

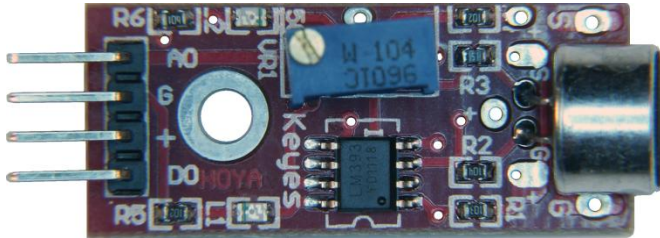
342162

24 Sensors Kit (Contents)

1x Ball switch
1x Buzzer (active)
1x Buzzer (passive)
1x LM35 temperature sensor
1x Flame sensor
1x IR collision detector
1x IR Line Follower
1x IR receiver
1x IR transmitter
1x Laser module
1x LED - 3W white
1x LED - RGB
1x Light Sensor (photo resistor)
1x Linear Hall sensor
1x magnetic reed switch
1x Microphone sound sensor
1x Relay 5v
1x Rotary Encoder
1x Real Time Clock (RTC) module
1x Small button switch
1x Temp & Humidity
1x Touch sensor
1x Vibration sensor
2x Light Cup (LED + mercury tilt)

Components:

Sensors and modules



Microphone Sound Sensor

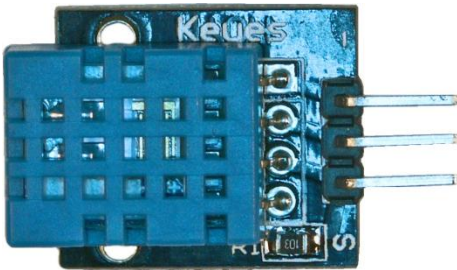
Pin connections:

- G - connect to Ground
- + - connect to 5V
- AO - Analog Out - connect to Arduino for analog input
- DO - Digital Out - Connect to Arduino as digital trigger input; adjust sensitivity via the screw on the potentiometer.

For code examples, see:

<http://www.princetonics.com/sound-sensitive-lights-w-sound-sensor-arduino/>

https://tkkrlab.nl/wiki/Arduino_KY-038_Microphone_sound_sensor_module



Temperature and Humidity sensor

Three connections:

- (-) = ground (-) Note the square solder pad.
- (center pin) = +5V
- (S) = Signal (digital, serial output)

For the DHT11 library and information, see:

<http://playground.arduino.cc/main/DHT11Lib>

<http://www.johnboucha.com/arduino-dht11-temperature-humidity/>

https://tkkrlab.nl/wiki/Arduino_KY-015_Temperature_and_humidity_sensor_module

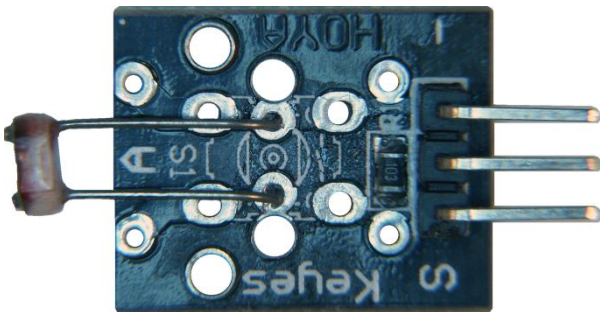


Photo Resistor Light Sensor

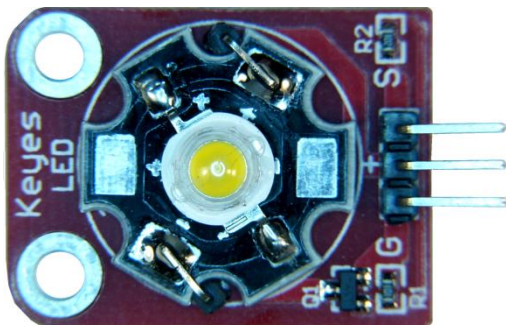
Three connections: Out - Signal (analog voltage level), +5V, and - Ground.

