

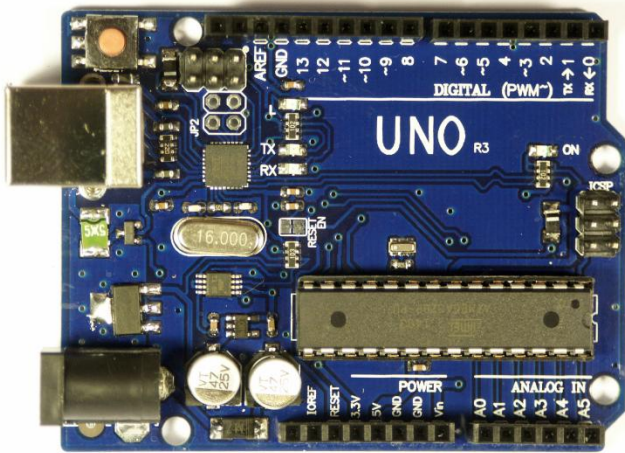


342014

Basic Starter Kit (Contents)

- 1x 50K Ω Pot
- 1x 6-cell AA Battery pack
- 1x 7-seg LED 4x module
- 1x 8x8 dot LED array
- 1x Buzzer (active)
- 1x Buzzer (passive)
- 1x Flame sensor
- 1x IC 74HC595N 16-pin DIP
- 1x IR receiver
- 1x IR remote control
- 1x LED - RGB
- 1x LM35 Temp Sensor
- 1x USB cable
- 2x 7-seg LED 1x module
- 2x Ball tilt sensor
- 3x Photo Resistor
- 4x Small button switch
- 5x 10K Ω resistor
- 5x 1K Ω resistor
- 5x LED - Blue
- 5x LED - Red
- 5x LED - Yellow
- 830-pin Breadboard
- 8x 220 Ω resistor
- Arduino Uno compatible board
- Dupont connector wires

Inland Uno R3:



UNO R3 Summary:

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

See <http://arduino.cc> for detailed specifications, overviews, schematics, etc. Core functions, code examples, and links to many of the device libraries can be found in the learning section; refer to the manufacturer's site if using other add-on shields or sensors.

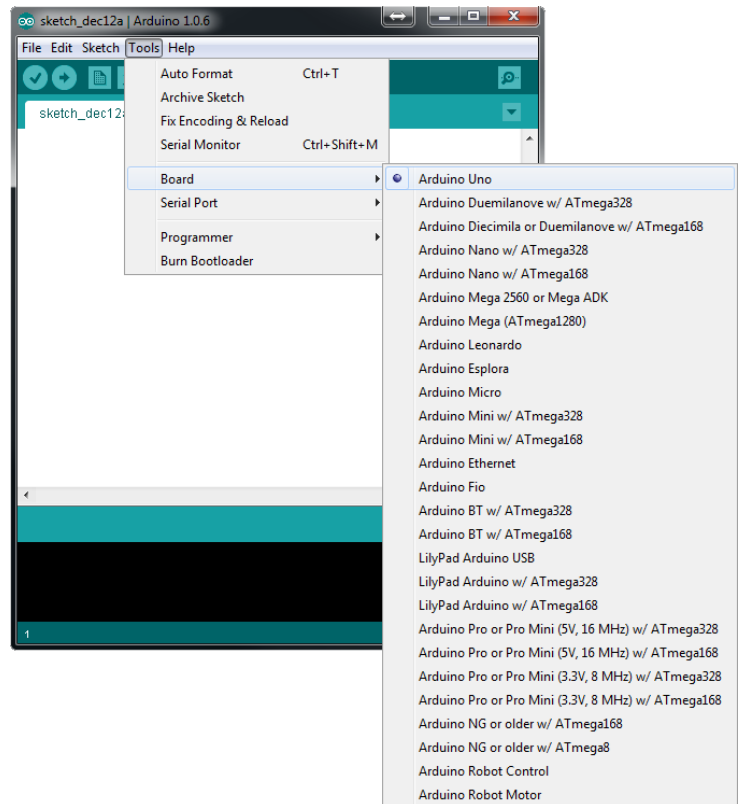
The latest Arduino Integrated Development Environment (IDE) necessary for programming your UNO R3 board can be obtained at <http://arduino.cc/en/Main/Software> (the **Download** menu choice on Arduino.cc)

Examples for many basic components can be found under the Examples menu. As you install libraries for additional shields, new examples may be available.

Follow the getting started guide found on the arduino.cc web site. Click **Learning**, and select **Getting started**. Click on the link for Windows, Mac OS X, or Linux for more specific directions.

