TA0018 37 in 1 Kit

MLB

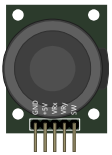
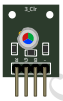
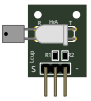
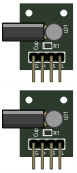
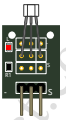
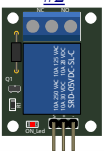
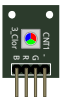
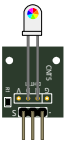
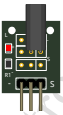
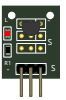
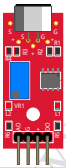
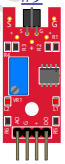
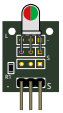
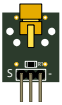
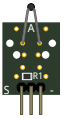
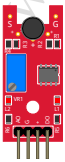
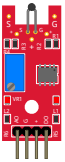
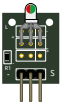
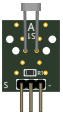
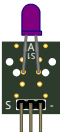
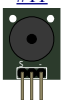
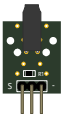
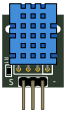
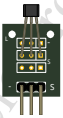
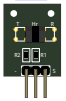
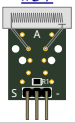
Contents

Hardware
Website: www.auselectronicsdirect.com.au
E-Mail: info@auselectroncsdirect.com
Phone: 02 9723 5902

Arduino: .....www.arduino.cc .....Arduino **I**ntegrated **D**esign **E**ditor and programming examples.
Inkscape: ....www.inkscape.org: ....**S**calable **V**ector **G**raphics editor.
Code Blocks: www.codeblocks.org . **I**ntegrated **D**esign **E**ditor for multiple programming languages.
Open Office:. www.openoffice.org: . Office suite (Word processor, Spreadsheet, Database, etc.).

37 in 1 Kit - Overview

When you first receive the kit, it should be arranged in the following order in its plastic case.
#1 should be on the top left, closest to the hinged edge, #36&37 should be on the bottom right.
Note! The table below has been compressed to fit the page, the actual container is wider.

#1 	#7 	#13 	#19 	#25 (×2) 	#31 
#2 	#8 	#14 	#20 	#26 	#32 
#3 	#9 	#15 	#21 	#27 	#33 
#4 	#10 	#16 	#22 	#28 	#34 
#5 	#11 	#17 	#23 	#29 	#35 
#6 	#12 	#18 	#24 	#30 	#36  #37 

Modules shown at 1/2 (50%) scale. Click on the image to navigate to its page within this reference.

37 in 1 Kit - Overview…

The “37 in 1 Kit” is a collection of modules specifically designed for Arduino® development boards. It contains "input" modules which measure real world properties such as temperature, light, sound, etc. to electronic outputs, which are converted by the Arduino board to numeric values. There are also "output" modules such as **L**ight **E**mitting **D**iodes, buzzers, etc., which produce a physical result when they receive an electronic signal from an Arduino board. Learning how to use these modules will provide a good background in the basic concepts of electronics and computer programming suitable for careers in robotics and related industries.

This reference focusses on the basics of how each module is connected to an Arduino board, and how they are accessed via programming code. The real fun begins when multiple modules are used together, and with 37 modules, the possibilities are almost endless.

Note! While some of the modules appear to perform the same task, they use different types of components and / or electronic circuitry to achieve their results.

Modules by Category

Within this reference the modules have been arranged by their category, as this is how they appear when the companion sketch files "37_Kit_Sketches.zip" are installed into the Arduino **I**ntegrated **D**evelopment **E**nvironment.

The lists below describe in brief what each module is for, and how they can be distinguished between other modules with the same physical appearance.

Audio

- #11: Active Buzzer Module: An enclosed 12 millimetre [mm] diameter polarised buzzer to generate sound. *The active buzzer is taller than the passive buzzer.*
- #12: Passive Buzzer Module: An enclosed 11.8 mm diameter, 16 ohm [Ω] speaker. Maximum current through the speaker is 25 milliamperes. Use **P**ulse **W**idth **M**odulation to generate sound. *The passive buzzer is shorter than the active buzzer.*
- #4: Small Microphone Module: A 6 mm diameter electret (electrostatic magnet) microphone mounted parallel to the **P**rinted **C**ircuit **B**oard. It produces an analogue output signal based on the volume of the sound, and a digital output signal when a threshold level is reached. The sensitivity can be adjusted via the on-board multi-turn potentiometer. *The small microphone is less sensitive than the large microphone.*
- #3: Large Microphone Module: A 9.7 mm diameter electret microphone mounted perpendicular to the **PCB**. It produces an analogue output signal based on the volume of the sound, and a digital output signal when a threshold level is reached. The sensitivity can be adjusted via the on-board multi-turn potentiometer.

Electromagnetic

- #30: Analogue Hall Effect Sensor Module: A magnetic sensor **I**ntegrated **C**ircuit which senses magnetic fields close to the face of the **IC** mounted at the top of the module. It outputs a digital signal when a magnetic field has been detected.
- #8: Linear Hall Effect Sensor Module: A magnetic sensor **IC** which senses the polarity and strength of magnetic fields close to the face of the IC which is mounted at the top of the module. The sensitivity can be adjusted via the on-board multi-turn potentiometer. It outputs an analogue signal based on the intensity and polarity of the magnetic field, and a digital signal when a preset level has been reached.
- #31: Magnetic Sensor Module: An active magnetic sensor IC which senses the polarity and strength of magnetic fields close to the face of the IC which is mounted at the top of the module. It outputs a digital signal based on the intensity and polarity of the magnetic field. It includes an LED to display when a magnetic field is present.
- #18: Mini Reed Switch Module: A **N**ormally **O**pen reed switch which will close its contacts when a magnet is brought close to the side of its glass case. The reed switch is connected to the 2 outer pins, and a 10 kiloohm resistor is included which connects the centre pin to the signal "S" pin for use as a pull up / pull down resistor.
- #17: Reed Switch Module: A **N**ormally **O**pen reed switch which will close its contacts when a magnet is brought close to the side of its glass case. It provides both analogue and digital outputs, with the multi-turn pot able to be used as a pull up / pull down resistor.
- #2: SPDT Relay Module: A **S**ingle **P**ole **D**ouble **T**hrow relay capable of switching loads of up to 10 amperes [A] of current (up to 250 **V**olts **A**lternating **C**urrent, or 30 **V**olts **D**irect **C**urrent). The state of the relay is controlled using a 5 **VDC** signal.

Environment [4]

- #33: Analogue Temperature Sensor Module: A **N**egative **T**emperature **C**oefficient thermistor which converts temperature to electrical resistance. The 2 outer pins connect directly to the **NTC** thermistor, and the centre pin connects to the "S" pin for use as a pull up / pull down resistor.

- #10: Digital Temperature Sensor Module: A **N**egative **T**emperature **C**oefficient thermistor mounted at the top of the **PCB** which converts temperature to resistance. This is converted to voltage and output as an analogue signal proportional to its temperature. The sensitivity can be adjusted via the on-board multi-turn potentiometer. A digital signal is also available which indicates when the temperature has reached the preset level.
- #32: I²C Digital Temperature Sensor Module: A module containing a Dallas Semiconductor temperature sensor **I**ntegrated **C**ircuit which communicates the temperature at the **IC** via the I²C (Inter Integrated Circuit) protocol.
- #29: DHT11 Temperature and Humidity Sensor Module: A module containing a **D**igital **H**umidity and **T**emperature (Type **11**) sensor which communicates the temperature and humidity via the I²C protocol.

Interface

- #1: Dual Axis Analogue Joystick Module: A spring loaded 2 axis joystick which returns analogue values based on its position in the horizontal and vertical planes, and also includes a switch which activates when the joystick shaft is pressed downwards (in-line with the shaft).
- #22: Push Button Switch Module: A PCB mounted 6 mm normally open push button switch which connects the two outer pins when pressed. A 10 kiloohm [kΩ] resistor is included on-board, which connects the centre pin to the signal "S" pin for use as a pull up / pull down resistor.
- #24: Rotary Encoder Module: A rotary shaft which opens and closes contacts within its base as it is rotates. By calculating the sequence of these contacts, the direction and rotation angle can be calculated. It also includes a normally open switch which is activated by pressing downwards in-line with the shaft.
- #9: Touch Sensor Module: A touch wire connected to a transistor senses when it is in contact with skin. The sensitivity can be adjusted via the on-board multi-turn potentiometer. It outputs an analogue signal based on the skin conductivity / pressure, and a digital signal when a preset level has been reached.

Optical

- #16: 2 Colour 3 mm THT LED Module: A 3 mm **T**hrough **H**ole **T**ype opaque common cathode **L**ight **E**mitting **D**iode with 2 individually controllable red and green colour elements.
- #15: 2 Colour 5 mm THT LED Module: A 5 mm **THT** opaque common cathode **LED** with 2 individually controllable red and green colour elements.
- #13: 3 Colour 5mm THT LED Module: A 5 mm **THT** waterclear **LED** with 3 individually controllable red, green, and blue colour elements.
- #14: 3 Colour SMD LED Module: A 5 mm **S**urface **M**ount **D**evice **LED** with 3 individually controllable red, green, and blue colour elements.
- #20: 7 Colour 5mm THT Flashing LED Module: A 5 mm **THT** waterclear **LED** which cycles through 7 colours at various intervals.
- #7: Flame Sensor Module: A sensor in an **LED** package which senses light in the spectral range of naked flames. The sensitivity can be adjusted via the on-board multi-turn potentiometer. It outputs an analogue signal based on the intensity of the emissions, and a digital signal when a preset level has been reached.
- #19: Heartbeat Sensor Module: A 5 mm **THT** opaque infrared **LED** and photo-transistor pair, which monitor the opacity of a human finger tip placed between them. As blood is pumped through the fingertip the opacity changes, which is output as an analogue output signal.
- #5: Infrared Line Tracking Module: A short distance (up to 10 mm) infrared reflection detector (transmitter / receiver pair) mounted parallel to the **PCB**, which can have its sensitivity adjusted via the on-board multi-turn potentiometer. It is able to distinguish between light and dark objects such as a thick dark line drawn on a light surface. It outputs an analogue signal proportional to the infra-red reflection level of an object, and a digital signal when a preset level has been reached.
- #6: Infrared Obstacle Avoidance Module: A medium distance (up to 70 mm) infrared reflection detector (transmitter / receiver pair) mounted perpendicular to the **PCB** which can have its sensitivity adjusted via the on-board multi-turn potentiometer. It outputs an analogue signal proportional to the amount of infrared reflection from an object. The module can be set to sense continually ("EN" enable jumper fitted), or manually, by sending a logic high signal to the enable pin.
- #35: Infrared Receiver Module: An infrared receiver which decodes pulses of infra-red light for short range communications at a frequency of 38 kilohertz [kHz] (e.g. television, DVD player remote controls).
- #34: Infrared Transmitter Module: A 5 mm **THT** infrared **LED** suitable for communicating with devices (televisions, DVD players, etc.) which support common infrared protocols.

- #21: LASER Module: A **LASER** (**L**ight **A**mplification by **S**timulated **E**mission of **R**adiation) module which produces a polarised beam of light in the 650 nanometre [nm] (red) colour range. *Warning! While the output is less than 1 milliwatt, this can still cause permanent injury if shone into the eyes.*
- #28: Light Dependent Resistor Module: A **L**ight **D**ependent **R**esistor (photoresistor) which varies in resistance proportional to the amount of light received at the face of the **LDR**. The 2 outer pins connect directly to the **LDR**, and the centre pin connects to the "S" pin for use as a pull up / pull down resistor.
- #36: Photo-Interrupter Module: An integrated infrared **LED** / photo-transistor pair with a small air gap which detects when the **LED**'s light is obstructed. The output is via a digital signal at the "S" pin.

Other:

- #25: Magic Light Cup Module: A pair of modules containing a 5 mm **THT** red **LED**, and a tilt switch. Each module is placed in the bottom of a clear glass with one glass being held vertically, and the other being held horizontally. When the position of the glasses is reversed, it creates the effect of light pouring from one glass into the other.

Physical (Movement):

- #27: Ball Switch Module: A switch containing an enclosed metal ball which is free to move along its non conducting enclosure. When the ball is at either end, it bridges two contacts to produce a short circuit. The 2 outer pins connect directly to the ball switch, and the centre pin connects to the "S" pin for use as a pull up / pull down resistor.
- #37: Hit Sensor Module: A horizontally mounted spring enclosed in an opaque rectangular container, with one end of the spring free to vibrate and touch a contact producing a momentary short circuit. Sensitive to being bumped perpendicular to the spring.
- #23: Shock Sensor Module: A small spring within a cylindrical case which is fixed at one end. Any sudden movement to the side of the case will cause the unmounted end of the spring to make contact with the case and create a momentary short circuit. The 2 outer pins connect directly to the shock switch, and the centre pin connects to the (S) pin for use as a pull up / pull down resistor.
- #26: Tilt Switch Module: A heatshrink covered metal cylinder containing 2 small metal balls which are free to move along the length of the cylinder. When the ball is at the end where the connecting leads exit the cylinder, it bridges the ends of these leads and closes the circuit. The on-board **L**ight **E**mitting **D**iode illuminates when the internal contacts are bridged.

Quick Start

To get a module working, all of the required sketches (the term for Arduino programming source code files) are contained within the zip archive "37_Kit_Sketches.zip" which accompanies this reference. These sketches include the suggested electrical connections for the module, and also include comments explaining each line of code. This zip archive also includes additional projects which perform more rigorous tests, as well as "real world" projects which combine multiple modules. Copy the archive to your default sketch folder, then unzip it. Once the files have been extracted, a new sub-folder is created which contains all of the sketches listed by their category. The original archive file can then be moved to another location so you will always have a backup.

Software (Code) Examples

The example code within this reference is the bare minimum required to test the basic functionality of the module using an Arduino UNO (or compatible) board connected to a **P**ersonal **C**omputer running the Arduino **I**ntegrated **D**evelopment **E**nvironment. Other Arduino boards can be used as long as their inputs / outputs are 5 VDC (e.g Leonardo, Mega, Nano), although some pins may need to be changed, as not all boards use the same pins for **P**ulse **W**idth **M**odulation and interrupts.

The code samples (sketches) within the archive file include comments for every line, however, these comments have been removed from the examples in the book to reduce the number of pages.

Each of the projects have been tested using an UNO r3 development board, with the electrical and the physical properties measured using a a variety of purpose built digital meters.

Additional Downloadable Content

Available from the same location as this reference.

- **37_Kit_Sketches.zip**: All projects from this reference as well as additional "real world" projects.
- **37_Kit_Libraries.zip**: Libraries required to access some of the advanced features of the modules.

37 in 1 Kit - Connection Table

Module	Pin #1	Pin #2	Pin #3	Pin #4	Pin #5
Active Buzzer	S < D2	middle NC	- > GND		
Passive Buzzer	S < D2	middle NC	- > GND		
Small Microphone	AO > A0	G > GND	+ < 5V	DO > D2	
Large Microphone	AO > A0	G > GND	+ < 5V	DO > D2	
Analogue Hall Effect	- > GND	middle < 5V	S > A0		
Linear Hall Effect	AO > A0	G > GND	+ < 5V	DO > D2	
Magnetic Sensor	- > GND	middle < 5V	S > D2		
Mini Reed Switch	S > D2	middle NC	- > GND		
Reed Switch	AO > A0	G > GND	+ < 5V	DO > D2	
SPDT Relay	S < D2	+ < 5V	- > GND		
Analogue Temperature Sensor	S > A0	middle < 5V	- > GND		
Digital Temperature Sensor	AO > A0	G > GND	+ < 5V	DO < D2	
I²C Digital Temperature Sensor	- > GND	middle < 5V	S > D2		
DHT 11 Sensor	S < D2	middle < 5V	- > GND		
Dual Axis Analogue Joystick	GND > GND	+5V < 5V	VRx > A0	VRy > A1	SW > D2
Push Button Switch	S > D2	middle NC	- > GND		
Rotary Encoder	GND > GND	+ < 5V	SW > D4	DT > D3	CLK > D2
Touch Sensor	AO > A0	G > GND	+ < 5V	DO > D2	
2 Colour 3 mm THT LED	- > GND	middle < 150Ω < D3	S < 150Ω < D2		
2 Colour 5 mm THT LED	- > GND	middle < 150Ω < D3	S < 150Ω < D2		
3 Colour 5 mm THT LED	R < D4	G < D3	B < D2	- > GND	
3 Colour 5mm SMD LED	B< 110Ω < D4	R< 180Ω < D3	G< 110Ω < D2	- > GND	
7 Colour THT Flashing LED	S < D2	middle NC	- > GND		
Flame	AO > A0	G > GND	+ < 5V	DO > D2	
Heartbeat	S > A0	middle < 5V	- > GND		
Infrared Line Tracking	G > GND	V+ < 5V	S > D2		
Infrared Obstacle Avoidance	GND > GND	+ < 5V	OUT > D2	EN < NC	
Infrared Receiver	- > GND	middle < 5V	S > D2		
Infrared Transmitter	S < D3	middle NC	- > GND		
LASER	S < D2	middle NC	- > GND		
Light Dependent Resistor	S > A0	middle < 5V	- > GND		
Photo-Interrupter	- > GND	middle < 5V	S > D2		
Magic Light Cup	G > GND	+ < 5V	S > D4, D7	L < P5, P6	
Ball Switch-Ball	S > D2	middle NC	- > GND		
Hit Sensor	S > D2	middle NC	- > GND		
Shock Sensor	S > D2	middle NC	- > GND		
Tilt Switch	- > GND	middle < 5V	S > D2		

Legend

<: data / power from Arduino to module, >: data / power from module to Arduino. a: any analogue pin, d: any digital pin, "p" any pin capable of Pulse Width Modulation, "V" / "+" = 5V (5 VDC), "G" / "-" = "GND" (Ground), "R" = AREF (Analogue Reference). Some modules do not have a label marked on the **PCB**, and they are listed above by their position. e.g. "middle". NC: No Connection.

Conventions

Within this reference the following conventions have been used so that all general text, programming code, and electrical diagrams share a consistent theme. Where possible, any existing standards have been used, and any modifications to those standards are outlined within this section.

General Documentation

Electronic devices includes physical and electrical specifications, performance data, CAD dimensional drawings, connection tables, and sample programming code. The performance section has been measured by the author using appropriate test equipment, and where possible the specifications from any datasheet, are also included.

Within technical documentation there are many acronyms, abbreviations and shortened forms used. Any acronym, abbreviation, shortened form, and measurement unit has been written in its full form the first time it appears within a section. To help distinguish between them, the following text styles / colours have been used;

- Acronyms, which are created from the first letters of each word, are in **bold blue text**. e.g. **L**ight **E**mitting **D**iode, **V**olts **D**irect **C**urrent.
- Abbreviations, which are created from one or more sequential characters of each word are in **bold green text**. e.g. **pot**entiometer **pot**,
- Shortened forms, which use any characters from the word(s), and may also include characters not within the words are in bold brown text with the full form in brackets e.g. electret (electrostatic magnet), trimpot (trimming potentiometer).
- Measurement units and symbols are in purple text. The first time the unit appears, the unit symbol will be adjacent to the full name of the measurement unit in square brackets. e.g. millimetres [mm], ohms [Ω].

Coding Conventions

Comments: Are placed in the line above the code they refer to.

Code Formatting (Syntax Highlighting): The colour and styles for code within this document are based on those used by the Arduino IDE, with some additional styles to help differentiate between harder to see symbols.

Table a: Code Formatting Conventions

Code Element	Colour	Style	Case	Background	Sample
Assignment Operator	Blue	-	-	Yellow	`I = 1`
† Comment	Grey	Italic	-	-	`// Comment`
Comparison Operators	Purple	Bold	-	-	`if( i == 1 )`
Constants	Blue	Bold	Upper ‡	-	**iTEST**
Data Type	Cyan	Bold	-	-	**void**
† Keyword 1	Orange	Bold	-	-	**Serial**
† Keyword 2	Orange	-	Camel	-	println
† Keyword 3	Green	-		-	#include
† Literal1	Cyan	-		-	**HIGH**
Numbers	Magenta	-		-	1234567890
Strings	Blue	Italic		-	*"ABCD"*
Structure	Cyan	-		-	void
Symbols	Bold	Bold		-	**( ) { } + - . ;**
Variables	Blue	-	Camel	-	iTest

† Indicates styles which are used by the Arduino IDE, ‡ With the exception of the lower case prefix(es) which defines the type of data being stored.

Variable / Constant / Function Naming

Variables and constants have a prefix in lower case which relates to the type of data which they contain. The remainder of the characters in the name are in "camel case", which is derived from the description of the variable / constant. e.g. a floating point variable which stores an analogue value from a temperature sensor = "`fAnalogueTemperature`".

"Camel Case" is where the first character each word is capitalised, and all spaces between words are removed.

Conventions…

Two special cases of prefix are used as modifiers. "a" is used to denote an array, with the next part of the prefix specifying the data type for each array element. e.g. asName would be an array of strings. "u" is used in conjunction with numeric data types to specify that they are unsigned, e.g. ucNumeric would be an unsigned char. Where these prefixes are used together, the "a" array prefix is specified first. e.g. aucCharacter.
Other prefixes used; b: Boolean, c: char, f: floating point, i: integer, l: long, p: pin, r: real, s: string.

Arduino Pin Conventions

Text / background colours are used to show the type of connection. Where direct references to the Atmel processors are used, the background of the text is as per the Atmel documentation.

Table b: Colour Code Conventions

Electronic Colour Codes		Arduino Pin Code Colours		Atmel Pin Code Colours	
Colour	**Value (Multiplier) [Tolerance]**	**Colour**	**Title**	**Colour†**	**Title**
Black	0 (× 1)	Black	Ground	Orange	Power
Brown	1 (× 10) [1%]	Red	5 VDC Power	Black	Ground
Red	2 (× 100) [2 %]	Orange	3.3 VDC Power	Light Blue	Digital
Orange	3 (× 1000)	Green	Analogue	Green	Analogue
Yellow	4 (× 10 000)	Blue	Digital	Dark Blue	Programming
Green	5 (× 100 000)	Grey	Digital (PWM)	Dark Grey	Crystal Oscillator
Blue	6 (× 1 000 000)	Magenta	Analogue Ref	Cream	Regulated Output
Violet	7	Silver	No Connection	Grey	Reset
Grey	8			Mauve	Port ‡
White	9			Light Green	PC Interrupt ‡
Silver	(× 0.01) [± 10%]			Pink	Serial ‡
Gold	(× 0.1) [± 5%]			Silver	Function ‡

† Approximate names, with colours are from Atmel documentation. ‡ Not Atmel standard, assigned by author.

Breadboard Wiring

As there are only 10 colours of jumper wires commonly available, and, with 3 already being reserved for 5 VDC (red) , 3.3 VDC (orange), and ground (black), more complex circuits will have to reuse some colours. *To increase the number of unique jumper wire colours, lighter coloured jumper wires can have a line drawn along their length using a darker colour permanent marker.*
Where possible jumper wire colours are assigned in the following order.

- Digital Connections: Brown, Violet, Grey, White
- Analogue Connections: Blue, Yellow, Green

Where Serial Peripheral Interface is in use, SDA (Serial Data) is assigned Blue,, and SCL (Serial Clock) is assigned Yellow.

CAD Drawings

The CAD dimensional drawings provide dimensions for holes to mount the component, and the physical space the module requires to be enclosed, including the positions and size of any large parts.

The colours on the CAD drawings are the closest match to the real world component, which would still be clearly visible when printed in black and white. Lighter colours have been darkened, and any white silkscreen text or component labels have been blackened. All dimensions are in blue, with their text oriented in the same angle as the main dimension lines.

Most drawings are at full scale (1:1) so a printed copy of the page can be used to cut out a template. Confirm the size by checking any dimension on the printout is the same as the stated dimension.

Arduino Libraries

Some of the modules require that libraries are installed to access all of their features. Libraries are a powerful feature of a programming language which allow sections of code from external files to be used as if that code was contained within the current sketch file. They are written in C++ which is a superset of the Arduino sketch language, but they differ slightly in how they are stored within the file system. The C++ language splits each library into separate "header" and "code" files, with the "header" file(s) containing the (function / constant) declarations which will be used within the sketch files, and the "code" file(s) which contain the actual code which perform the actions. The benefit of using libraries is that they shield the Arduino programmer from complex low level code, and also considerably reduce the size of sketch files. Another benefit, is that when a library is updated to fix bugs, or, to add additional features, any sketch file which references the libraries will also be updated the next time it is opened and re-compiled.

All of the required libraries are contained within the zip archive "37_Kit_Libraries.zip" which accompanies this reference. Extract its contents to a suitable location within your computer system, then install the libraries using one of the procedures described below. If you choose procedure 3 or 4, the procedure will have to be performed for each of the 4 libraries.

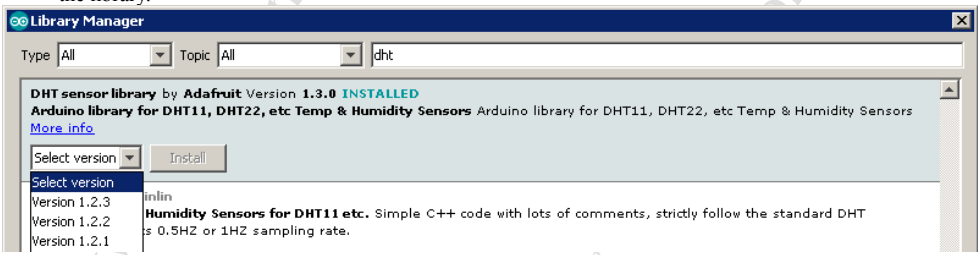
Inside the zip archive are separate 4 zip archives (one for each library):

- **DallasTemperature-3.7.6.zip**: Dallas Semiconductor Temperature Library. This is used to decode the temperature specific data of the I^2C Digital Temperature Module. (Also requires the "OneWire" library below).
- **DHT_sensor_library-1.3.0.zip**: Decodes the temperature and humidity data for the DHT11 Digital Humidity and Temperature Sensor module.
- **IRremote-2.0.1.zip:** Used to communicate between the Infrared Transmitter / Receiver modules, and also to communicate with other commercial devices which use infrared remote controls such as televisions, DVD players, and remote control toys.
- **OneWire-master.zip**: Used to communicate with devices which use the "One Wire" I^2C (Inter-Integrated Circuit) protocol such as the Dallas Semiconductor Temperature Integrated Circuit.

