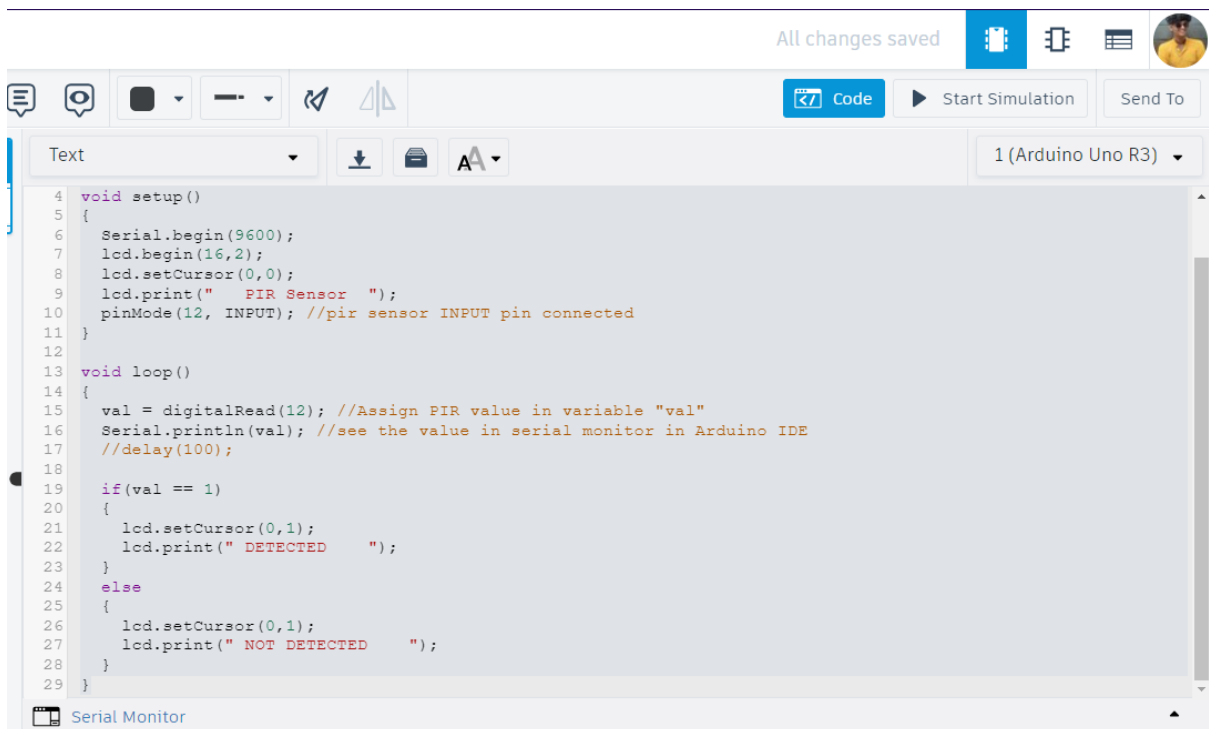


PIR Sensor with Arduino

CODE:

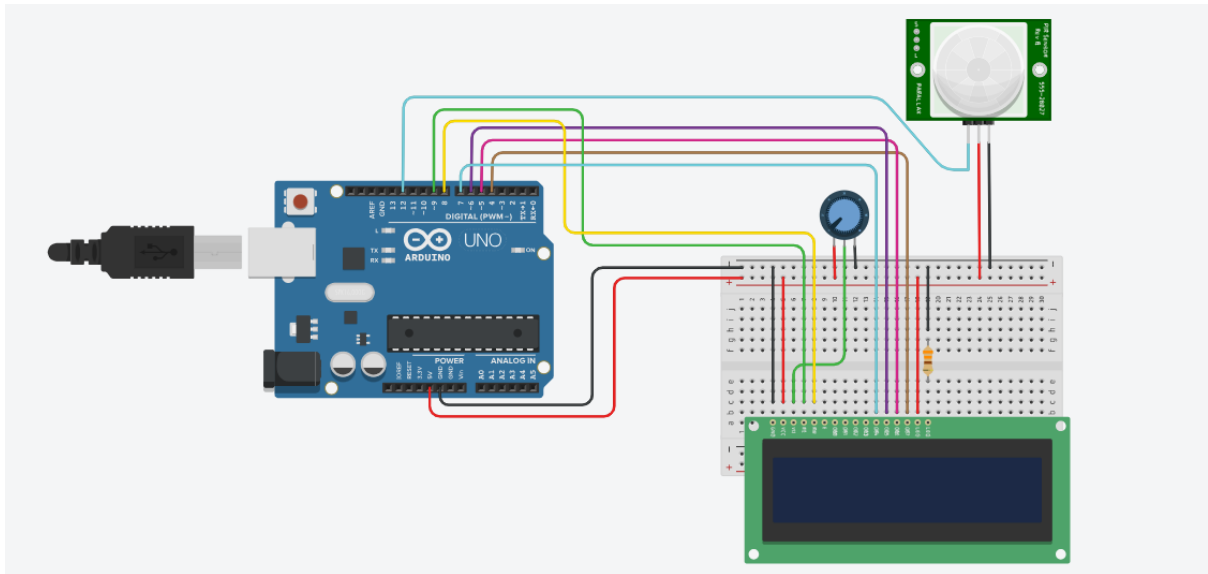


```
1 #include "LiquidCrystal.h" //Library of LCD
2 LiquidCrystal lcd(9,8,7,6,5,4); //pin of lcd
3 int val =0;
4 void setup()
5 {
6   Serial.begin(9600);
7   lcd.begin(16,2);
8   lcd.setCursor(0,0);
9   lcd.print("   PIR Sensor   ");
10  pinMode(12, INPUT); //pir sensor INPUT pin connected
11 }
12
13 void loop()
14 {
15   val = digitalRead(12); //Assign PIR value in variable "val"
16   Serial.println(val); //see the value in serial monitor in Arduino IDE
17   //delay(100);
18
19   if(val == 1)
20   {
21     lcd.setCursor(0,1);
22     lcd.print(" DETECTED   ");
23   }
24   else
25   {
26     lcd.setCursor(0,1);
```



```
4 void setup()
5 {
6   Serial.begin(9600);
7   lcd.begin(16,2);
8   lcd.setCursor(0,0);
9   lcd.print("   PIR Sensor   ");
10  pinMode(12, INPUT); //pir sensor INPUT pin connected
11 }
12
13 void loop()
14 {
15   val = digitalRead(12); //Assign PIR value in variable "val"
16   Serial.println(val); //see the value in serial monitor in Arduino IDE
17   //delay(100);
18
19   if(val == 1)
20   {
21     lcd.setCursor(0,1);
22     lcd.print(" DETECTED   ");
23   }
24   else
25   {
26     lcd.setCursor(0,1);
27     lcd.print(" NOT DETECTED ");
28   }
29 }
```

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
ECHO pin
const int LED_PIN = 11; //Arduino pin Connected to Ultrasonic LED's pin
const int DISTANCE_THRESHOLD = 50; //centimeters

//variables will 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-microsecond 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;

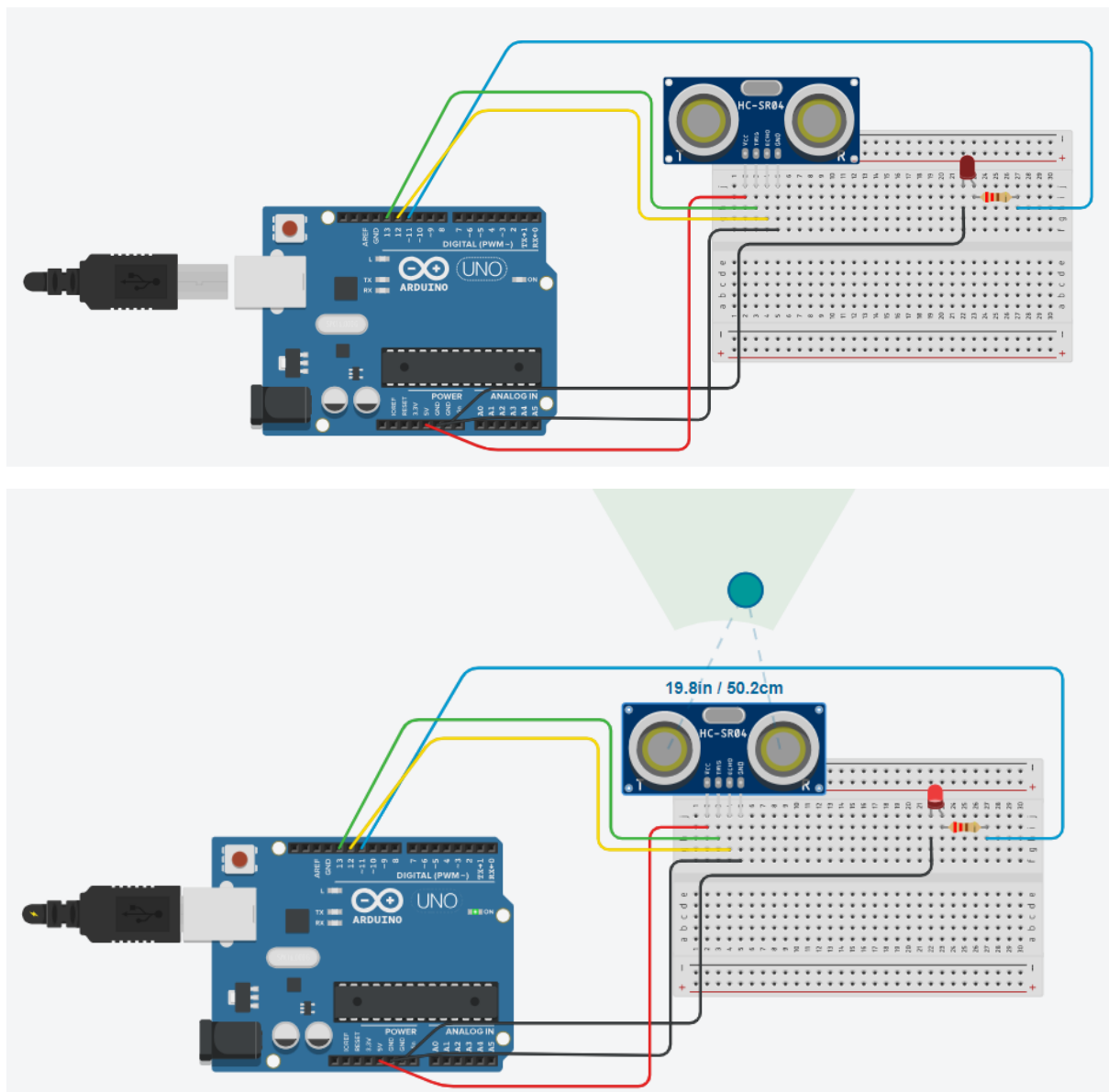
    if(distance_cm < DISTANCE_THRESHOLD)
        digitalWrite(LED_PIN, HIGH); // turn on LED
    else
        digitalWrite(LED_PIN, LOW); // turn off LED

    //print the value of serial monitor
    Serial.print("distance: ");
    Serial.print(distance_cm);
    Serial.println(" cm");

    delay(500);
}

```

OUTPUT:



TIN
KER
CAD

Interfacing Ultrasonic sensor with Arduino

All changes saved

Simulator time: 00:02:00

Code

Stop Simulation

Send To

1 (Arduino Uno R3)

Ultrasonic Distance Sensor (...)

Name 1

17.8in / 45.3cm

```
13 pinMode(LED_PIN, OUTPUT); //set arduino pin to output mode
14 }
15
16 void loop()
17 {
18     //generate 10-microsecond pulse to TRIG pin
19     digitalWrite(TRIG_PIN,HIGH);
20     delayMicroseconds(10);
21     digitalWrite(TRIG_PIN,LOW);
22
23     //measure duration of pulse from ECHO pin
24     duration_us = pulseIn(ECHO_PIN, HIGH); //capture the lenth of t
25     distance_cm = 0.017 * duration_us;
26
27     if(distance_cm < DISTANCE_THRESHOLD)
28     {
```

Serial Monitor

distance: 35.61 cm
distance: 44.91 cm
distance: 44.91 cm
distance: 44.91 cm
distance: 44.88 cm
distance: 44.91 cm

Send Clear