

Duinotech 37 in 1 Sensor Kit

Generally, any of the analog or digital pin assignments noted below can be changed to any other pin of the same type. The examples below are given to correspond to the code snippets. The only exception to this is not to use D0 and D1 if possible, as they might interfere with sketch uploading and serial communication on the Uno, Mega and Nano. If you need to use D0 and D1, then disconnect these pins until after the sketch is uploaded. The modules have been tested on 5V systems unless otherwise noted. Where a part number is given in brackets, then this is the Jaycar part number of the individual sensor module by itself.

Code Samples:

Most modules will output a digital or analog output signal, and for those that do, the following code examples will work. You may need to change the pin to correspond to your wiring, and then open the Serial Monitor to view the output from the Arduino. You can also use the Serial Plotter (Tools>Serial Plotter) to view analog values as a graph.

Some of the modules need a library, and this will have to be downloaded separately. There will be an example sketch inside this library where this is necessary.

Analog:

```
void setup() {
  Serial.begin(9600);
}
void loop() {
  Serial.println(analogRead(A0));
  delay(50);
}
```

Digital:

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

void loop() {
  Serial.println(digitalRead(4)?"HIGH":"LOW");
  delay(200);
}
```

Joystick Module (XC4422)

Module	DuinoTech	Function
GND	GND	Ground Connection
+5V	5V	Power
VRX	A0	X Output
VRY	A1	Y Output
SW	D7	Pushbutton Output



The Joystick module will work equal well on 5V and 3.3V Arduinos. VRX gives an analog output proportional to X-position, VRY gives an analog output proportional to Y-position, and SW is connected to ground when the Pushbutton is pushed. The easiest way to interface to the button is to configure the pin in setup:

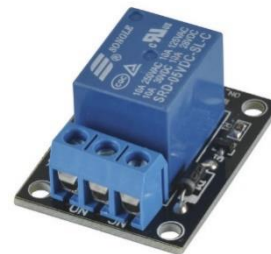
```
pinMode(7, INPUT_PULLUP);
```

then the following will read 0 when the button is pushed and 1 when the button is released:

```
digitalRead(7)
```

Relay Module (XC4419)

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D8	Control Signal



Relay module is designed to work on 5V power and 5V signal, but modules seem to work fine on 3.3V systems too. When the S pin is high, the relay is energised (and COM is connected to NO), when the S pin is low, the relay is de-energised (and COM is connected to NC).

Small Microphone Module

Large Microphone Module (XC4438)

Module	DuinoTech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



These modules provide two outputs- an analog signal which comes directly from the microphone, and a digital one, which is the result of the analog signal being compared with the setting of the potentiometer. The state of the digital output is shown in the state of LED L2. Typically, the potentiometer is adjusted so that it triggers the digital output at a certain level. We want the L2 LED to be flickering between on at the slightest sound. If the L2 LED is on, turn the brass screw anticlockwise. If L2 is off, turn the screw clockwise until it turns on, then turn anti-clockwise until it just goes off. The LED should now flash on if the microphone is tapped. If you would like it to be less sensitive to sounds, it can be turned further anti-clockwise.

Line Tracking Sensor (XC4474)

Module	DuinoTech	Function
G	GND	Ground Connection
V+	5V	Power
S	D4	Digital Signal



The Line Tracking Sensor outputs a digital signal depending on how much reflection it is getting off the surface in front of it and also the position of the potentiometer. You would need two of these sensors to implement a line tracking robot (one to test whether the robot is drifting left, and one to test if it's drifting right).

Obstacle Avoidance Sensor (XC4524)

Module	DuinoTech	Function
GND	GND	Analog Signal
+	5V	Ground Connection
Out	D4	Power
EN	not needed	



Although it appears similar to other modules that have adjustable sensitivity, this module is different. Be careful not to change the potentiometer closest to the GND pin, as this is a factory calibration. The other potentiometer closest to the 'EN' marking is for changing the IR emitter power, and thus the effective sensing range. Turn the potentiometer clockwise for longer range (but not fully clockwise, as the output locks on), and anti-clockwise for shorter range. Note that the effective range will also depend on how reflective the obstacles are.

When connected to power, the 'Pled' LED will light, and the 'Sled' LED will light up if an obstacle is in front of the sensor- this makes testing easier. The 'Out' output is reversed, and will be Low (0V) when there is an obstacle, and High (5V) when there is no obstacle.

Flame Sensor Module

Module	DuinoTech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



This module is sensitive to infrared light, and since flames usually emit a lot of infra-red light, this module can be used to sense the presence of a flame. As with other similar looking sensors, this module has both digital and analog outputs, and the threshold for where the digital output changes can be adjusted via the blue potentiometer. The state of the digital output is also shown on the L2 LED, so the threshold can be adjusted just by hooking it up to a power supply. If the L2 LED is on, turn the brass screw anticlockwise. If L2 is off, turn the screw clockwise until it turns on, then turn anti-clockwise until it just goes off. Keep in mind that the strength of the signal received will depend on the intensity of the flame and the distance from the flame.

Hall Effect Sensor (Digital and Analog)

Module	DuinoTech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



This Hall Effect sensor provides both analog and digital outputs. The analog output behaves like the analog sensor below, while the digital part has an adjustable threshold. The state of the digital output is also shown on the L2 LED, so the threshold can be adjusted just by hooking it up to a power supply. If the L2 LED is on, turn the brass screw anticlockwise. If L2 is off, turn the screw clockwise until it turns on, then turn anti-clockwise until it just goes off.

With the adjustable threshold, you could set the module to trigger at a preset distance from a magnet. Note that like most Hall Effect sensors, this module is sensitive to magnetic field direction as well as strength.

Hall Effect Sensor

Hall Effect Switch Module (XC4434)

There are two almost identical looking modules, both of which are Hall Effect devices (ie they react to magnetic fields). The module with the IC marked '49E' has the analog output and the module with the IC marked '44E' has a digital output. Otherwise, they look identical.

Analog Module:

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	A0	Analog Signal



This sensor can also operate on 3.3V supplies.

This Hall Effect Sensor outputs an analog signal related to the magnetic field strength. In the presence of no magnetic field, the output will be half the supply voltage. Applying a magnet one way will cause the voltage to decrease, the other way will cause the voltage to increase. The output changes by about 1V per 60mT.

Digital Module:

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D4	Digital Signal



This sensor outputs a digital signal that changes from high to low in the presence of a magnetic field. It is normally high (5V), and goes low (0V) in the presence of a magnet. If you find you aren't getting a changing output, try reversing the magnet, as the sensor only responds to a magnetic field in one direction. The LED on the module goes on when the output is low, so it can be tested without a sketch being loaded.

Touch Sensor

Module	Duinodech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



The Touch Sensor Module provides both analog and digital outputs, but in practice, only the digital output is needed- it will register high when the lead wrapped around the body of the transistor is touched, and low otherwise, and the L2 LED mimics the D0 output. If you find that the LED is on when it should be off, turn the potentiometer anti-clockwise. If the LED doesn't come on when it should, try turning the potentiometer clockwise.

Temperature Sensor Module

Module	Duinodech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



This module, like most of the 4 pin types, has both analog and digital outputs. The state of the digital output is also shown on the L2 LED, so the threshold can be adjusted just by hooking it up to a power supply. If the L2 LED is on, turn the brass screw anticlockwise. If L2 is off, turn the screw clockwise until it turns on, then turn anti-clockwise until it just goes off. The digital output will now trigger near the current temperature.

The analog output might can be converted with some mathematics to a useful temperature value, as shown below.

```
#include <math.h>
double Thermistor(int RawADC){
  double Temp;
  Temp = log(((1024000/RawADC) - 10000));
  Temp = 1/(0.001129148+(0.000234125+(0.000000876741*Temp*Temp ))*Temp);
  Temp = Temp - 273.15;
  return Temp;
}

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

void loop(){
  Serial.print(Thermistor(analogRead(A0)));
  Serial.println("c");
  delay(1000);
}
```

Digital Temperature Sensor

Module	DuinoTech	Function
-	GND	Ground Connection
(not marked)	5V	Power
S	D10	Digital signal



The Digital Temperature Sensor module contains a small integrated circuit which communicates with your Arduino via the S pin on the module, using the 'OneWire' protocol- there is more information on this at <http://playground.arduino.cc/Learning/OneWire>. Because of this, we also need a library to control the interface- this can be installed through Library Manager by clicking Sketch>Include Library>Manage Libraries and search for 'onewire' and installing the library simply called 'OneWire', or by downloading and installing manually from <https://github.com/PaulStoffregen/OneWire>.

Fortunately, the DS18B20 IC on this module is quite common, and the DS18x20 example from File>Examples>OneWire will work. Open it, connect as above, and open the Serial Monitor at 9600 Baud. Information from the OneWire device will then be displayed, including temperature and the device address.

Buzzer Module (XC4424)

Passive Buzzer Sounder

Module	DuinoTech	Function
-	GND	Ground Connection
+	Not needed	
S	D8	Waveform signal



These two modules are ideal for adding a sound output to a project. If the module is connected as above, the following code will play the note A indefinitely:

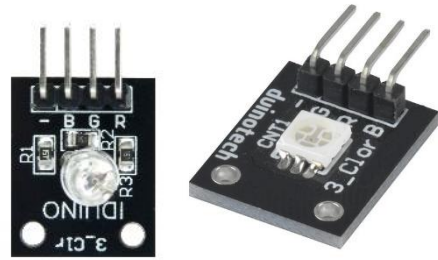
```
tone(8,441);
```

For more examples using the built-in tone library, see <https://www.arduino.cc/en/Reference/Tone> and the examples under File>Examples>02.Digital.

RGB LED Module

SMD RGB LED Module

Module	Duinotech	Function
-	GND	Ground Connection
G	D2	Green LED
R	D3	Red LED
B	D4	Blue LED



Both of these modules can be used to display a wide range of colours. Using simple digital outputs, you can make the following colours:

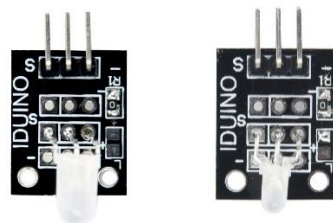
Colour	R pin	G pin	B pin
Off	Low	Low	Low
Red	High	Low	Low
Green	Low	High	Low
Blue	Low	Low	High
Cyan	Low	High	High
Purple	High	Low	High
Yellow	High	High	Low
White	High	High	High

Using the `analogWrite()` function (which is only available on certain pins, usually marked with ~ on the Arduino board), you can make up pretty much any colour which can be displayed on a computer monitor. See <https://www.arduino.cc/en/Reference/AnalogWrite> for more information.

Two Colour LED Module (5mm)

Two Colour LED Module (3mm)

Module	Duinotech	Function
-	GND	Common Cathode
+	D2	Green LED Anode
S	D3	Red LED Anode

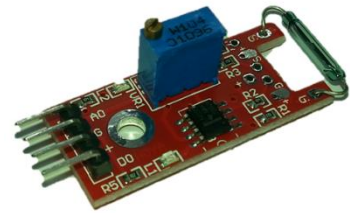


The Two Colour LED Module is a variation of the RGB LED Module, but only containing the red and green LED's, and also lacking the dropping resistors to limit current to the LEDs. To avoid damaging the LEDs, a 150 Ohm resistor should be placed between the digital outputs and the module (the GND connection can be made direct). The following output combinations will work:

Colour	R pin	G pin
Off	Low	Low
Red	High	Low
Green	Low	High
Yellow	High	High

Reed Switch Module

Module	DuinoTech	Function
A0	A0	Analog Signal
G	GND	Ground Connection
+	5V	Power
D0	D4	Digital Signal



This module has analog and digital outputs, but because a reed switch is effectively an on/off device, even the analog output is really digital, and adjusting the blue potentiometer on this module won't have any noticeable effect on the output. The digital output will be high when a magnet is near the reed switch, while the analog output will be low when a magnet is near.

Mini Reed Switch Module

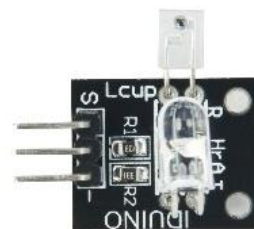
Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D4	Digital Signal



The Mini Reed Switch Module is a simplified version of the above module, and the output will change when a magnet is brought near the reed switch.

Heartbeat Sensor

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	A0	Signal



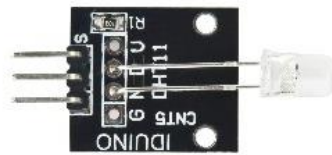
Although it looks very simple, this sensor can detect a pulse or heartbeat when a finger is placed between the LED and photosensor. Connect the sensor as shown above, then upload the following sketch to the Arduino and open the Serial Plotter (Tools>Serial Plotter). This is a great way to visualise what is happening with an analog sensor signal, and could be used with any analog sensor.