Getting Started:

1. Download the Arduino Environment (IDE) and install or unzip/extract the application directory.
2. Connect the UNO board to one of your computer's USB port.
3. Install the drivers (If the computer does not automatically download and install the necessary USB drivers, point the hardware setup to the "**drivers**" directory of the Arduino IDE application.)
4. Launch the Arduino IDE application
5. Open a sketch example such as "Blink"
6. Select your **Board** from the Tools menu.
7. Select the **Serial Port** used by the board
8. Upload the sketch to the board

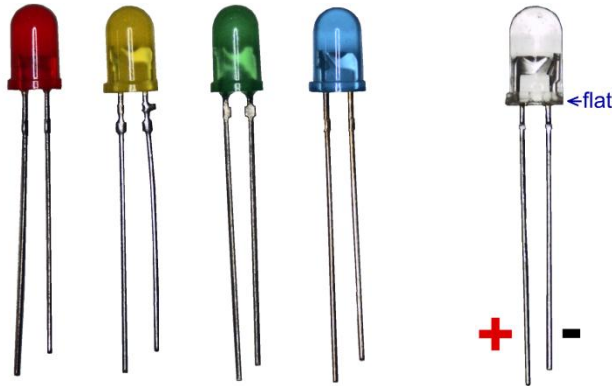


Sketch (code) Examples are included as part of the IDE. If you install device libraries for other components or shields, additional examples may be included and will show up in the list under the IDE File menu.

(See: <http://arduino.cc/en/Tutorial/HomePage> for an overview of the core functions and libraries.)

Components:

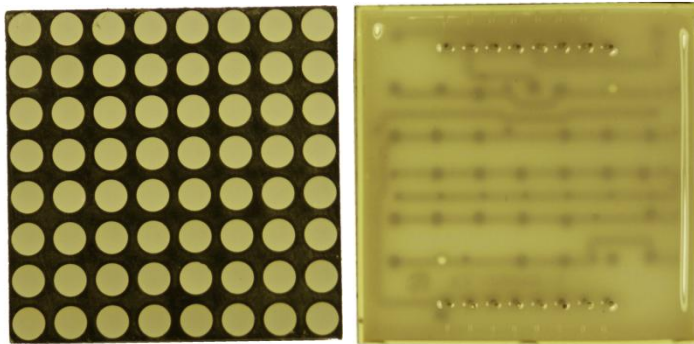
LEDs



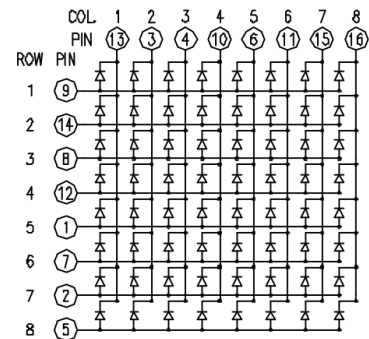
LED - Light Emitting Diodes

- 1) Connect a current-limiting resistor (220 ohm) between the LED's positive pin and the 5v pin. Connect the LED's negative pin directly to your Arduino output pin. -OR-
- 2) Connect a current-limiting resistor (220 ohm) between the Arduino output pin and the LED's positive pin. Connect the LED's negative pin directly to a Ground (GND) pin.

Note: LEDs may have "water clear" or color tinted lens.

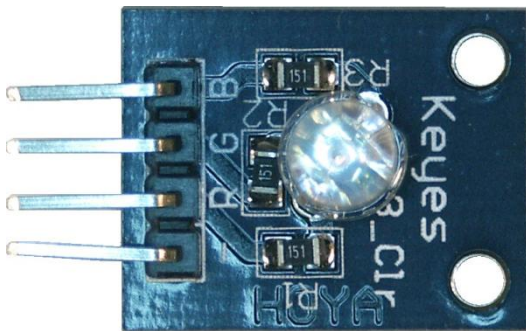


LED - 8x8 Matrix (1588BS or similar)



Connect Columns to Arduino Data pins that can be pulled to ground, connect columns using current limiting resistors to pins that will output positive voltage to illuminate the selected LED. See:

<http://arduino.cc/en/Tutorial/RowColumnScanning>



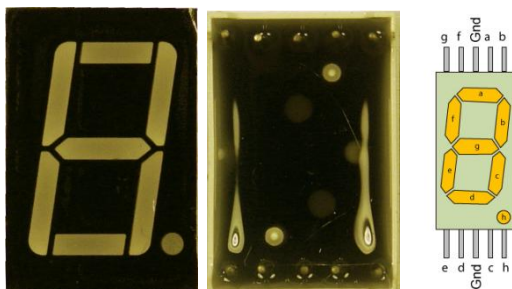
LED - RGB Module

Current-limiting resistors ($151 = 1.5 \times 10 = 150$ ohm) are already attached to the positive Red, Green, and Blue pins of the LED. Connect the negative (-) pin to your ground, and the R, G, and B pins to your Arduino output pins.