Installing Libraries

There are several ways of installing libraries within the Arduino Integrated Development Environment:

1. **Default Arduino Installation Folder:** Extract the contents of each individual zip archive to the "libraries" sub-folder of the Arduino IDE installation directory. *Note! As the Arduino installation program deletes any existing files within its installation directory, any libraries installed to this location will have to be re-installed if you update the Arduino IDE.*
2. **Default User Sketch Folder:** Extract the contents of each individual zip archive to the "libraries" sub-folder of the Arduino Sketchbook folder. The default User Sketchbook location can be located from the Preferences Window of the Arduino IDE (keyboard shortcut "Ctrl + ,"). Installing libraries to this folder has the advantage of being included within your own documents folder when backing up your files.
3. **Using the "Add Zip Library function of the Arduino IDE**. From the "Sketch" menu, select the "Include Library", sub-menu, then "Add .ZIP Library". An "Open File" window will be displayed, where you can navigate to the location of the library's zip archive file. Select the library zip archive file, then press "Open", to install the library.
4. **Using the "Library Manager" window of the Arduino IDE** *Note! An Internet connection is required.* From the "Tools" menu select "Manage Libraries", or use the keyboard shortcut "Ctrl + Shift + I". Enter the name of the library in the edit control at the right hand side, and any matching libraries will be displayed. Using the "drop down" control, select the latest version of the library, then click the install button to install the library.



*Screen shot of the "Arduino **IDE** Library Manager" displaying libraries matching "dht".*

Active Buzzer Module

This module contains an enclosed 12 millimetre [mm] diameter polarised piezoelectric buzzer with an output audio frequency of approximately 2.5 kilohertz [kHz]. The output amplitude (volume) can be decreased by either using a resistor (see Performance below). The module may have a sticker covering the face of the buzzer. This is only to protect it during manufacture, and should be removed before use.

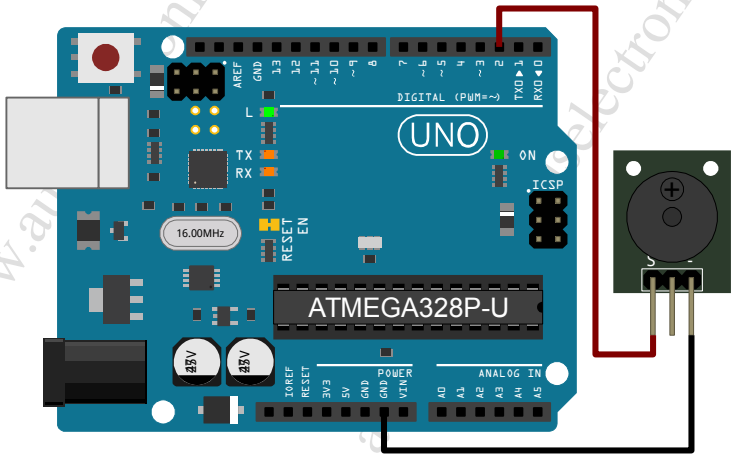
Note! The active buzzer is taller than the passive buzzer used on the Passive Buzzer Module.

***Important!** This buzzer module is capable of generating audio output levels which are loud enough to cause damage to human hearing. Keep the module at least 500 mm away from ears.*

Table 3: Active Buzzer Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Positive power to buzzer.
middle	NC		No connection.
-	GND	■	Ground connection for buzzer.

D2: can be any digital pin.



The sketch below can be used to control the active buzzer module via the Arduino Serial Monitor.

```
int pBuzzer = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pBuzzer, OUTPUT );
  Serial.println( "Enter 1 to turn on buzzer for 1 second." );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '1' ) {
      Serial.println( "Buzzer On" );
      digitalWrite( pBuzzer, HIGH );
      delay( 1000 );
      Serial.println( "Buzzer Off" );
      digitalWrite( pBuzzer, LOW );
    }
  }
}
```

Active Buzzer Module…

Module Specifications

PCB Dimensions (H × W × D): 25.5 × 15.6 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D): 25.5 × 15.6 × 12.9 mm
Weight: 2.80 grams [g]
Input Voltage: 3.3 to 5 **V**olts **D**irect **C**urrent

Buzzer / Module Performance

Current Draw (active): 23.7 milliamps [mA] @ 4.94 **VDC**
Current Draw (3.27 VDC): 15.6 mA @ 3.27 **VDC**

Table 4: Active Buzzer Decibels by Distance, Voltage and Resistors

Voltage:	**3.3 VDC**			**5.0 VDC**		
Distance:	**50** mm	**100** mm	**200** mm	**50** mm	**100** mm	**200** mm
Direct Connection 0 ohm [Ω]**:**	102	96	91	102	97	96
100 Ω**:**	98	92	88	102	98	93
220 Ω**:**	89	82	79	102	94	90
330 Ω**:**	80	78	74	90	87	79
560 Ω**:**	-	-	-	86	79	75

*Values in decibels [db] measured at various distances directly in-line with buzzer output (perpendicular to **P**rinted **C**ircuit **B**oard).*

Buzzer Specifications

Input Voltage: 3.5 ~ 5.5 **VDC**
Current: 30 mA @ 5 **VDC**
Frequency: 2.5 kHz ± 0.3 kHz
Minimum Output Level: 85 decibels [db] @ 100 millimetres [mm]
Working Temperature: -20 degrees Celsius [ºC] ~ 70 ºC

Module Mounting

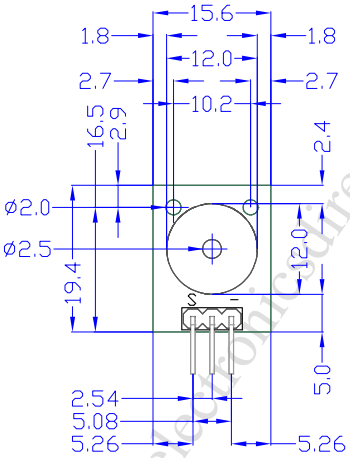
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

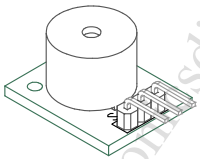
Projects

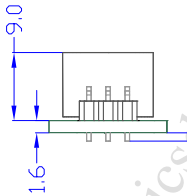
Folder: **Modules\Audio\Active_Buzzer**

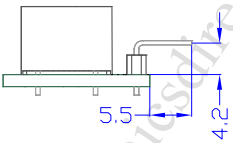
- **Active_Buzzer_SM**: Allows the buzzer to be controlled via the Arduino Serial Monitor.

Active Buzzer Module - Dimensions









Passive Buzzer Module

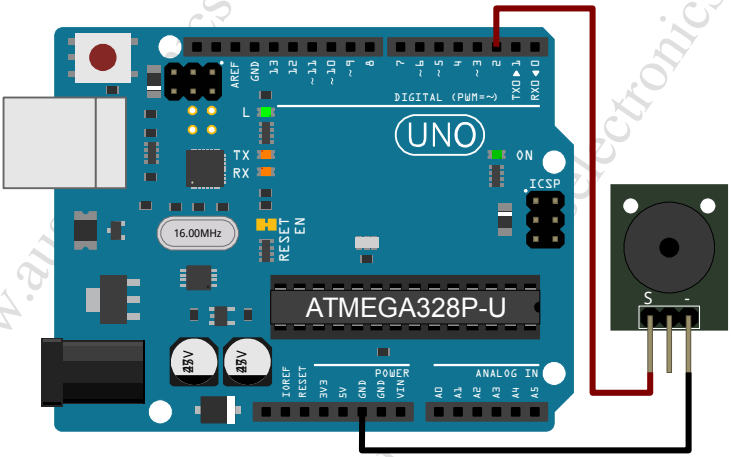
This module contains an enclosed 11.8 millimetre [mm] diameter, 16 ohm [Ω] speaker. Maximum current through the speaker is 25 milliamperes [mA]. **P**ulse **W**idth **M**odulation can be used to generate a variety of different sounds.

Note! The passive buzzer is shorter than the active buzzer used on the Active Buzzer Module.

Table 5: Passive Buzzer Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Signal to left side of speaker.
middle	NC		No connection.
-	GND	■	Signal to right side of speaker.

D2: can be any digital pin.



The sketch below can be used to control the passive buzzer module via the Arduino Serial Monitor.

```
int pBuzzer = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pBuzzer, OUTPUT );
  Serial.println( "Enter 1 to turn on buzzer for ~1 second." );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '1' ) {
      Serial.println( "Buzzer On" );
      digitalWrite( pBuzzer, HIGH );
      for ( int i = 0; i < 500; i++ ) {
        digitalWrite( pBuzzer, HIGH );
        delay( 1 );
        digitalWrite( pBuzzer, LOW );
        delay( 1 );
      }
      Serial.println( "Buzzer Off" );
    }
  }
}
```

Passive Buzzer Module…

Module Specifications

PCB Dimensions (H × W × D) :	18.9 × 15.5 × 1.6 mm
Enclosing Dimensions (H × W × D) :	25.0 × 15.5 × 11.1 mm
Weight:	2.44 grams [g]
Module Resistance:	15.2 ohms [Ω]

Module Performance

Current Draw:	37. milliamps @ 4.58 VDC (50% duty cycle)

Module Mounting

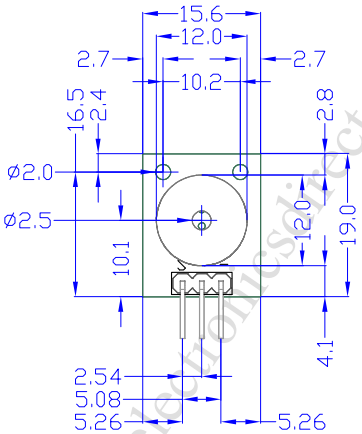
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the PCB, suitable spacers and insulation must be used.

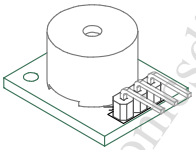
Projects

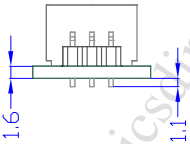
Folder: **Modules\Audio\Passive_Buzzer**

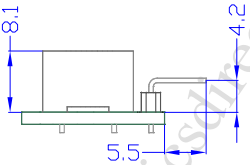
- **Passive_Buzzer_SM**: Allows the buzzer to be controlled via the Arduino Serial Monitor.
- **Passive_Buzzer_PWM_Sweep**: Generates a sweep of defined musical note frequencies using **PWM** to vary the duty cycle of the output signals. Frequencies are displayed in the Arduino Serial Monitor.

Passive Buzzer Module - Dimensions









Small Microphone Module

This module contains a 6 millimetre [mm] diameter electret (electrostatic magnet) microphone mounted parallel to the **P**rinted **C**ircuit **B**oard which converts sound waves received at the face of the microphone (black felt) to an analogue voltage. The module also includes a digital output which is triggered when the sound rises above a preset level. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns).

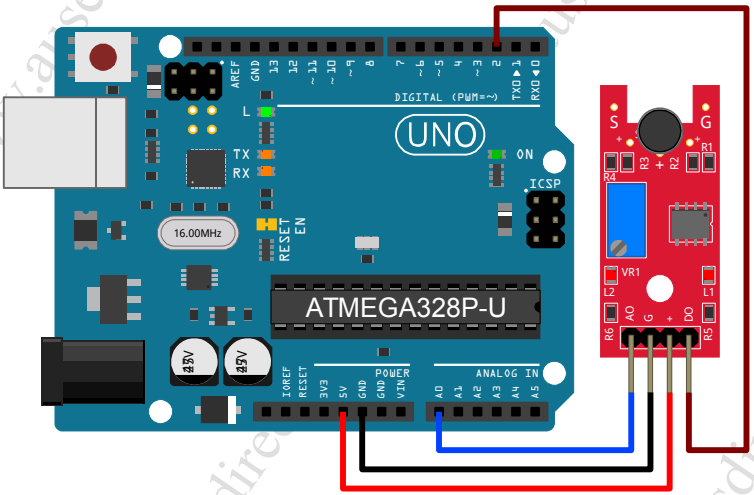
Arduino boards are able to manipulate the data from microphones to analyse both their frequencies and amplitude. Microphones can also be used as triggers to create sound controlled switches, and for monitoring sound levels for alarm systems. The microphone will need to be calibrated against a known sound level in order to display accurate decibel readings.

Note! Setting the sensitivity of the microphone also adjusts the trigger point for the digital output, so if these are required to be set at different levels, the trigger will have to be managed through programming.

Table 6: Small Microphone Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue output signal from the microphone.
G	GND		Ground connection.
+	5V		5 **V**olts **D**irect **C**urrent positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below can be used to display the results to the Serial Monitor / Plotter.

```
int pDigitalMic = 2;
int pAnalogueMic = A0;
void setup() {
  Serial.begin( 9600 );
  pinMode( pDigitalMic, INPUT );
}
void loop() {
  Serial.print( analogRead( pAnalogueMic ), DEC );
  Serial.print(", ");
  Serial.println(digitalRead( pDigitalMic ), DEC );
}
```

Small Microphone Module…

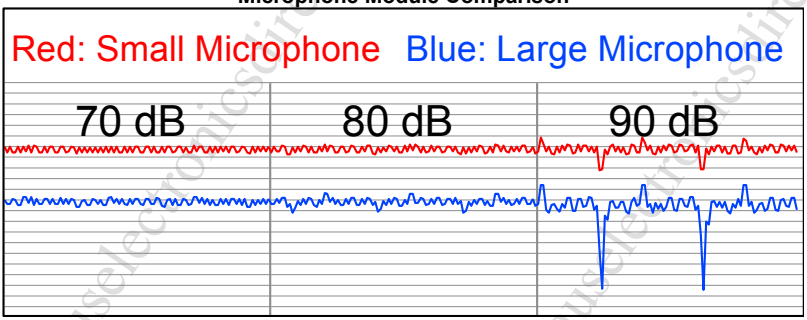
Module Specifications

PCB Dimensions (H × W × D):	15.5 × 19.5 × 1.6 mm
Enclosing Dimensions (H × W × D):	42.3 × 15.94 x 14.5 mm
Weight:	3.55 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (not triggered):	4.3 milliamps [mA] @ 5.02 **VDC**
Current Draw(triggered):	6.7 mA @ 5.01 **VDC**

Microphone Module Comparison



The image above is the oscilloscope trace of the output of both small and large microphones when they are placed 50 mm away from a speaker playing a triangular waveform. There is very little difference between the microphones at 70 dB, but at 80 dB, the output from the large microphone is around 1.5 times greater than the small microphone, and at 90 dB, the output of the large microphone is around 5 times greater.

Module Mounting

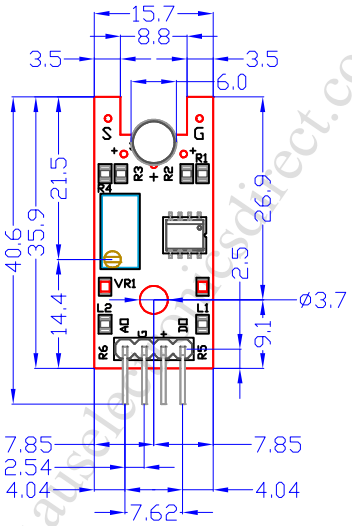
The module has a single 3.7 mm diameter mounting hole close to the centre of the **PCB**. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

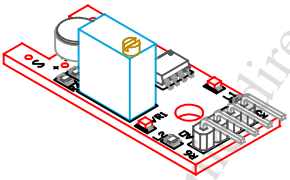
Projects

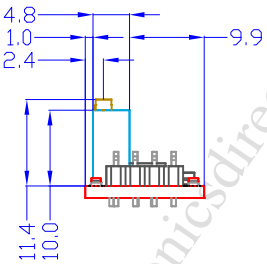
Folder: **Modules\Audio\Small_Microphone**

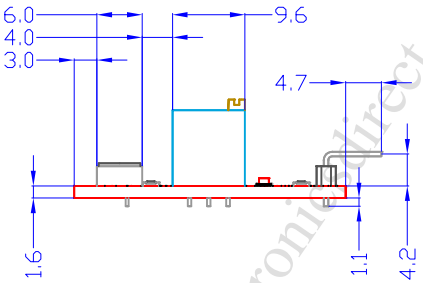
- **Small_Microphone_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Small Microphone Module - Dimensions









Large Microphone Module

This module contains a 9.7 millimetre [mm] diameter electret (electrostatic magnet) microphone mounted perpendicular to the **P**rinted **C**ircuit **B**oard which converts sound waves received at the face of the microphone (black felt) to an analogue voltage. The module also includes a digital output which is triggered when the sound rises above a preset level. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns).

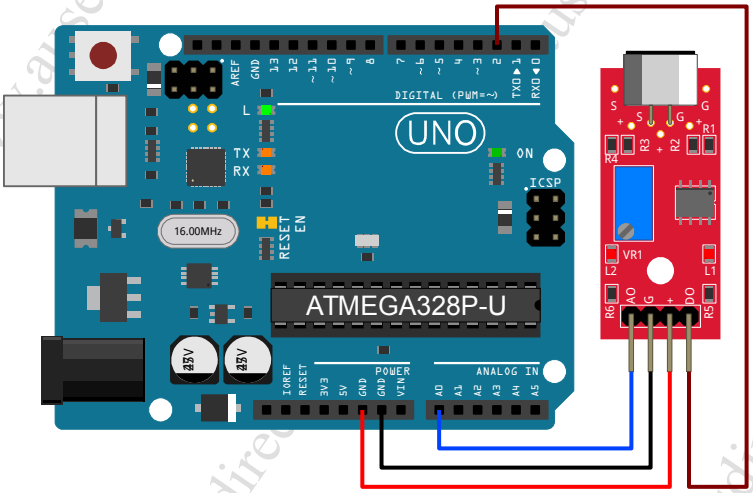
Arduino boards are able to manipulate the data from microphones to analyse both their frequencies and amplitude. Microphones can also be used as triggers to create sound controlled switches, and for monitoring sound levels for alarm systems. The microphone will need to be calibrated against a known sound level in order to display accurate decibel readings.

Note! Setting the sensitivity of the microphone also adjusts the trigger point for the digital output, so if these are required to be set at different levels, the trigger will have to be managed through programming.

Table 7: Large Microphone Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue output signal from the microphone.
G	GND		Ground connection.
+	5V		5 **V**olts **D**irect **C**urrent positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pDigitalMic = 2;
int pAnalogueMic = A0;
void setup() {
  Serial.begin( 9600 );
  pinMode( pDigitalMic, INPUT );
}
void loop() {
  Serial.print( analogRead( pAnalogueMic ), DEC );
  Serial.print(", ");
  Serial.println(digitalRead( pDigitalMic ), DEC );
}
```

Large Microphone Module…

Module Specifications

PCB Dimensions (H × W × D) :	35.9 × 15.9 × 1.6 mm
Enclosing Dimensions (H × W × D) :	42.3 × 15.9 x 14.5 mm
Weight:	3.56 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (not triggered):	4.3 milliamps [mA] @ 5.05 **VDC**
Current Draw(triggered):	6.7 mA @ 5.04 **VDC**

See the diagram within the Small Microphone section for a comparison of both microphone modules.

Module Mounting

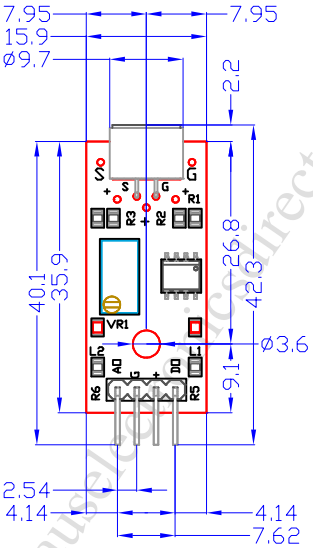
The module has a single 3.6 mm diameter mounting hole close to the centre of the **PCB**. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

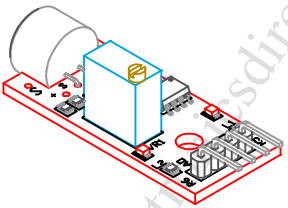
Projects

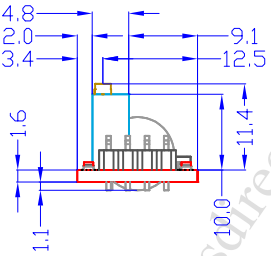
Folder: **Modules\Audio\Large_Microphone**

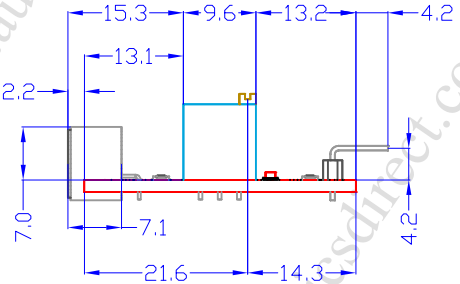
- **Large_Microphone_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Large Microphone Module - Dimensions









Analogue Hall Effect Sensor Module

This module contains a "49E, 825BG" bipolar linear Hall Effect **I**ntegrated **C**ircuit which senses magnetic fields close to the sensing face (the face containing label "49E 825BG").

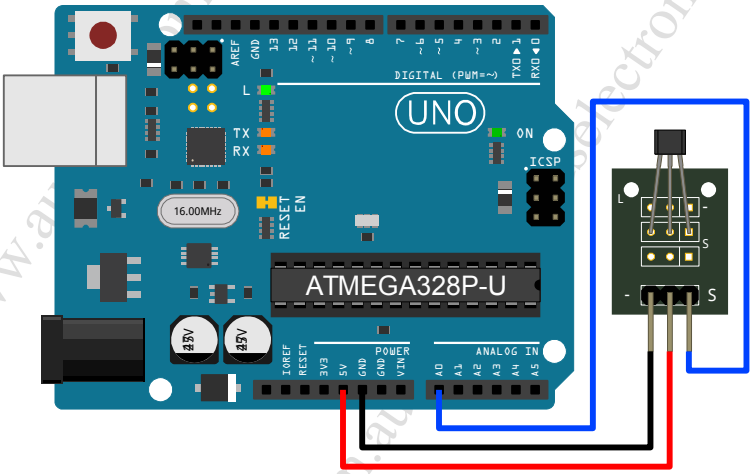
Hall Effect sensors are used to monitor the position of rotating mechanisms such as anemometers, fluid flow rate impellers, as well as monitoring the positions of objects which move in a linear path.

This module differs from the Linear Hall Effect Sensor Module: in that it has a single analogue output, and the sensitivity is not adjustable.

Table 8: Analogue Hall Effect Sensor Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	5 **V**olts **D**irect **C**urrent power supply to Hall Effect **I**ntegrated **C**ircuit.
S	A0	■	Digital output signal from Hall Effect **I**ntegrated **C**ircuit.

A0: can be any analogue pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pHallEffect = A0;
void setup() {
  Serial.begin(9600);
}
void loop() {
  Serial.println(analogRead(pHallEff), DEC);
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.1 × 15.5 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	29.4 × 15.5 x 7.3 mm
Weight:	1.26 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (idle):	5.80 milliamperes [mA]
Output Voltage (idle):	2.52 **VDC**
Current Draw (positive magnetic field):	5.75 mA
Output Voltage (positive magnetic field):	4.36 **VDC**

Current Draw (negative magnetic filed):	5.89 mA
Output Voltage (negative magnetic field):	0.82 **VDC**

49E 825BG Linear Hall Effect Sensor Specifications

Input Voltage (V_{CC}):	3 to 6.5 **VDC**
Output Current:	10 mA
Operating Temperature:	-40 to 85 degrees Celsius [ºC]

Module Mounting

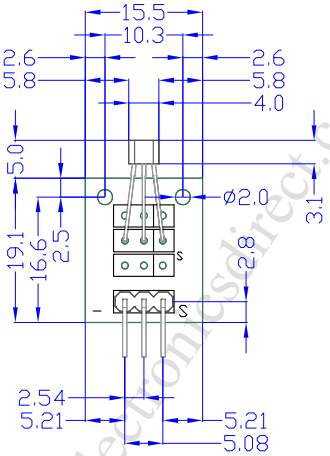
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

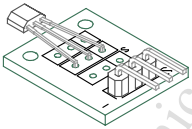
Projects

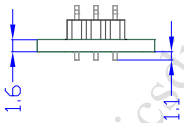
Folder: **Modules\Electromagnetic\Analogue_Hall_Effect_Sensor**

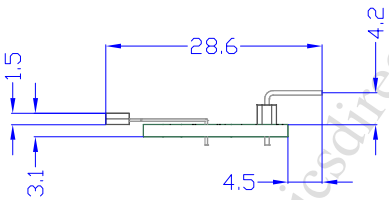
- **Analogue_Hall_Effect_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Analogue Hall Effect Module - Dimensions









Linear Hall Effect Sensor Module

This module contains a "49E, 825BG" bipolar linear Hall Effect **I**ntegrated **C**ircuit which senses magnetic fields close to the sensing face (the face containing label "49E 825BG"). It outputs an analogue signal based on the intensity and polarity of the magnetic flux, and a digital signal when a preset level has been reached. The sensitivity can be adjusted via the on-board multi-turn potentiometer. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns).

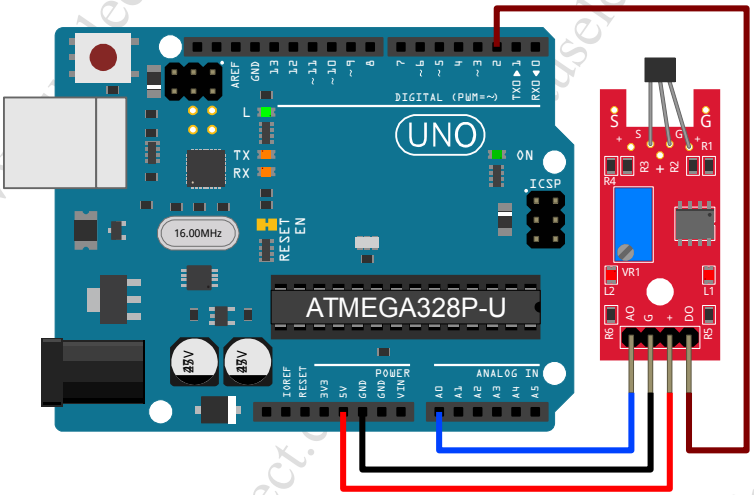
Hall Effect sensors are used to monitor the position of rotating mechanisms such as anemometers, fluid flow rate impellers, as well as monitoring the positions of objects which move in a linear path.

This module differs from the Analogue Hall Effect Sensor Module in that it has both analogue and digital outputs, as well as being able to have its sensitivity adjusted.

Table 9: Linear Hall Effect Sensor Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue signal output from the Hall Effect sensor.
G	GND		Ground connection.
+	5V		5 **V**olts **D**irect **C**urrent positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pDigitalHall = 2;
int pAnalogueHall = A0;
void setup () {
  Serial.begin( 9600 );
  pinMode ( pDigitalHall, INPUT );
}
void loop () {
  Serial.print( analogRead( pAnalogueHall ), DEC );
  Serial.print( ", " );
  Serial.println( digitalRead( pDigitalHall ), DEC );
}
```