Analog signal level will be higher in bright light, lower in darkness. For example code, see:

<http://playground.arduino.cc/Learning/PhotoResistor>

https://tkkrlab.nl/wiki/Arduino_KY-018_Photo_resistor_module

<http://www.cutedigi.com/articles/sensor-ky018-arduino-sample-code.html>



LED module - 3W white

The bright white LED module has three pin connections:

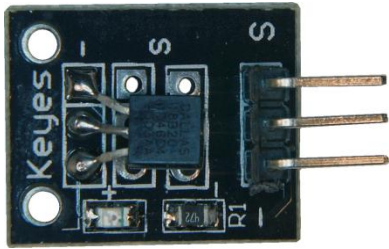
- G = ground
- + = +5V
- S = signal

The 3W White LED module uses a surface-mount transistor to control the brightness of the LED while drawing minimal current from the Arduino signal pin output (S).

The Arduino IDE includes several examples for use with LEDs, see:

File, Examples, 01.Basic, **Fade**

File, Examples, 01.Basic, **Blink**



Temperature Sensor

The Temperature sensor uses three connections (-) Ground, (+) +5V, and (S) Signal. It has an operating range of -10C to +85C. The module also includes an LED and utilizes I2C serial communication through the (S) Signal pin.

Dallas 18B20 information:

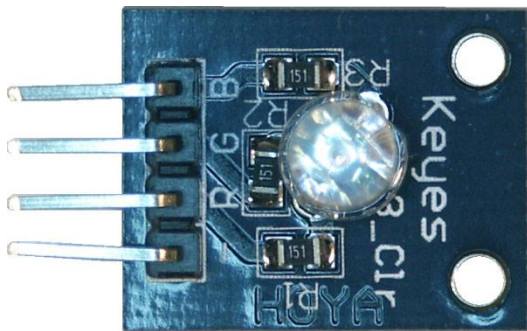
<http://www.maximintegrated.com/en/products/analog/sensors-and-sensor-interface/DS18B20.html>

For code examples, see:

<http://playground.arduino.cc/Learning/OneWire>

<http://playground.arduino.cc/Main/GetOneWireTemp>

<http://playground.arduino.cc/ComponentLib/Thermistor>



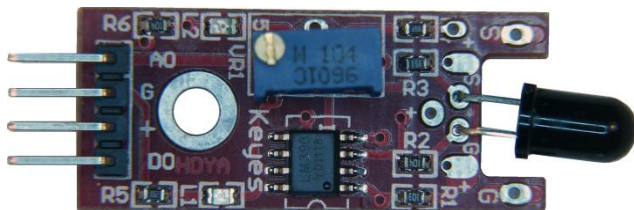
LED - RGB Module

Current-limiting resistors ($151 = 15 \times 10 = 150 \text{ ohm}$) are already attached to the positive Red, Green, and Blue pins of the LED. Connect the negative (-) pin to your ground, and the R, G, and B pins to your Arduino output pins.

If using PWM (Pulse Width Modulation) capable outputs, you can effectively mix the RGB primary colors to produce thousands or different output colors in the single LED. See:

File, Examples, 01.Basic, **Fade** sketch example in the IDE

<http://playground.arduino.cc/Main/RGBLEDPWM>

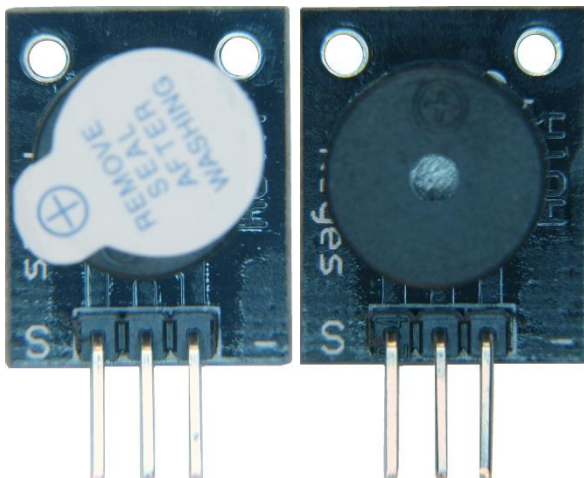


Flame Sensor (YG1006 or similar)

The Flame sensor is a high-speed and highly sensitive NPN Silicon photo transistor based on the YG1006. It can be used to detect fire or other wavelength at 760nm ~ 1100nm light. Response time is 15us, supply voltage is 3.3-5V; output is analog (A0) or digital (D0). To use D0 as a digital trigger input; adjust sensitivity via the screw on the potentiometer.