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

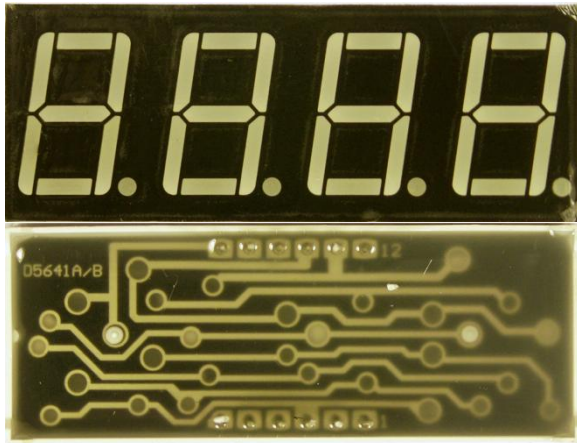
File, Examples, 01.Basic, **Fade** sketch example in the IDE

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

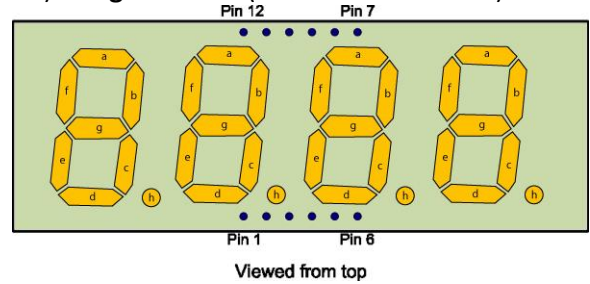


(1-bit) 7-segment LED (TOS5121AS or similar)

Pin 1 is bottom left. Pins 3 and 8 are a common ground. Connect other pins to your Arduino with a current limiting resistor.



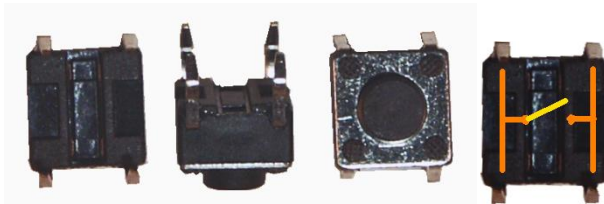
(4-bit) **7-segment LED** (SH5461AS or similar)



Pins 12, 9, 8, 6 are grounds for each segment; LED segments share pins 11(a), 7(b), 4(c), 2(d), 1(e), 10(f), 5(g), 3(h). Transistors are recommended to handle current that could exceed the maximum output of the Arduino pins. See:

<http://learn.parallax.com/4-digit-7-segment-led-display-arduino-demo>

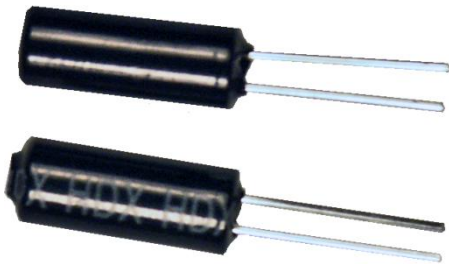
Switches



Small button switch - momentary contact, NO

For the switch connection, you can use either pair located on one side. The connection is Normally Open (off) until the button is pushed.

Sensors and modules



Ball Tilt Sensor

This is a very simple switch with a ball inside of the tube. When the sensor is tipped upward past the horizontal, the ball will short the contacts, closing the switch. With the top (away from the pins) is tilted down relative to the horizontal, the switch opens.



Flame Sensor (YG1006 or similar)

The Flame sensor is a high-speed and highly sensitive NPN Silicon photo transistor based on the YG1006. It can be used to detect fire or other wavelength at 760nm ~ 1100nm light. Response time is 15us, supply voltage is 3.3-5V; output is analog.

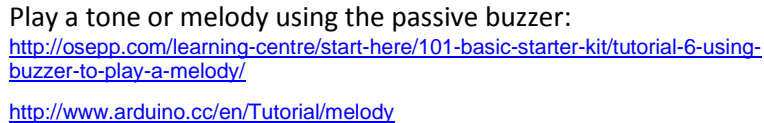


IR Receiver (VS1838B or similar)

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

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

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



Use as a speaker, buzzer or other audible indicator.