Linear Hall Effect Sensor Module…

Module Specifications

PCB Dimensions (H × W × D):	35.9 × 16.0 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	44.9 ×16.0 × 14.1 mm
Weight:	2.81 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (triggered):	12.6 milliamps [mA]
Current Draw (not triggered):	10.1 mA

SS49E Specifications

Supply Voltage:	3 **VDC** (min) to 6.5 **VDC** (max)
Current:	4.2 mA (nominal), 8.0 mA (max)
Response Time:	3 microseconds [µs]
Sensitivity:	2.0 (min) to 3.0 (max) millivolts per gauss [mV/G]
Output Voltage (-1500 G) :	0.86 **VDC**
Output Voltage (+1500 G):	4.21 **VDC**

Module Mounting

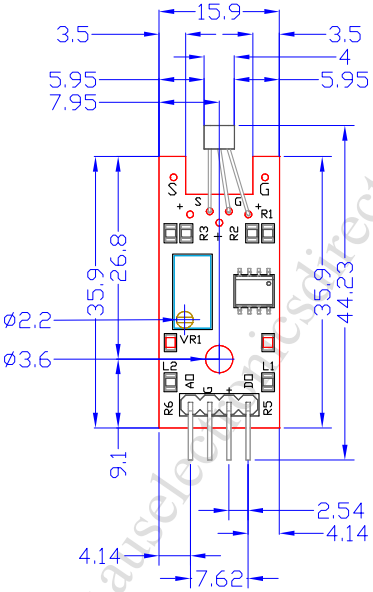
The module has a single 3.6 mm diameter hole close to the connection pin end of the **P**rinted **C**ircuit **B**oard. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

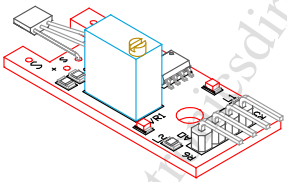
Projects

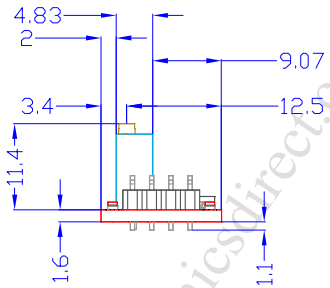
Folder: **\Modules\Electromagnetic\Linear_Hall_Effect_Sensor**

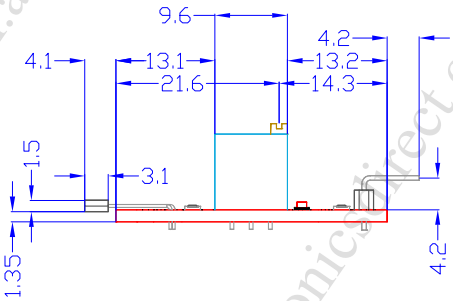
- **Linear_Hall_Effect_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Linear Hall Effect Sensor Module - Dimensions









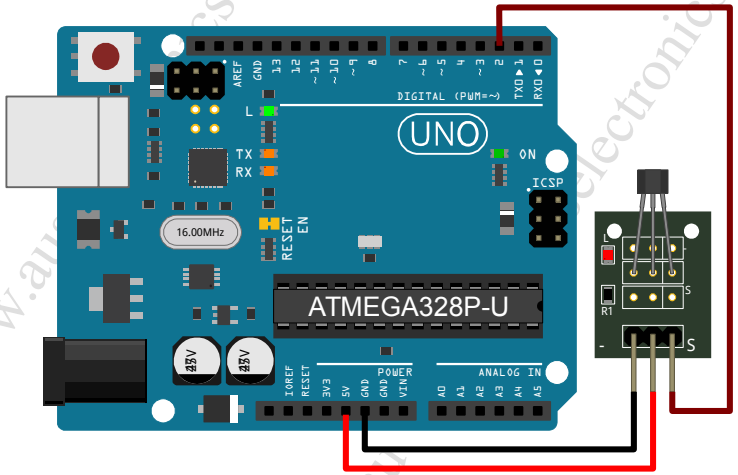
Magnetic Sensor Module

This module is uses a 3144 Hall Effect **I**ntegrated **C**ircuit which senses the presence of a magnetic field close to the face of the component. The sensing face is at the centre of the flat area of the IC containing the label "3144". When a magnetic field is present, it outputs a digital logic low signal, and the on-board red **S**urface **M**ount **D**evice **L**ight **E**mitting **D**iode illuminates.

Table 10: Magnetic Sensor Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	5 VDC power supply to the 3144 Hall Effect **IC** and on-board circuitry.
S	D2	■	Digital signal output from 3144 Hall Effect **IC**.

D2: can be any digital pin



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pAnalogueMagnetic = A0;
void setup () {
  Serial.begin( 9600 );
}
void loop () {
  Serial.println( analogRead( pAnalogueMagnetic ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.4 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	29.8 ×15.4 × 7.2 mm
Weight:	1.24 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (triggered):	3.0 milliamps [mA]
Current Draw (not triggered):	8.2 mA
Digital Output (triggered):	0.67 **VDC**
Digital Output (not triggered):	3..63 **VDC**

Magnetic Sensor Module…

3144 IC Specifications

Supply Voltage: 4.5 VDC (min) to 24 VDC (max)
Current Draw: 4.4 mA (nominal) to 9.0 mA (max)

Module Mounting

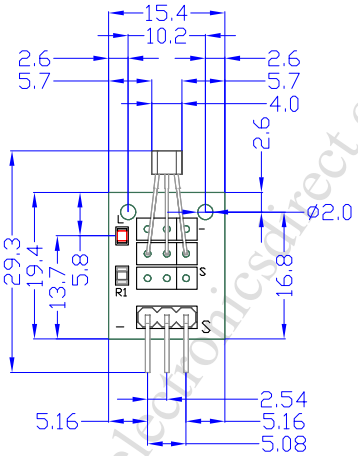
The module has 2 ×2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the Printed Circuit Board, suitable spacers and insulation must be used.

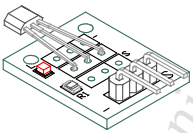
Projects

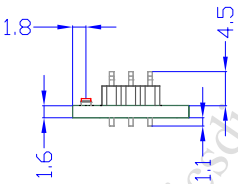
Folder: **\Modules\Electromagnetic\Magnetic_Sensor**

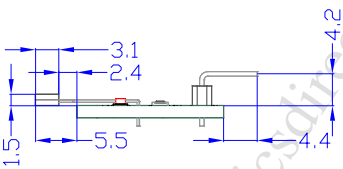
- **Magnetic_Sensor_SM**: Displays the state of the module via the Arduino Serial Monitor / Plotter.

Magnetic Sensor Module - Dimensions









Mini Reed Switch Module

This module contains a sealed **N**ormally **O**pen reed switch which will close its contacts when a magnetic field is brought close to the side of its glass case. The reed switch is connected to the 2 outer pins, and a 10 kiloohm [kΩ] resistor is included which connects the centre pin to the signal "S" pin for use as a pull up / pull down resistor.

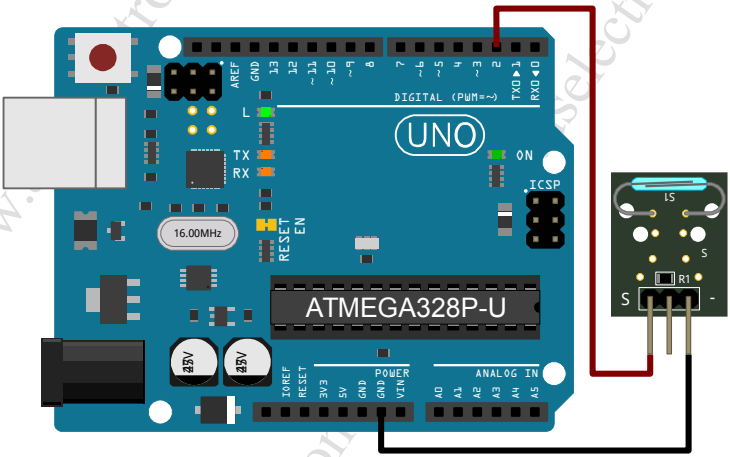
Reed switches are commonly used as limit or proximity switches where electrical monitoring of a moving object is required. They are basically a mechanical version of the "Hall Effect" sensor, both of which change their output when a magnetic field is present.

This module differs from the Reed Switch Module which contains additional electronic circuitry to set the sensitivity, as well as the Reed Switch Module outputting both analogue and digital signals.

Table 11: Mini Reed Switch Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Connects to left side of the reed switch.
middle	NC		Connects to left side of the reed switch via a 10 kiloohm [kΩ] resistor.
-	GND	■	Connects to right side of reed switch.

D2: can be any digital pin.



The sketch below displays the state of the reed switch to the Arduino Serial Monitor / Plotter.

```
int pDigitalReed = 2;
void setup () {
  Serial.begin( 9600 );
  pinMode ( pDigitalReed, INPUT );
  digitalWrite ( pDigitalReed, HIGH );
}
void loop () {
  Serial.println( digitalRead( pDigitalReed ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.1 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	24.5 × 15.4 × 7.3 mm
Weight:	1.16 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Mounting

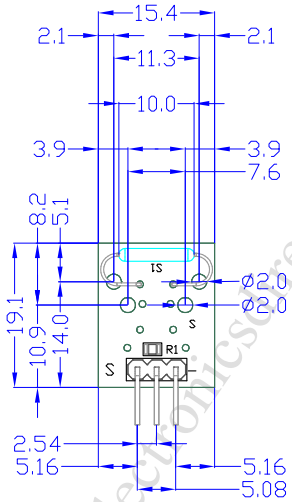
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

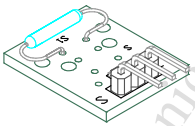
Projects

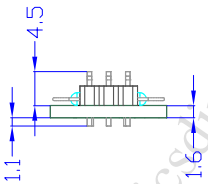
Folder: **\Modules\Electromagnetic\Mini_Reed_Switch**

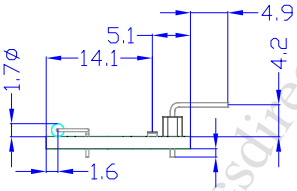
- **Mini_Reed_Switch_SM**: Displays the state of the mini reed switch on the Arduino Serial Monitor / Plotter.

Mini Reed Switch Module - Dimensions









Reed Switch Module

This module contains a sealed **N**ormally **O**pen reed switch which will close its contacts when a magnetic field is brought close to the side of its glass case. It provides both analogue and digital outputs, with the multi-turn pot able to adjust the sensitivity. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns).

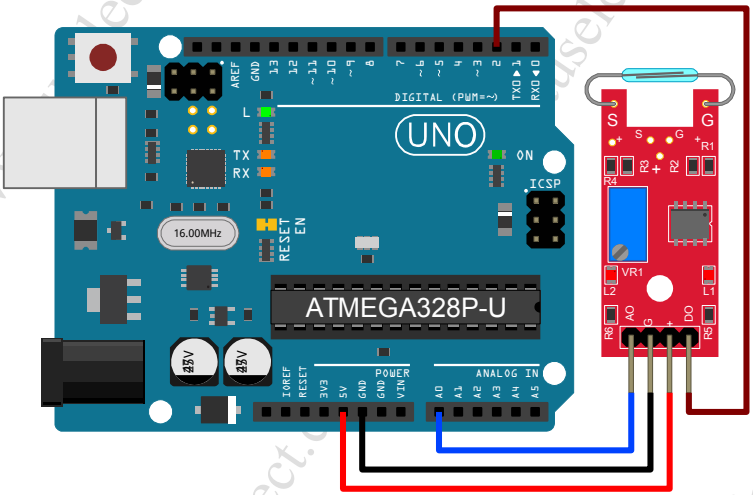
Reed switches are commonly used as limit or proximity switches where electrical monitoring of a moving object is required. They are basically a mechanical version of the "Hall Effect" sensor, both of which change their output when a magnetic field is present.

This module differs from the Mini Reed Switch Module as it contains additional electronic circuitry to set the sensitivity, as well as outputting both analogue and digital signals.

Table 12: Reed Switch Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue signal output from reed switch circuitry.
G	GND		Ground connection.
+	5V		5 **V**olts **D**irect **C**urrent positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pDigitalReed = 2;
int pAnalogueReed = A0;
void setup () {
  Serial.begin( 9600 );
  pinMode ( pDigitalReed, INPUT );
  digitalWrite ( pDigitalReed, HIGH );
}
void loop () {
  Serial.print( analogRead( pAnalogueReed ), DEC );
  Serial.print( ", " );
  Serial.println( digitalRead( pDigitalReed ), DEC );
}
```

Reed Switch Module…

Module Specifications

PCB Dimensions (H × W × D):	35.4 × 15.5 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	42.9 × 19.2 × 14.4 mm
Weight:	2.91 grams [g]
Input Voltage:	5 **VDC**

Module Performance

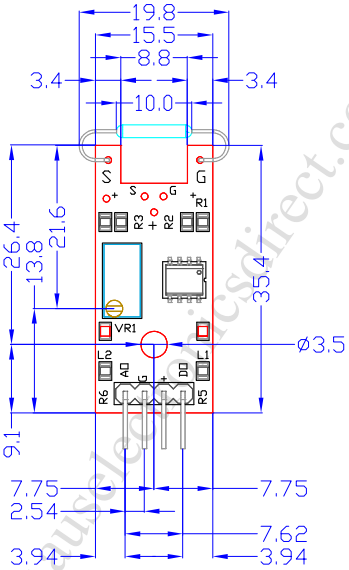
Current Draw (not triggered):	4.3 milliamps [mA] @ 5.04 **VDC**
Current Draw (triggered):	6.9 mA @ 5.03 **VDC**

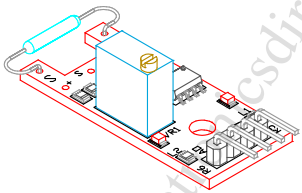
Projects

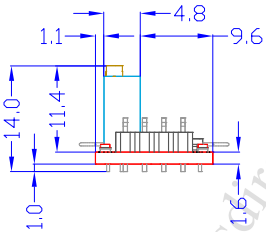
Folder: **Modules\Electromagnetic\Reed_Switch**\

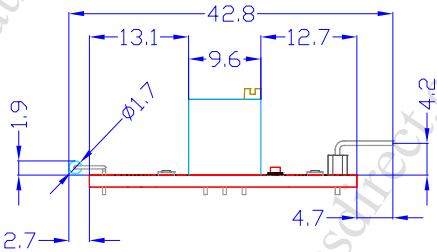
- **Reed_Switch_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Reed Switch Module - Dimensions









SPDT Relay Module

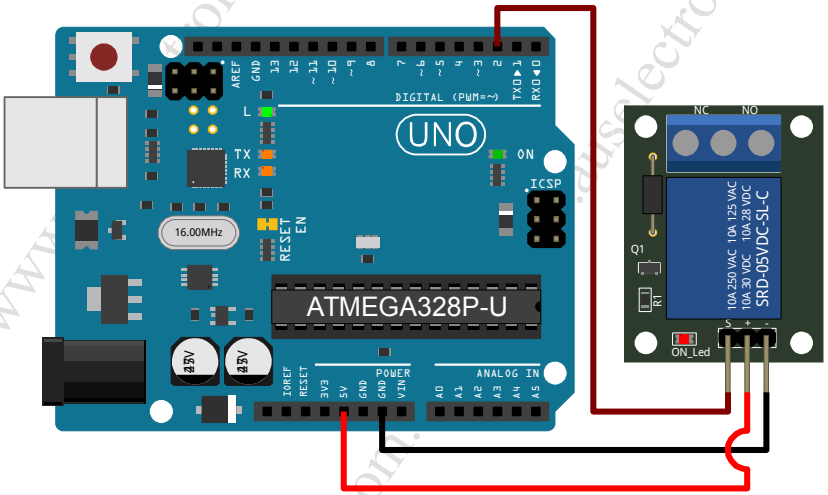
This module contains a Single Pole Double Throw relay capable of switching loads of up to 10 amps [A] (up to 250 Volts Alternating Current, or up to 30 Volts Direct Current). The state of the relay is controlled using a 5 VDC signal from a digital pin. The primary component is a "Songle SRD-05VDC-SL-C" (or equivalent), which is driven by a Surface Mount Device transistor. A diode is included to protect against reverse current spikes when the relay coil transits from the energised to the de-energised state.

The relay has both Normally Open, and Normally Closed contacts so it can be used to switch between two electrical paths of a circuit. A Light Emitting Diode is included which lights when the relay is the energised.

Table 13: SPDT Relay Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Signal for relay activation.
+	5V	■	5 VDC positive supply for board circuitry.
-	GND	■	Ground connection.

D2: can be any digital pin.



The sketch below can be used to control the Relay Module via the Arduino Serial Monitor.

```
int pRelay = 2;
void setup () {
  digitalWrite ( pRelay, LOW );
  Serial.begin( 9600 );
  while ( !Serial ) {
    ;
  }
  pinMode ( pRelay, OUTPUT );
  digitalWrite(pRelay, LOW);
  Serial.println( "0 to turn off, 1 to turn on" );
}
void loop () {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '0' ) {
      Serial.println( "Off" );
      digitalWrite ( pRelay, LOW );
    }
```

```
    if ( cInput == '1' ) {
      Serial.println( "On" );
      digitalWrite( pRelay, HIGH );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D):	33.9 × 26.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	38.5 × 26.4 × 19.6 mm
Weight:	14.51 grams [g]
Input Voltage:	5 **VDC**

Relay Manufacturers Specifications

Relay Model:	Songle SRD-05VDC-SL-C
Nominal Coil Voltage:	5 **VDC** Max (120% = 6 **VDC**)
Coil resistance:	70 ohm Ω
Coil Nominal Current:	71.4 milliamps [mA]
Structure:	Sealed
Coil Sensitivity:	0.36 watts [W]
Energise Time:	10 milliseconds [ms]
De-energise Time:	5 ms
Contact Resistance:	100 milliohm [mΩ] Max
Contact Life Expectancy:	10^7 (operation no load), 10^5 (at rated coil voltage)

Module Performance

Current Draw (Relay energised):	79 mA (relay coil + on-board circuitry)
Current Draw (Relay not energised):	0 mA

Module Mounting

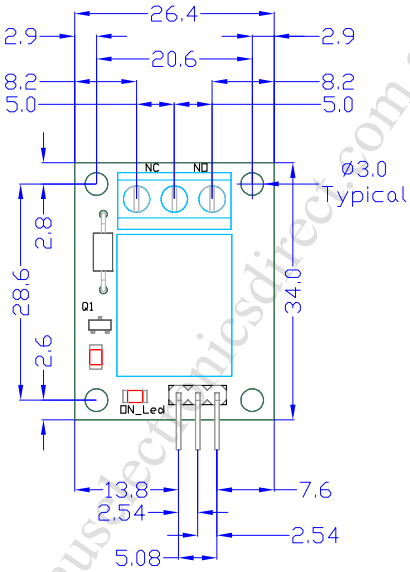
The module has 4 × 3 mm diameter holes at each corner of the **P**rinted **C**ircuit **B**oard. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

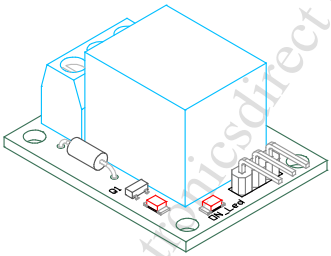
Projects

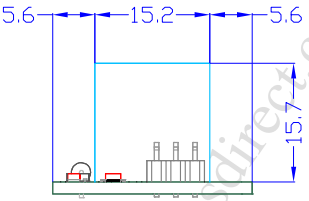
Folder: **\Modules\Electromagnetic\SPDT_Relay**

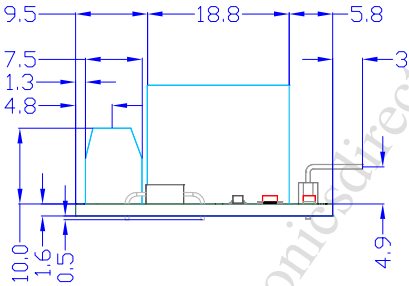
- **SPDT_Relay_SM**: Controls the relay by entering "0", or "1" in the Arduino Serial Monitor.
- **SPDT_Relay_Timed**: Toggles the state of the relay based on "time on" and "time off" intervals.

Relay Module - Dimensions









Analogue Temperature Sensor Module

This module contains a **N**egative **T**emperature **C**oefficient thermistor which converts temperature to electrical resistance. The value is converted to a temperature using the "Steinhart-Hart" equation below.

"Steinhart-Hart" Equation $$\frac{1}{T} = A + B \ln R + C (\ln R)^3$$

- ***T***: Temperature kelvins [K]
- ***R***: Resistance at "T" ohms [Ω]
- ***A***, ***B***, and ***C***: Steinhart-Hart coefficients.

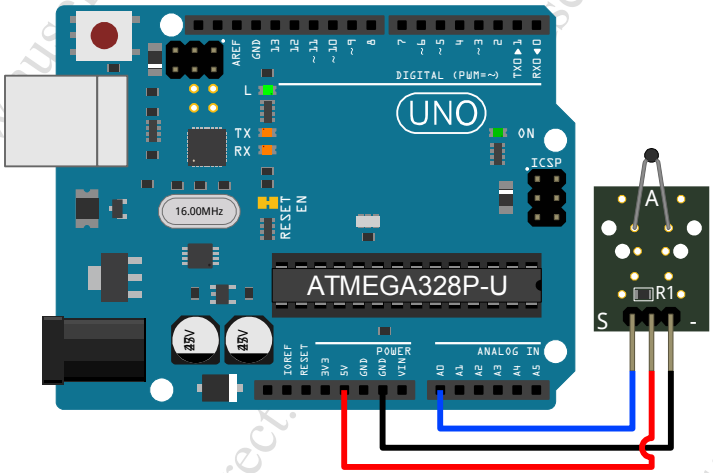
See Project code for usage of the "Steinhart-Hart" equation.

Because of their small size and low cost, thermistors are used in a large range of devices such as kitchen appliances, automotive sensors, and for process control automation.

Table 14: Analogue Temperature Sensor Module Connections

Device	Arduino	Wire	Description
S	A0		Connects to left side of the **NTC** thermistor.
middle	5V		Connects to left side of the **NTC** thermistor via a 10 kiloohm [kΩ] resistor.
-	GND		Connects to right side of the **NTC** thermistor.

A0: can be any analogue pin



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pTempAnalogue = A0;
void setup () {
  Serial.begin( 9600 );
}
void loop () {
  Serial.println( analogRead( pTempAnalogue ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.4 × 15.2 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	29.7 ×15.2 × 7.2 mm
Weight:	1.19 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Analogue Temperature Sensor Module…

Module Performance

Current Draw: 0.4 milliamps [mA] @ 4.61 **VDC**

Module Mounting

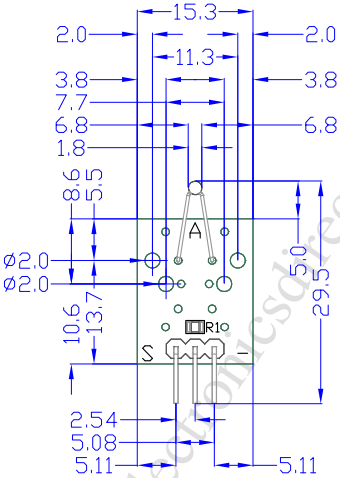
The module has 4 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

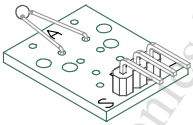
Projects

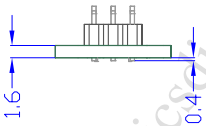
Folder: **Modules****Environment****Analogue_Temperature_Sensor**\

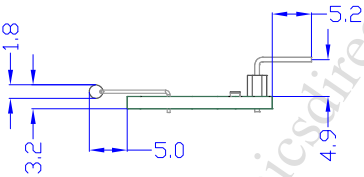
- **Analogue_Temperature_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.
- **Analogue_Temperature_Sensor_Celsius_SM**: Displays the results in degrees Celsius [ºC] to the Arduino Serial Monitor / Plotter.
- **Analogue_Temperature_Sensor_Fahrenheit_SM**: Displays the results in degrees Fahrenheit [ºF] to the Arduino Serial Monitor / Plotter.

Analogue Temperature Sensor Module - Dimensions









Digital Temperature Sensor Module

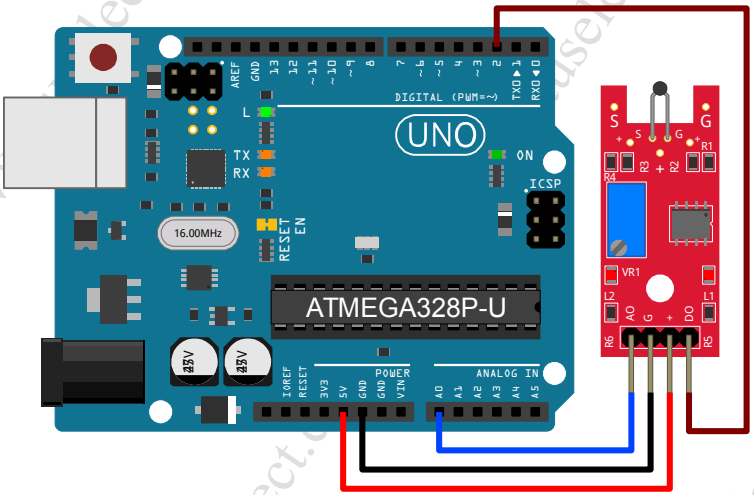
This module contains a **N**egative **T**emperature **C**oefficient thermistor which converts temperature to resistance. The on-board circuitry converts this resistance to a voltage between 0 and 5 **V**olts **D**irect **C**urrent, and output an analogue signal proportional to the temperature. A digital signal is also available which indicates when the temperature has reached the preset level.

The sensitivity can be adjusted via the on-board multi-turn potentiometer, however, this also adjusts the analogue value output of the temperature. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns). When the module is calibrated, the threshold trigger is set to ~25 degrees Celsius. Calibration procedure is explained at the end of this section.

Table 15: Digital Temperature Sensor Module Connections

Device	Arduino	Wire	Description
AO	A0		Analogue signal output from thermistor circuitry.
G	GND		Ground connection.
+	5V		5 **VDC** positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pDigitalTemp = 2;
int pAnalogueTemp = A0;
void setup() {
  Serial.begin( 9600 );
  pinMode( pDigitalTemp, INPUT );
}
void loop() {
  Serial.print( analogRead( pAnalogueTemp ), DEC );
  Serial.print( ", " );
  Serial.println( digitalRead( pDigitalTemp ), DEC );
}
```

Digital Temperature Sensor Module…

Module Specifications

PCB Dimensions (H × W × D):	35.7 × 15.6 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	41.3 × 15.6 × 14.3 mm
Weight:	2.74 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (triggered):	6.1 milliamps [mA] @ 4.53 **VDC**
Current Draw (not triggered):	4.3 mA @ 4.52 **VDC**

Module Mounting

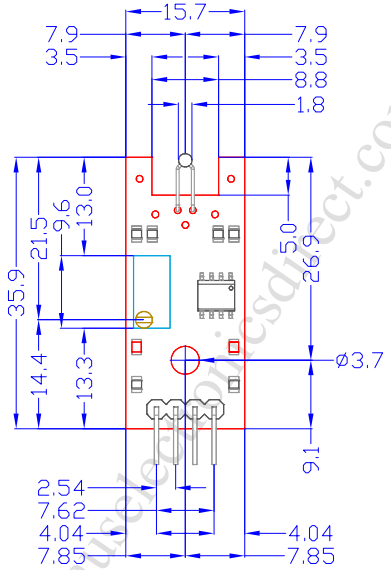
The module has a single 3.7 mm diameter hole close to the connection pin end of the **P**rinted **C**ircuit **B**oard. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

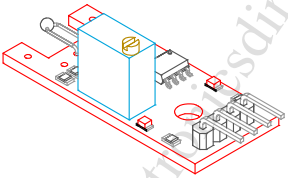
Projects

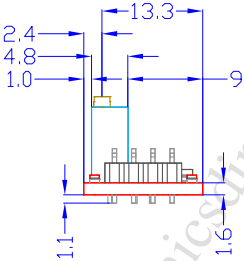
Folder: **Modules\Environment\Digital_Temperature_Sensor**

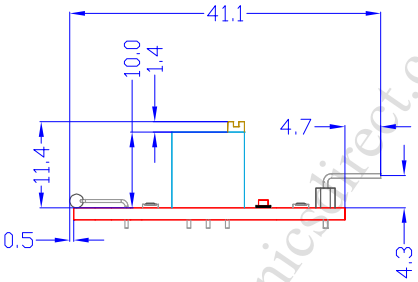
- **Digital_Temperature_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.
- **Digital_Temperature_Sensor_Celsius_SM**: Displays the results in degrees Celsius [ºC] to the Arduino Serial Monitor / Plotter.
- **Digital_Temperature_Sensor_Fahrenheit_SM**: Displays the results in degrees Fahrenheit [ºF] to the Arduino Serial Monitor / Plotter.

