+Sl.no. 1

Name:Active Buzzer sensor

Principle:generates sound at only one frequency. This buzzer operates at an audible frequency of 2 KHz.

Type of protocol:SERIAL DIGITAL & ANALOG

PIN Connection:ALL pins

Type of Sensor: Active

Codes:

void setup(){

pinMode(8,OUTPUT);

}

void loop(){

digitalWrite(8,HIGH);

delay(500);

digitalWrite(8,LOW);

delay(500);

}

Ratings:



Sl.no. 2

Name:led

Principle: a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. The color of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gapof the semiconductor.

Type of protocol:SERIAL DIGITAL & ANALOG

PIN Connection:ALL pins

Type of Sensor:active

Codes:

void setup(){

pinMode(8,OUTPUT);

}The Grove - Light sensor integrates a photo-resistor(light dependent resistor) to detect the intensity of light. The resistance of photo-resistor decreases when the intensity of light increases. A dual OpAmp chip LM358 on board produces voltage corresponding to intensity of light(i.e. based on resistance value). The output signal is analog value, the brighter the light is, the larger the value.

This module can be used to build a light controlled switch i.e. switch off lights during day time and switch on lights during night time.

void loop(){

digitalWrite(8,HIGH);

delay(500);

digitalWrite(8,LOW);

delay(500);

}

Ratings:

Operating Voltage 5V

Sl.no. 3

Name: push button

Principle:switching action

Type of protocol:SERIAL DIGITAL

PIN Connection:ALL pins

Type of Sensor:passive

Codes:

void setup(){

Serial.begin(9600);

pinMode(8,INPUT);

}

void loop(){

int state;

state = digitalRead(8);

delay(500);

if(state == HIGH)

Serial.print("switch on");

else

Serial.print("switch off");

Serial.print("\n");

}

Ratings:



Sl.no. 4

Name: sound sensor

Principle:it uses the vibrations and changes into current otherwise voltage

Type of protocol:SERIAL ANALOG

PIN Connection:Analog pins

Type of Sensor:passive

Codes:

const int pinAdc = A0;

void setup()

{

Serial.begin(115200);

//Serial.println("Grove - Sound Sensor Test...");

}

void loop()

{

long sum = 0;

for(int i=0; i<32; i++)

{

sum += analogRead(pinAdc);

}

sum >>= 5;

Serial.println(sum);

delay(1000);

}

Ratings:

OLE object

Sl.no. 5

Name: Capacitive Touch Sensor

Principle: working principle of capacitors

Type of protocol:SERIAL DIGITAL

PIN Connection:ALL Digital Pins

Type of Sensor:passive

Codes:

const int TouchPin=2;

const int ledPin=3;

void setup() {

pinMode(ledPin,OUTPUT);

Serial.begin(9600);

}

void loop() {

int sensorValue = digitalRead(TouchPin);

if(sensorValue==1)

{

Serial.print("switch is on");

}

else

{

Serial.print("switch is off");

}

Serial.print("\n");

delay(50);

}

Ratings:

Operating Voltage: 2.0 - 5.5V

* Operating Current(Vcc=3V):1.5 - 3.0μA
* Operating Current(VDD=3V):3.5 - 7.0μA
* Output Response Time: 60 – 220mS

Sl. no. 6

Name: Light Sensor

Principle:The Light sensor integrates a photo-resistor(light dependent resistor) to detect the intensity of light. The resistance of photo-resistor decreases when the intensity of light increases. A dual OpAmp chip LM358 on board produces voltage corresponding to intensity of light(i.e. based on resistance value). The output signal is analog value, the brighter the light is, the larger the value.

This module can be used to build a light controlled switch i.e. switch off lights during day time and switch on lights during night time.