For an Arduino code example, see:

https://tkkrlab.nl/wiki/Arduino_KY-026_Flame_sensor_module



Passive & Active buzzers

Use as a speaker, buzzer or other audible indicator.

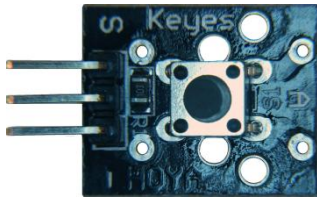
The Active buzzer has a protective tag over the opening, note the + identifies the positive pin of the device, as the rear is covered with epoxy. The active buzzer will generate a tone as soon as power is supplied to the device.

Passive buzzers must have a modulated signal supplied to the device (like a speaker) and would only generate a "click" if DC voltage is applied.

Play a tone or melody using the passive buzzer:

<http://osepp.com/learning-centre/start-here/101-basic-starter-kit/tutorial-6-using-buzzer-to-play-a-melody/>

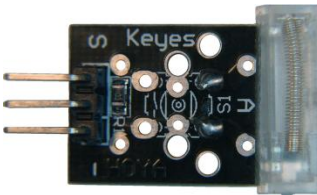
<http://www.arduino.cc/en/Tutorial/melody>



Small Button Switch

3 connections: Signal, + positive (+5V in) and - negative (ground). When this module is connected, the surface mount resistor (103 = 10K ohms) pulls the signal pin up to +5V. When the button is pushed, the signal pin is pulled to ground through the switch.

To use the Arduino example sketch "Button", configure your IDE to the correct board and COM port, and then load the "Button" sketch: *File, Examples, 02.Digital, Button*
The default configuration in this sketch is set to use a button switch on D2 and the LED on D13. When you press the button, the LED on the Arduino board should light up.

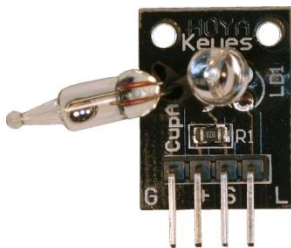


Vibration Sensor

The vibration sensor consists of a spring connected to one contact and a pin inside the spring connected to the other contact. Vibration of the spring will cause the spring to come into contact with the pin, closing the switch circuit briefly.

The surface mount resistor (103 = 10K ohms) pulls the signal pin up to +5V. When the contacts close, the signal pin is pulled to ground.

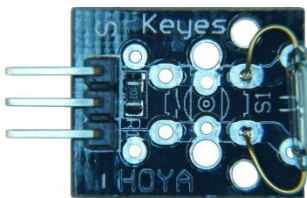
For an Arduino code example, try:
File, Examples, 02.Digital, Debounce



Light Cup (LED + Mercury Tilt Switch)

Four connections: (G) ground, (+) +5V, (S) signal from the mercury switch, and (L) for LED +. You should use a current limiting resistor between the (L) pin and your positive voltage source or Arduino signal pin. The negative pin of the LED connects to the (G) ground.

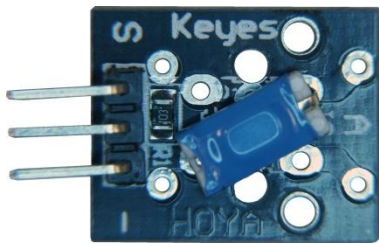
The surface mount resistor (103 = 10K ohms) pulls the signal pin up to +5V. When the mercury shorts the two contacts, the signal pin is pulled to ground.



Magnetic Reed Switch

The reed switch consists of two metal contacts held apart inside a glass tube. One of the contacts is iron that will be attracted to a magnet. To activate this switch, bring a small magnet near the tube to close the contacts. The more powerful your magnet (such as a neodymium magnet), the further away you can be from the sensor and still trip the switch.

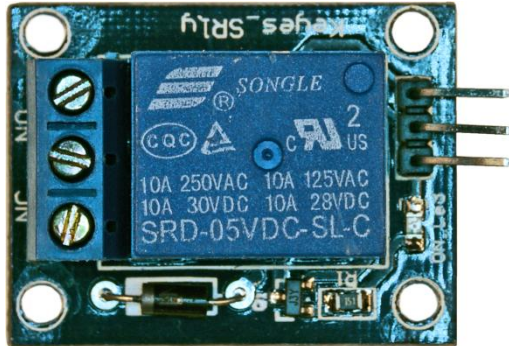
The surface mount resistor (103 = 10K ohms) pulls the signal pin up to +5V. When the contacts close, the signal pin is pulled to ground.



Ball Tilt Sensor

This is a very simple switch with a ball inside of the chamber. When the sensor is tipped past the horizontal, the ball will touch the top and bottom contacts, closing the switch.

The surface mount resistor (103 = 10K ohms) pulls the signal pin up to +5V. When the contacts close, the signal pin is pulled to ground.



5 Volt Relay module

Three input pins:

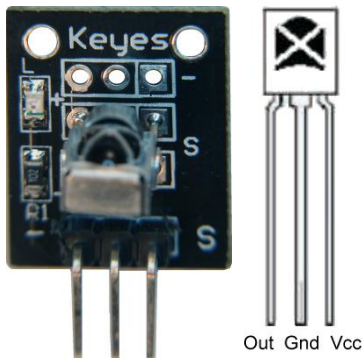
(+) +5V

(-) Ground

(S) Signal - connect to your Arduino "signal" pin to trip the relay.

To use an external 5V source to power the relay coil or servo motors, connect the ground of the external source to the Arduino ground.

Three output (screw) pins: Center is common, NC indicates Normally Closed (ON), NO indicates Normally Open (OFF). When relay engages, the NC contact will open, the NO contact will close.

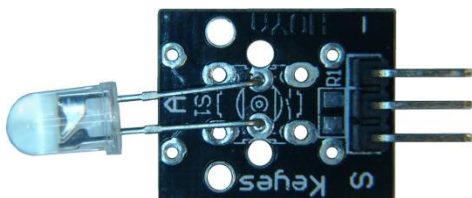


IR Receiver (VS1838B or similar)

Connect the Vcc pin to your 5V pin and the Gnd to a Ground pin. The Signal pin connects to an Arduino input pin and will change when the sensor detects an Infrared signal.

An IR remote control will send coded pulses based on which button you press, or an IR LED will produce a continuous illumination.

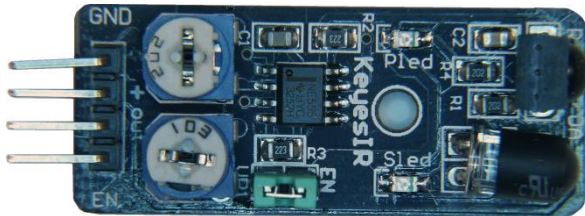
The surface-mount LED connects to the signal pin of the receiver. It will illuminate when the sensor detects an active infrared source.



IR Transmitter

The transmitter is an infrared LED. There is no current limiting resistor on the module, so you should place one between the (S) signal pin and your source.

Only 2 pins are used: (-) Ground and (S) Signal (the center pin is not connected.) When the Signal pin supplies positive voltage, the Infrared LED will illuminate. *Tip: you can verify operation by viewing the LED through a digital camera or smartphone camera. CCD cameras are sensitive into the infrared, and your display should show when the LED is on.*



IR Collision Detector

Typical use is with robots as collision avoidance detection, but it can also be used to detect a high-contrast mark on a rotating wheel, belt or tank tread (optical encoder). Sensitivity and LED brightness can be adjusted using the two potentiometers on the module.

The module consists of an infrared LED unit and an infrared detector that can trigger a signal based on the intensity of reflected IR light. Note the heat-shrink tubing over the LED - used to block side-shining IR light from the sensor. Pled = power LED; will illuminate when +5V and ground are connected. Sled = Signal LED; will illuminate if detector is enabled and reflected (or transmitted) IR light is detected.

Pin connections:

(GND) Ground

(+) +5v

(out) Output signal from detector

(EN) Enable External Vcc connection to detector; active if jumper "EN" is removed.