Digital Temperature Module - Dimensions









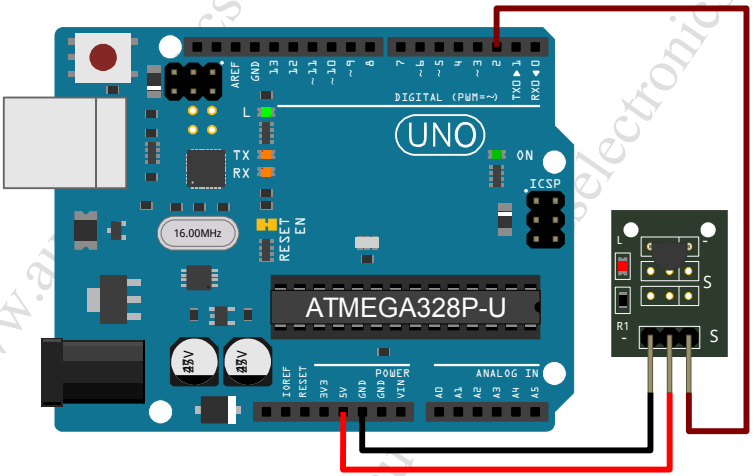
I²C Digital Temperature Sensor Module

This module contains a Dallas Semiconductor temperature sensor **I**ntegrated **C**ircuit DS18B20 or equivalent, which communicates the temperature via the I²C (Inter-Integrated Circuit) protocol. Each **IC** has a unique 64 bit serial code to allow multiple modules to be connected on the same I²C bus. The **L**ight **E**mitting **D**iode on the **P**rinted **C**ircuit **B**oard will flash when a temperature measurement has been sent.

Table 16: I²C Digital Temperature Sensor Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	5 **V**olts **D**irect **C**urrent power supply to Dallas Semiconductor **IC**.
S	D2	■	Digital output signal from Dallas Semiconductor temperature **IC**.

D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
#include <DallasTemperature.h>
#include <OneWire.h>
int pDigitalTemp = 2;
OneWire oneWire( pDigitalTemp );
DallasTemperature sensor( &oneWire );
void setup() {
  Serial.begin( 9600 );
  sensor.begin();
}
void loop() {
  sensor.requestTemperatures();
  Serial.println( sensor.getTempCByIndex( 0 ) );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.4 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	24.7 × 15.4 × 7.1 mm
Weight:	1.30 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw:	0.67 milliamps [mA] @ 4.68 **VDC**

I²C Digital Temperature Sensor Module…

Dallas Temperature IC Performance

Supply Voltage:	3.0 to 5.5 **VDC**
Temperature Range:	-55 to 125 degrees Celsius [ºC]
Accuracy:	± 0.5 ºC from -10 to +85 ºC

Module Mounting

The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

Projects

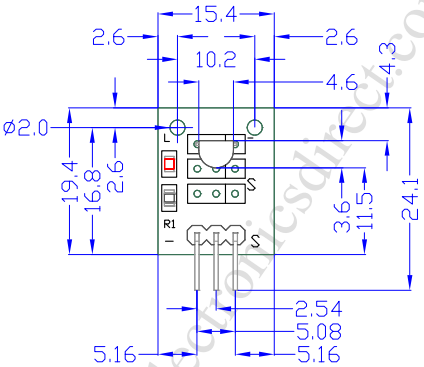
Folder: **\Modules\Environment\I2C_Digital_Temperature_Sensor**

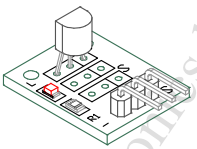
- **I2C_Digital_Temperature_Sensor_Celsius_SM:** Displays the results in Celsius to the Arduino Serial Monitor / Plotter.
- **I2C_Digital_Temperature_Sensor_Fahrenheit_SM**: Displays the results in Fahrenheit to the Arduino Serial Monitor / Plotter.

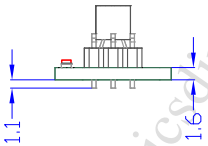
Libraries

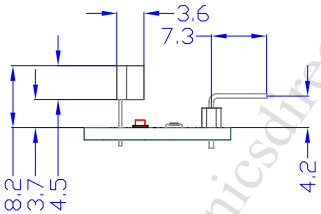
- **OneWire**: Used to communicate via the I2C protocol.
- **DallasTemperature**: Used to decode the temperature data from the DS18B20.

I²C Digital Temperature Sensor Module - Dimensions









DHT11 Temperature and Humidity Sensor Module

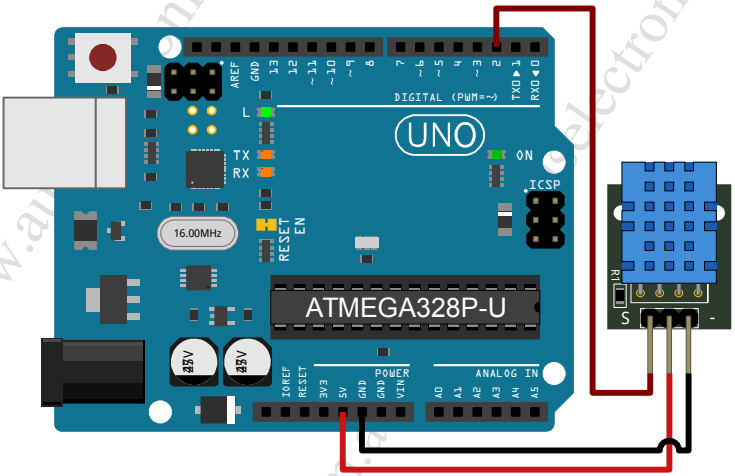
This module contains a **D**igital **H**umidity and **T**emperature Type **11** sensor which communicates its readings via the Inter-Integrated Circuit (I^2C) protocol. The **DHT11** component incorporates a capacitive humidity sensor and a **N**egative **T**emperature **C**oefficient thermistor, and outputs an 8 bit digital signal. It requires at least one second from initial power up for initialisation, and a delay of at least 2 seconds between subsequent readings.

This type of module is common in many digital weather stations, and also appears in some desktop digital clocks.

Table 17: DHT11 Temperature and Humidity Sensor Module Pin Connections

Device	Arduino	Wire	Description
S	D2		Digital output signal from the DHT11.
middle	5V		5 VDC power supply to the DHT11.
-	GND		Ground connection.

D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
#include "DHT.h"
#define pDHT 2
#define DHTTYPE DHT11
DHT dht( pDHT, DHTTYPE );
void setup() {
  Serial.begin( 9600 );
  dht.begin();
}
void loop() {
  float fHumidity = dht.readHumidity();
  float fTemperature = dht.readTemperature();
  if ( isnan( fTemperature ) || isnan( fHumidity ) ) {
    Serial.println( "Failed to read from DHT" );
  } else {
    Serial.print( fHumidity );
    Serial.print(",");
    Serial.println( fTemperature );
  }
  delay( 3000 );
}
```

DHT11 Temperature and Humidity Sensor Module…

Module Specifications

PCB Dimensions (H × W × D):	19.2 × 16.4 × 1.5 millimetre [mm]
Enclosing Dimensions (H × W × D):	28.5 × 16.4 × 8.5 mm
Weight:	2.14 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (Measuring):	1.43 milliamps [mA]
Current Draw (Stand-by):	150 microamps [µA]

DHT11 Component Specifications

Temperature Range:	0 - 50 degrees Celsius [ºC] ±2%
Relative Humidity Range:	20 - 90 % ±5%
Current (Measuring):	0.5 mA (min), 2.5 mA (max)
Current (Stand-by):	0.1 mA (min), 0.15 mA (max)

Module Mounting

There are 2 × 2 mm diameter mounting holes on the **P**rinted **C**ircuit **B**oard, which require that the **DHT11** component is carefully bent away from the board to insert mounting screws. Because the **DHT11** is designed to sit flat on the **PCB**, the holes on the **PCB** can be countersunk using a hand held countersink bit to accept countersunk head screws. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

Projects

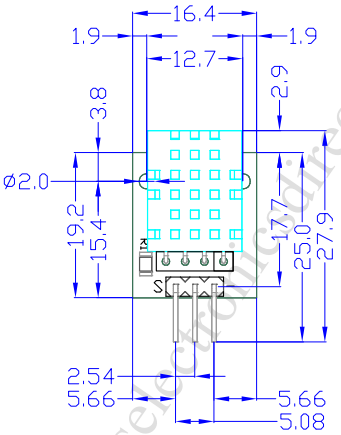
Folder: **Modules\Environment\DHT11_Temperature-Humidity_Sensor**

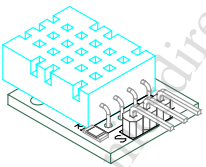
- **DHT11_Temperature-Humidity_Sensor_SM**: Displays the results from the **DHT11** module to the Arduino Serial Monitor at 3 second intervals.

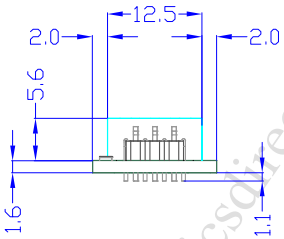
Libraries

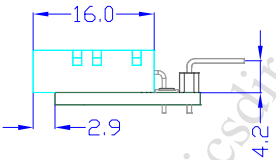
The "DHT" library is required for the Arduino to communicate with the **DHT11**. This adds the functions "`readTemperature()`", and "`readHumidity()`".

DHT11 Temperature and Humidity Sensor Module - Dimensions









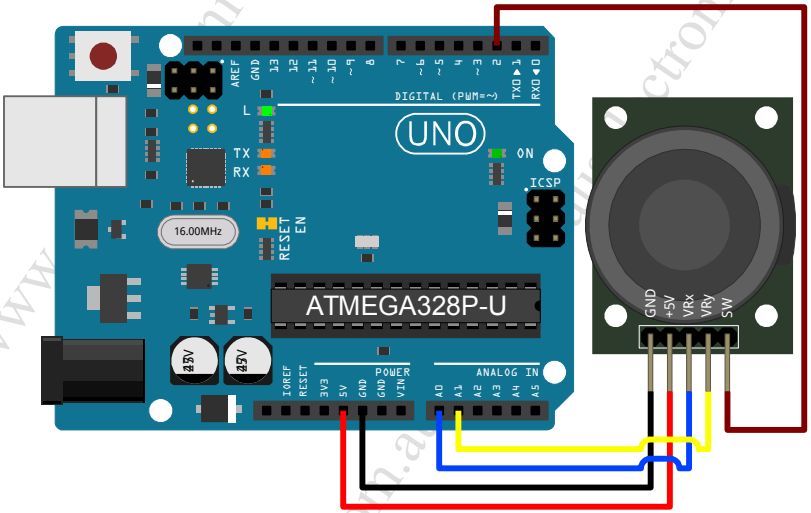
Dual Axis Analogue Joystick Module

This module contains a spring loaded dual axis joystick which returns analogue values based on the position of the stick in the horizontal and vertical planes. It also includes an integrated push button switch which activates when the joystick shaft is pressed downwards in-line with the stick.

Table 18: Dual Axis Analogue Joystick Module Pin Connections

Device	Arduino	Wire	Description
GND	GND	■	Ground connection.
+5V	5V	■	5 Volts Direct Current power supply to potentiometers.
VRx	A0	■	Connects to horizontal axis potentiometer (10 kiloohm [kΩ]).
VRy	A1	■	Connects to vertical axis potentiometer (10 kΩ).
SW	D2	■	Connects to ground when the switch is depressed.

A0, A1 can be any analogue pin, D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pJoyX = A0;
int pJoyY = A1;
int pJoySw = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pJoySw, INPUT );
  digitalWrite( pJoySw, HIGH );
}
void loop() {
  Serial.print( analogRead( pJoyX ) );
  Serial.print( "," );
  Serial.print( analogRead( pJoyY ) );
  Serial.print( "," );
  Serial.println( digitalRead( pJoySw ) );
}
```

Dual Axis Analogue Joystick Module…

Module Specifications

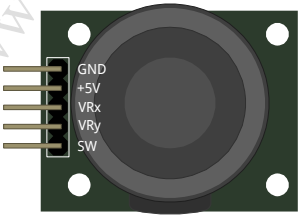
PCB Dimensions (H × W × D) :	34.0 × 26.3 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	39.4 × 27.5 × 32.9 mm
Weight:	10.07 grams [g]
Input Voltage:	~ 5 **VDC**

Module Performance

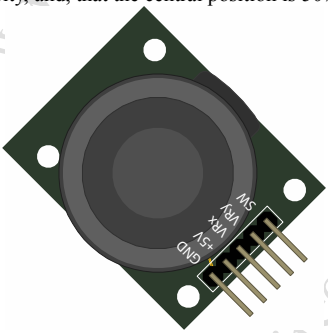
Note! The X and Y axis marked on the PCB are for when the module is oriented with the pins at the left hand side.

Joystick X Full Left Position Voltage:	0.0 **VDC** @ 5.06 **VDC** Supply
Joystick X Centre Position Voltage :	2.51 **VDC** @ 5.06 **VDC** Supply
Joystick X Full Right Position Voltage:	5.06 **VDC** @ 5.06 **VDC** Supply
Joystick Y Full Top Position Voltage:	0.0 **VDC** @ 5.06 **VDC** Supply
Joystick Y Centre Position Voltage :	2.46 **VDC** @ 5.06 **VDC** Supply
Joystick Y Full Bottom Position Voltage:	5.06 **VDC** @ 5.06 **VDC** Supply
Joystick X Full Left Position Resistance:	58 ohms [Ω] @ -33º
Joystick X Centre Position Resistance:	3732 Ω @ 0º
Joystick X Full Right Position Resistance:	4913 Ω @ +33º
Joystick Y Full Top Position Resistance:	23 Ω @ -33º
Joystick Y Centre Position Resistance:	3619 Ω @ 0º
Joystick Y Full Bottom Position Resistance:	4901 Ω @ + 33º
Joystick Range of movement:	66º (±33º from centre) both axis

When using the joystick to control the drive of a vehicle with separate drive and steering systems the module will be oriented as shown in the diagram below left. When used for vehicles which use tracks or skid steering, the module can be rotated clockwise 225ºas shown in the diagram below right, so that the upper position returns the highest values, and the lower position returns the lowest values. This maps the direction directly to what is required for each drive motor, although this does affect the linearity, and, that the central position is 50% drive.



Normal Orientation (Pins facing left)
X = Steering
Y = Drive Speed



225º Clockwise Orientation (Pins facing bottom right)
L = Left (X), R = Right (Y)

X: 0%, Y: 0%	X: 50%, Y: 0%	X: 100%, Y: 0%	L: 100%, R: 50%	L: 100%, R: 100%	L: 50%, R: 100%
X: 0%, Y 50%	X: 50%, Y 50%	X: 100%, Y 50%	L: 100%, R: 0%	L: 50%, R 50%	L: 0%, R 100%
X: 0%, Y:100%	X:50%, Y:100%	X: 100%, Y: 100%	L: 50%, R: 0%	L:0%, R:0%	L: 0%, R: 50%

Module Mounting

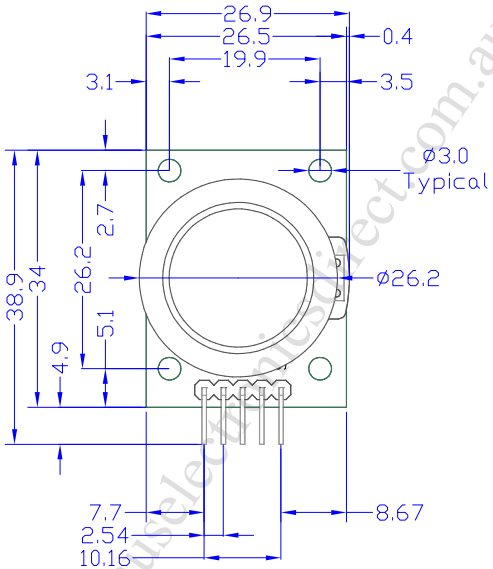
The module has 4 × 3 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

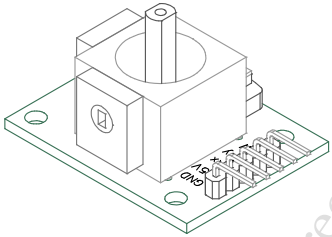
Projects

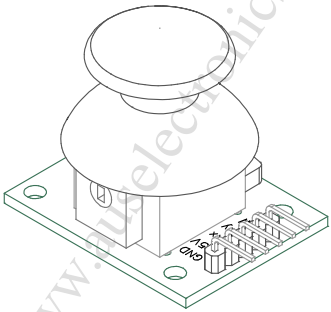
Folder: **Modules\Interface\Dual_Axis_Analogue_Joystick**

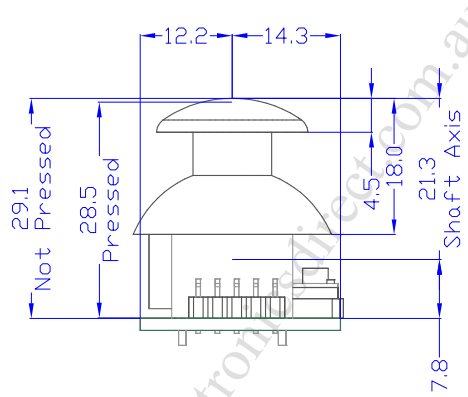
- **Dual_Axis_Analogue_Joystick_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

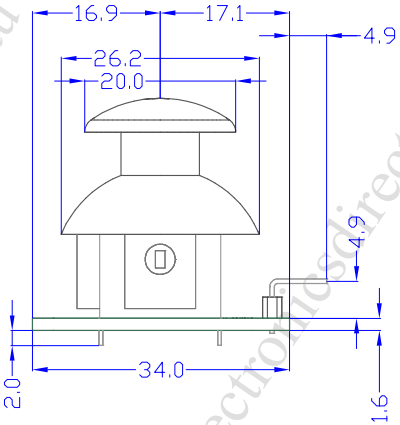
Dual Axis Analogue Joystick Module - Dimensions











Push Button Switch Module

This module uses a **P**rinted **C**ircuit **B**oard mounted 6 millimetre [mm] square **N**ormally **O**pen push button switch which connects the two outer pins when pressed. A 10 kiloohm [kΩ] resistor is included which connects the centre pin to the signal "S" pin for use as a "pull up" or "pull down" resistor.

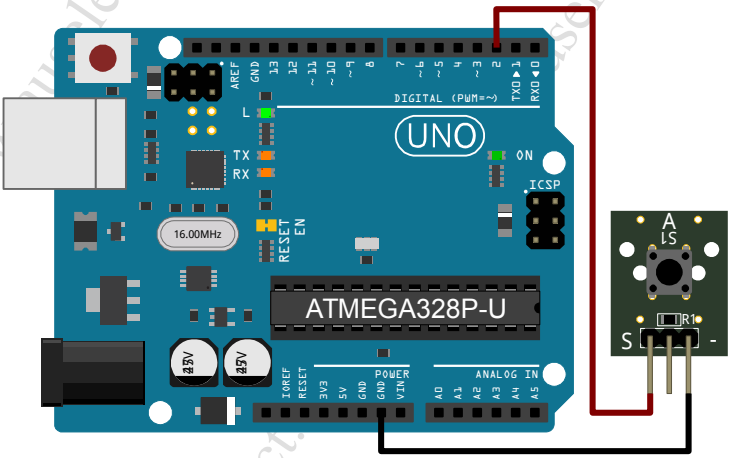
Internally, they contain a small concave disc which is forced down onto the 2 contacts when pressure is applied to the plastic shaft. This type of push button switch is common in many low voltage electronic devices and they typically have a long service life as long as they are not abused. e.g. If the switch is pressed with excess force, the concave disc can deform, and no longer make reliable contact.

As with any switch having mechanical contacts, it will be subject to "contact bounce". This occurs during the final steps of closing, when the contacts are moving into their final connected position. Once enough pressure has been applied the connection will become stable. This can be overcome via coding, or, by adding additional circuitry.

Table 19: Push Button Switch Module Pin Connections

Device	Arduino	Colour	Description
S	D2		Connects to left side of button switch.
middle	NC		*Optional connection to left side of button switch via 10 kΩ resistor.*
-	GND		Connects to right side of button switch.

D2 can be any digital pin. This connection method and sketch below use the Arduino's internal "pull up" resistor.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pButton = 2;
void setup() {
  pinMode( pButton, INPUT );
  digitalWrite( pButton, HIGH );
  Serial.begin( 9600 );
}
void loop() {
  Serial.println( digitalRead( pButton ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.2 × 15.5 × 1.6 mm
Enclosing Dimensions (H × W × D) :	24.7 × 15.5 × 9.8 mm
Weight:	1.45 grams [g]
Input Voltage:	<12 **V**olts **D**irect **C**urrent

Push Button Switch Module…

Module Performance

Switch Resistance (Pins "S" & "-"):	1.0 Ω
On-board Resistor (Pins "S" & middle):	10.0 kΩ

Mounting

The module has 2 × 2 mm suitable mounting holes on the **P**rinted **C**ircuit **B**oard. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

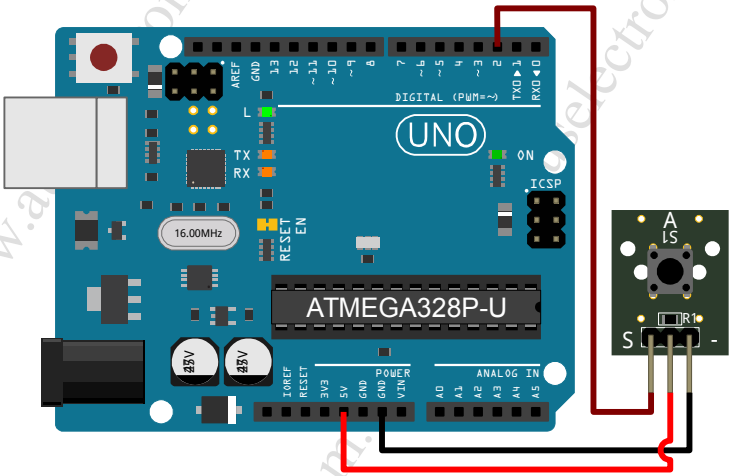
Projects

Folder: **Modules\Interface\Push_Button_Switch**

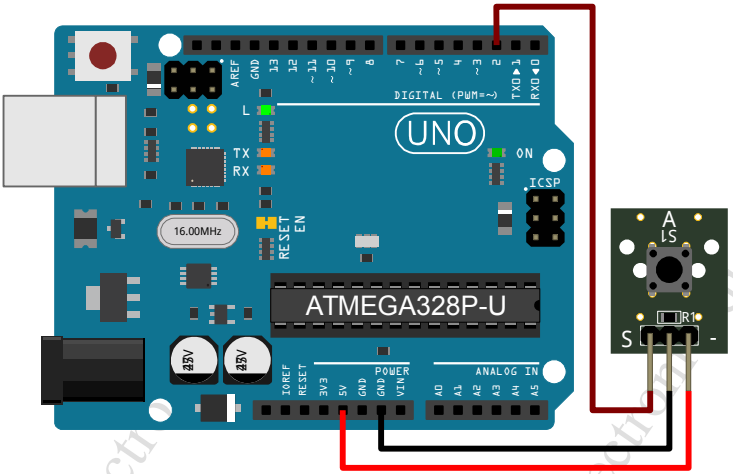
- **Push_Button_Switch_SM**: Displays the switch state to the Arduino Serial Monitor / Plotter.

Pull Up / Pull Down Connections

Many modules include an on-board resistor which can be used as a "Pull Up" or "Pull Down" resistor. This forces the Arduino's digital input pin to be either "high" (~5 **VDC**), or, low (~0 **VDC**).



Connection when using the module's on-board resistor as a "Pull Up" resistor.