```
void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  Serial.println(analogRead(A0));  
  delay(50);  
}
```

If you can't see a signal, try aiming the LED so that it points more towards the middle of the photosensor.

Seven Colour Flashing

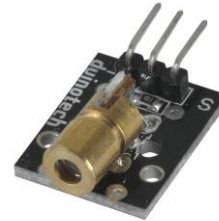
Module	DuinoTech	Function
-	GND	Ground Connection
+	Not needed	
S	D8	Signal



This module contains an LED which automatically cycles through seven colours when powered on through a digital pin from the Arduino. It could be used to create a very eye-catching indicator light.

Laser Module (XC4490)

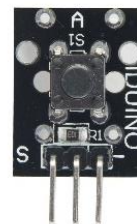
Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D8	Signal



This module is just like what you'd find in a laser pointer- a small module able to create a highly focussed red dot. Set the S pin high to activate the laser, and low to turn it off.

Tactile Switch Module

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D8	Output



The tactile switch module gives you a plug and play user input for your project. Will output a high level when not pressed, and a low level when pressed. Will work on 3.3V and 5V systems.

Rotary Encode Module

Module	DuinoTech	Function
CLK	D7	Rotary phase A
DT	D6	Rotary phase B
SW	D5	Pushbutton
+	5V	Power
GND	GND	Ground



The Rotary Encode Module is an easy way to add a control interface to a project- in fact many 3d printers have nothing more than a rotary encoder as their user input device. The pushbutton is a simple switch that is connected to ground when the button is pushed. To use the button, put this line in your setup:

```
pinMode(5, INPUT_PULLUP);
```

then the following will read 0 when the button is pushed and 1 when the button is released:

```
digitalRead(5)
```

The following code will read the rotary part of the module and display the position on the Serial Monitor:

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

void loop() {
  static int s=0;
  int d=checkrotary(6,7);
  s=s+d;
  if(d){Serial.println(s);}
}

int checkrotary(int rotary1, int rotary2){ //pins
  static byte old=0;
  byte new1=digitalRead(rotary1);
  byte new2=digitalRead(rotary2);
  byte vnew=(new1*2)|(new2^new1);
  byte d=old-vnew;
  old=vnew;
  switch(d){
    case 1: return 1;
    case 3: return -1;
    case 255: return -1;
    case 253: return 1;
    default: return 0;
  }
}
```

Tilt Switch Module

Ball Switch Module

Shock Sensor Module

Vibration Sensor Module



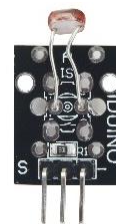
Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D4	Digital Signal

These four modules use a small moving conductive element to make or break a circuit, thus giving a high or low output signal. The Tilt Switch Module has a small amount of conductive liquid, while the Ball Switch Module has a small ball bearing, and both can detect whether they have been rotated. The Shock Sensor Module and Vibration Sensor Module have a small spring, and are triggered when exposed to a shock or jolt. A good way to test how they respond to movement is to hook them up (one at a time) as described above, and run the following sketch to display on the Serial Monitor:

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.println(digitalRead(4)?"HIGH":"LOW");  
  delay(200);  
}
```

Light Dependent Resistor Module

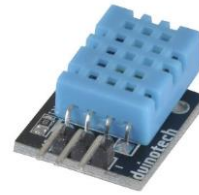
Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	A0	Analog Signal



The LDR Module is ideal for measuring changes in ambient light intensity, and works due to the changing resistance of the LDR. The second resistor on the board forms a voltage divider, so the output can be fed directly into an analog input. The analog sketch can be used to get an idea of the values you will get under certain conditions. Upload the sketch, and open the Serial Monitor at 9600 Baud, and see what values you get by covering and uncovering the sensor.

Temperature and Humidity Module (XC4520)

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D5	Communication Pin



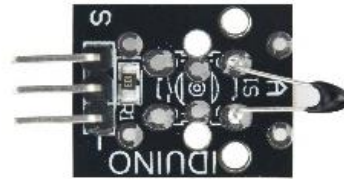
In this small package is an IC which can read temperature and humidity from the surroundings and communicate them with an Arduino- two sensor readings on one IO pin. Because we can't simply read an analog value, we need to use a library. We use the library recommended on the Arduino website: <http://playground.arduino.cc/Main/DHTLib> (the IC inside the sensor is called a DHT11, hence the name of the library).

The library can be downloaded and installed from:

<https://github.com/RobTillaart/Arduino/tree/master/libraries/DHTlib>, and then the dht11_test example can be loaded from Files>Examples>DHTlib. Upload the example file, and open the Serial Monitor at 115200 Baud. The sketch will display the temperature and humidity- you can see both change by breathing onto the module (this will warm it and increase the humidity). Due to limitations of the IC, the data can only be updated about once every two seconds.

Analog Temperature Sensor

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	A0	Analog Signal



The Analog Temperature Sensor uses a thermistor (or a resistor that changes resistance with temperature) as the sensing element. While this makes reading the data easy, the value needs to have a bit of mathematics applied to turn it into the value we're looking for.