Object detection example:

http://www.tinkbox.ph/sites/mytinkbox.com/files/downloads/INFRARED_PROXIMITY_COLLISION_MODULE.pdf



IR Line Follower

Typical use is with robots for detecting and following a line. Other uses include using the module as an optical encoder to detect revolutions of a wheel or belt, etc. Sensitivity can be adjusted using the potentiometer on the module. The LED is connected to the (S) Signal pin, and illuminates when an IR source (or reflection) is detected.

The module consists of an infrared LED unit and an infrared detector that can trigger a signal based on the intensity of reflected IR light. Line following robots typically use a minimum of 2 line following sensors to keep the robot "centered" with the line between the two sensors.

Pin connections:

(G) Ground

(V+) + 5V

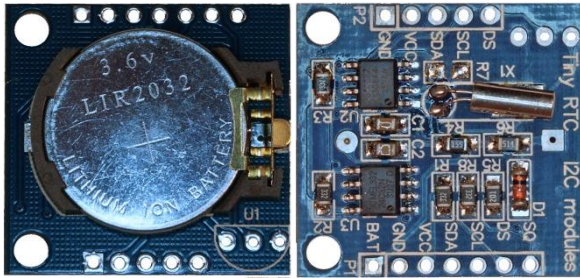
(S) Signal: low level when detecting reflected IR light, high when out of range (i.e. edge detection) or low reflectivity (i.e. black tape.)

Line following robot using 3 line sensor modules:

<http://adhocnode.com/line-following/>

Line following example with user-built line sensors:

<http://www.circuitstoday.com/line-follower-robot-using-arduino>



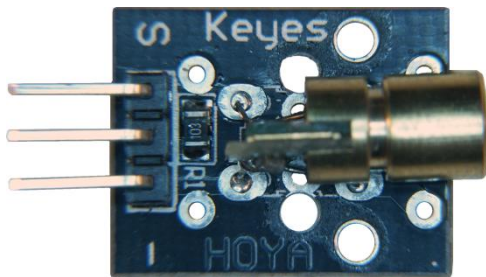
Real Time Clock Module

The Tiny RTC communicates with a microprocessor via the I2C serial interface. The real-time clock/calendar provides seconds, minutes, hours, day, date, month, and year information. Pin connections:

- BAT: Battery voltage monitor (not used)
- GND: Ground
- VCC: +5V
- SDA: I2C data (connect to A4)
- SCL: I2C clock (connect to A5)
- DS: DS18B20 Temp. Sensor output (connect to D2)
- SQ: Square wave output (not used)

For library and code example, see:

http://www.hobbyist.co.nz/?q=real_time_clock
<http://playground.arduino.cc/code/time>



Laser Module

The Laser is a basic 650nm red LED laser that can be used as a light source for a photo resistor module to create a laser "trip wire", or it can be modulated through the (S) signal line to transmit a digital signal. You can also connect this using just the Ground and Signal pin (the center-pin resistor will further limit the current and provide a dimmer output.)

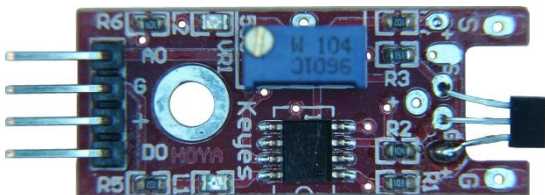
Pin connections:

- (S) Signal
- (center pin) +5V
- (-) Ground

With the ground and +5V pin connected, the laser will illuminate because the resistor between the center pin and (S) signal pin completes the circuit. If a signal is provided (+5V) on the signal pin, the laser remains on, but if the signal pin goes negative (ground), the laser will turn off.

Laser Tripwire example:

<http://computers.tutsplus.com/tutorials/how-to-build-a-laser-tripwire-with-arduino-cms-21485>



Linear Hall Sensor

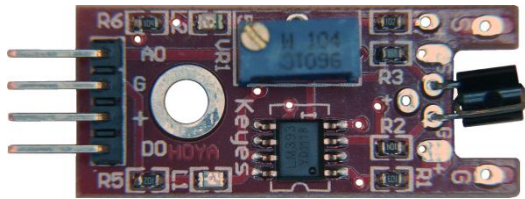
(SS49E or equivalent Linear Hall Effect Sensor) The Hall Effect sensor is very sensitive to magnetic fields, and instead of generating just an on/off condition like a reed switch, field strength may be monitored as an analog signal. To use DO as a digital trigger input; adjust sensitivity via the screw on the potentiometer.

For typical specifications see:

http://dsc.lcsr.jhu.edu/main/images/3/31/SS49e_Hall_Sensor_Datasheet.pdf
<http://www.diodes.com/datasheets/AH49E.pdf>

For Arduino code examples, see:

<http://playground.arduino.cc/Code/HallEffect>



Touch Sensor

The touch sensor module uses an MPSA13 (or equivalent) NPN Darlington Transistor. The base of the transistor is left unconnected and wraps around the insulated package to serve as an antenna. When the exposed lead is touched, the module will act like a momentary switch where the signal can be monitored on the DO pin, adjust sensitivity via the screw on the potentiometer.

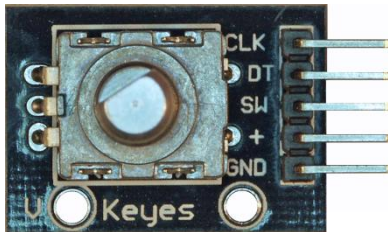
Pin connections:

(AO) Analog Output - You could monitor this to detect the gradual change from the transistor as you come near the transistor lead.

(G) Ground

(+) +5V

(DO) Digital Output - this will produce a digital signal similar to a button switch.



Rotary Encoder

The encoder generates two sets of square wave (on/off) pulses from the Clock and Data pins. These can be used to track the rotation of the shaft in either a clockwise or counterclockwise rotation. The encoder also incorporates a momentary push-button switch that is actuated by pressing down on the shaft.

Pin connections:

(CLK) Clock

(DT) Data

(SW) Button switch

(+) +5V

(GND) Ground

For Arduino code examples, see:

<http://playground.arduino.cc/Main/RotaryEncoders>

https://tkkrnlab.nl/wiki/Arduino_KY-040_Rotary_encoder_module

<http://arduinotronics.blogspot.com/2013/09/using-rotary-encoder-with-arduino.html>

Additional Resources:

Several sites have hook-up and information and code examples on a variety of sensors, similar to, and including the ones found in this kit. Some sensors may be loose components or integrated into different board designs. If the documented sensor uses the same electronic component, then any code sketch documented may work with the sensors found in your kit. However, depending on the circuit design, the adjustments or sensitivity range may need to be modified slightly to achieve the desired result. Sites documenting these and other sensors include:

Arduino Playground Examples and additional libraries (code sketches available from the IDE *File, Examples* menu):

<http://www.arduino.cc/en/Tutorial/HomePage>

Arduino Playground Tutorials: <http://playground.arduino.cc/Learning/Tutorials>

Forum.HobbyComponents.com: <http://forum.hobbycomponents.com/viewtopic.php?f=73&t=1320>

LinkSprite Wiki - Advanced Sensors Kit for Arduino: http://linksprite.com/wiki/index.php5?title=Advanced_Sensors_Kit_for_Arduino

TkkrLab.nl (Tukkerlab)Wiki: https://tkkrlab.nl/wiki/Arduino_37_sensors

University of Rhode Island (PDF coursework): <http://www.ele.uri.edu/courses/ele205/Arduino%20-%20Learning.pdf>

Freeduino.org: <http://www.freeduino.org/>

Arduino for Projects (PDF with 1193 projects): <http://duino4projects.com/arduino-projects-pdf/>

Lady Ada - Introduction to Arduino- step-by-step lessons: <http://www.ladyada.net/learn/arduino/index.html>

Tronixstuf Arduino Tutorials: <http://tronixstuff.com/tutorials/>

Earthshine Electronics Beginners Guide to Arduino:

https://docs.google.com/file/d/0Bw_ruMOtRDDgNXI3OTFGZXhIZ2c/edit?usp=sharing

Sheepdog's Guide to Arduino Programming: <http://sheepdogguides.com/arduino/FA1main.htm>