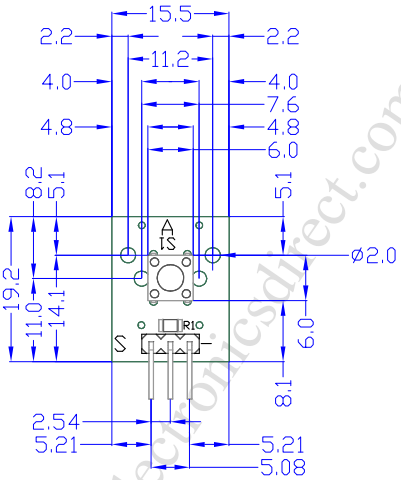


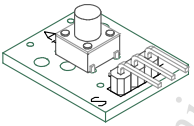
Connection when using the module's on-board resistor as a "Pull Down" resistor.

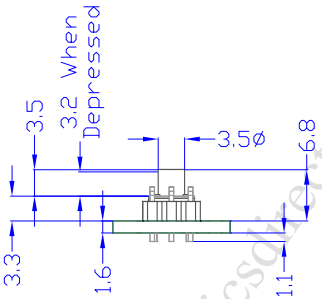
The sketch below is for when the Push Button Module is used with a Pull-Up / Pull-Down resistor.

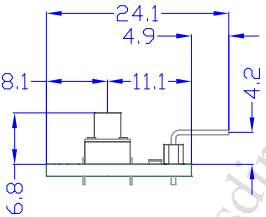
```
int pButton = 2;
void setup() {
  pinMode( pButton, INPUT );
  // This line below is not required when using an external resistor.
  // digitalWrite( pButton, HIGH );
  Serial.begin( 9600 );
}
void loop() {
  Serial.println( digitalRead( pButton ), DEC );
}
```

Push Button Switch Module - Dimensions



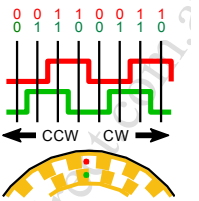






Rotary Encoder Module

This module contains an incremental rotary encoder. A rotary encoder detects when its shaft is rotated, and, in which direction it was rotated in. It also includes a normally open switch which is activated by pressing down in-line with the shaft.



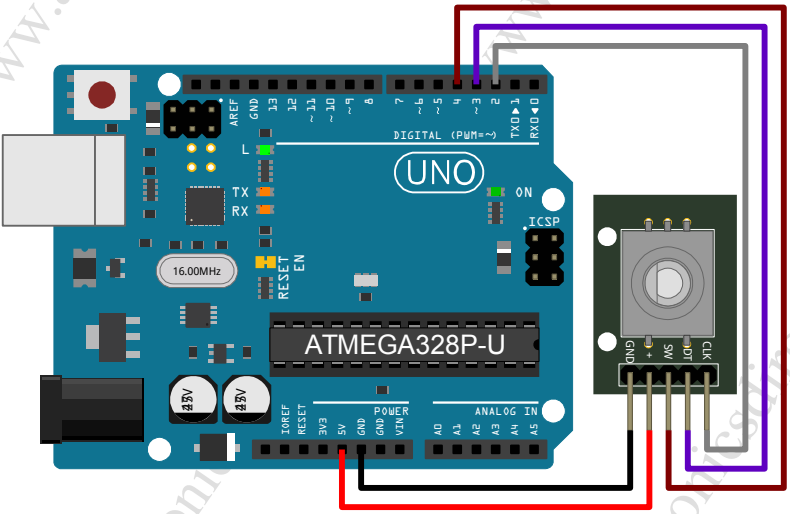
Internally, the encoder contains 2 discs, one containing the rotating contact pads, and the other containing 2 static contacts. In the image at right the lower circular section represents the disc containing the contact pads which rotate (in gold), with the red and green dots representing the static contacts. The contact pads are arranged on the static disc in a specific sequence, and with each switch's contact ring out of phase with the other. This system is known as quadrature encoding as there are 4 (quad) possible states. By comparing the previous state of the contacts against the new state, the direction and rotation angle can be calculated.

The main uses for this type of control are for navigating display menu's, however variations of the disc assembly are often used to track the position of rotary motors.

Table 20: Rotary Encoder Module Pin Connections

Device	Arduino	Wire	Description
GND	GND	■	Ground connection.
+	5V	■	Positive 5 Volts Direct Current supply.
SW	D4	■	Connects to ground when switch is depressed.
DT	D3	■	Connects to rotary encoder contact A (data).
CLK	D2	■	Connects to rotary encoder contact B (clock).

D2, D3, D4 can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pCLK = 2;
int pDT = 3;
int pSW = 4;
int iEncoderPos = 0;
boolean bLastState = LOW;
void setup() {
  pinMode( pDT, INPUT );
  pinMode( pCLK, INPUT );
  pinMode( pSW, INPUT );
```

```
  digitalWrite( pDT, HIGH );
  digitalWrite( pCLK, HIGH );
  digitalWrite( pSW, HIGH );
  Serial.begin ( 9600 );
}
void loop() {
  boolean bEncoderCLK = digitalRead( pCLK );
  if ( ( bLastState == HIGH ) && ( bEncoderCLK == LOW ) ) {
    if ( digitalRead( pDT ) == LOW ) {
      iEncoderPos--;
    } else {
      iEncoderPos++;
    }
  }
  bLastState = bEncoderCLK;
  Serial.print ( iEncoderPos );
  Serial.print ( "," );
  Serial.println ( digitalRead( pSW ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D) :	26.4 × 19.0 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	31.5 × 19.0 × 29.9 mm
Weight:	5.99 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Mounting

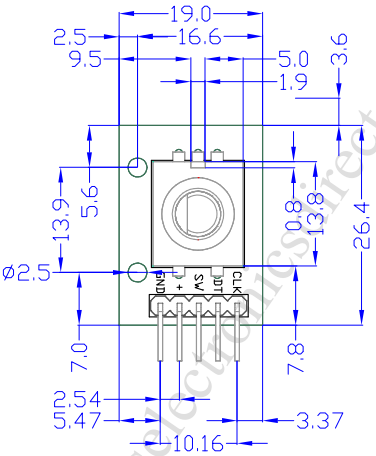
The module has 2.5 millimetre diameter mounting holes at the left hand side. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

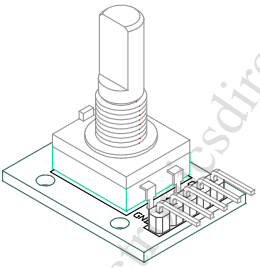
Projects

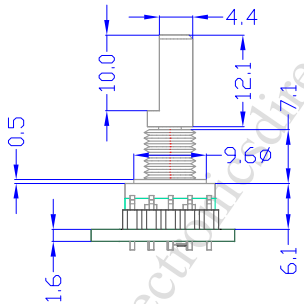
Folder: **\Modules\Interface\Rotary_Encoder**

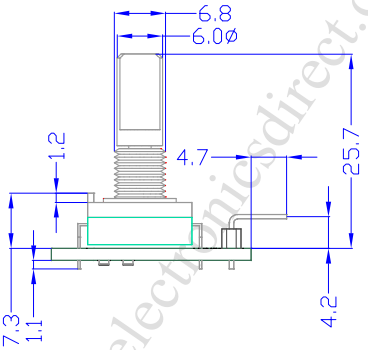
- **Rotary_Encoder_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Rotary Encoder Module - Dimensions









Touch Sensor Module

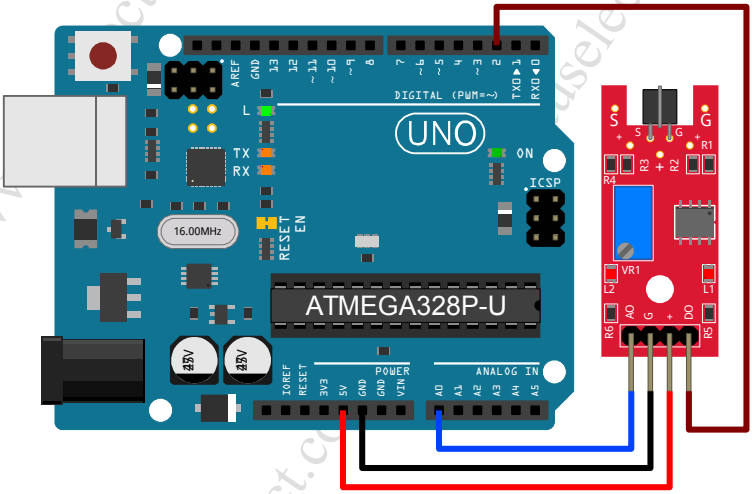
This module contains a "Darlington" transistor (with its base wrapped around the case) which senses when it is in contact with skin. It outputs an analogue signal based on the skin conductivity / pressure, and a digital signal when a preset level has been reached. The sensitivity can be adjusted via the on-board multi-turn potentiometer. Turning the adjustment screw clockwise increases the sensitivity, (the multi-turn potentiometer has a range of approximately 25 turns).

A "Darlington" transistor is a pair of transistors with the emitter of the first transistor connected to the base of the second transistor so as to greatly increase the gain. This allows the transistor to amplify the very small changes in resistance present in human skin.

Table 21: Touch Sensor Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue signal output from the touch sensor circuitry.
G	GND		Ground connection.
+	5V		5 Volts Direct Current positive supply for board circuitry.
DO	D2		Digital output of threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pTouchDigital = 2;
int pTouchAnalogue = A0;
void setup() {
  pinMode( pTouchDigital, INPUT );
  digitalWrite( pTouchDigital, HIGH );
  Serial.begin( 9600 );
}
void loop() {
  Serial.print( analogRead( pTouchAnalogue ), DEC );
  Serial.print( ", " );
  Serial.println( digitalRead( pTouchDigital ), DEC );
}
```

Touch Sensor Module…

Module Specifications

PCB Dimensions (H × W × D):	35.9 × 15.5 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	40.9 × 15.5 × 14.2 mm
Weight:	2.89 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (not triggered):	4.1 milliamps [mA] @ 5.01 **VDC**
Current Draw (triggered):	4.8 mA @ 4.96 **VDC**

Module Mounting

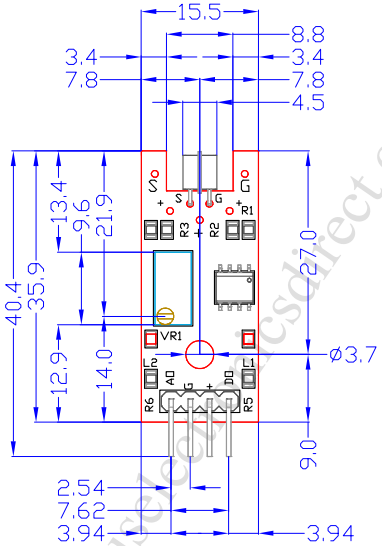
The module has a single 3.7 mm diameter hole close to the connection pin end of the **P**rinted **C**ircuit **B**oard. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

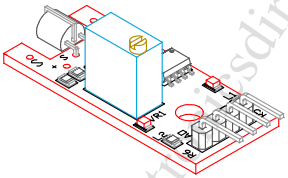
Projects

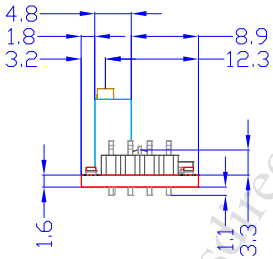
Folder: **Modules\Interface\Touch_Sensor**

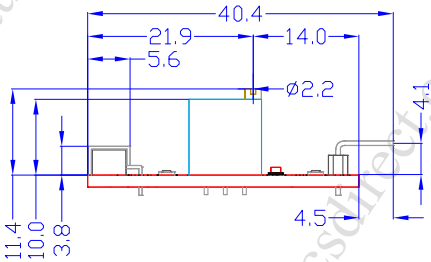
- **Touch_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Touch Sensor Module - Dimensions









2 Colour 3mm THT LED Module

This module includes a 3 millimetre [mm] Through Hole Type opaque common cathode Light Emitting Diode with 2 individually controllable red and green colour elements. The brightness of each LED component can be varied by connecting each component to an Arduino pin which supports Pulse Width Modulation (pins 3, 5, 6, 9, 10, 11 on the UNO).

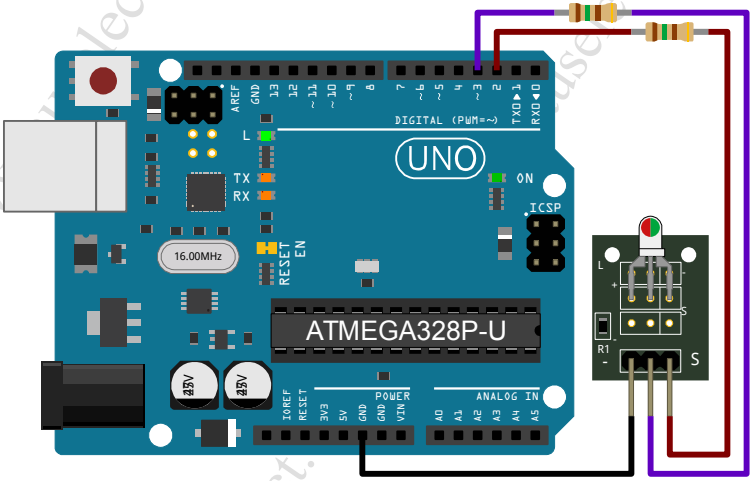
Important! *This module requires a "dropper" resistor (150 ohm or higher for continuous use) for each of the LED's individual red and green components, otherwise they will be permanently damaged.*

The manufacturer's specifications for the nominal (normal continuous use) forward current of each individual red and green component is 20 milliamps [mA]. The "dropper" resistor to achieve this current at 5 Volts Direct Current is 150 ohm [Ω]. It is possible to run the LED components at 30 mA for a short period without permanent damage, in which case 120 Ω resistors can be used.

Table 22: 2 Colour 3mm THT LED Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Common ground for both LED components.
middle	D3	■	Positive power for red component of the LED (Requires 150 Ω resistor).
S	D2	■	Positive power for green component of the LED (Requires 150 Ω resistor).

D2, D3 : can be any digital pin.



*The sketch below can be used to control the 2 Colour 3mm **THT LED** Module via the Arduino Serial Monitor.*

```
int pRed = 3;
int pGreen = 2;
void setup() {
  Serial.begin( 9600 );
  while ( !Serial ) {
  }
  digitalWrite ( pGreen, LOW );
  digitalWrite ( pRed, LOW );
  pinMode( pRed, OUTPUT );
  pinMode( pGreen, OUTPUT );
  Serial.println( "R=red on, r=red off, G=green on, g=green Off." );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == 'R' ) {
```

```
      Serial.println( "Red On" );
      digitalWrite ( pRed, HIGH );
    }
    if ( cInput == 'r' ) {
      Serial.println( "Red off" );
      digitalWrite ( pRed, LOW );
    }
    if ( cInput == 'G' ) {
      Serial.println( "Green On" );
      digitalWrite ( pGreen, HIGH );
    }
    if ( cInput == 'g' ) {
      Serial.println( "Green off" );
      digitalWrite ( pGreen, LOW );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.2 × 15.4 × 1.6 mm
Enclosing Dimensions (H × W × D) :	28.7 × 15.4 × 14.7 mm
Weight:	1.31 grams [g]
Input Voltage:	5 **VDC**

LED Manufacturers Specifications

Viewing Angle:	65º
Red component wavelength:	605 nanometres [nm]
Red component forward current (IF):	20 mA nominal, 30 mA maximum
Red component forward voltage (VF):	2.1 **VDC**
Green component wavelength:	567 nm
Green component forward current (IF):	20 mA nominal, 30 mA maximum
Green component forward voltage (VF):	2.3 **VDC**

Module Performance

Current Draw (Red component):	14.1 mA @ 4.99 **VDC** using 150 Ω resistor
	17.8 mA @ 4.99 **VDC** using 120 Ω resistor
Current Draw (Green component):	17.5 mA @ 4.95 **VDC** using 150 Ω resistor
	22.2 mA @ 4.95 **VDC** using 120 Ω resistor

Module Mounting

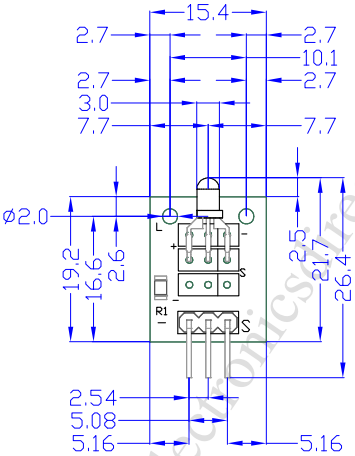
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

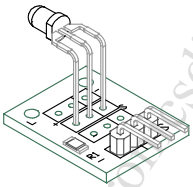
Projects

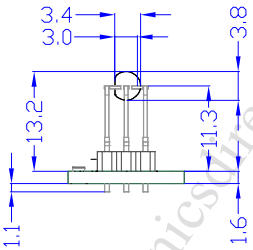
Folder: **\Modules\Optical\2_Colour_3_mm_THT_LED**

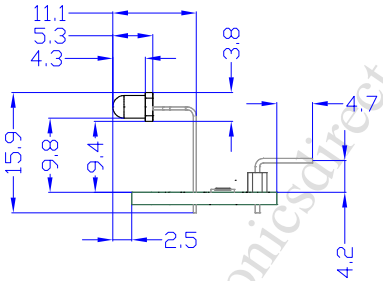
- **2_Colour_3_mm_THT_LED_SM**: Controls the state of the **LED**'s using the Arduino Serial Monitor.
- **2_Colour_3_mm_THT_LED_Timed**: Alternates the red and green components at timed intervals.
- **2_Colour_3_mm_THT_LED_PWM**: Uses **PWM** to vary the brightness of the red and green components at timed intervals.

2 Colour 3mm THT LED Module - Dimensions









2 Colour 5mm THT LED Module

This module contains a 5 millimetre [mm] Through Hole Type opaque common cathode Light Emitting Diode with 2 individually controllable red and green colour elements. The brightness of each LED component can be varied by connecting each component to a Arduino pin which supports Pulse Width Modulation (pins 3, 5, 6, 9, 10, 11 on the UNO).

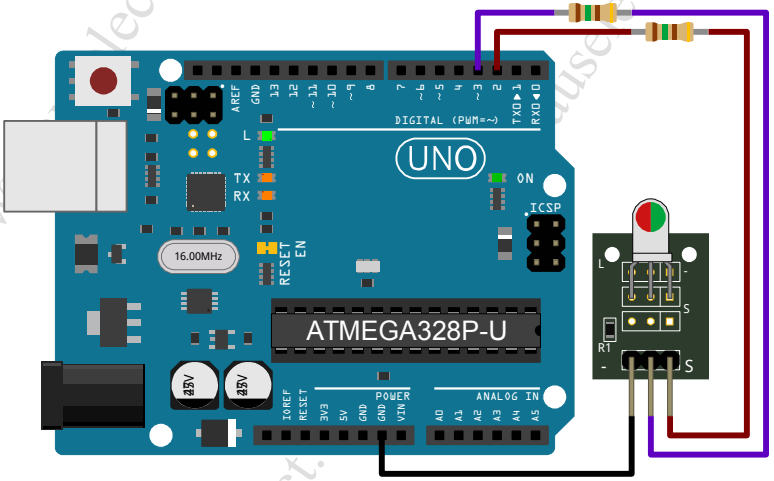
Important! *This module requires a "dropper" resistor (150 ohm or higher for continuous use) for each of the LED's individual red and green components, otherwise they will be permanently damaged.*

The manufacturer's specifications for the nominal (normal continuous use) forward current of each individual red and green component is 20 milliamps [mA]. The "dropper" resistor to achieve this current at 5 Volts Direct Current is 150 ohm [Ω]. It is possible to run the LED components at 30 mA for a short period without permanent damage, in which case 120 Ω resistors can be used.

Table 23: 2 Colour 5mm THT LED Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Common ground for both LED components.
middle	D3	■	Positive power for red component of LED (Requires 150 Ω resistor).
S	D2	■	Positive power for green component of LED (Requires 150 Ω resistor).

D2, D3 : can be any digital pin.



*The sketch below can be used to control the 2 Colour 5mm **THT LED** Module via the Arduino Serial Monitor.*

```
int pRed = 3;
int pGreen = 2;
void setup() {
  Serial.begin( 9600 );
  while ( !Serial ) {
  }
  digitalWrite ( pGreen, LOW );
  digitalWrite ( pRed, LOW );
  pinMode( pRed, OUTPUT );
  pinMode( pGreen, OUTPUT );
  Serial.println( "R=red on, r=red off, G=green on, g=green Off." );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == 'R' ) {
```

```
      Serial.println( "Red On" );
      digitalWrite ( pRed, HIGH );
    }
    if ( cInput == 'r' ) {
      Serial.println( "Red off" );
      digitalWrite ( pRed, LOW );
    }
    if ( cInput == 'G' ) {
      Serial.println( "Green On" );
      digitalWrite ( pGreen, HIGH );
    }
    if ( cInput == 'g' ) {
      Serial.println( "Green off" );
      digitalWrite ( pGreen, LOW );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.3 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	26.5 × 15.4 × 19.6 mm
Weight:	1.44 grams [g]
Input Voltage:	5 **VDC**

LED Manufacturers Specifications

Viewing Angle:	90º
Red component wavelength:	625 nanometres [nm]
Red component forward current (IF):	20 mA nominal, 30 mA maximum
Red component forward voltage (VF):	2.1 **VDC**
Green component wavelength:	567 nm
Green component forward current (IF):	20 mA nominal, 30 mA maximum
Green component forward voltage (VF):	2.2 **VDC**

Module Performance

Current Draw (Red component):	16.6 mA @ 4.91 **VDC** using 150 Ω resistor
	24.9 mA @ 4.93 **VDC** using 120 Ω resistor
Current Draw (Green component):	17.5 mA @ 4.96 **VDC** using 150 Ω resistor
	23.1 mA @ 4.96 **VDC** using 120 Ω resistor

Module Mounting

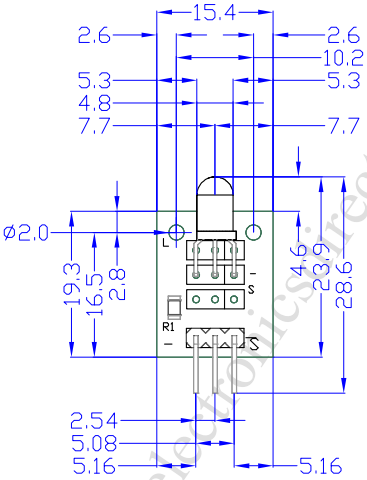
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

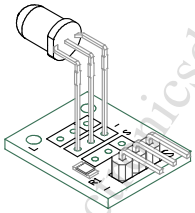
Projects

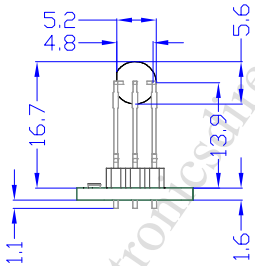
Folder: **Modules\Optical**

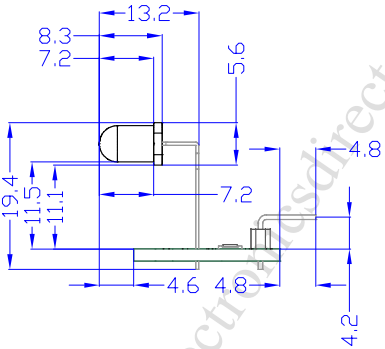
- **2_Colour_5_mm_THT_LED_SM**: Controls the state of the **LED**'s using the Arduino Serial Monitor.
- **2_Colour_5_mm_THT_LED_Timed**: Alternates the red and green components at timed intervals.
- **2_Colour_5_mm_THT_LED_PWM**: Uses **PWM** to vary the brightness of the red and green components at timed intervals.

2 Colour 5mm THT LED Module - Dimensions









3 Colour 5mm THT LED Module

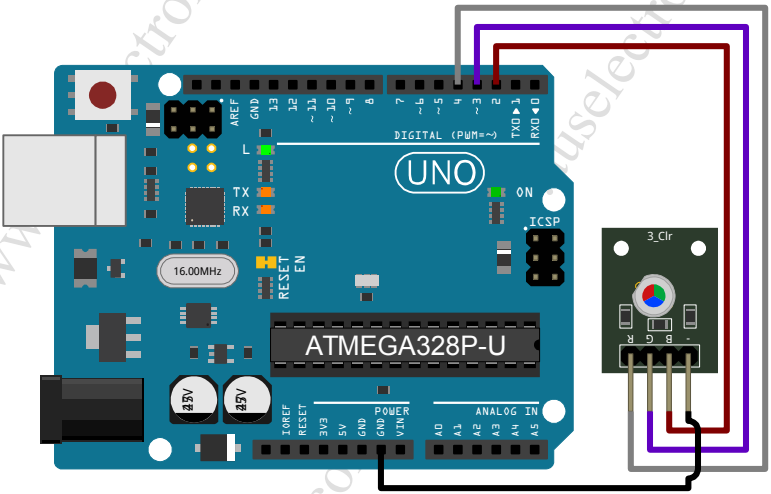
This module contains a 5 millimetre [mm] Through Hole Type waterclear Light Emitting Diode with 3 individually controllable red, green, and blue colour elements. This module includes the "dropper" resistors for the LED components. The individual LED components can be connected to Arduino pins which support Pulse Width Modulation to create almost any colour.

This type of module is bright enough to provide ambient light. It can also be used to provide a visual indication of the state of a process, e.g. red = fault, green = OK, blue = undetermined.

Table 24: 3 Colour 5mm THT LED Module Pin Connections

Device	Arduino	Wire	Description
R	D4		Positive 5 Volts Direct Current power for Red component of LED.
G	D3		Positive 5 VDC power for Green component of LED.
B	D2		Positive 5 VDC power for Blue component of LED.
-	GND		Common ground for all components of the LED.

D2, D3, D4 : can be any digital pin.



The sketch below can be used to control the RGB 5mm THT LED Module via the Arduino Serial Monitor.