Type of protocol:Serial analog

PIN Connection:All analog pins

Type of Sensor: Passive

Codes:

void setup(){

Serial.begin(9600);

pinMode(A0,INPUT);

}

void loop(){

int value = analogRead(A0);

Serial.print("\nnThe value = ");

Serial.print(value);

delay(500);

}

Ratings:

| Itemiteiiii | Value |
| --- | --- |
| Operating voltage | 3~5V |
| Operating current | 0.5~3 mA |
| Response time | 20-30 milliseconds |
| Peak Wavelength | 540 nm |
| Weight | 4 g |

Sl.no. 7

Name: 3-Axis digital accelerometer (+-1.5 g)

Principle:

Measures the proper accelerations oaf a device.

an accelerometer behaves as a damped mass on a spring. When the accelerometer experiences an acceleration, the mass is displaced to the point that the spring is able to accelerate the mass at the same rate as the casing. The displacement is then measured to give the acceleration.

Type of protocol:Serial

PIN Connection: I2C Pins

Type of Sensor:Active

Codes:

#include <Wire.h> //for I2C pin interface

#include "MMA7660.h" // for accelerometer

MMA7660 acc; // declare a variable

void setup()

{

acc.init();

}

void loop()

{

int8\_t x; //inbuilt variables

int8\_t y;

int8\_t z;

float ax,ay,az;

acc.getXYZ(&x,&y,&z);

acc.getAcceleration(&ax,&ay,&az);

delay(500);

}

Ratings:

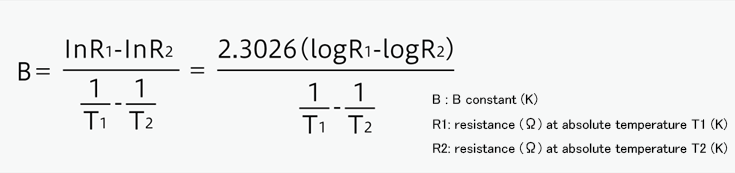
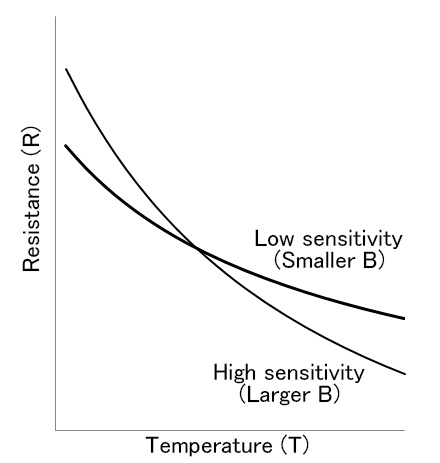
Working voltage: 3.0 - 5.5V

* Off Mode Current: 0.4μA
* Standby Mode Current: 2μA
* Active Mode Current: 47 μA at 1 ODR
* Test Range: ±1.5g
* Sensitivity: 21LSB/g
* Suli-compatible Library

Sl.no.8

Name: Temperature sensor

Principle:The Grove - Temperature Sensor uses a [Thermistor](http://www.legacydistribution.co.uk/downloads/NTC-thermistors/TTC03.pdf) to detect the ambient temperature. The resistance of a thermistor will increase when the ambient temperature decreases. It's this characteristic that we use to calculate the ambient temperature. The detectable range of this sensor is -40 - 125ºC, and the accuracy is ±1.5ºC.

Type of protocol:Serial

pin connection:Analog Pins

Type of sensor:Active

codes:

#include <math.h> //for log

int a;

float temperature;

int B=3975; //B value of the thermistor

float resistance;

void setup()

{

}

void loop()

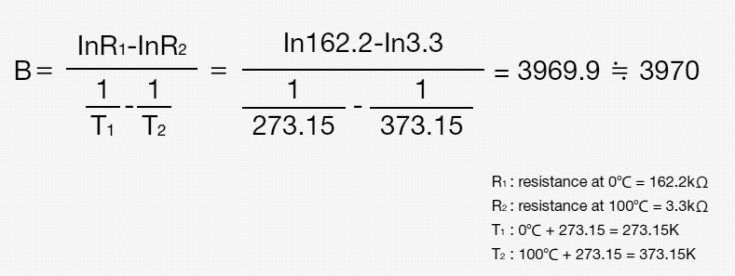
{

a=analogRead(0);

resistance=(float)(1023-a)\*10000/a; //get the resistance of the sensor;

temperature=1/(log(resistance/10000)/B+1/298.15)-273.15; // to K → C

delay(1000);

