

SEMESTER -4



INTERNET OF THINGS

NAME : JUTTHADA SAI KIRAN
REG.NO: 122010315047
SECTION: B15
ROLL NO: 47
EMAIL I'D: kjutthad@gitam.in

DATE:04-02-2022

EXPERMENT NO 2

AIM: .Read data from a sensor. Experiment with both analog and digital sensors.

OBJECTIVES: Student should get the knowledge of Temperature Sensor and IR sensor

OUTCOMES: Student will be developed programs using Arduino IDE and Arduino Board for Temperature Sensor and IR sensor.

CONNECTING TO A TEMPERATURE SENSOR:

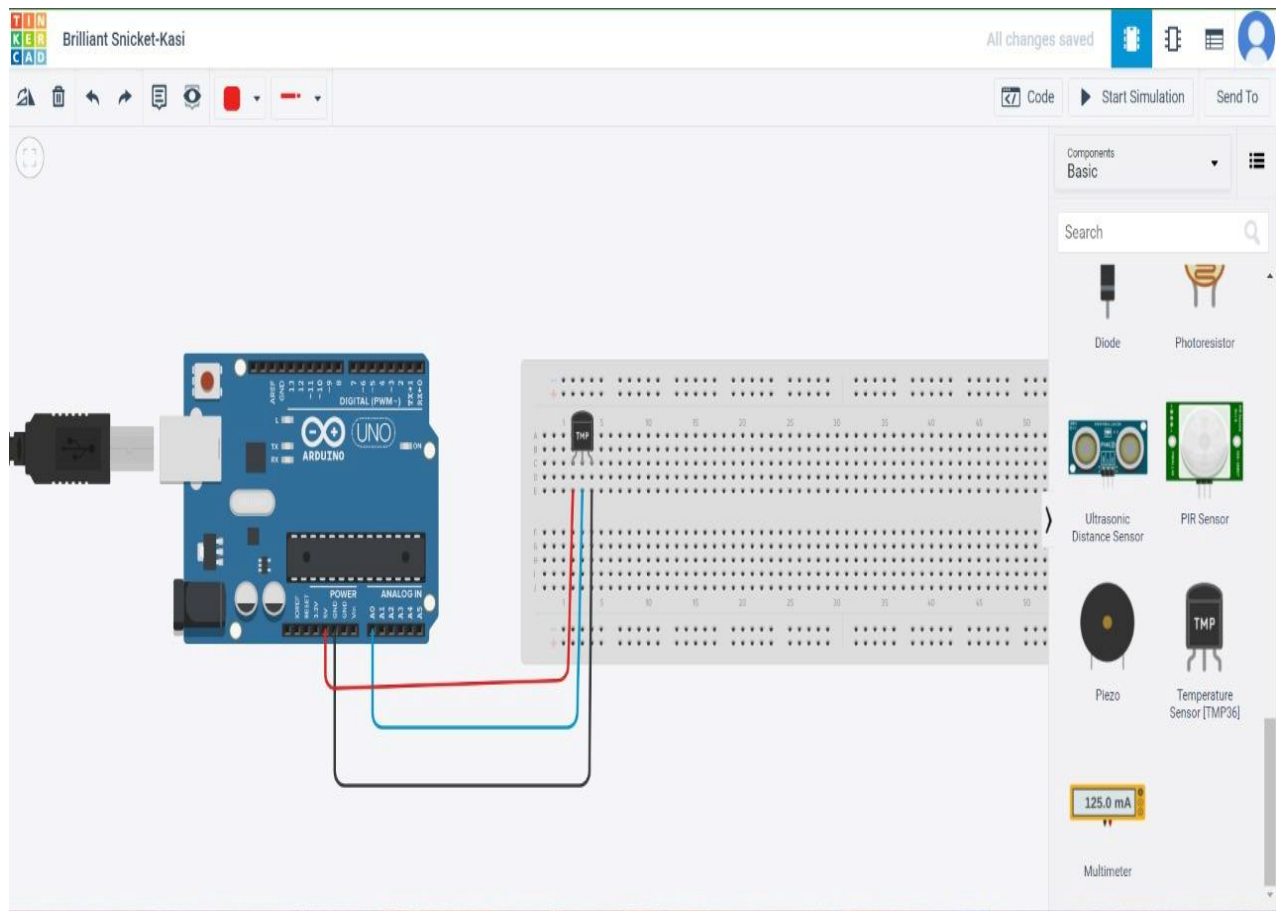
These sensors have little chips in them and while they're not that delicate, they do need to be handled properly.

