#define SensorPin A0                     //pH meter Analog output to Arduino Analog Input 0

**#define Offset 0.00**                      **//Calibration Value Sensor Value - pH** Value Neutral (7) deviation compensate for sensor

unsigned long int avgValue;              //Store the average value of the sensor feedback void

int arduinoled =13;                     //Arduino Onboard LED

int redPin = 11;                        //Sets Arduino Pin to LED Module Color Red

int greenPin = 10;                      //Sets Arduino Pin to LED Module Color Green

int bluePin = 9;                        //Sets Arduino Pin to LED Module Color Blue

void setup()

{

pinMode(redPin, OUTPUT);                //Sets Arduino pin 11 (Red) to only sending Signals out to the LED Module

pinMode(greenPin, OUTPUT);              //Sets Arduino pin 10 (Green) to only sending Signals out to the LED Module

pinMode(bluePin, OUTPUT);               //Sets Arduino pin 9 (Blue) to only sending Signals out to the LED Module

pinMode(arduinoled ,OUTPUT);            //Sets Arduino pin 13 (Ardiuno Onboard LED) to only sending Signals

Serial.begin(9600);                     //Connects Arduino Data Output to Serial Monitor in the Arduino IDE

Serial.println("Ready");                //Test the serial monitor

}

void loop()

{

//Get 10 sample value from the sensor for smooth the value

int buf[10];                            //buffer for read analog

for(int i=0;i<10;i++)

{

buf[i]=analogRead(SensorPin);           //Analog Read takes in 10 readings and assigns to a 1 Dimensional Array

delay(10);                              //10 millisecond Delay to allow sensor processing time

}

for(int i=0;i<9;i++)                   //Sort Sensor Readings from small to large to find average

{

for(int j=i+1;j<10;j++)

{

if(buf[i]>buf[j])

{

int temp=buf[i];

buf[i]=buf[j];

buf[j]=temp;

}}}

avgValue=0;

for(int i=2;i<8;i++)                      //take the average value of 6 center sample

avgValue+=buf[i];

float phValue=(float)avgValue\*5.0/1024/6; //convert the analog into millivolt

phValue=3.5\*phValue+Offset;               //convert the millivolt into pH

//Outputs pH Sensor Average Value

Serial.print("pH Value: ");

Serial.print(phValue,2);

Serial.println();

//Turns on LED on Arduino when a average value is calculated

digitalWrite(arduinoled ,HIGH);

delay(800);

digitalWrite(arduinoled , LOW);

//Neutral = LED Color Green

if (phValue>=6.8 && phValue<=7.2)       // Range for sensor error

{

setColor (0,255,0);                     //3 numbers represent RGB (Red, Gree, Blue) scale 0-255; Values passed to Function Setcolor turning on the Red

}

//Bases = LED Color Yellow

else if (phValue < 7)

{

setColor (255, 50, 0);                //3 numbers represent RGB (Red, Gree, Blue) scale 0-255; Values passed to Function Setcolor turning on the Red

}

// Acids = LED Color Red

else

{

setColor (255, 0,0);                  //3 numbers represent RGB (Red, Green, Blue) scale 0-255; Values passed to Function Setcolor turning on the Red

}

}

void setColor(int red, int green, int blue)       //Function to assign LED Color Range 0-255 on the Red, Green, Blue Color Spectrum

{

analogWrite(redPin, red);

analogWrite(greenPin, green);

analogWrite(bluePin, blue);

}