}

Ratings:

Voltage: 3.3 ~ 5V

* Max power rating at 25℃: 300mW
* Zero power resistance: 10 KΩ
* Operating temperature range: -40 ~ +125 ℃

Sl.no.9

Name: Rotary Sensor

Principle:

A rotary encoder, also called a shaft encoder, is an [electro-mechanical](https://en.wikipedia.org/wiki/Electro-mechanical) device that converts the [angular](https://en.wikipedia.org/wiki/Angle) position or motion of a shaft or axle to analog or digital output signals[[1]](https://en.wikipedia.org/wiki/Rotary_encoder" \l "cite_note-1).

Type of protocol:SERIAL ANALOG

PIN Connection: Analog pins

Type of Sensor: Passive

Codes:

#define ROTARY\_ANGLE\_SENSOR A0

#define LED 3 //the Grove - LED is connected to PWM pin D3 of Arduino

#define ADC\_REF 8 //reference voltage of ADC is 5v.If the Vcc switch on the seeeduino

//board switches to 3V3, the ADC\_REF should be 3.3

#define GROVE\_VCC 5 //VCC of the grove interface is normally 5v

#define FULL\_ANGLE 300 //full value of the rotary angle is 300 degrees

void setup()

{

Serial.begin(9600);

pinMode(ROTARY\_ANGLE\_SENSOR, INPUT);

pinMode(LED,OUTPUT);

}

void loop()

{

float voltage;

int sensor\_value = analogRead(ROTARY\_ANGLE\_SENSOR);

voltage = (float)sensor\_value\*ADC\_REF/1023;

float degrees = (voltage\*FULL\_ANGLE)/GROVE\_VCC;

Serial.println("The angle between the mark and the starting position:");

Serial.println(degrees);

int brightness;

brightness = map(degrees, 0, FULL\_ANGLE, 0, 255);

analogWrite(LED,brightness);

delay(500);

}

Ratings:

Voltage

Min:4.75

Max5.25:

Typical:5.0

Rotary Angle

Min:0

Max:300

Sl.no.10

Name: Piezo Vibration Sensor

Principle:

Similar to an accelerometer with a wide range of frequency receptors.

Type of protocol:SERIAL ANALOG

PIN Connection: All Digital and Analog Pins

Type of Sensor: Passive

Codes:

const int ledPin=3;

void setup() {

Serial.begin(9600);

pinMode(ledPin,OUTPUT);

}

void loop() {

int sensorState = digitalRead(2);

Serial.println(sensorState);

delay(100);

if(sensorState == HIGH)

{

digitalWrite(ledPin,HIGH);

}

else

{

digitalWrite(ledPin,LOW);

}

delay(1000);

}

Ratings:

Wide dynamic range：0.1Hz~180Hz

* Adjustable sensitivity
* High receptivity for strong impact

Sl.no. 11

Name: RGB\_LCD

Principle:

Type of protocol:SERIAL ANALOG

PIN Connection: I2c pins

Type of Sensor: passive and active

Codes:

/\*

##this is the compilation of all the codes in the RGB\_LCD in a single file

\*/

#include <rgb\_lcd.h> //for I2C PINS

#include <Wire.h> // LIBRARY FOR RGB\_LCD

rgb\_lcd lcd; //DECLARE A VARIABLE

byte smiley[8] = { // any character can be created by just switching between the bits in the suitable order

0b00000,

0b00000,

0b01010,

0b00000,

0b00000,

0b10001,

0b01110,

0b00000

};

byte armsDown[8] = {

0b00100,

0b01010,

0b00100,

0b00100,

0b01110,

0b10101,

0b00100,

0b01010

};

byte armsUp[8] = {

0b00100,

0b01010,

0b00100,

0b10101,

0b01110,

0b00100,

0b00100,

0b01010

};;

void setup(){

lcd.begin(16, 2);

#if 1

// create a new character

cd.createChar(1, smiley);

;