```
#include <math.h>
double Thermistor(int RawADC){
  double Temp;
  Temp = log(((1024000/RawADC) - 1000));
  Temp = 1/(0.001129148+(0.000234125+(0.000000876741*Temp*Temp ))*Temp);
  Temp = Temp - 273.15;
  return Temp;
}

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

void loop(){
  Serial.print(Thermistor(analogRead(A0)));
  Serial.println("c");
  delay(1000);
}
```

IR Transmitter Module (XC4426)

Module	DuinoTech	Function
-	GND	Ground Connection
+	Not connected	
S	D3	Output



You can use a simple digital pin to turn the IR LED off and on, but if you want to send codes that can be received by an appliance or the IR Receiver Module, you'll need a library- we've been using this one: <https://github.com/z3t0/Arduino-IRremote>. You might also find it under the Library Manager (click Sketch>Include Library>Manage Libraries and search for 'irremote').

D3 is the defined pin for this library for the Uno, but it might be different on other boards- see the library documentation for more info. To test an IR Receiver Module, you could use the IRsendDemo example, or if you've already worked out which protocol and codes your system is using, look for the appropriate matching example.

IR Receiver Module (XC4427)

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	D11	Signal from receiver

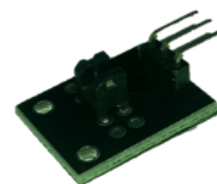


The IR Receiver Module simply outputs the demodulated version of any IR signal it receives. This is usually a digital code which may have a different format depending on the type of code used in the corresponding IR transmitter. Fortunately, there are Arduino libraries to handle this. We have been using this one: <https://github.com/z3t0/Arduino-IRremote>. You might also find it under the Library Manager (click Sketch>Include Library>Manage Libraries and search for 'irremote').

Once this library is installed, open and upload the 'IRrecvDumpV2' example, wire up the IR Receiver Module as shown, and open the Serial Monitor at 9600 Baud. You can then use this to find out what buttons on your remote correspond to what codes. Alternatively, the IR Transmitter Module can also be used with the same library to generate IR codes.

Photo Interrupter Module

Module	DuinoTech	Function
-	GND	Ground Connection
+	5V	Power
S	A0	Signal from module



The Photo Interrupter Module contains an IR LED and photo-transistor, and is designed to detect an object in the gap. Although the active area is very small, the signal is very strong and precise, and is used in things like inkjet printers to tell the printer when the printhead is in the correct position- you can imagine it has to be accurate to within a fraction of a millimetre to produce consistent print results. Connect the module as above, and see from the Serial Monitor how it responds. Because it uses an IR LED, you may find it even reacts to transparent objects.

Magic Light Cup Modules

Module	Duinotech	Function
G	GND	Ground Connection
+		No connection
S	D7	Signal from module
L	D6	LED

Module	Duinotech	Function
G	GND	Ground Connection
+		No connection
S	D4	Signal from module
L	D3	LED



This pair of modules is designed to work together as an optical effect- when you load the sketch below, one of the LED's will light up, and the light can be 'poured' by tilting the modules as though pouring from one to module to the other.

```
void setup (){
  pinMode (3, OUTPUT);
  pinMode (6, OUTPUT);
  pinMode (4, INPUT_PULLUP);
  pinMode (7, INPUT_PULLUP);
}

void loop (){
  static int b=0;
  if(digitalRead(4)&(!digitalRead(7))){
    b++;
    if(b>255){b=255;}
  }
  if(digitalRead(7)&(!digitalRead(4))){
    b--;
    if(b<0){b=0;}
  }
  analogWrite(6,b);
  analogWrite(3,255-b);
  delay(20);
}
```