```
int pRed = 4;
int pGreen = 3;
int pBlue = 2;
void setup() {
  pinMode( pBlue, OUTPUT );
  pinMode( pRed, OUTPUT );
  pinMode( pGreen, OUTPUT );
  digitalWrite ( pBlue, LOW );
  digitalWrite ( pRed, LOW );
  digitalWrite ( pGreen, LOW );
  Serial.begin( 9600 );
  while ( !Serial ) {
  }
  Serial.println( "R, G, B = turn colour on, r, g, b = turn colour off." );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
```

```
    if ( cInput == 'B' ) {
      Serial.println( "Blue On" );
      digitalWrite ( pBlue, HIGH );
    }
    if ( cInput == 'b' ) {
      Serial.println( "Blue off" );
      digitalWrite ( pBlue, LOW );
    }
    if ( cInput == 'R' ) {
      Serial.println( "Red On" );
      digitalWrite ( pRed, HIGH );
    }
    if ( cInput == 'r' ) {
      Serial.println( "Red off" );
      digitalWrite ( pRed, LOW );
    }
    if ( cInput == 'G' ) {
      Serial.println( "Green On" );
      digitalWrite ( pGreen, HIGH );
    }
    if ( cInput == 'g' ) {
      Serial.println( "Green off" );
      digitalWrite ( pGreen, LOW );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.2 × 15.4 × 1.6 mm
Enclosing Dimensions (H × W × D):	24.5 × 15.4 × 15.0 mm
Weight:	1.54 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (Red component):	19.1 milliamps [mA] @ 4.92 **VDC**
Current Draw (Green component):	13.3 mA @ 4.96 **VDC**
Current Draw (Blue component):	13.0 mA @ 4.88 **VDC**

LED Manufacturers Specifications

Viewing Angle:	60º
Red component wavelength:	643 nanometres [nm]
Green component wavelength:	528 nm
Blue component wavelength:	468 nm

Module Mounting

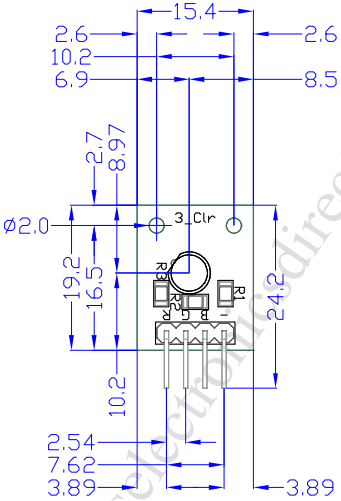
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

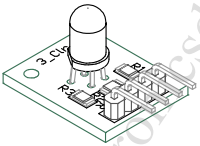
Projects

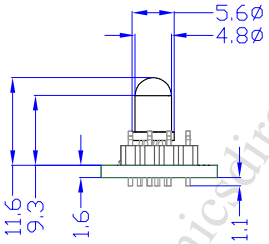
Folder: **\Modules\Optical\3_Colour_5mm_THT_LED**

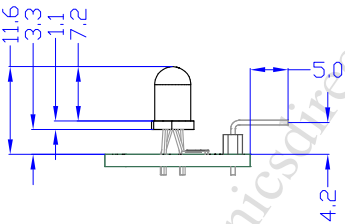
- **3_Colour_5mm_THT_LED_SM**: Control the module via the Arduino Serial Monitor.
- **3_Colour_5mm_THT_LED_PWM_SM:** Control the module using **P**ulse **W**idth **M**odulation via the Arduino Serial Monitor.

3 Colour 5mm THT LED Module - Dimensions









3 Colour 5mm SMD LED Module

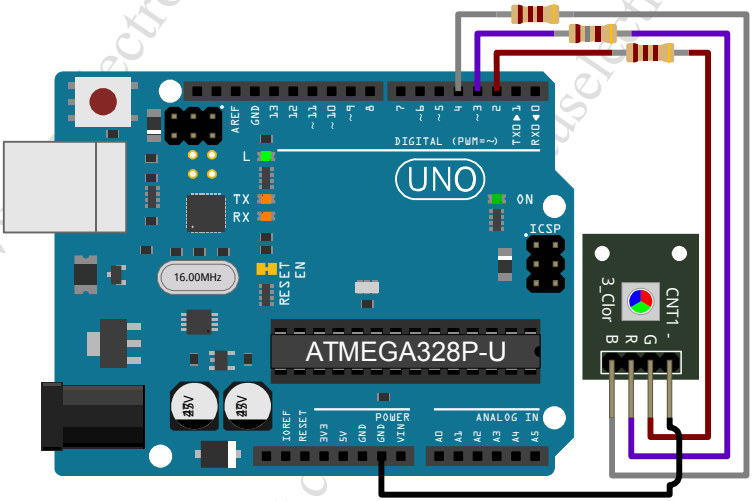
This module contains a 5 millimetre [mm] square **S**urface **M**ount **D**evice **L**ight **E**mitting **D**iode with 3 individually controllable red, green, and blue colour components. The individual **LED** components can be connected to Arduino pins which support **P**ulse **W**idth **M**odulation and combined at various levels to reproduce almost any colour. This type of module can be used to provide a visual indication of the state of a process, e.g. red = fault, green = OK, blue = undetermined.

Important! *This module requires a "dropper" resistor of at least 110 ohm [Ω] for green and blue components, and 180 Ω for the red component), otherwise they will be permanently damaged.*

Table 25: 3 Colour 5mm SMD LED Module

Device	Arduino	Wire	Description
B	D4		Positive 5 **V**olts **D**irect **C**urrent power via 110 Ω for blue **LED** component.
R	D3		Positive 5 **VDC** power via 180 Ω for red segment of **LED**.
G	D2		Positive 5 **VDC** power via 110 Ω for green segment of **LED**.
-	GND		Common ground for all segments of the **LED**.

D2, D3, D4 : can be any digital or analogue pin.



The sketch below can be used to control the RGB 5mm SMD LED Module via the Arduino Serial Monitor.

```
int pBlue = 4;
int pRed = 3;
int pGreen = 2;
void setup() {
  pinMode( pBlue, OUTPUT );
  pinMode( pRed, OUTPUT );
  pinMode( pGreen, OUTPUT );
  digitalWrite ( pBlue, LOW );
  digitalWrite ( pRed, LOW );
  digitalWrite ( pGreen, LOW );
  Serial.begin( 9600 );
  while ( !Serial ) {
  }
  Serial.println( "R, G, B = turn colour on, r, g, b = turn colour off." );
}
void loop() {
  if ( Serial.available() > 0 ) {
```

```
    int cInput = Serial.read();
    if ( cInput == 'B' ) {
      Serial.println( "Blue On" );
      digitalWrite ( pBlue, HIGH );
    }
    if ( cInput == 'b' ) {
      Serial.println( "Blue off" );
      digitalWrite ( pBlue, LOW );
    }
    if ( cInput == 'R' ) {
      Serial.println( "Red On" );
      digitalWrite ( pRed, HIGH );
    }
    if ( cInput == 'r' ) {
      Serial.println( "Red off" );
      digitalWrite ( pRed, LOW );
    }
    if ( cInput == 'G' ) {
      Serial.println( "Green On" );
      digitalWrite ( pGreen, HIGH );
    }
    if ( cInput == 'g' ) {
      Serial.println( "Green off" );
      digitalWrite ( pGreen, LOW );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.3 × 15.4 × 1.6 mm
Enclosing Dimensions (H × W × D) :	24.5 × 15.4 × 7.3 mm
Weight:	1.25 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (Blue component):	12.5 milliamps [mA] @ 5.06 **VDC** (via 110 Ω Resistor)
Current Draw (Red component):	15.7 mA @ 5.05 **VDC** (via 180 Ω Resistor)
Current Draw (Green component):	15.6 mA @ 5.05 **VDC** (via 110 Ω Resistor)

Module Mounting

The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

LED Manufacturers Specifications

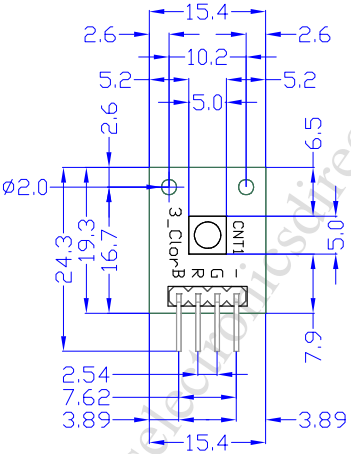
Manufacturer / Model:	Way Jun Technology / LED-0805RGBC
Red component wavelength:	620 - 635 nanometres [nm]
Green component wavelength:	500 - 530 nm
Blue component wavelength:	460 - 475 nm
Red component forward voltage (V_F):	1.8 volts [V] nominal, 2.4 V maximum
Green component forward current (V_F):	3.0 V nominal, 3.6 V maximum
Blue component forward current (V_F):	3.0 V nominal, 3.6 V maximum

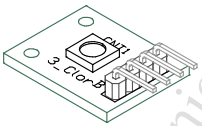
Projects

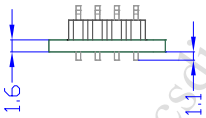
Folder: **\Modules\Optical\3_Colour_SMD_LED**

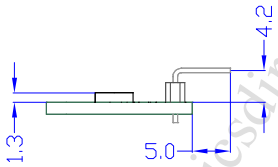
- **3_Colour_SMD_LED_SM**: Controls the module via the Arduino Serial Monitor.
- **3_Colour_SMD_LED_PWM_SM:** Control the module using **P**ulse **W**idth **M**odulation via the Arduino Serial Monitor.

3 Colour 5mm SMD LED Module - Dimensions









7 Colour 5mm THT Flashing LED Module

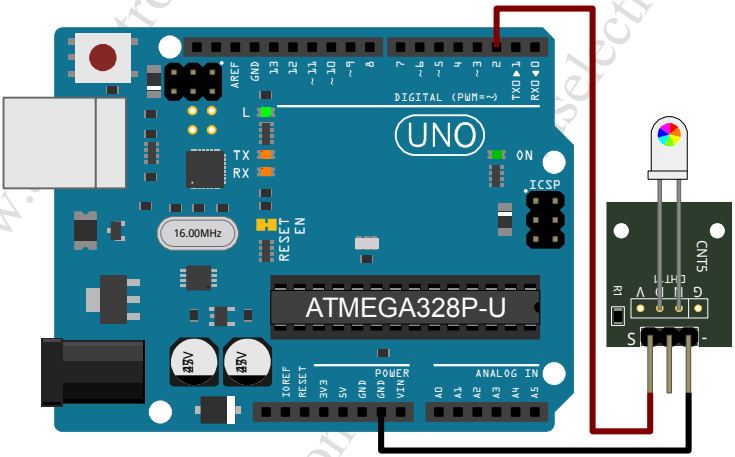
This module contains a 5 millimetre [mm] Through Hole Type waterclear Light Emitting Diode which cycles through 7 colours at various intervals. The initial pattern is 21 short flashes (3 cycles of 7 colours), then it cycles through 6 one second intervals with each of the 3 components (red, green, and blue) slowly increasing in intensity. At the end of each of the one second cycles where the light is at full intensity, the current can reach up to 50 milliamps [mA] for short periods. This is at the upper range of what an Arduino board can safely supply, and therefore it should not be used in a project where the LED will run continuously for long periods. The cycle restarts whenever the power supply is removed or disrupted.

This type of LED is commonly used in small ornaments, or for shop displays.

Table 26: 7 Colour 5mm THT Flashing LED Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Positive 5 Volts Direct Current supply to LED.
middle	NC		*Optional connection to "S" pin via 10 kiloohm [kΩ] resistor.*
-	GND	■	Ground connection.

D2: can be any digital pin.



The sketch below can be used to control the 7 Colour LED Module via the Arduino Serial Monitor.

```
int pLED = 2;
void setup () {
  digitalWrite ( pLED, LOW );
  Serial.begin( 9600 );
  while ( !Serial ) {
    ;
  }
  pinMode( pLED, OUTPUT );
  Serial.println( "0 to turn off, 1 to turn on" );
}
void loop () {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '0' ) {
      Serial.println( "Off" );
      digitalWrite( pLED, LOW );
    }
    if ( cInput == '1' ) {
      Serial.println( "On" );
```

```
      digitalWrite( pLED, HIGH );
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.2 × 16.6 × 1.6 mm
Enclosing Dimensions (H × W × D) :	36.8 × 16.6 × 7.5 mm
Weight:	1.53 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (Quick Cycle):	up to 24 mA
Current Draw (Longer Cycle):	up to 50 mA

Module Mounting

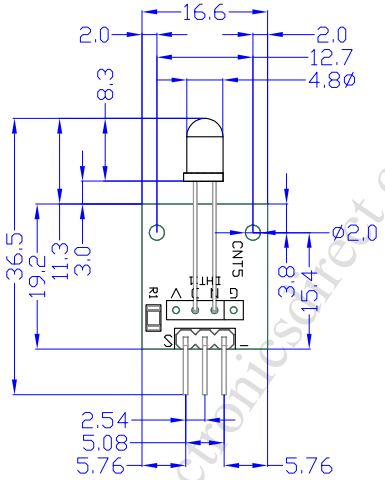
The module has 2 ×2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

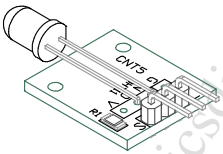
Projects

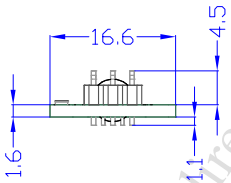
Folder: **\Modules\Optical\7_Colour_5mm_THT_Flashing_LED**

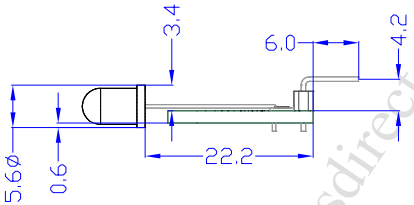
- **7_Colour_5mm_THT_Flashing_LED_SM**: Controls the 7 Colour 5mm **THT** Flashing **LED** Module via the Arduino Serial Monitor.
- **7_Colour_5mm_THT_Flashing_LED_Timed**: Turns the 7 Colour 5mm **THT** Flashing **LED** Module on and off at pre defined timer intervals.

7 Colour 5mm THT Flashing LED Module - Dimensions









Flame Sensor Module

This module contains a flame sensor in an **L**ight **E**mitting **D**iode package, which senses light in the spectral range of naked flames (760 to 1100 nanometres [nm]). It outputs an analogue signal based on the flame's intensity, distance, and direction. It also includes a digital output when a preset (threshold) level has been reached.

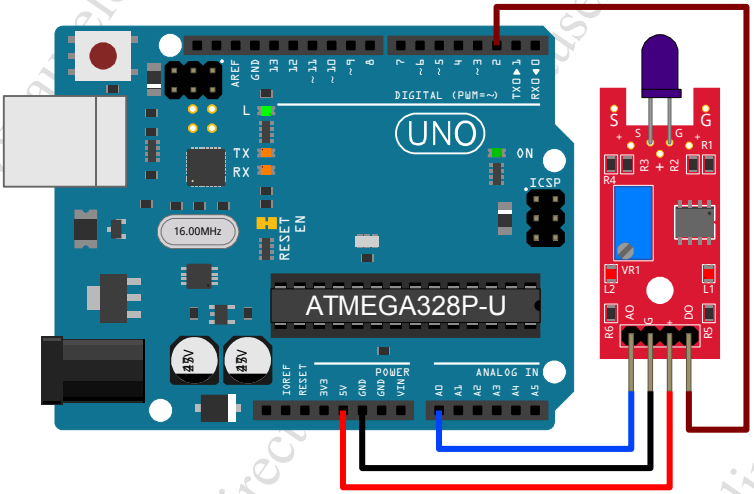
Flame sensors are typically used to monitor the state of the pilot light in gas based appliances. They are also used within fire detection systems, although when used for safety, they are used in conjunction with other components to provide additional levels of redundancy.

When set to maximum sensitivity (multi-turn potentiometer rotated fully clockwise), it can detect a candle in a dark room at around 2 metres [m], when the flame is directly line with the **LED**. As this module measures a specific range of light, the background (ambient) will also have an effect on its sensitivity. The field of view ranges from an angle of 180º at close proximity, to an angle of 2º at around 2 m.

Table 27: Flame Sensor Module Pin Connections

Device	Arduino	Wire	Description
AO	A0		Analogue output signal from Flame Sensor Module.
G	GND		Ground connection.
+	5V		5 **V**olts **D**irect **C**urrent positive supply for board circuitry.
DO	D2		Digital output signal for threshold trigger.

A0: can be any analogue pin, D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pDigitalFlame = 2;
int pAnalogueFlame = A0;
void setup() {
  Serial.begin( 9600 );
  pinMode( pDigitalFlame, INPUT );
}
void loop() {
  Serial.print( analogRead( pAnalogueFlame ), DEC );
  Serial.print(", ");
  Serial.println( digitalRead( pDigitalFlame ), DEC );
}
```

Flame Sensor Module…

Module Specifications

PCB Dimensions (H × W × D) :	35.9 × 15.6 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	47.5 × 15.6 × 14.3 mm
Weight:	2.93 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (Not triggered):	4.1 milliamps [mA] @ 4.9 **VDC**
Current Draw (Triggered):	6.8 mA

10 bit (0 - 1023) data for a candle positioned at 150 mm grid from the Flame Sensor Module.

The chart above is the raw flame detection data from the module, when a candle was placed in the centre of each 150 × 150 mm grid. The module was calibrated so the threshold was set just below the trigger level of the ambient light. The colours are derived from a spectrum with red hues being the most sensitive, through to blue colours being the least sensitive. The red concentric arcs are at 150 mm spacing.

Note As the sensor receives equally from both sides, only one side was tested, and the results were mirrored.

Module Mounting

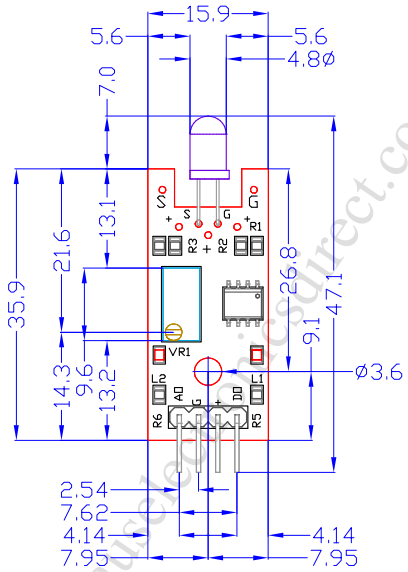
The module has a single 3.6 mm diameter mounting hole close to the end with the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

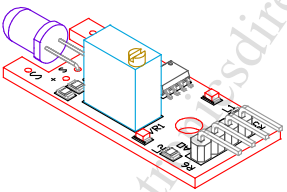
Projects

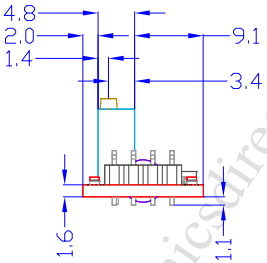
Folder: **\Modules\Optical\Flame_Sensor**

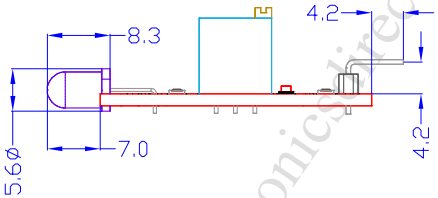
- **Flame_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Flame Sensor Module - Dimensions









Heartbeat Sensor Module

This module contains a 5 millimetre [mm] Through Hole Type opaque infra-red Light Emitting Diode and photo-transistor pair, which senses the opacity of a human finger tip placed between them.

This method of measuring blood flow is known as "photoplethysmography". As blood is pumped through the fingertip the opacity changes, with the results output via the the analogue output pin "S".

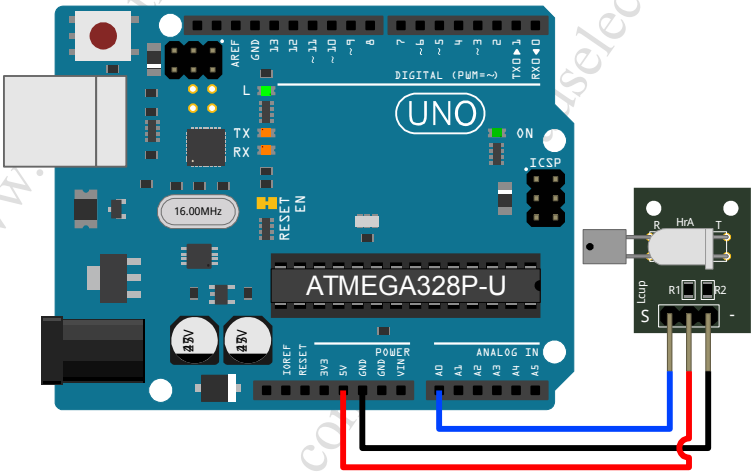
Because skin opacity varies, and, the size / shape of fingers differ, this module will require some trial and error to get the best results. Covering the module and finger from ambient light will also help with accuracy. Using the Serial Plotter of the Arduino Integrated Development Environment gives a good indication of the changes in opacity (pulse).

This module is only for entertainment only, not for medical purposes.

Table 28: Heartbeat Sensor Module Pin Connections

Device	Arduino	Wire	Description
S	A0		Analogue output signal from photo transistor circuitry.
middle	5V		Positive 5 Volts Direct Current supply for board circuitry.
-	GND		Ground connection.

A0: can be any analogue pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pHeartbeat = A0;
void setup() {
  Serial.begin( 9600 );
}
void loop() {
  Serial.println( analogRead( pHeartbeat ) );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.2 × 15.5 × 1.6 mm
Enclosing Dimensions (H × W × D):	24.9 × 22.2 × 17.7 mm
Weight:	1.26 grams [g]
Input Voltage:	5 Volts Direct Current

Module Performance

Current Draw:	12.0 milliamps [mA] @ 5.06 VDC

Heartbeat Sensor Module - Continued

Module Mounting

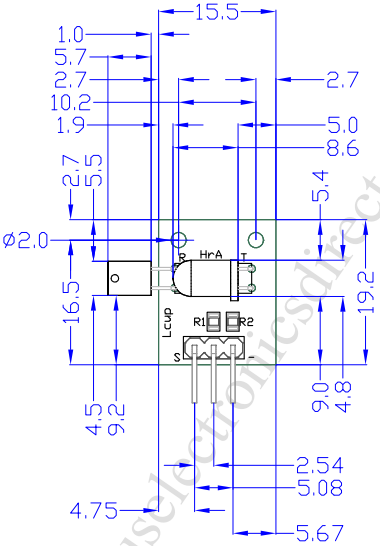
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

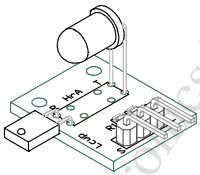
Projects

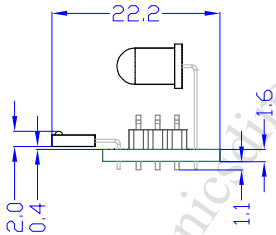
Folder: **Modules\Optical\Heartbeat_Sensor**:

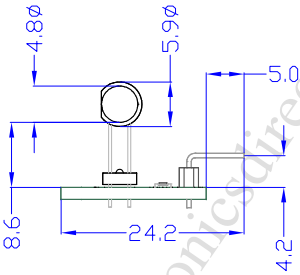
- **Heartbeat_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Heartbeat Sensor Module - Dimensions









Infrared Line Tracking Module

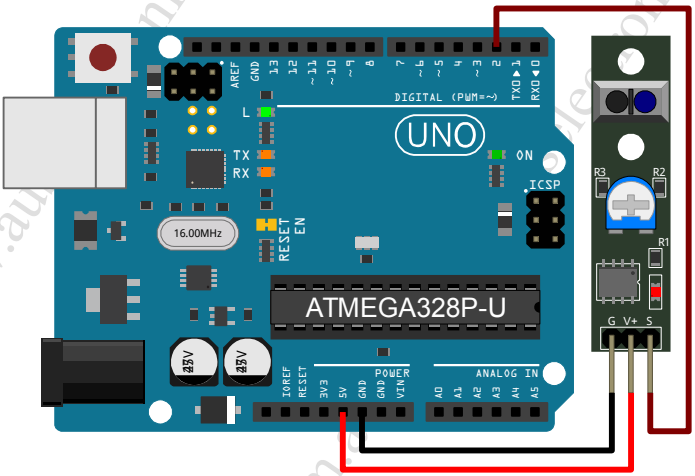
This module contains an integrated infrared reflection detector (transmitter / receiver pair) mounted parallel to the **P**rinted **C**ircuit **B**oard which detects between light and dark objects a short distance (up to 10 millimetres [mm]) from the detector's face. Its sensitivity is adjusted via the on-board 270º rotation potentiometer.

It is able to distinguish between light and dark objects such as a thick dark line drawn on a light surface. It outputs an analogue signal proportional to the infra-red reflection level of an object, and a digital signal when a preset level has been reached.

Table 29: Infrared Line Tracking Module Pin Connections

Device	Arduino	Wire	Description
G	GND	■	Ground connection.
V+	5V	■	Positive 5 **V**olts **D**irect **C**urrent supply for board circuitry.
S	D2	■	Digital output signal from infrared sensor.

D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pLineTrack = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pLineTrack, INPUT );
}
void loop() {
  Serial.println( digitalRead( pLineTrack ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	41.7 × 10.5 × 1.6 mm
Enclosing Dimensions (H × W × D):	47.5 × 10.5 × 11.4 mm
Weight:	2.24 grams [g]
Input Voltage:	5 **VDC**

Module Performance

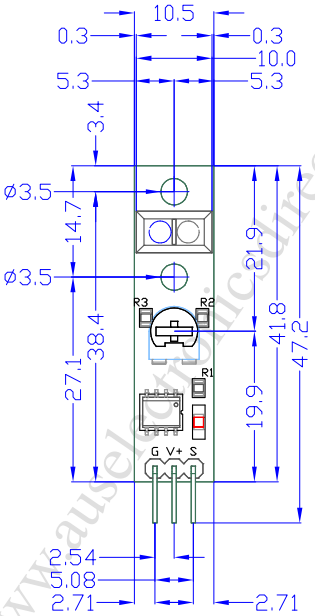
Current Draw (not triggered):	10.9 milliamps [mA] @ 5.01 **VDC**
Current Draw (triggered):	12.8 mA @ 5.03 **VDC**

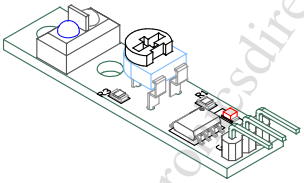
Projects

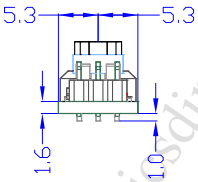
Folder: **Modules\Optical\Infrared_Line_Tracking**

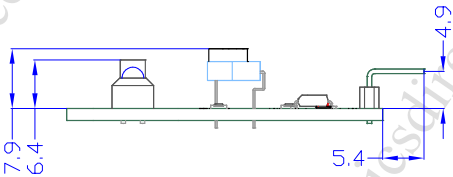
- **Infrared_Line_Tracking_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Infrared Line Tracking Module - Dimensions









Infrared Obstacle Avoidance Module

This module contains a medium distance infrared reflection detector (matched transmitter / receiver pair) mounted perpendicular to the **P**rinted **C**ircuit **B**oard. It outputs an analogue signal proportional to the amount of infra-red reflection from an object within a range of approximately 70 millimetres [mm]. The module can be set to sense continually ("**EN**" ("**Enable**" jumper fitted), or be triggered manually, by sending a signal to the "**EN**" pin. To use software (manual) detection, remove the "**EN**" jumper, then the detections are triggered by setting a digital pin to high. The signals will sent continually while there is a logic high signal present at the "**EN**" pin.

The trimpot (trimmer potentiometer) closest to the "**EN**" pin, adjusts the sensitivity (detection distance), with full anticlockwise being the most sensitive. The other trimpot is used for the frequency adjustment for the infrared transmitter. The sensitivity will be affected by the ambient light in the area where it will be used, the reflectivity of the surface, as well as the colours of the object being detected.

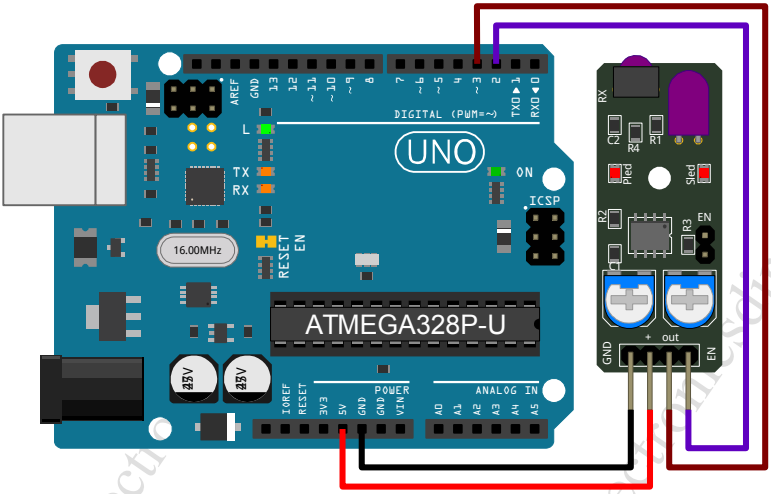
This type of component is commonly used in robotics for close range object detection, as well as being extensively used in the field of process control automation where it is used to sense the positions of rotary and linear actuators.

Note! The trimmer potentiometer closest to the "GND" controls the infrared transmitter frequency and has been set at the factory so it should not need adjusting. Make a note of its position. If it does get moved accidentality, the frequency across the transmitter's infrared Light Emitting Diode pins should be 38 kHz. It can be set by trial and error, by setting the trimmer potentiometer to 50% rotation, and then rotating the frequency adjustment trimpot in either direction until the best level of detection is found.

Table 30: Infrared Obstacle Avoidance Module Pin Connections

Device	Arduino	Wire	Description
GND	GND	■	Ground connection.
+	5V	■	Positive 5 **V**olts **D**irect **C**urrent supply for board circuitry.
OUT	D2	■	Digital output signal from infrared sensor.
EN	D3	■	*Optional: Manually trigger a read when EN jumper is not used.*

D2, D3: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pAvoidInput = 2;
int pAvoidEnable = 3;
void setup() {
  Serial.begin( 9600 );
  pinMode( pAvoidInput, INPUT );
}
```

```
void loop() {
  digitalWrite( pAvoidEnable, HIGH );
  Serial.println( digitalRead( pAvoidInput ), DEC );
  digitalWrite( pAvoidEnable, LOW );
}
```