// create a new character

lcd.createChar(3, armsDown);

// create a new character

lcd.createChar(4, armsUp);

#endif

// set up the lcd's number of columns and rows:

lcd.write(0);

}

void loop(){

cd.setCursor(0,0); // set the cursor:

// read the potentiometer on A0:

int sensorReading = analogRead(A0);

// map the result to 200 - 1000:

int delayTime = map(sensorReading, 0, 1023, 20, 1000);

// set the cursor to the bottom row, 5th position:

lcd.setCursor(4, 1);

// draw the little man, arms down:

lcd.write(3);

delay(delayTime);

lcd.setCursor(4, 1);

// draw him arms up:

lcd.write(4);

delay(delayTime);

lcd.println("hello everyon"); //to print on RGB

lcd.autoscroll(); //to start autoscroll

delay(1000);

lcd.noAutoscroll(); //to stop autoscroll

delay(3000);

lcd.noBlink(); //not ot blink

delay(500);

lcd.blink(); //to blink

lcd.noCursor(); // Turn off the cursor:

delay(500);

lcd.cursor(); // Turn on the cursor:

lcd.clear(); // to clear the screen

// Turn off the display:

lcd.noDisplay();

delay(500);

// Turn on the display:

lcd.display();

delay(500);

//set RGB color

lcd.setRGB(r,g,b); //set the RGB values

lcd.setPWM(REG\_COLOR,#VALUE); //SET THE BACKGROUND COLOR OF RGB

//prints the mesage in hte serial monitor

if(Serial.available()){

lcd.setCursor(5,0); //set the cursor

lcd.clear();

delay(100);

while(Serial.available() > 0){

lcd.write(Serial.read());

lcd.rightToLeft(); //the next element is placed right to left

lcd.leftToRight(); //the next element is placed left to right

lcd.home(); // set cursor to (0,,0)

}

Ratings:

Sl.no.12

Name: UltraSonic\_Range Finder

Principle: Works on the principle of Sonars as of the oscillating wave is send for a distant object and strikes back to the receiving terminal such that the time delay for its return is used to calculate the distance it travels

Type of protocol:SERIAL ANALOG

PIN Connection: A0

Type of Sensor: passive Sensor

Codes:

#include <rgb\_lcd.h>

rgb\_lcd lcd;

#include <Ultrasonic.h>

Ultrasonic ultra1(7);

const int LED1 13;

const int LED2 12;

const int LED3 11;

const int LED4 10;

void setup() {

lcd.begin(16,2);

Serial.begin(9600);

pinMode(LED1,OUTPUT);

pinMode(LED2,OUTPUT);

pinMode(LED3,OUTPUT);

pinMode(LED4,OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

long ultc1;

ultc1 = ultra1.MeasureInCentimeters();

digitalWrite(LED1,LOW);

digitalWrite(LED2,LOW);

digitalWrite(LED3,LOW);

digitalWrite(LED4,LOW);

if(ultc1 < 85){

ultc1=ultc1+2;//from 0-84

digitalWrite(LED1,HIGH);

}

else if(ultc1 < 140){

ultc1=ultc1+3;//from 85-139

digitalWrite(LED2,HIGH);

}

else if(ultc1 < 195){

ultc1=ultc1+4;//from 140-194

digitalWrite(LED3,HIGH);

}

else if (ultc1 <=330){

ultc1=ultc1+2;//from 195-330

digitalWrite(LED4,HIGH);

}

else{

ultc1=0;

digitalWrite(LED1,HIGH);

digitalWrite(LED2,HIGH);

digitalWrite(LED3,HIGH);

digitalWrite(LED4,HIGH);

}

Serial.print("Distance:");

Serial.println(ultc1);

Serial.print("Cms");

lcd.setCursor(0,0);

lcd.print("Cur dist = ");

lcd.print(ultc1);

lcd.print("cms");

delay(200);

lcd.clear();

}

Ratings:

Sl.no.

Name:

Principle:

Type of protocol:SERIAL ANALOG

PIN Connection:

Type of Sensor:

Codes:

Ratings: