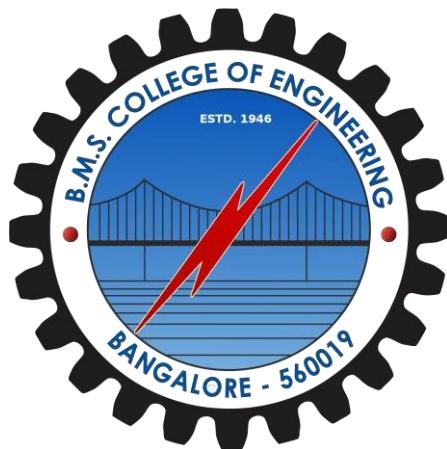


IOT LAB MANUAL-2020



**Name: Shivam Vaish
USN: 1BM18CS152**

CONSOLIDATED LAB PROGRAMS

Index:

Expt No	Title	Page No.
1	Traffic Lights	3-5
2	Push Button	6-8
3	Fade LED	9-11
4	Potentiometer	12-14
5	Temperature	15-17
6	Fire Alarm	18-20
7	Tilt Sensor	21-23
8	Ultrasonic Distance	24-26
9	Gas Leakage	27-29
10	IR Remote	30-32
11	LDR Sensor	33-35
12	PIR	36-38
13	Weather	39-41
14	Humidity	42-44
15	LCD with RGB LEDs	45-47

1. Traffic Light

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD traffic controller All changes saved Code Stop Simulation Export Share Simulator time: 00:01:51

The circuit diagram shows an Arduino Uno connected to a breadboard. The breadboard has three red LEDs connected to pins 10, 11, and 12. A pushbutton is connected between pin 13 and ground. The Arduino Uno is connected to the breadboard via a USB cable.

```
#define yellow 10
#define green 11
#define red 12
void setup()
{
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(10, OUTPUT);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
void turnon(int mypin,int mytime)
{
  digitalWrite(mypin,HIGH);
  delay(mytime);
  digitalWrite(mypin,LOW);
}
void loop()
{
  turnon(red, 30000);
  turnon(yellow, 15000);
  turnon(green, 30000);
  turnon(yellow, 15000);
}
```

Serial Monitor

Windows taskbar: Type here to search, Start button, File Explorer, Edge, File Explorer, Mail, Google Chrome, File Explorer, 1740, ENG, 03-11-2020, Notifications.

Code

define yellow 10
define green 11
define red 12

void setup()
{
 pinMode (12, OUTPUT);
 pinMode (11, OUTPUT);
 pinMode (10, OUTPUT);
}

void turnOn (int mypin, int mytime)
{
 digitalWrite (mypin, HIGH);
 delay (mytime);
 digitalWrite (mypin, LOW);
}

void loop()
{
 turnOn (red, 3000);
 turnOn (yellow, 1500);
 turnOn (green, 3000);
 turnOn (yellow, 1500);
}

Shivam Vaish
IBM18CS152

```
#define yellow 10
#define green 11
#define red 12
void setup()
{
    pinMode(12, OUTPUT);
    pinMode(11,OUTPUT);
    pinMode(10,OUTPUT);
    digitalWrite(12,LOW);
    digitalWrite(11,LOW);
    digitalWrite(10,LOW);
}
void turnon(int mypin,int mytime)
{
    digitalWrite(mypin,HIGH);
    delay(mytime);
    digitalWrite(mypin,LOW);

}
void loop()
{
    turnon(red, 3000);
    turnon(yellow, 1500);

    turnon(green, 3000);
    turnon(yellow, 1500);
}
```

2. Push Button

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-push-button All changes saved

Text Start Simulation Export Share 1 (Arduino Uno R3)

How it works
1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

```
1 void setup()
2 {
3     pinMode(12, OUTPUT);
4     digitalWrite(10,LOW);
5     pinMode(2,INPUT);
6 }
7 void loop()
8 {
9     if(digitalRead(2)==HIGH)
10    {
11        digitalWrite(12,HIGH);
12    }
13    else
14    {
15        digitalWrite(12,LOW);
16    }
17 }
18 }
```

Serial Monitor

Type here to search 1809 03-11-2020

Code

Shivam Vaish
13M18CS152

```
void Setup()
{
    pinMode(12, OUTPUT);
    digitalWrite(12, LOW);
    pinMode(2, INPUT);
}
```

```
void loop()
{
    if(digitalRead(2) == HIGH)
    {
        digitalWrite(12, HIGH);
    }
    else
    {
        digitalWrite(12, LOW);
    }
}
```

```
void setup()
{
  pinMode(12, OUTPUT);
  digitalWrite(10,LOW);
  pinMode(2,INPUT);
}
void loop()
{
  if(digitalRead(2)==HIGH)
  {
    digitalWrite(12,HIGH);
  }
  else
  {
    digitalWrite(12,LOW);
  }
}
```

3. Fade Bulb

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

All changes saved

TIN KER CAD 1bm18cs152-fade-bulb

Code Start Simulation Export Share 1 (Arduino Uno R3)

How it works

1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Text

```
1 int led = 12;
2 int brightness = 0;
3 int fadeAmount = 5;
4
5 void setup() {
6   pinMode(led, OUTPUT);
7 }
8
9 void loop() {
10   analogWrite(led, brightness);
11
12   brightness = brightness + fadeAmount;
13
14   if (brightness == 0 || brightness == 255) {
15     fadeAmount = -fadeAmount;
16   }
17   delay(30);
18 }
```

Serial Monitor

The screenshot shows a Tinkercad workspace for a breadboard circuit. On the left, a breadboard is populated with components: a red LED connected to digital pin 12 via a resistor, a potentiometer connected to digital pin 0, and a 10k pull-down resistor connected to ground. A USB cable is connected to the Arduino Uno board. The Arduino Uno has its pins labeled: AREF, GND, 3, 2, 1, 0, 9, 8, 7, 6, 5, 4, 3, 2, 1, TX, RX, GND, POWER, ANALOG IN. The code in the center is for a fading LED using analogWrite(). The status bar at the bottom indicates the date and time: 03-11-2020, 18:18, ENG.

CodeShivam Vaish
1BM18CS152

```
int led = 12;
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);
}
```

```
int led = 12;
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);
}
```

4. Potentiometer

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-potentiometer All changes saved

Text 1 (Arduino Uno R3)

```
1 int led = 12;
2 int brightness = 0;
3
4 void setup() {
5   Serial.begin(9600);
6   pinMode(led, OUTPUT);
7 }
8
9 void loop() {
10   brightness=map(analogRead(A0),0,1023,0,255);
11   analogWrite(led,brightness);
12   Serial.print("Brightness: ");
13   Serial.println(brightness);
14 }
```

How it works
1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Serial Monitor

Brightness: 244
Brightness:

Type here to search 1838 ENG 03-11-2020

Code

Shivamveish

IBM/18CS152

```
int led = 12;
int brightness = 0;

void setup()
{
    Serial.begin (9600);
    pinMode (led, OUTPUT);
}

void loop ()
{
    brightness = map (analogRead (A0), 0, 1023, 0, 255);
    analogWrite (led, brightness);
    Serial.print (" Brightness: ");
    Serial.println (brightness);
    delay (50);
}
```

```
int led = 12;  
int brightness = 0;  
  
void setup() {  
    Serial.begin(9600);  
    pinMode(led, OUTPUT);  
}  
  
void loop() {  
  
    brightness=map(analogRead(A0),0  
,1023,0,255);  
    analogWrite(led, brightness);  
    Serial.print("Brightness: ");  
    Serial.println(brightness);  
}
```

5. Temperature Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

Simulator time: 00:01:21 All changes saved

Code Stop Simulation Export Share

1 (Arduino Uno R3)

How it works

1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Text

```
1 #define temp A0
2 void setup()
3 {
4     Serial.begin(9600);
5 }
6
7 void loop()
8 {
9     float milivolts=analogRead(temp)*0.004882814;
10    float celsius=(milivolts-0.5)*100.0;
11    Serial.print(celsius);
12    Serial.println(" C");
13    float faran=(celsius*9.0/5.0)+32.0;
14    Serial.print(faran);
15    Serial.println(" F");
16 }
17
18 }
```

Serial Monitor

124.80° C
256.65° F

Type here to search 0133 04-11-2020

Send Clear

```
#define temp A0
void setup()
{
    Serial.begin(9600);
}
void loop()
{
    float milivolts=analogRead(temp)*0.004882814;
    float celsius=(milivolts-0.5)*100.0;
    Serial.print(celsius);
    Serial.println(" C");
    float faran=(celsius*9.0/5.0)+32.0;
    Serial.print(faran);
    Serial.println(" F");
}
```

124.80° C
256.65° F
124.80° C
256.65° F
124.80° C
256.65° F
124.80° C
256.65° F

```
# define temp A0
void setup()
{
    Serial.begin(9600);
}

void loop()
{
    float millivolts = analogRead(temp) + 0.004882814;
    float celsius = (millivolts - 0.5) * 100;
    Serial.print(celsius);
    Serial.println("C");
    float faran = (celsius * 9.0/5.0) + 32.0;
    Serial.print(faran);
    Serial.println("F");
    delay(100);
}
```

```
#define temp A0
void setup()
{
    Serial.begin(9600);

}
void loop()
{
    float
    milivolts=analogRead(temp)*0.004
    882814;
    float celsius=(milivolts-
    0.5)*100.0;
    Serial.print(celsius);
    Serial.println(`C`);
    float
    faran=(celsius*9.0/5.0)+32.0;
    Serial.print(faran);
    Serial.println(`F`);
    delay(100);
}
```

6. Fire Detector

The screenshot shows a Tinkercad project titled "1bm18cs152-fire-detector". The circuit diagram on the left features a breadboard with a TMP36 temperature sensor connected to pins A0 and A1 of the Arduino Uno. The Arduino is connected to a breadboard power source and a red LED with a resistor. The code editor in the center contains an Arduino sketch that reads the analog input from the TMP36 sensor, converts it to Celsius, prints the value, and then outputs a sound if the temperature is above 37.0°C. The serial monitor on the right shows the temperature values being printed.

```
serial.begin(9600);
void loop()
{
    float milivolts=analogRead(temp)*0.004882814;
    float celsius=(milivolts-0.5)*100.0;
    Serial.print(celsius);
    Serial.println(" C");
    float farane=(celsius*9.0/5.0)+32.0;
    Serial.print(faran);
    Serial.println(" F");
    if(celsius>37.0)
    {
        digitalWrite(buzzer,HIGH);
        tone(buzzer,500,1000);
        Serial.println("FIRE FIRE FIRE");
    }
    else
    {
        digitalWrite(buzzer,LOW);
        Serial.println("CHILL CHILL CHILL");
    }
    delay(100);
}
```

Text

```
-9.96 °C  
14.07 °F  
CHILL CHILL CHILL  
-9.96 °C  
14.07 °F  
CHILL CHILL CHILL  
-9.96 °C  
14.0
```

Code Stop Simulation Export Share

1 (Arduino Uno R3)

How it works

1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Type here to search

0154 04-11-2020 ENG

```
# define temp A0
# define buzzer 12

Void Setup()
{
    PinMode ( buzzer, OUTPUT);
    Serial.begin ( 9600);
}

Void Loop()
{
    float millivolts = analogRead (temp) * 0.004882884;
    float celsius = (millivolts - 0.5) * 100.0;
    Serial.print (celsius);
    Serial.println (" °C");
    float faran = (celsius * 9.0/5.0) + 32.0;
    Serial.print (faran);
    Serial.println (" °F");
    if (celsius > 37.0)
    {
        digitalWrite (buzzer, HIGH);
        tone (buzzer, 500, 1000);
        Serial.println (" Fire Fire Fire ");
    }
    else
    {
        digitalWrite (buzzer, LOW);
        Serial.println (" Chill ");
    }
    delay (100);
}
```

```
#define temp A0
#define buzzer 12
void setup()
{
    pinMode(buzzer, OUTPUT);
    Serial.begin(9600);

}

void loop()
{
    float milivolts=analogRead(temp)*0.004882814;
    float celsius=(milivolts-0.5)*100.0;
    Serial.print(celsius);
    Serial.println(`C`);
    float faran=(celsius*9.0/5.0)+32.0;
    Serial.print(faran);
    Serial.println(`F`);
    if(celsius>37.0)
    {
        digitalWrite(buzzer,HIGH);
        tone(buzzer,500,1000);
        Serial.println("FIRE FIRE FIRE");
    }
    else
    {
        digitalWrite(buzzer,LOW);
        Serial.println("CHILL CHILL CHILL");
    }
    delay(100);
}
```

7. Tilt Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-tilt Simulator time: 00:01:34

Code Stop Simulation Export Share

Text

```
1 #define tilt 7
2 #define led 13
3
4
5
6 void setup()
7 {
8   pinMode(tilt, INPUT);
9   pinMode(led, OUTPUT);
10  Serial.begin(9600);
11
12 }
13
14 void loop()
15 {
16   if(digitalRead(tilt))
17   {
18     digitalWrite(led, LOW);
19   }
20   else
21   {
22     digitalWrite(led, HIGH);
23   }
24   delay(100);
25 }
```

Serial Monitor

```
1
1
1
1
1
1
1
```

Code

Shivam Vaish
1BM18CS152

```
# define tilt 17
# define led 13
void setup()
{
    pinMode (tilt, INPUT);
    pinMode (led, OUTPUT);
    Serial.begin (9600);
}

void loop ()
{
    if (digitalRead (tilt))
    {
        digitalWrite (led, LOW);
    }
    else
    {
        digitalWrite (led, HIGH);
    }
    delay (100);
}
```

```
#define tilt 7
#define led 13

void setup()
{
    pinMode(tilt, INPUT);
    pinMode(led, OUTPUT);
    Serial.begin(9600);

}

void loop()
{
    if(digitalRead(tilt))
    {
        digitalWrite(led, LOW);
    }
    else
    {
        digitalWrite(led, HIGH);
    }
    delay(100);
}
```

8. Ultrasonic Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-ultrasonic All changes saved Code Stop Simulation Export Share Simulator time: 00:00:40.405

Ultrasonic Distance Sensor
Name 2

```
void setup()
{
    Serial.begin(9600);
}
void loop()
{
    pinMode(triggerPin, OUTPUT);
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(5);
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    pinMode(echoPin, INPUT);
    long duration=pulseIn(echoPin, HIGH);
    cm = 0.01723 * duration;
    inches = (cm / 2.54);
    Serial.print(inches);
    Serial.println("in, ");
    Serial.print(cm);
    Serial.println("cm");
    delay(100); // Wait for 100 millisecond(s)
}
```

Serial Monitor
121in,
336cm
121in,
336cm
121in,
336cm
121in,
336cm
121in,
336cm

Type here to search 02:22 04-11-2020 ENG

CodeShivam Vaish
18M18CS152

```
#define triggerpin 2
#define echoPin 3
long inches = 0;
long cm = 0;
Void Setup ()
{
    Serial.begin(9600);
}
Void loop ()
{
    pinMode(triggerpin, OUTPUT);
    digitalWrite(triggerpin, LOW);
    delayMicroseconds(5);
    digitalWrite(triggerpin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerpin, LOW);
    pinMode(echoPin, INPUT);
    long duration = pulseIn(echoPin, HIGH);
    cm = 0.01723 * duration;
    inches = cm / 2.54;
    Serial.print(cm);
    Serial.println("cm");
    Serial.print(inches);
    Serial.println("inches");
    delay(100);
}
```

```
#define triggerPin 2
#define echoPin 3
long inches = 0;

long cm = 0;

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

}

void loop()
{
    pinMode(triggerPin, OUTPUT);
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(5);
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    pinMode(echoPin, INPUT);
    long duration=pulseIn(echoPin, HIGH);
    cm = 0.01723 * duration;
    inches = (cm / 2.54);
    Serial.print(inches);
    Serial.println("in, ");
    Serial.print(cm);
    Serial.println("cm");
    delay(100); // Wait for 100 millisecond(s)
}
```

9. Gas Leakage

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad?tenant=circuits

TIN KER CAD 1bm18cs152-weather Simulator time: 00:00:15 Saving... Code Stop Simulation Export Share

Gas Sensor
Name 1

The circuit diagram shows an Arduino Uno connected to a gas sensor module. The gas sensor module is mounted on a breadboard. A red wire connects pin 9 of the Arduino to the VCC pin of the breadboard. A green wire connects the GND pin of the Arduino to the GND rail of the breadboard. A blue wire connects the A0 pin of the Arduino to the analog input of the gas sensor module. A red wire also connects the digital output of the gas sensor module to pin 13 of the Arduino. A black wire connects the ground of the gas sensor module to the GND rail of the breadboard. A grey cloud icon representing smoke is positioned above the breadboard.

```
1 #define led 13
2 #define gas A0
3 int threshold=300;
4
5 void setup()
6 {
7   pinMode(led, OUTPUT);
8   Serial.begin(9600);
9 }
10
11 void loop()
12 {
13   int gasvalue=analogRead(gas);
14   Serial.println(gasvalue);
15   if(gasvalue>=threshold)
16   {
17     digitalWrite(led, HIGH);
18     Serial.println("Smoke Detected");
19   }
20   else
21   {
22   }
23 }
```

Serial Monitor

96
96
96
96
96
96
96
96

Type here to search 12:30 04-11-2020 ENG

Code

Shivam Vaish
18M18CS152

```
#define led 13
#define gas A0
int threshold = 300;
void setup()
{
    pinMode(led, OUTPUT);
    Serial.begin(9600);
}

void loop()
{
    int gasValue = analogRead(gas);
    Serial.println(gasValue);
    if (gasValue >= threshold)
    {
        digitalWrite(led, HIGH);
        Serial.println("Smoke detected");
    }
    else
    {
        digitalWrite(led, LOW);
    }
    delay(50);
}
```

```
#define led 13
#define gas A0
int threshold=300;

void setup()
{
    pinMode(led, OUTPUT);
    Serial.begin(9600);

}

void loop()
{
    int gasvalue=analogRead(gas);
    Serial.println(gasvalue);
    if(gasvalue>=threshold)
    {
        digitalWrite(led, HIGH);
        Serial.println("Smoke
Detected");
    }
    else
    {
        digitalWrite(led,LOW);
    }
    delay(50);

}
```

10. IR Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-ir-remote Simulator time: 00:00:20 Saving... Code Stop Simulation Export Share

IR remote
Name: 3

```
8  Irecv irrecv(irsensor);
10 Servo myservo;
11 decode_results results;
12 void setup()
13 {
14   myservo.attach(servol);
15   irrecv.enableIRIn();
16   Serial.begin(9600);
17 }
18 void loop()
19 {
20   if (irrecv.decode(&results))
21   {
22     switch(results.value)
23     {
24       case 0xFD20DF:
25         myservo.write(0);
26         break;
27       case 0xFD609F:
28         myservo.write(180);
29         break;
30       default:
31         Serial.println("Invalid Input");
32     }
33   }
34 }
```

Serial Monitor

Type here to search 1012 04-11-2020

Send Clear

Code

Shivam Vaish
1BM18CS152

```
#include <IRremote.h>
#include <Servo.h>

#define IRSensor 10
#define Servo 11

IRrecv irrecv(IRsensor);
Servo myservo;
decode_results results;
void setup()
{
    myservo.attach(Servo);
    irrecv.enableIRIn();
    Serial.begin(9600);
}

void loop()
{
    if (irrecv.decode(&results))
    {
        switch(results.value)
        {
            case 0xFD20DF:
                myservo.write(0);
                break;
            case 0xFD609F:
                myservo.write(180);
                break;
            default:
                Serial.println("Invalid Input");
        }
        irrecv.resume();
    }
}
```

```
#include <IRremote.h>
#include <Servo.h>

#define irsensor 10
#define servo1 11

IRrecv irrecv(irlsensor);
Servo myservo;
decode_results results;
void setup()
{
    myservo.attach(servo1);
    irrecv.enableIRIn();
    Serial.begin(9600);

}

void loop()
{
    if (irrecv.decode(&results)){
        switch(results.value)
        {
            case 0xFD20DF:
                myservo.write(0);
                break;
            case 0xFD609F:
                myservo.write(180);
                break;
            default:
                Serial.println("Invalid Input");
        }
        irrecv.resume();
    }
}
```

11. LDR Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-potentiometer All changes saved Code Stop Simulation Export Share Simulator time: 00:00:09.511

Photoresistor Name 4

```
#define led 12
int brightness = 0;

void setup() {
  Serial.begin(9600);
  pinMode(led,OUTPUT);
}

void loop() {
  brightness=analogRead(A0);
  if(brightness<400)
  {
    digitalWrite(led,HIGH);
  }
  else
  {
    digitalWrite(led,LOW);
  }
  Serial.print("Brightness: ");
  Serial.println(brightness);
}
```

How it works

1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Serial Monitor

Brightness: 6
Bri

Send Clear

Type here to search

1906 ENG 03-11-2020

Code

Shivam Vaish
18m10cs152

```
#define led 12
int brightness = 0;
void setup()
{
    Serial.begin(9600);
    pinMode(led, OUTPUT);
}
void loop()
{
    brightness = analogRead(A0);
    if (brightness < 400)
        digitalWrite(led, HIGH);
    else
        digitalWrite(led, LOW);
    Serial.print(" Brightness");
    Serial.println(brightness);
}
```

```
# dy
# dy
book
Void
{
}
Void
{
}
```

```
#define led 12
int brightness = 0;

void setup() {
    Serial.begin(9600);
    pinMode(led,OUTPUT);
}

void loop() {

    brightness= analogRead(A0);
    if(brightness<400)
    {
        digitalWrite(led,HIGH);
    }
    else
    {
        digitalWrite(led,LOW);
    }
    Serial.print("Brightness: ");
    Serial.println(brightness);

}
```

12. PIR Sensor

Circuit design Fantastic Turing-BI x + tinkercad.com/things/fvGzwCfEzP2-fantastic-turing-blad/editel?tenant=circuits

TIN KER CAD 1bm18cs152-pir All changes saved Code Stop Simulation Export Share 1 (Arduino Uno R3)

Simulator time: 00:02:47

PIR Sensor Name: 1

How it works

1. Add breakpoints by clicking on the line numbers.
2. Start simulation.
3. Hover over the variables while paused to see their value.

Text

```
8 //-----,
9 pinMode(sensor, INPUT);
10
11 void loop() {
12   if(digitalRead(sensor)==HIGH)
13   {
14     digitalWrite(led,HIGH);
15     if(!state)
16     {
17       Serial.println("motion detected");
18       state=true;
19     }
20   }
21   else
22   {
23     digitalWrite(led,LOW);
24     if(state)
25     {
26       Serial.println("motion stopped");
27       state=false;
28     }
29   }
30   delay(100);
31 }
32 }
```

Serial Monitor

```
motion detected
motion stopped
motion detected
motion stopped
motion detected
motion stopped
motion detected
motion stopped
```

Type here to search 0105 04-11-2020 ENG

Send Clear

The screenshot shows a Tinkercad circuit project titled "1bm18cs152-pir". The circuit diagram includes a PIR sensor module mounted on a breadboard, an Arduino Uno, and a breadboard power supply. The PIR sensor's output is connected to digital pin 2 of the Arduino. The Arduino is connected to a computer via USB. The code in the editor is written in C++ for the Arduino platform. It reads the state of the PIR sensor (pin 2) and toggles an LED (pin 13) based on whether motion is detected or stopped. The serial monitor shows the output of the code, which alternates between "motion detected" and "motion stopped" messages at regular intervals. The Tinkercad interface also shows various tools and options for editing the circuit and sharing it.

Code

1BM18CS152
Shivam Vaish

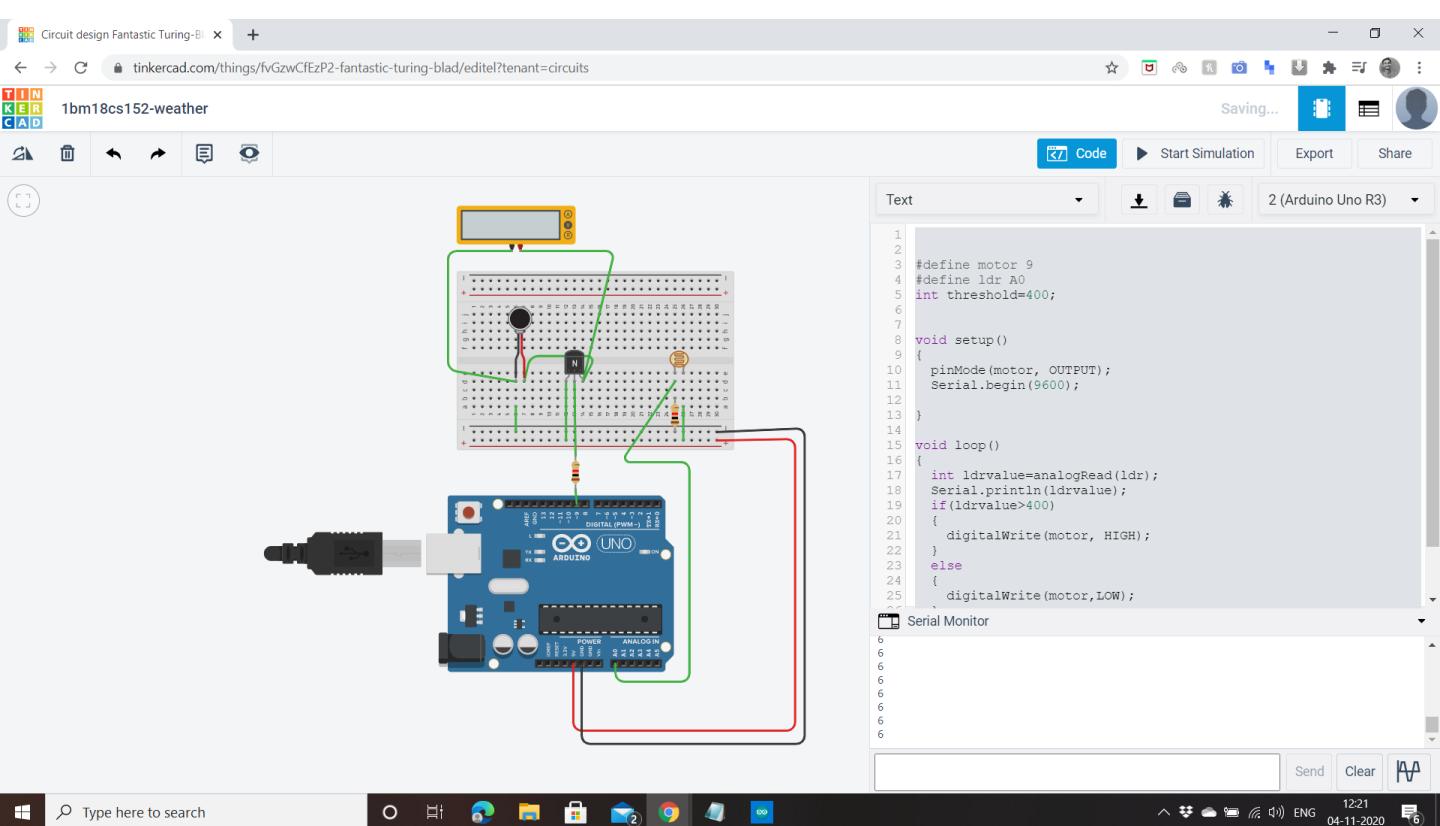
```
# define led 12
# define sensor 6
bool state = false;
void setup()
{
    Serial.begin(9600);
    pinMode(led, OUTPUT);
    pinMode(sensor, INPUT);
}
void loop()
{
    if (digitalRead(sensor) == HIGH)
    {
        digitalWrite(led, HIGH);
        if (!state)
        {
            Serial.println(" Motion detected");
            state = true;
        }
    }
    else
    {
        digitalWrite(led, LOW);
        if (state)
        {
            Serial.println(" motion stopped");
            state = false;
        }
    }
    delay(100);
}
```

```
#define led 12
#define sensor 6
bool state=false;

void setup() {
    Serial.begin(9600);
    pinMode(led,OUTPUT);
    pinMode(sensor, INPUT);
}

void loop() {
    if(digitalRead(sensor)==HIGH)
    {
        digitalWrite(led,HIGH);
        if(!state)
        {
            Serial.println("motion
detected");
            state=true;
        }
    }
    else
    {
        digitalWrite(led,LOW);
        if(state)
        {
            Serial.println("motion
stopped");
            state=false;
        }
    }
    delay(100);
}
```

13. Weather using Vibrator



Code

Shivam Vaish
1BM18CS152

```
#define Motor_9
#define ldr A0
int threshold = 400;
void setup()
{
    pinMode(motor, OUTPUT);
    Serial.begin(9600);
}
void loop()
{
    int ldrvalue = analogRead(ldr);
    Serial.println(ldrvalue);
    if (ldrvalue > 400)
    {
        digitalWrite(motor, HIGH);
    }
    else
    {
        digitalWrite(motor, LOW);
    }
    delay(100);
}
```

```
#define motor 9
#define ldr A0
int threshold=400;

void setup()
{
    pinMode(motor, OUTPUT);
    Serial.begin(9600);

}

void loop()
{
    int ldrvalue=analogRead(ldr);
    Serial.println(ldrvalue);
    if(ldrvalue>400)
    {
        digitalWrite(motor, HIGH);
    }
    else
    {
        digitalWrite(motor, LOW);
    }
    delay(50);

}
```

14. Humidity Sensor

The screenshot shows a Tinkercad project titled "1bm18cs152-humidity". The circuit diagram on the left features an Arduino Uno at the bottom. A blue servo motor is positioned above it. A green wire connects pin 9 on the Uno to the servo's control input. A red wire connects the Uno's 5V power rail to the servo's power input. A black wire connects the servo's ground terminal to the Uno's GND rail. On the right side, a blue BME280 humidity sensor is connected to the Uno. Its V_{DD} pin is connected to the Uno's 3.3V rail via a green wire. Its GND pin is connected to the Uno's GND rail via a black wire. Its SDA pin is connected to pin A4 on the Uno via a green wire. Its SCL pin is connected to pin A5 on the Uno via a red wire. The Uno's USB port is shown on the far left.

Code:

```
#include <Servo.h>
Servo myservo;
void setup()
{
  myservo.attach(6);
  Serial.begin(9600);
}
void loop()
{
  myservo.write(map(analogRead(A0),0,1023,0,180));
  delay(200);
}
```

Serial Monitor:

```
0
0
0
0
0
0
0
```

Code

Shivam Vaish
1BM18CS152

```
#include <Servo.h>
Servo myservo;
Void setup()
{
    myservo.attach(6);
    Serial.begin(9600);
}
Void loop()
{
    myservo.write(map(analogRead(A0), 0, 1023, 0, 180));
    delay(200);
}
```

```
#include <Servo.h>
Servo myservo;
void setup()
{
    myservo.attach(6);
    Serial.begin(9600);
}
void loop()
{

myservo.write(map(analogRead(A
0),0,1023,0,180));
    delay(200);
}
```

15. LCD Screen with RGB LEDs

The screenshot shows a Tinkercad project titled "1bm18cs152-lcd-display". The circuit diagram on the left features an Arduino Uno at the bottom, connected to a breadboard. On the breadboard, there is an LCD screen (16x2 characters) and three RGB LED modules. The connections include power rails, ground, and specific pins from the Arduino Uno mapped to the LCD and the RGB LEDs. The code editor on the right contains the following C++ code for an Arduino Uno R3:

```
#include <LiquidCrystal.h>
#define red 10
#define green 8
#define blue 9
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
    lcd.begin(16,2);
    lcd.setCursor(0,0);
    pinMode(red, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(blue, OUTPUT);
}
void loop()
{
    lighton("RED",red);
    lighton("GREEN",green);
    lighton("BLUE",blue);
}
void lighton(char light[],int pin)
{
    lcd.print(light);
    analogWrite(red,0);
    analogWrite(green,0);
    analogWrite(blue,0);
    analogWrite(pin,255);
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
}
```

The code uses the LiquidCrystal library to interface with the LCD and analogWrite to control the brightness of the RGB LEDs. The Arduino Uno is connected to a Windows 10 desktop computer at the bottom, which has a taskbar visible.

```
#include <LiquidCrystal.h>
#define red 10
#define green 8
#define blue 9

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

void setup()
{
    lcd.begin(16,2);
    lcd.setCursor(0,0);
    pinMode(red, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(blue, OUTPUT);
}

void loop()
{
    lightOn("RED", red);
    lightOn("GREEN", green);
    lightOn("BLUE", blue);
}

void lightOn(char light[], int pin)
{
    lcd.print(light);
    analogWrite(red,0);
    analogWrite(green,0);
    analogWrite(blue,0);
    analogWrite(pin,255);
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
}
```

```
#include <LiquidCrystal.h>
#define red 10
#define green 8
#define blue 9
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
    lcd.begin(16,2);
    lcd.setCursor(0,0);
    pinMode(red, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(blue, OUTPUT);
}
void loop()
{
    lighton("RED",red);
    lighton("GREEN",green);
    lighton("BLUE",blue);
}
void lighton(char light[],int pin)
{
    lcd.print(light);
    analogWrite(red,0);
    analogWrite(green,0);
    analogWrite(blue,0);
    analogWrite(pin,255);
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
}
```