Module Specifications

PCB Dimensions (H × W × D):	40.7 × 16.6 × 1.6 mm
Enclosing Dimensions (H × W × D):	46.7 × 16.6 × 11.3 mm
Weight:	4.13 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (not triggered):	4.4 milliamps [mA] @ 5.06 **VDC**
Current Draw (triggered):	6.2 mA @ 5.06 **VDC**

Module Mounting

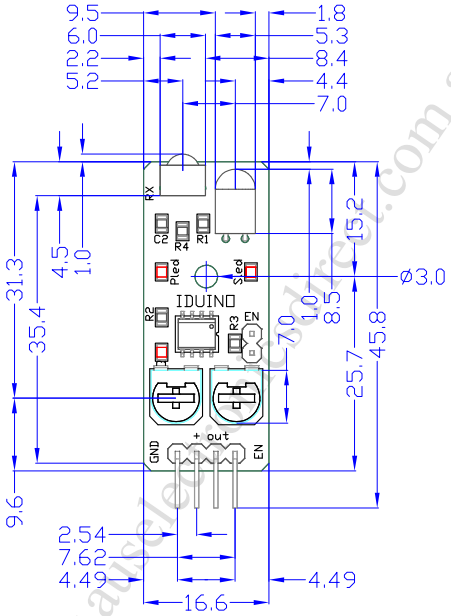
The module has a single 3 mm diameter mounting hole. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

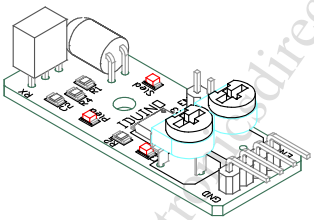
Projects

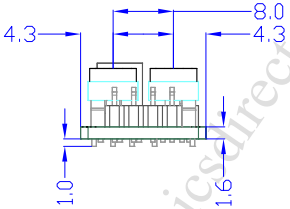
Folder: **\Modules\Optical\Infrared_Obstacle_Avoidance**

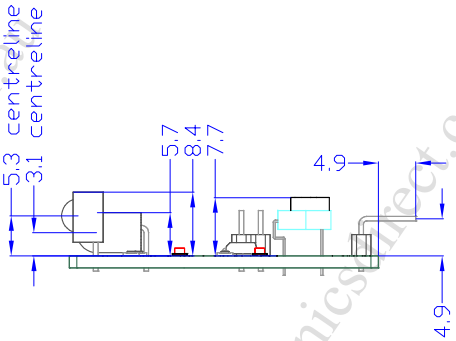
- **Infrared_Obstacle_Avoidance_SM**: Display the results to the Arduino Serial Monitor / Plotter.
- **Infrared_Obstacle_Avoidance_SM_Manual**: Uses a timer to set the enabled state, and, when the module is enabled, results are sent to the Serial Monitor.

Infrared Obstacle Avoidance Module - Dimensions









Infrared Receiver Module

This module contains an infra-red receiver **I**ntegrated **C**ircuit which decodes pulses of infra-red light for short range communications at a frequency of 38 kilohertz [kHz] (e.g. television, DVD player remote controls).

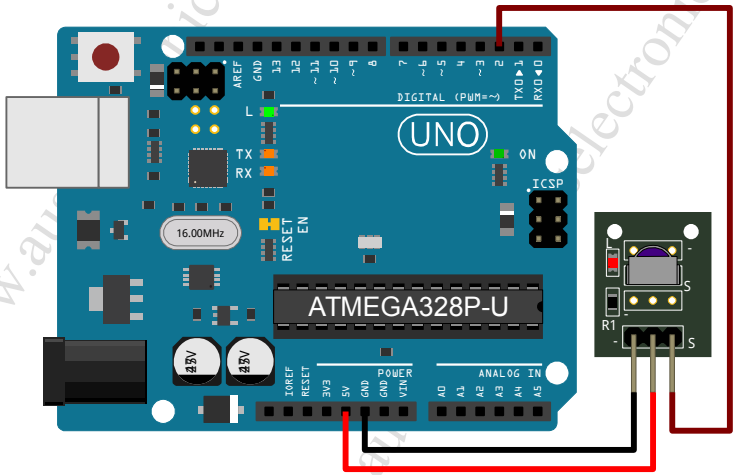
Some laptop computers also include infrared communications for short range low bandwidth data transfer.

This module can be used in conjunction with the Infrared Transmitter Module to create short range custom communication systems.

Table 31: Infrared Receiver Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	5 **V**olts **D**irect **C**urrent power supply to infrared receiver.
S	D2	■	Data output signal from infrared receiver.

D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
#include <IRremote.h>
int pIRReceive = 2;
IRrecv irrecv ( pIRReceive );
decode_results results;
void setup () {
  Serial.begin( 9600 );
  irrecv.enableIRIn();
}
void loop() {
  if (irrecv.decode ( & results ) ) {
    Serial.println( results.value, HEX );
    irrecv.resume();
  }
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.2 × 15.5 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	24.3 × 15.5 × 14.6 mm
Weight:	1.7 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Infrared Receiver Module…

Module Performance

Current Draw (idle):	0.46 milliamps [mA] @ 5.06 **VDC**
Current Draw (receiving):	0.94 mA @ 5.06 **VDC**

Module Mounting

The module has 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

Projects

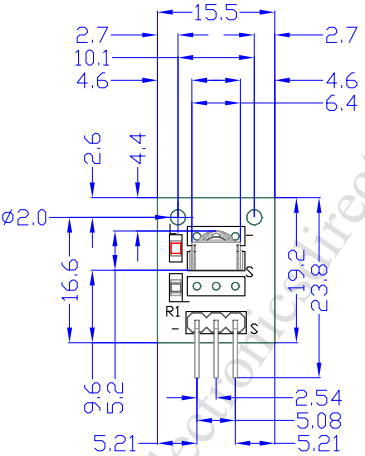
Folder: **Modules\Optical\Infrared_Receiver**

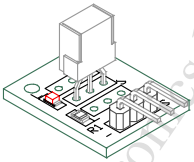
- **Infrared_Receiver_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

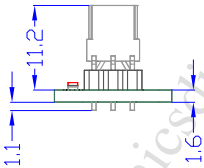
Libraries

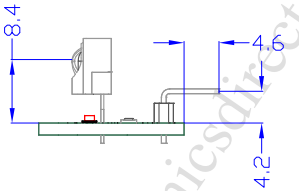
- **IRRemote**: Configures the timing for transmitting the infrared remote control codes.

Infrared Receiver Module - Dimensions









Infrared Transmitter Module

This module uses a 5 millimetre [mm] Through Hole Type infrared Light Emitting Diode for sending pulses to devices which communicate using common infrared transmission protocols. This module is designed to pulse the LED at short intervals as it draws a large current (~300 milliamps [mA]) under continuous use.

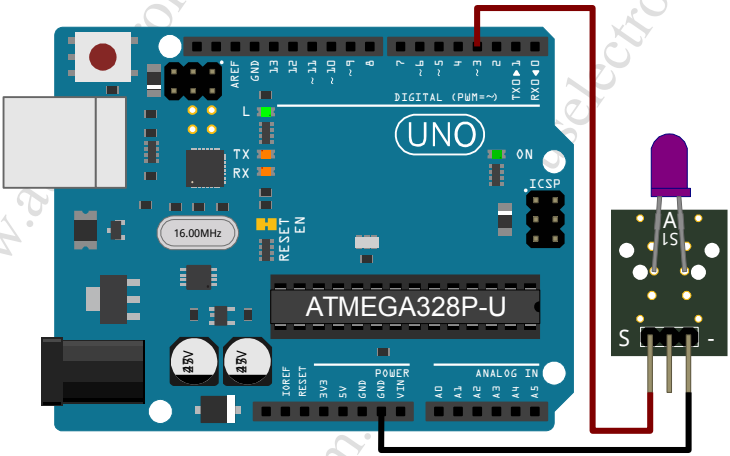
Infrared communication is widely used for the remote controls of common household devices such as televisions, DVD players, and is also used on some laptop computers for short range, low bandwidth data transfer.

This module can be used in conjunction with the Infrared Receiver Module to create short range custom communication systems.

Table 32: Infrared Transmitter Module Pin Connections

Device	Arduino	Wire	Description
S	D3	■	Connects to positive terminal of the infrared LED.
middle	NC		No connection.
-	GND	■	Connects to ground terminal of the infrared LED.

D3: must be used as it is hard-coded within the IRRemote library.



The sketch below can be used to transmit a "Sony" infrared code by sending "1" from the Arduino Serial Monitor.

```
#include <IRremote.h>
int pIRLED = 3;
IRsend irsend;
void setup () {
  Serial.begin( 9600 );
}
void loop() {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '1' ) {
      Serial.println( "Sent" );
      irsend.sendSony(0xa90, pIRLED);
    }
  }
}
```

Module Specifications

PCB Dimensions (H × W × D) :	19.4 × 15.4 × 1.6 mm
Enclosing Dimensions (H × W × D) :	33.3 × 15.4 × 7.2 mm
Weight:	1.37 grams [g]
Input Voltage:	5 Volts Direct Current

Infrared Transmitter Module…

Module Performance

Current Draw (continuous):	~300 milliamps [mA]

Infrared LED Specifications

Forward Voltage:	1.25 **VDC** (typical) 1.5 **VDC** (max)
Peak Forward Current:	1.0 amps [A]
Viewing Angle:	20º
Peak Wavelength:	940 nanometres [nm]

Module Mounting

The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **PCB**, suitable spacers and insulation must be used.

Projects

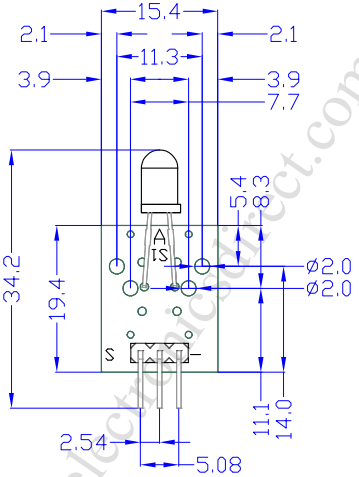
Folder: **Modules\Optical\Infrared_Transmitter**

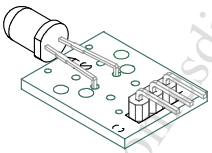
- **Infrared_Transmitter_SM**: Transmits a "Sony" infrared code by sending "1" from the Arduino Serial Monitor.

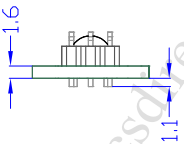
Libraries

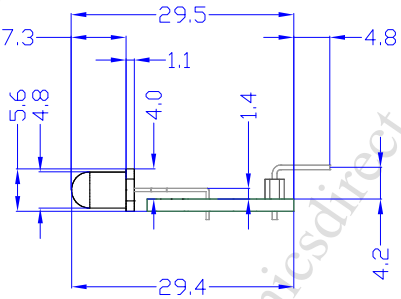
- **IRRemote**: Configures the timing for transmitting the infrared remote control codes.

Infrared Transmitter Module - Dimensions









LASER Module

A LASER (**L**ight **A**mplification by **S**timulated **E**mission of **R**adiation) module which produces a polarised beam of light in the 650 nanometre [nm] (red colour) range.

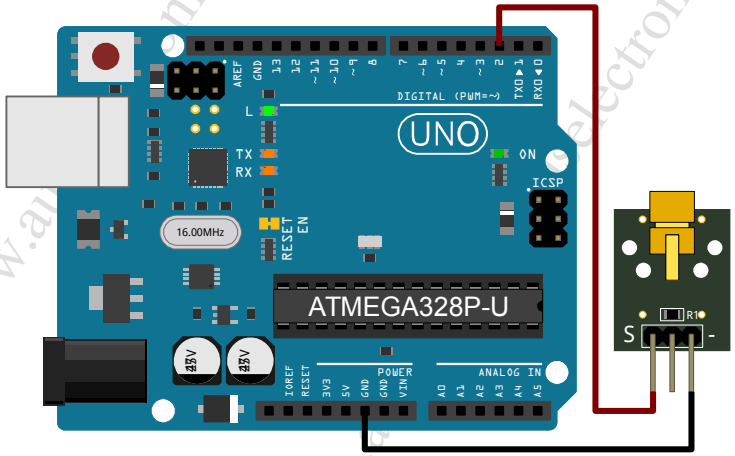
Low voltage **LASER**'s are used in Barcode Scanners, **D**igital **V**ersatile **D**isc players, and in many devices where fast optical based switching is required.

Warning! While the output of the LASER module is less than 1 milliwatt, this can still cause permanent injury if shone into the eyes.

Table 33: LASER Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Positive 5 **V**olts **D**irect **C**urrent supply to **LASER** module.
middle	NC		Connects to "S" pin via 10 kiloohm [kΩ] resistor.
-	GND	■	Ground connection.

D2: can be any digital pin.



The sketch below can be used to control the LASER module via the Arduino Serial Monitor.

```
int pLASER = 2;
void setup () {
  Serial.begin( 9600 );
  while ( !Serial ) {
    ;
  }
  pinMode ( pLASER, OUTPUT );
  digitalWrite ( pLASER, LOW );
  Serial.println( "0 to turn off, 1 to turn on" );
}
void loop () {
  if ( Serial.available() > 0 ) {
    int cInput = Serial.read();
    if ( cInput == '0' ) {
      Serial.println( "Off" );
      digitalWrite ( pLASER, LOW );
    }
    if ( cInput == '1' ) {
      Serial.println( "On" );
      digitalWrite ( pLASER, HIGH );
    }
```

```
  }
}
```

Module Specifications

PCB Dimensions (H × W × D):	18.8 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	25.8 × 15.4 × 9.3 mm
Weight:	2.43 grams [g]
Input Voltage:	5 **VDC**

Module Performance

Current Draw (**LASER** on):	27.9 milliamps [mA] @ 5.02 **VDC**

Module Mounting

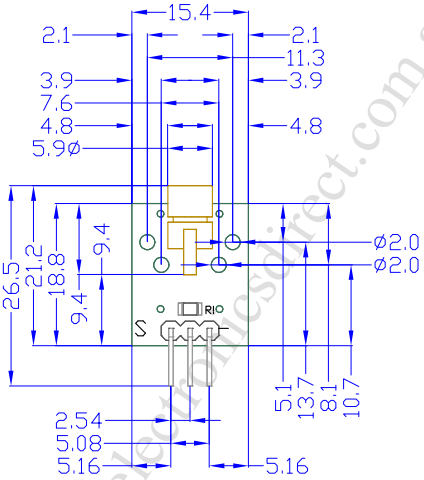
There are 2 × 2 mm mounting holes closes to the edge of the **P**rinted **C**ircuit **B**oard. The bare component leads protrude through the **PCB**, so suitable spacers and insulation will be required.

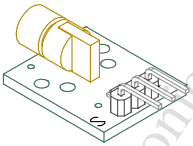
Projects

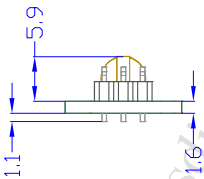
Folder: **Modules\Optical\LASER**

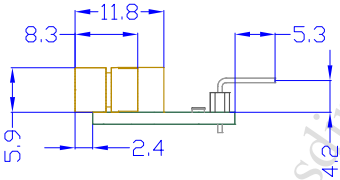
- **LASER_SM**: Controls the **LASER** module via the Arduino Serial Monitor.
- **LASER_Timed**: Controls the **LASER** using preset on and off durations.

LASER Module - Dimensions









Light Dependent Resistor Module

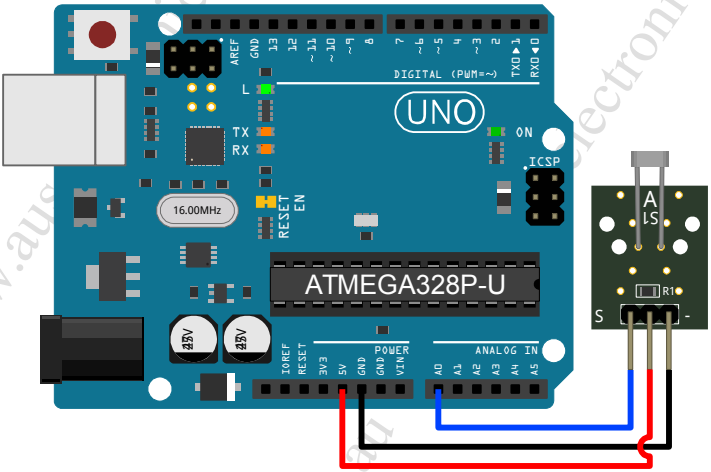
This module includes a **L**ight **D**ependent **R**esistor (also referred to as a photoresistor) which varies in resistance proportional to the amount of light received at its face.

LDR's are commonly used to activate the flash on camera's when the required light is below a preset level. They are also used to intelligently adjust the screen brightness of smart devices and laptop computers based on the ambient light level.

Table 34: Light Dependent Resistor Module Pin Connections

Device	Arduino	Wire	Description
S	A0		Connects to left side of the **LDR** component.
middle	5V		Connects to "S" pin via on-board 10 kiloohm [kΩ] resistor.
-	GND		Connects to right side of the **LDR** component.

A0: can be any analogue pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pLDR = A0;
void setup() {
  Serial.begin( 9600 );
}
void loop() {
  Serial.println( analogRead( pLDR ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D) : 19.0 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) : 28.6 × 15.4 × 7.5 mm
Weight: 1.29 grams [g]
Input Voltage: 5 **V**olts **D**irect **C**urrent

Module Performance

Current Draw (0% light): 0.02 milliamps [mA]
Current Draw (100% light): 0.41 mA

LUX	0 Dark	25	50	100	1000	8000 Light
10 bit value	970	331	256	230	40	12
%	94.7%	32.3 %	25.0 %	22.5 %	3.9 %	1.17 %

Module Mounting

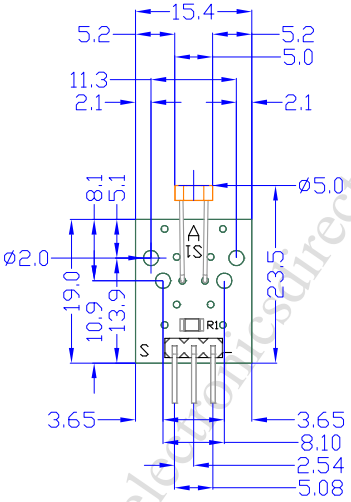
The module has 4 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

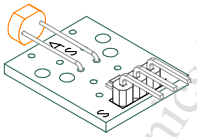
Projects

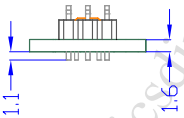
Folder: **Modules\Optical\Light_Dependent_Resistor**

- **Light_Dependent_Resistor_SM**: Displays the results to the Arduino Serial Monitor /Plotter.

Light Dependent Resistor - Dimensions







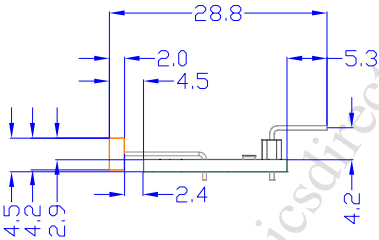


Photo Interrupter Module

This module includes a photo-interrupter component, which is a matched infrared **L**ight **E**mitting **D**iode / photo-transistor pair which face each other, with a small gap between them. When the **LED**'s light is obstructed the module returns a logic high signal.

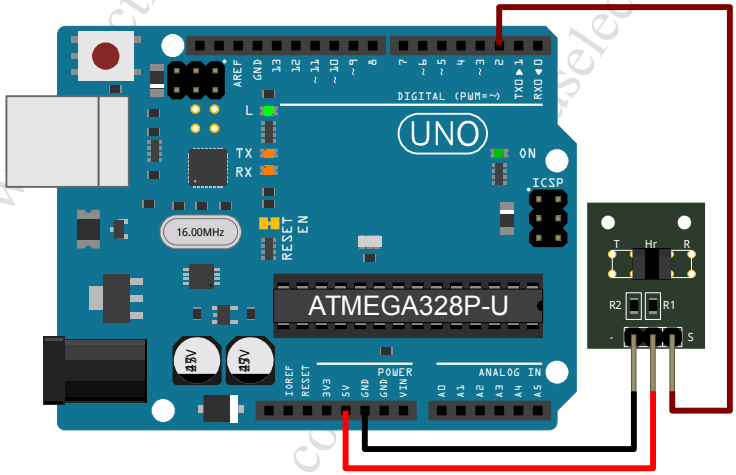
Photo-interrupters are most commonly used to determine the position of a rotary shaft. (e.g. printers, automotive ignition distributors). They can also be used as limit switches for linear devices such as the end stops. Because they use optics, they should not be used in an environment where dirt or dust can settle in the air gap and interfere with the infra-red beam.

Table 35: Photo Interrupter Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	5 **V**olts **D**irect **C**urrent power supply to photo interrupter component.
S	D2	■	Digital output signal from photo interrupter component.

D2: can be any digital pin.

*The image below shows the pin labels when viewing the module from the photo-interrupter side. On the actual **P**rinted **C**ircuit **B**oard of the module, they are printed on the underside.*



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pPhotoInterrupter = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pPhotoInterrupter, INPUT );
}
void loop() {
  Serial.println( digitalRead( pPhotoInterrupter ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.3 × 15.6 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	25.3 × 15.6 × 8.4 mm
Weight:	1.32 grams [g]
Input Voltage:	5 **VDC**
Output Voltage (Open):	0.3 **VDC**
Output Voltage (Closed):	4.2 **VDC**

Photo Interrupter Module…

Module Performance

Current Draw (not triggered): 21.5 milliamps [mA] @ 4.94 **VDC**
Current Draw (triggered): 20.2 mA @ 4.94 **VDC**

Module Mounting

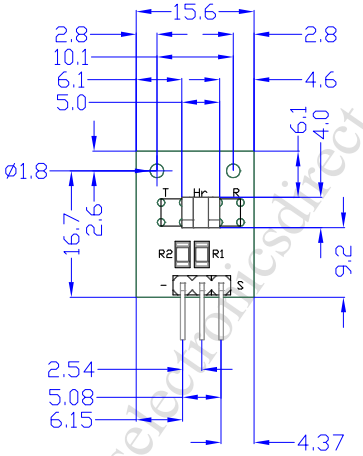
The module has 2 ×1.8 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the PCB, suitable spacers and insulation must be used.

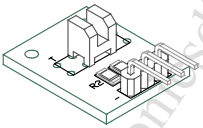
Projects

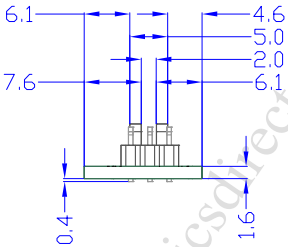
Folder: **Modules\Optical\Photo_Interrupter**

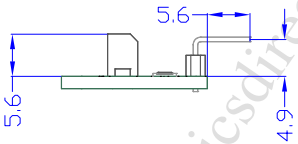
- **Photo_Interrupter_SM**: Displays the results to the Arduino Serial Monitor / Plotter.
- **Photo_Interrupter_LCD_Shield**: Displays the results to the screen of a **L**iquid **C**rystal **D**isplay Shield.