Be careful of static electricity when handling them and make sure the power supply is connected up correctly and is between 2.7 and 5.5V DC - so don't try to use a 9V battery.

They come in a "TO-92" package which means the chip is housed in a plastic hemi-cylinder with three legs. The legs can be bent easily to allow the sensor to be plugged into a breadboard. You can also solder to the pins to connect long wires.

READING THE ANALOG TEMPPERATURE DATA:

The sensor has 3 pins 2 pins are used to power the sensor and third one is the analog output. To read the temperature value from the sensor, connect the output pin directly into an Analog (ADC) input of Arduino board.



Remember that you can use anywhere between 2.7V and 5.5V as the power supply. For this example, I'm showing it with a 5V supply but note that you can use this with a 3.3v supply just as easily. No matter what supply you use, the analog voltage reading will range from about 0V (ground) to about 1.75V.

If you're using a 5V Arduino, and connecting the sensor directly into an Analog pin, you can use these formulas to turn the 10-bit analog reading into a temperature:

*Voltage at pin in milliVolts = (reading from ADC) * (5000/1024)*

This formula converts the number 0-1023 from the ADC into 0-5000mV (= 5V) If you're using a 3.3V Arduino, you'll want to use this:

*Voltage at pin in milliVolts = (reading from ADC) * (3300/1024)*

This formula converts the number 0-1023 from the ADC into 0-3300mV (= 3.3V) Then, to convert millivolts into temperature, use this formula

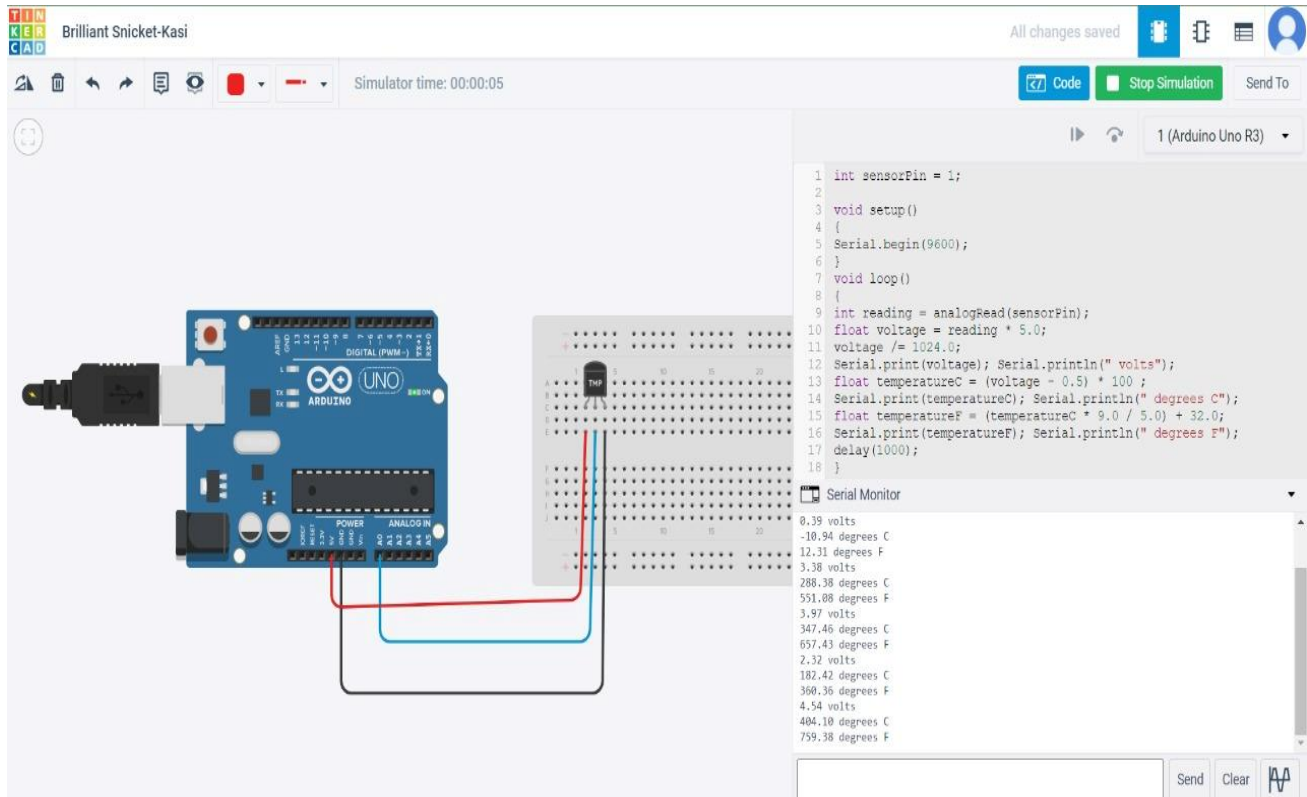
:

Centigrade temperature = [(analog voltage in mV) - 500] / 10

Simple Thermometer

This example code for Arduino shows a quick way to create a temperature sensor, it simply prints to the serial port what the current temperature is in both Celsius and Fahrenheit

WITH TEXT:



The screenshot displays the Arduino IDE interface. On the left, a virtual breadboard shows an Arduino Uno R3 connected to a temperature sensor module. The sensor's VCC pin is connected to the Arduino's 5V pin, GND to GND, and the signal pin to A0. The code editor on the right contains the following C++ code:

```
1 int sensorPin = 1;
2
3 void setup()
4 {
5   Serial.begin(9600);
6 }
7 void loop()
8 {
9   int reading = analogRead(sensorPin);
10  float voltage = reading * 5.0;
11  voltage /= 1024.0;
12  Serial.print(voltage); Serial.println(" volts");
13  float temperatureC = (voltage - 0.5) * 100 ;
14  Serial.print(temperatureC); Serial.println(" degrees C");
15  float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;
16  Serial.print(temperatureF); Serial.println(" degrees F");
17  delay(1000);
18 }
```

Below the code, the Serial Monitor window shows the output of the program:

```
0.39 volts
-10.94 degrees C
12.31 degrees F
3.38 volts
288.38 degrees C
551.08 degrees F
3.97 volts
347.46 degrees C
657.43 degrees F
2.32 volts
182.42 degrees C
350.36 degrees F
4.54 volts
404.10 degrees C
759.38 degrees F
```

```
1 int sensorPin = 1;
2
3 void setup()
4 {
5   Serial.begin(9600);
6 }
7 void loop()
8 {
9   int reading = analogRead(sensorPin);
10  float voltage = reading * 5.0;
11  voltage /= 1024.0;
12  Serial.print(voltage); Serial.println(" volts");
13  float temperatureC = (voltage - 0.5) * 100 ;
14  Serial.print(temperatureC); Serial.println(" degrees C");
15  float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;
16  Serial.print(temperatureF); Serial.println(" degrees F");
17  delay(1000);
18 }
```

WITH BLOCKS:

1. Connect the circuit as shown below

2. Create a variable name reading and acquire the analog value of A0 into it. observe the output in serial monitor. arrange the code blocks as shown and start simulation. you will see the temperature in serial monitor. vary the input and observe the output.

The screenshot shows the TinkerCAD simulation environment. On the left, an Arduino Uno is connected to a TMP36 temperature sensor via a breadboard. The sensor's VCC pin is connected to the 5V pin of the Arduino, GND to GND, and the output pin to analog pin A0. The code blocks on the right are as follows:

```
set reader to read temperature sensor on pin A0 in units °C
print to serial monitor reader with newline
```

The serial monitor at the bottom shows a graph of the output, which fluctuates between approximately 45 and 180. The graph has a y-axis ranging from -90 to 180. The x-axis is labeled with 58.