جميع الاكواد البرمجية لجميع المتحسسات

المعروضة ضمن الفديو

الدرس 1

```
#define dht_apin A0 // Analog Pin sensor is connected to

dht DHT;

 void setup()

 }

Serial.begin(9600 ;(

delay(500);//Delay to let system boot Serial.println("DHT11 Humidity & temperature Sensor\n\n"); delay(1000);//Wait before accessing Sensor }//end "setup"()

 void loop()

// }Start of Program

 DHT.read11(dht_apin;(

 Serial.print("Current humidity = "); Serial.print(DHT.humidity;(

 Serial.print;(" %")

 Serial.print("temperature = "); Serial.print(DHT.temperature(

 Serial.println("C;("

 delay(5000);//Wait 5 seconds before accessing sensor again. //Fastest should be once every two seconds.

 //{ end loop
```

2

```
int Led=13;//define LED interface

 int Shock=3//define vibration sensor interface int val;//define digital
varible val void setup()

 {

pinMode(Led,OUTPUT);//define LED as output
pinMode(Shock,INPUT);//define shock as input } void loop()

 {

val=digitalRead(Shock);//

if(val==HIGH)//

 {

digitalWrite(Led,LOW);

 }

else

{

digitalWrite(Led,HIGH);

 }

}
```

الدرس 3

```
LED int SENSOR = 3 ; // define the Hall sensor int val ; // define numeric
variables   void setup () {

Serial.begin(9600);

 pinMode (Led, OUTPUT) ; // define LED as output pinMode (SENSOR,
INPUT) ; // define the Hall magnetic sensor line as input
```

```
}  void loop ()
 {
val = digitalRead (SENSOR) ; // read sensor line
 if (val == LOW) // when the Hall sensor detects a magnetic field, Arduino LED lights up
 {
digitalWrite (Led, HIGH);
 Serial.println("Magnetic field detected");
}
Else
 {
digitalWrite (Led, LOW);
 Serial.println("No magnetic field detected");
}
delay(1000);
}
```

الدرس 4

```
int inPin = 2;
 outPin = 13;
 int state = HIGH;
 int reading;
 int previous = LOW;
 long time = 0;
```

```
 long debounce = 200;
void setup()
{
pinMode(inPin, INPUT);
pinMode(outPin, OUTPUT);
}
 void loop() {
reading = digitalRead(inPin);
if (reading == HIGH && previous == LOW && millis() - time > debounce)
{
if (state == HIGH) state = LOW; else state = HIGH; time = millis();
}
digitalWrite(outPin, state);
previous = reading;
 }
```

الدرس 5

```
#include <IRremote.h>
IRsend irsend;
 void setup()
{
 Serial.begin(9600);
}
```

```
void loop()
{
  for (int i = 0; i < 50; i++) {
    irsend.sendSony(0xa90, 12); // Sony TV power code
     delay(40);
  }
}
```

الدرس 6

```
int buzzer = 8; // set the buzzer control digital IO
void setup() {
pinMode(buzzer, OUTPUT); // set pin 8 as output
}
void loop()
{       for (int i = 0; i < 80; i++) {  // make a sound

digitalWrite(buzzer, HIGH);
 // send high signal to buzzer              delay(1); // delay 1ms
        digitalWrite(buzzer, LOW); // send low signal to buzzer  delay(1);
}       delay(50);
for (int j = 0; j < 100; j++)
 {
```

```
 //make another sound
        digitalWrite(buzzer, HIGH);
        delay(2); // delay 2ms
        digitalWrite(buzzer, LOW);
delay(2);}
delay(100);}
```

7

```
int laserPin = 13;


void setup() {
        pinMode(laserPin, OUTPUT);  // Define the digital output
interface pin 13
}


void loop() {
        digitalWrite(laserPin, HIGH); // Open the laser head
        delay(1000); // Delay one second
        digitalWrite(laserPin, LOW); // Close the laser head
        delay(1000);
}
```

8

```
int redpin = 11; //select the pin for the red LED
```

```
int bluepin =10; // select the pin for the  blue LED
int greenpin = 9;// select the pin for the green LED
int val;
void setup() {
  pinMode(redpin, OUTPUT);
  pinMode(bluepin, OUTPUT);
  pinMode(greenpin, OUTPUT);
  Serial.begin(9600);
}
void loop()
{
  for(val = 255; val > 0; val--)
  {
    analogWrite(redpin, val);  //set PWM value for red
    analogWrite(bluepin, 255 - val); //set PWM value for blue
    analogWrite(greenpin, 128 - val); //set PWM value for green
    Serial.println(val); //print current value
    delay(1);
  }
for(val = 0; val < 255; val++)
  {
    analogWrite(redpin, val);
    analogWrite(bluepin, 255 - val);
```

```
   analogWrite(greenpin, 128 - val);
   Serial.println(val);
   delay(1);
  }
}
```

9

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
 void setup()
{
  Serial.begin(9600);
  Serial.println("Dallas Temperature IC Library Demo");
  sensors.begin();
}
 void loop()
{ sensors.requestTemperatures() to issue a global temperature
  // request to all devices on the bus
  Serial.print("Requesting temperatures...");
  sensors.requestTemperatures(); // Send the command to get
temperatures
```

```
  Serial.println("DONE");
   Serial.print("Temperature for Device 1 is: ");
  Serial.print(sensors.getTempCByIndex(0));
}
```

10

```
int Led = 13; // define LED pin
int buttonpin = 3; // define photo interrupter signal pin
int val; //define a numeric variable

void setup()
{
        pinMode(Led, OUTPUT); // LED pin as output
        pinMode(buttonpin, INPUT); //photo interrupter pin as input
}

void loop()
{
        val=digitalRead(buttonpin); //read the value of the sensor
        if(val == HIGH) // turn on LED when sensor is blocked
        {
                digitalWrite(Led,HIGH);
        }
```

```
        else
        {
                digitalWrite(Led,LOW);
        }
}
```

11

```
int BlueLed = 9;
int KwikSwitch = 8;
int Relay = 7;
int val;
void setup()
{   pinMode (BlueLed, OUTPUT);
pinMode (Relay, OUTPUT);
pinMode (KwikSwitch, INPUT);}
void loop()   {     val = digitalRead (KwikSwitch) ;
if (val == HIGH)  {
 digitalWrite (BlueLed, HIGH);
digitalWrite (Relay, HIGH
else  {
  digitalWrite (BlueLed, LOW);
  digitalWrite (Relay, LOW);
```

```
}
}
```

12

```
int Led = 13 ;// define LED Interface
 int  tilepin = 7; // define the tilt switch sensor interfaces
 int val ;// define numeric variables val
void setup ()
 {
pinMode (Led, OUTPUT) ;// define LED as output interface
pinMode (tilepin, INPUT) ;//define the output interface tilt switch sensor
}
void loop ()
 {
val = digitalRead (tilepin) ;// digital interface will be assigned a value
if (val == HIGH) //When the tilt sensor detects a signal when the switch, LED lashes {
digitalWrite (Led, HIGH);                    }
else { digitalWrite (Led, LOW);                    }
}
```

13

```
int Led_pin = 13 ;
```

```
int Sensor_pin = 2 ;
int Value ;
void setup ( )
{      pinMode ( Led_pin, OUTPUT ) ;
  pinMode ( Sensor_pin, INPUT ) ;
}
void loop ( )
 {  Value = digitalRead ( Sensor_pin ) ;
  if (Value == HIGH )     {
    digitalWrite ( Led_pin, HIGH ) ;  }
  else    {    digitalWrite ( Led_pin, LOW ) ;  }  }
```

```
 int Sensor_pin = A0 ;
int Led_pin = 13 ;
int Value = 0 ;
void setup ( ) {
  pinMode (ed_pin, OUTPUT ) ;
  Serial.begin ( 9600 ) ;
}
void loop ( ) {    Value = analogRead ( Sensor_pin ) ;
  digitalWrite ( Led_pin, HIGH ) ;
```

```
 delay (Value ) ;
 digitalWrite ( Led_pin, LOW ) ;
 delay ( Value ) ;
 Serial.print( "SENSOR PIN A0:" )
 Serial.println ( Value, DEC ) ;                    }
```

14

```
int redpin = 11; // pin for red signal
int greenpin = 10; // pin for green signal
int val;
void setup() {
        pinMode(redpin, OUTPUT);
        pinMode(greenpin, OUTPUT);     }
void loop() {
        for(val = 255; val > 0; val--) {     analogWrite(redpin, val);
                analogWrite(greenpin, 255 - val);      delay(15);   }
        for(val = 0; val < 255; val++) {
                analogWrite(redpin, val); //brighten red
                analogWrite(greenpin, 255 - val); //dim green
                delay(15);}         }
```

15

```
int knockPin = 10;
```

```
int knockVal = HIGH;
boolean bAlarm = false;
unsigned long lastKnockTime;
int knockAlarmTime = 500;
void setup ()
{   Serial.begin(9600);   pinMode (knockPin, INPUT) ; }
void loop ()
{   knockVal = digitalRead (knockPin) ;
   if (knockVal == LOW)   {   lastKnockTime = millis();
 if (!bAlarm){ Serial.println("KNOCK, KNOCK");
    bAlarm = true;    }
 }    else   {  if( (millis()-lastKnockTime) > knockAlarmTime  &&
bAlarm)   {  Serial.println("no knocks"); bAlarm = false;  }    }   }
```

16

```
int Led_pin = 13 ;
int Sensor_pin = 2 ;
int Value ;
void setup ( )
{    pinMode ( Led_pin, OUTPUT ) ;
 pinMode ( Sensor_pin, INPUT ) ;
void loop ( ) {
 Value = digitalRead ( Sensor_pin ) ;
 if (Value == HIGH )     {
```

```
   digitalWrite ( Led_pin, HIGH ) ; }
  else  {
   digitalWrite ( Led_pin, LOW ) ;      }
}
```

17

```
int Led = 13 ; // define LED Interface
int buttonpin = 3; // define Metal Touch Sensor Interface
int val ;
void setup ()
{
  pinMode (Led, OUTPUT) ;
 pinMode (buttonpin, INPUT) ; }
void loop ()
{  val = digitalRead (buttonpin) ;
  if (val == HIGH)   {   digitalWrite (Led, HIGH);
  }  else  {   digitalWrite (Led, LOW);

 }
}
```

18

```
int Led = 13 ; // define LED Interface
```

```
int buttonpin = 3; // define Metal Touch Sensor Interface
int val ;
void setup ()
{
 pinMode (Led, OUTPUT) ;
 pinMode (buttonpin, INPUT) ; }
void loop ()
{  val = digitalRead (buttonpin) ;
 if (val == HIGH)   {    digitalWrite (Led, HIGH);
 } else {    digitalWrite (Led, LOW);


 }
}
```

19

```
int sensorPin = A5;
double Thermistor(int RawADC) {
 double Temp;
 Temp = log(10000.0*((1024.0/RawADC-1)));
 Temp = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 * Temp
* Temp ))* Temp );
 Temp = Temp - 273.15;
  //Temp = (Temp * 9.0)/ 5.0 + 32.0; // Convert Celcius to Fahrenheit
  return Temp;
 }
```

```
void setup() {
Serial.begin(9600);    }
                    void loop() {
int readVal=analogRead(sensorPin);
 double temp =  Thermistor(readVal);
 Serial.println(temp);  // display tempature
 Serial.println(readVal);  // display tempature
  delay(500);    }
```

20

```
int sensorPin = A5; // select the input pin for the potentiometer
int ledPin = 13; // select the pin for the LED
int sensorValue = 0; // variable to store the value coming from the
sensor
void setup() {
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  sensorValue = analogRead(sensorPin);
  digitalWrite(ledPin, HIGH);
  delay(sensorValue);
  digitalWrite(ledPin, LOW);
  delay(sensorValue);
    Serial.println(sensorValue, DEC);     }
```

21

```
int R = 2;
 void setup()
{
 Serial.begin(115200);
 Serial.println("Arduino Examples - 7 Color Flash");
 pinMode(R, OUTPUT); digitalWrite(R, LOW);
}
void loop()
{
 digitalWrite(R, HIGH); // on (for 5 Second)
 delay(5000);
}
```

22

```
int Led = 13 ;// define LED Interface
int buttonpin = 8; // define D0 Sensor Interface
int val = 0;// define numeric variables val
void setup ()
{    pinMode (Led, OUTPUT) ;// define LED as output interface
  pinMode (buttonpin, INPUT) ;// output interface D0 is defined sensor
}
 void loop ()  {    val = digitalRead(buttonpin);// digital interface will be assigned a value of pin 3 to read val
```

```
 if (val == HIGH) // When the sound detection , LED flashes
 {      digitalWrite (Led, HIGH);    }
 else      {       digitalWrite (Led, LOW);   }
 }
```

```
 int sensorPin = A0; // select the input pin for the potentiometer
int ledPin = 13; // select the pin for the LED
int sensorValue = 0; // variable to store the value
void setup ()
{    pinMode (ledPin, OUTPUT);
 Serial.begin (9600);
}
 void loop ()
{    sensorValue = analogRead (sensorPin);
 digitalWrite (ledPin, HIGH);  delay (sensorValue);
 digitalWrite (ledPin, LOW);    delay (sensorValue);
 Serial.println (sensorValue, DEC);          }
```

23

```
#define mercuryTilt 5
#define LED 6
```

```
#define onBoardLed 13

boolean state = false;

boolean initialState;

void setup() {

pinMode(mercuryTilt, INPUT); pinMode(LED,OUTPUT);

pinMode(onBoardLed, OUTPUT);

initialState = digitalRead(mercuryTilt);

digitalWrite(LED, LOW); // start with LED off

}

void loop() { state = digitalRead(mercuryTilt);

if(state != initialState){  digitalWrite(LED, HIGH); // turn LED ON if
switch has been tilted

digitalWrite(onBoardLed, HIGH); // echo output to Arduino's LED

}  else{  digitalWrite(LED, LOW);  digitalWrite(onBoardLed, LOW);}
```

```
24

void setup() {

  Serial.begin(9600); // Enable Serial Communication }

void loop() {

  int sensorValue1 = analogRead(A0);

  int sensorValue2 = analogRead(A1);

  int sensorK = digitalRead(D3)

  Serial.print("The X and Y coordinate is:");

  Serial.print(sensorValue1, DEC);

  Serial.print(",");
```

```
 Serial.println(sensorValue2, DEC);
 Serial.println(" ");
 delay(200);
}
```

25

```
int Led = 13 ; // define LED Interface
int buttonpin = 3; // define the linear Hall magnetic
int val ; // define numeric variables val
void setup () { pinMode (Led, OUTPUT) ;
 pinMode (buttonpin, INPUT) ; }
void loop ()
{ val = digitalRead (buttonpin) ;
 if (val == HIGH)
{ digitalWrite (Led, HIGH); }
else { digitalWrite (Led, LOW);
}
 }
```

```
const int ledPin = 13;
int sensorPin = A0;
```

```
int digitalPin=7;
int sensorValue = 0;
boolean digitalValue=0;
void setup()
{  pinMode(digitalPin,INPUT); pinMode(ledPin,OUTPUT);
 Serial.begin(9600); }
void loop()
{  sensorValue = analogRead(sensorPin);
digitalValue=digitalRead(digitalPin);
 Serial.print("Sensor Value ");  Serial.println(sensorValue);
 Serial.print("Digital Value ");Serial.println(digitalValue);
 if( digitalValue==HIGH )  {
   digitalWrite(ledPin,LOW);//turn off the led
 }
 if( digitalValue==LOW)  else
 { digitalWrite(ledPin,HIGH);//turn on the led
 } delay(1000);}
```

26

```
int LedOutput = 12;
int SensorPin = 2; // Define as Sensor Pin Input
int Value;// Define as variable
void setup()
{   pinMode(LedOutput,OUTPUT);//Set as LED output
pinMode(SensorPin,INPUT);//Set as photo interrupter sensor
```

```
}
void loop()
{ Value=digitalRead(SensorPin);if(Value==HIGH) {
digitalWrite(LedOutput,HIGH); // Set ledoutput to HIGH or ON
} else
{ digitalWrite(LedOutput,LOW); }
  }
```

```
int AnalogSensorPin = A0;
int LedOutIndicator = 12; int Value = 0;
void setup () {   pinMode (LedOutIndicator, OUTPUT);
 Serial.begin (9600);
}
void loop () {   Value = analogRead (AnalogSensorPin);
 digitalWrite (LedOutIndicator, HIGH);
delay (Value);
 digitalWrite (LedOutIndicator, LOW);
  delay (Value);
 Serial.println (Value, DEC);
}
```

27

```
const int ledPin = 13;
const int avoidPin = 7;
void setup()
{     pinMode(ledPin, OUTPUT);   pinMode(avoidPin, INPUT); }
void loop()
{
 boolean avoidVal = digitalRead(avoidPin);
 if(avoidVal == LOW)
 {     digitalWrite(ledPin, HIGH);   }    else
{
   digitalWrite(ledPin, LOW);   }
  }
```

28

```
int sensorPin = A0;
double alpha = 0.75; int period = 100;double change = 0.0;
double minval = 0.0;
void setup ()
{
 Serial.begin (9600);
}
```

```
void loop ()
{   static double oldValue = 0; static double oldChange = 0;
   int rawValue = analogRead (sensorPin);
   double value = alpha * oldValue + (1 - alpha) * rawValue;
   Serial.print (rawValue);Serial.print (","); Serial.println (value);
   oldValue = value;delay (period);
}
```

29

```
float voltage_Sense;
float sensor_value;
void setup() {
 Serial.begin(9600);}
 void loop() {
 sensor_value = analogRead(A0);
 voltage_Sense = sensor_value/1024*5.0; // Voltage / 1024 * 5.0
  Serial.print("voltage_Sense = ");
Serial.print(voltage_Sense);
 Serial.println("Voltage");
delay(1000);
}
```

30

```
const int s0 = 8;  const int s1 = 9; const int s2 = 12;  const int s3 = 11;
```

```
const int out = 10;   int redLed = 2;  int greenLed = 3; int blueLed = 4;

int red = 0;  int green = 0;  int blue = 0;

void setup()

{

Serial.begin(9600);   pinMode(s0, OUTPUT);   pinMode(s1, OUTPUT);

 pinMode(s2, OUTPUT);   pinMode(s3, OUTPUT);    pinMode(out,
INPUT);

 pinMode(redLed, OUTPUT);   pinMode(greenLed, OUTPUT);
pinMode(blueLed, OUTPUT);

 digitalWrite(s0, HIGH);    digitalWrite(s1, HIGH);

}

 void loop()

 {

 color(); Serial.print("R Intensity:");Serial.print(red, DEC);

 Serial.print(" G Intensity: ");Serial.print(green, DEC);

 Serial.print(" B Intensity : ");Serial.print(blue, DEC);

 if (red < blue && red < green && red < 20)

 {

 Serial.println(" - (Red Color)"); digitalWrite(redLed,
HIGH);digitalWrite(greenLed, LOW);

 digitalWrite(blueLed, LOW);

 }
```

الكود البرمجي الجزء الثاني

```
else if (blue < red && blue < green)

 {
```

```
 Serial.println(" - (Blue Color)");

  digitalWrite(redLed, LOW); digitalWrite(greenLed, LOW);
digitalWrite(blueLed, HIGH);

 }

 else if (green < red && green < blue)

 {

 Serial.println(" - (Green Color)");    digitalWrite(redLed, LOW);

  digitalWrite(greenLed, HIGH);   digitalWrite(blueLed, LOW);

 }

 else   {Serial.println();

 }

 delay(300);

 digitalWrite(redLed, LOW);   digitalWrite(greenLed, LOW);
digitalWrite(blueLed, LOW);

 }

void color()

{

 digitalWrite(s2, LOW);digitalWrite(s3, LOW);red = pulseIn(out,
digitalRead(out) == HIGH ? LOW : HIGH);

 digitalWrite(s3, HIGH); blue = pulseIn(out, digitalRead(out) == HIGH ?
LOW : HIGH);

 digitalWrite(s2, HIGH); green = pulseIn(out, digitalRead(out) == HIGH
? LOW : HIGH);

}
```

النهايـــــــــــــــــــــــــة

استاذ فلاح كاطع
مطور ومبرمج وباحث في مجال
Embeded System and IOT
Web Client Frontend App.And Web Server Backend
DataBase Admin.

لاتنسى الاشتراك بالقناة لكي تصلك اخر المشاريع

مع الاكواد البرمجية والشروحات والافكار البسيطة

ويمكنك التواصل معي عبر الروابط

iraqprogrammer.wordpress.com

http://www.facebook.com/falahgs

mailto: digital_fgs@yahoo.com

https://github.com/falahgs