Photo Interrupter Module - Dimensions









Magic Light Cup Module

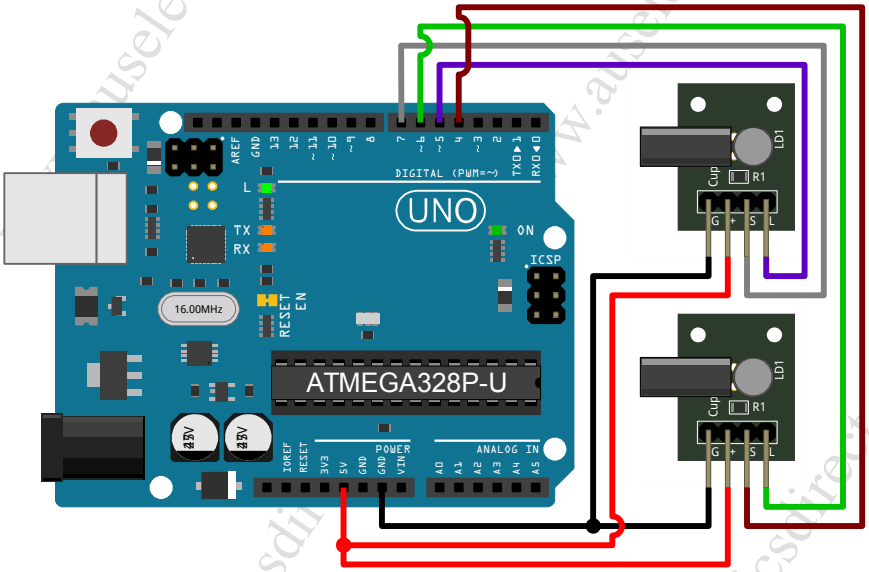
A pair of identical modules each containing a 5 mm Through Hole Type red Light Emitting Diode, and a rolling ball tilt switch.

Each module is placed in the bottom of a clear glass with one glass being held vertically, and the other being held horizontally. When the orientation of the glasses is reversed, it creates the effect of light pouring from one glass into the other.

Table 36: Magic Light Cup Module Pin Connections

Device	Arduino	Wire	Description
#1:G	GND		Ground connection.
#1:+	5V		5 VDC power supply. Link to S via 10 kiloohm resistor.
#1:S	D7		Connects to ground when tilt switch is conducting.
#1:L	P5		Positive 5VDC supply to LED.
#2:G	GND		Ground connection.
#2:+	5V		5 VDC power supply. Link to S via 10 kiloohm resistor.
#2:S	D4		Connects to ground when tilt switch is conducting.
#2:L	P6		Positive 5VDC supply to LED.

D2, D3: can be any digital pin. P5, P6: can be any Pulse Width Modulation capable digital pin.



The sketch below displays the results of each PWM output to the Arduino Serial Monitor / Plotter.

```
int pLEDA = 5;
int pLEDB = 6;
int pTiltA = 7;
int pTiltB = 4;
int bTiltA = 0;
int bTiltB = 0;
int iPWMA = 0;
int iPWMB = 255;
void setup() {
  Serial.begin( 9600 );
  pinMode( pLEDA, OUTPUT );
```

```
  pinMode( pLEDB, OUTPUT );
  pinMode( pTiltA, INPUT );
  pinMode( pTiltB, INPUT );
}
void loop() {
  bTiltA = digitalRead( pTiltA );
  if ( bTiltA == HIGH && iPWMA != 255 ) {
    iPWMA ++;
  }
  if ( bTiltA == LOW && iPWMA != 0 ) {
    iPWMA --;
  }
  analogWrite( pLEDB, iPWMA );
  Serial.print( iPWMA );
  Serial.print( "," );
  bTiltB = digitalRead( pTiltB );
  if ( bTiltB == HIGH && iPWMB != 0 ) {
    iPWMB --;
  }
  if ( bTiltB == LOW && iPWMB != 255 ) {
    iPWMB ++;
  }
  analogWrite( pLEDA, iPWMB );
  Serial.println( iPWMB );
  delay( 5 );
}
```

Module Specifications (each)

PCB Dimensions (H × W × D) :	19.5 × 15.5 × 1.6 millimetres [m]
Enclosing Dimensions (H × W × D) :	23.5 × 20.4 × 18.4 mm
Weight:	1.90 grams [g] each
Input Voltage:	5 **V**olts **D**irect **C**urrent

Module Performance (each)

Current Draw (LED lit):	34.3 milliamps [mA] @ 4.55 **VDC**

Module Mounting

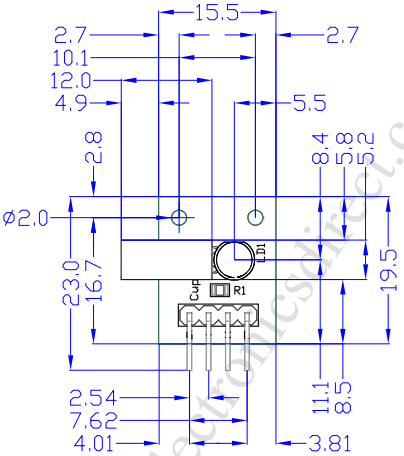
The module has 2 × 2 millimetre [mm] diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

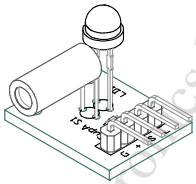
Projects

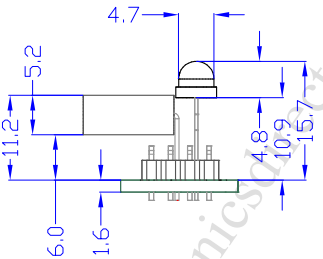
Folder: **\Other\Magic_Light_Cup**

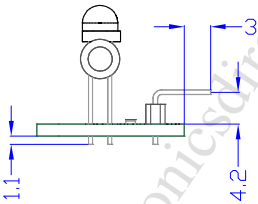
- **Magic_Light_Cup_SM**: Displays the results of each PWM output to the Arduino Serial Monitor / Plotter.

Magic Light Cup Module- Dimensions









Ball Switch Module

This module contains a ball switch, which detects when the module is tilted from its resting position (**P**rinted **C**ircuit **B**oard horizontal). Internally, the ball switch component contains an enclosed metal ball which is free to move within its enclosure, and, when the module is tilted around 45º from its horizontal position, it moves to either end and bridges a set of contacts to close the circuit.

Because the ball only has a small mass, it does not apply a great amount of pressure against the contacts, and therefore the resistance across the contacts can vary significantly. This is more pronounced if the switch is tilted slowly. This type of switch can also be affected by vibration.

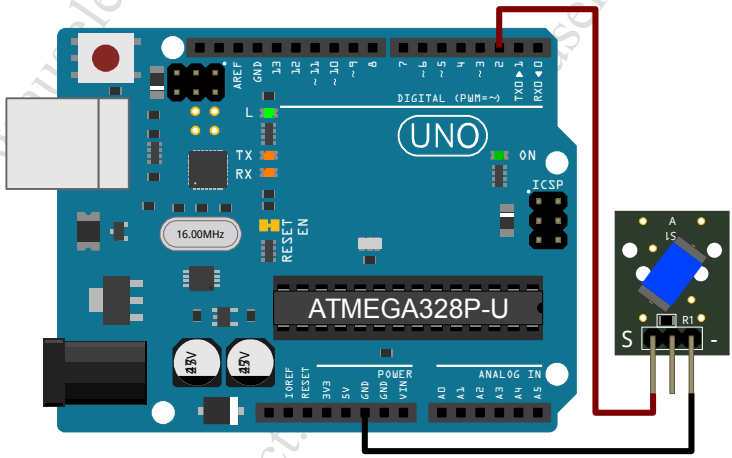
Ball switches are commonly used in motorcycle alarm systems, or, to monitor the maximum extent of rotation in devices such as robotic arms.

The difference between this module and the Tilt Switch Module is that this module senses angular movement in any direction.

Table 37: Ball Switch Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Connects to left side of Ball Switch.
middle	NC		Connects to left side of Ball Switch via on-board 10 kiloohm [kΩ] resistor.
-	GND	■	Connects to right side of Ball Switch.

D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial Monitor / Plotter.

```
int pBallSwitch = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pBallSwitch, INPUT );
  digitalWrite ( pBallSwitch, HIGH );
}
void loop() {
  Serial.println( digitalRead( pBallSwitch ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.1 × 15.2 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	24.8 × 15.2 × 7.2 mm
Weight:	1.40 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Ball Switch Module…

Module Performance

Trigger Angle: ~ 60º from horizontal plane

Module Mounting

The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the PCB, suitable spacers and insulation must be used.

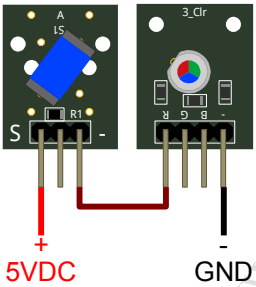
Projects

Folder: **\Modules\Physical\Ball_Switch**

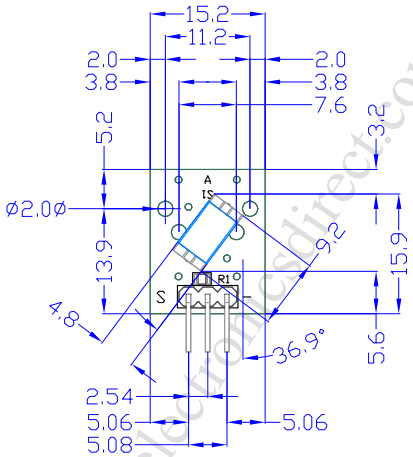
- **Ball_Switch_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

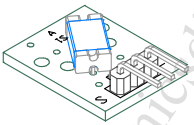
Notes

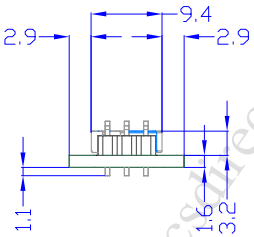
The module can be tested without an Arduino by connecting it as a switch for the 3 Colour 5 mm **T**hrough **H**ole **T**ype **L**ight **E**mitting **D**iode.

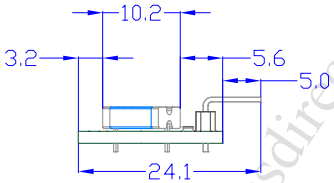


Ball Switch Module- Dimensions









Hit Sensor Module

This module contains a horizontally mounted spring sensor which has one end fixed, and the other end free to vibrate and touch a contact. The sensor is most sensitive when being bumped at right angles to the spring. The 2 outer pins connect directly to the hit sensor, and the centre pin connects to the "S" pin for use as a pull up / pull down resistor.

Because of the short amount of time the contacts remain closed, this component cannot be tested with a standard multi-meter, however, it can be connected in-line, as a switch, with a simple **LED**, resistor, battery circuit to visually display its results (schematic included at the end of this topic). To handle the issue of reading multiple fast results, Arduino code must either; check the state of the contacts continuously, or, use a pin which has the capability to be assigned to an interrupt.

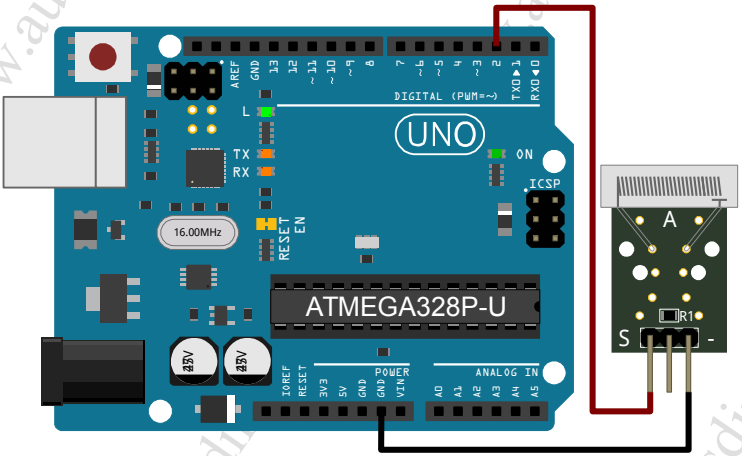
Hit and shock sensors are typically used in robotics, to sense when a collision has taken place. As with any device which monitors mechanical movement, background vibration will have a significant effect on its performance.

While this module is similar in functionality to the "Shock Sensor Module", they differ in the amount of force required for the internal spring to make contact.

Table 38: Hit Sensor Module Pin Connections

Device	Arduino	Wire	Description
S	D2		Connects to left side of the Hit Sensor.
middle	NC		Connects to left side of the Hit Sensor via 10 kiloohm [kΩ] resistor.
-	GND		Connects to right side of the Hit Sensor.

D2: can be any digital pin.



The sketch below can be used to display the results to the Arduino Serial / Plotter.

```
int pHitSwitch = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pHitSwitch, INPUT );
  digitalWrite( pHitSwitch, HIGH );
}
void loop() {
  Serial.println( digitalRead( pHitSwitch ), DEC );
}
```

Hit Sensor Module…

Module Specifications

PCB Dimensions (H × W × D):	19.3 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	30.8 × 18.2 × 8.2 mm
Weight:	1.70 grams [g]
Input Voltage:	<=5 **V**olts **D**irect **C**urrent

Module Mounting

The module has 4 × 2 mm diameter mounting holes towards the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

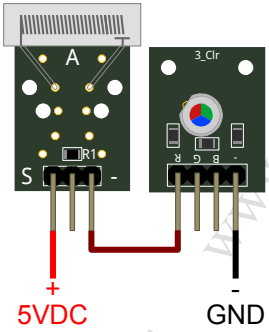
Projects

Folder: **\Modules\Physical\Hit_Switch**

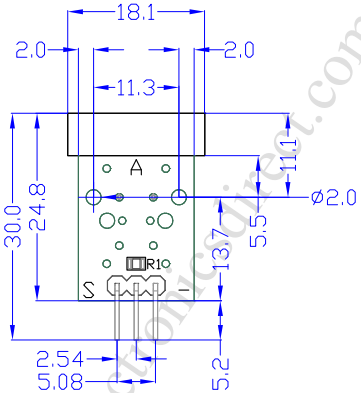
- **Switch-Hit_SM**: Displays the results to the Arduino Serial / Plotter.

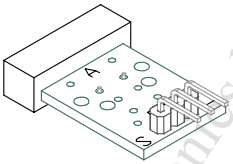
Notes

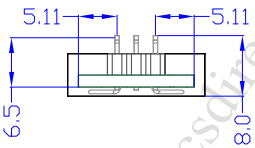
The module can be tested without an Arduino by connecting it as a switch for the 3 Colour 5mm THT LED Module.

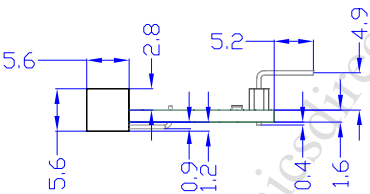


Hit Sensor Module - Dimensions









Shock Sensor Module

This module contains a horizontally mounted spring sensor which has one end fixed, and the other end free to vibrate and touch a contact. The sensor is most sensitive when being bumped at right angles to the spring. The 2 outer pins connect directly to the hit sensor, and the centre pin connects to the "S" pin for use as a pull up / pull down resistor.

Because of the short amount of time the contacts remain closed, this component cannot be tested with a standard multi-meter, however, it can be connected in-line, as a switch, with a simple **LED**, resistor, battery circuit to visually display its results (schematic included at the end of this topic). To handle the issue of reading multiple fast results, Arduino code must either; check the state of the contacts continuously, or, use a pin which has the capability to be assigned to an interrupt.

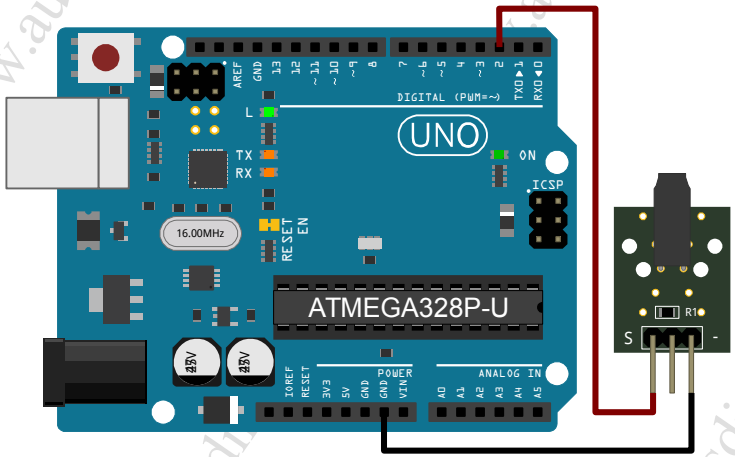
Hit and shock sensors are typically used in robotics, to sense when a collision has taken place. As with any device which monitors mechanical movement, background vibration will have a significant effect on its performance.

While this module is similar in functionality to the "Hit Sensor Module", they differ in the amount of force required for the internal spring to make contact.

Table 39: Shock Sensor Module Pin Connections

Device	Arduino	Wire	Description
S	D2	■	Connects to left side of the Shock Sensor.
middle	NC	■	Connects to left side of the Shock Sensor via 10 kiloohm [kΩ] resistor.
-	GND	■	Connects to right side of the Shock Sensor.

D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pShockSwitch = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pShockSwitch, INPUT );
  digitalWrite( pShockSwitch, HIGH );
}
void loop() {
  Serial.println( digitalRead( pShockSwitch ), DEC );
}
```

Shock Sensor Module…

Module Specifications

PCB Dimensions (H × W × D) :	19.3 × 15.4 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D) :	28.8 × 15.4 × 7.3 mm
Weight:	1.41 grams [g]
Input Voltage:	<=5 **V**olts **D**irect **C**urrent

Module Mounting

The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

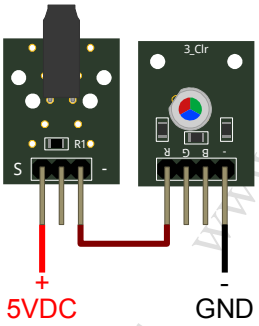
Projects

Folder: **\Modules\Physical\Shock_Sensor**

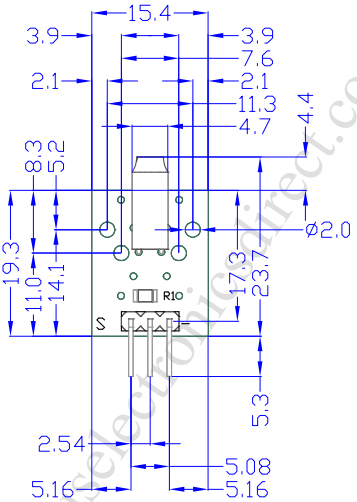
- **Shock_Sensor_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

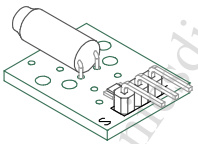
Notes

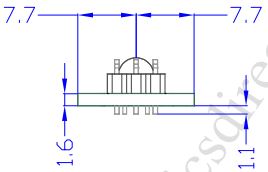
The module can be tested without an Arduino by connecting it as a switch for the 3 Colour 5mm THT LED Module.

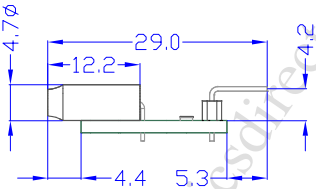


Shock Sensor Module - Dimensions









Tilt Switch Module

This module contains a ball type tilt switch, which is a heatshrink covered metal cylinder containing 2 small metal balls which are free to move along the length of the cylinder. When the ball is at the end where the connecting leads exit the cylinder, it bridges the ends of these leads and closes the circuit. The on-board **L**ight **E**mitting **D**iode illuminates when the internal contacts are bridged.

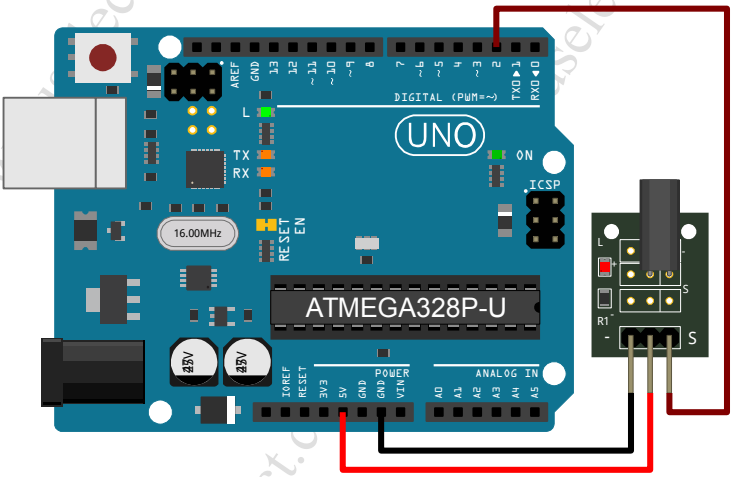
This type of tilt switch is best suited to applications where the angle of tilt, and speed of movement, are sufficient for the balls to "hit" the contacts with enough force to make a good connection. The plastic cylinder cap is only a "push" fit, so it should not be used in an environment where high levels of humidity or corrosive gasses could enter the cylinder and corrode the ball or its contacts.

The difference between this module and the Ball Switch module, is that this module only senses angular movement in line with the cylinder centre-line greater than 10° from the horizontal plane.

Table 40: Tilt Switch Module Pin Connections

Device	Arduino	Wire	Description
-	GND	■	Ground connection.
middle	5V	■	Positive 5 **V**olts **D**irect **C**urrent supply for board circuitry.
S	D2	■	Digital output signal from board circuitry.

D2: can be any digital pin.



The sketch below displays the results to the Arduino Serial Monitor / Plotter.

```
int pTiltSwitch = 2;
void setup() {
  Serial.begin( 9600 );
  pinMode( pTiltSwitch, INPUT );
  digitalWrite( pTiltSwitch, HIGH );
}
void loop() {
  Serial.println( digitalRead( pTiltSwitch ), DEC );
}
```

Module Specifications

PCB Dimensions (H × W × D):	19.1 × 15.5 × 1.6 millimetres [mm]
Enclosing Dimensions (H × W × D):	28.3 × 15.5 × 8.3 mm
Weight:	1.60 grams [g]
Input Voltage:	5 **V**olts **D**irect **C**urrent

Tilt Switch Module…

Module Performance

Trigger Angle:	~10º from horizontal plane
Current Draw (activated):	3.8 milliamps [mA] @ 5.05 **VDC**

Module Mounting

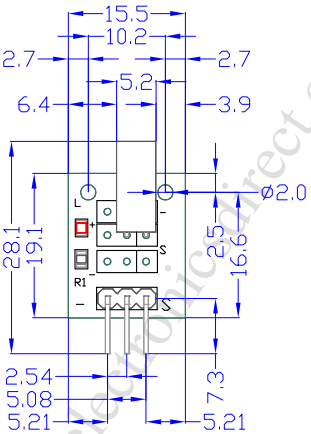
The module has 2 × 2 mm diameter mounting holes at the end opposite the pin connections. As the bare component leads protrude through the bottom of the **P**rinted **C**ircuit **B**oard, suitable spacers and insulation must be used.

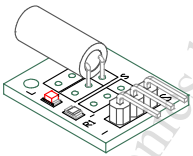
Projects

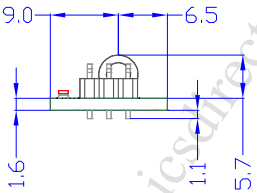
Folder: **\Modules\Physical\Tilt_Switch**

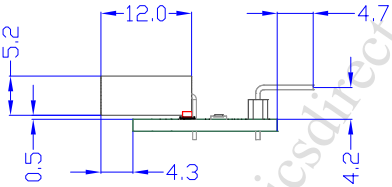
- **Tilt_Switch_SM**: Displays the results to the Arduino Serial Monitor / Plotter.

Tilt Switch Module - Dimensions









37 in 1 Kit - Available Spares

Some of the modules within this kit can be purchased from www.auselectronicsdirect.com.au separately for larger projects. Enter the SKU number into the "search box" at the top right of their web site, or, click on the links below to open your Internet Browser, with the keywords for the module automatically entered.

Modules Listed by Category

- **Audio:**
  - Small Microphone Module [SKU: TA0071]
- **Electromagnetic:**
  - Magnetic Sensor Module [SKU: TA0056]
  - Reed Switch Module [SKU: TA0058]
  - SPDT Relay Module [SKU: TA0044]
- **Environment:**
  - Analogue Temperature Sensor Module [SKU: TA0062]
- **Interface:**
  - Dual Axis Analogue Joystick Module [SKU: TA0051]
  - Rotary Encoder Module [SKU: TA0167]
- **Optical:**
  - 3 Colour 5 mm SMD LED Module [SKU: TA0076]
  - Flame Sensor Module [SKU: TA0159]
  - Infrared Line Tracing Module [SKU: TA0079]
  - Infrared Obstacle Avoidance Module [SKU: TA0046]
  - Infrared Receiver Module [SKU: TA0048]
  - Infrared Transmitter Module [SKU: TA0047]

Component Datasheets

The zip archive "37_Kit_Datasheets.zip" which accompanies this reference contains the datasheets for most of the components used in the modules in **P**ortable **D**ocument **F**ormat. As some components do not have distinguishing manufacture markings, the best match has been chosen based on the specifications of the components. Datasheet filenames are as defined by the manufacturer, so they can be entered into your preferred Internet Search Engine to search for updates.

- **Audio**
  - Active Buzzer: **ef532_ps.pdf**
  - Small Microphone: **lm393-n.pdf**, **3296.pdf**
  - Large Microphone: **lm393-n.pdf**, **3296.pdf**
- **Electromagnetic**
  - Analogue Hall Effect Sensor: **AH49E.pdf**
  - Linear Hall Effect Sensor: **AH49E.pdf**, **lm393-n.pdf**, **3296.pdf**
  - Magnetic Sensor: **3144.pdf**
  - Mini Reed Switch: **Reed Switches.pdf**
  - Reed Switch: **Reed Switches.pdf**, **lm393-n.pdf**, **3296.pdf**
  - SPDT Relay: **SRD-05VDC-SL-C-Datasheet.pdf**
- **Environment**
  - Analogue Temperature Sensor: **ntcappnote.pdf**
  - Digital Temperature Sensor: **ntcappnote.pdf**, **lm393-n.pdf**, **3296.pdf**
  - I^2C Digital Temperature Sensor: **DS18B20.pdf**
  - DHT11 Temperature and Humidity Sensor: **DHT11.pdf**
- **Interface**
  - Rotary Encoder: **TW-700198.pdf**
  - Touch Sensor: **lm393-n.pdf**,
- **Optical**
  - 3 Colour SMD LED: **SMD_LED.pdf**
  - Flame Sensor: **lm393-n.pdf**, **3296.pdf**
  - Infrared Line Tracking: **tcrt5000.pdf**,
  - Infrared Receiver: **30500.pdf**
  - LASER: **laser diode datasheet_0.pdf**
  - Light Dependent Resistor: **cds-resistor-pgm.pdf**,
  - Photo Interrupter: **ZD1901-dataSheetMain.pdf**,
- **Physical**
  - Ball Switch: **rb.pdf**