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

The passive buzzer does not have epoxy on the rear PCB, and the positive and negative connections are visible on the etched board. Passive buzzers must have a modulated signal supplied to the device (like a speaker) and would only generate a "click" if DC voltage is applied.



The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature.

Basic Temperature Sensor (+2° to +150°C):
+Vs=5V in, Ground, Vout = 0mV + 10.0mV/°C

For code examples, see:

<http://playground.arduino.cc/Main/LM35HigherResolution>

<http://www.learningaboutelectronics.com/Articles/LM35-temperature-sensor-circuit.php>

<http://pscmpf.blogspot.com/2008/12/arduino-lm35-sensor.html>

https://tkkrlab.nl/wiki/Arduino_KY-001_Temperature_sensor_module

8pcs 220 ohm
5pcs 1K ohm
5pcs 10K ohm

4-Band-Code

2%, 5%, 10% 560kΩ ± 5%

Color	1st Band	2nd Band	3rd Band	Multiplier	Tolerance
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	± 1% (F)
Red	2	2	2	100Ω	± 2% (G)
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	± 0.5% (D)
Blue	6	6	6	1MΩ	± 0.25% (C)
violet	7	7	7	10MΩ	± 0.10% (B)
Grey	8	8	8		± 0.05%
White	9	9	9		
Gold				0.1	± 5% (J)
Silver				0.01	± 10% (K)

5-Band-Code

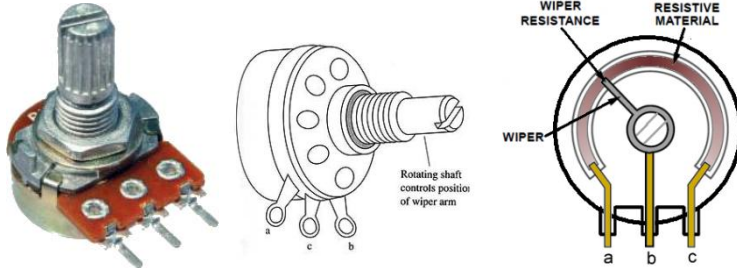
0.1%, 0.25%, 0.5%, 1% 2370 ± 1%

Resistors may come with 4 or 5 identifying color bands. (When in doubt, use a multimeter to verify the value.)



Photo Resistor

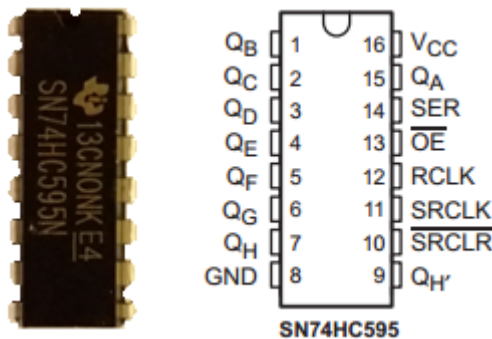
Resistance across the pins will be 1 meg ohm or higher in darkness, dropping to 60 ohms or less in bright light.



50K Potentiometer

Resistance between outer pins is 50K ohms. Resistance between one outer pin and the center (wiper) pin is 0-50K ohms based on position.

other



Tri-state 8-bit shift register IC SN74HC595N (or similar)

The datasheet refers to the 74HC595 as an "8-bit serial-in, serial or parallel-out shift register with output latches; 3-state." In other words, you can use it to control 8 outputs at a time while only taking up a few pins on your microcontroller.

See:

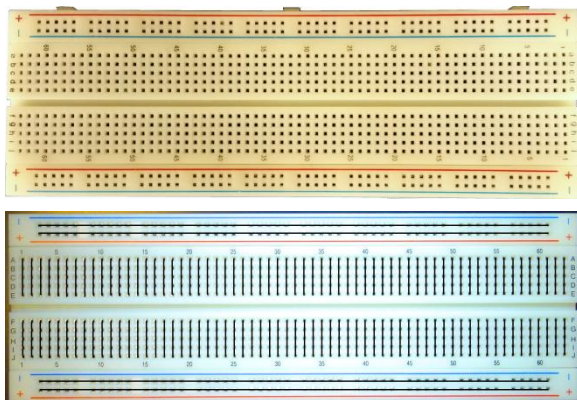
<http://arduino.cc/en/tutorial/ShiftOut>



IR Transmitter (Remote Control)

Multiple tutorials are available for different remotes, use serial monitoring to identify inputs from your remote control and use those values in your programs. For a typical setup, see:

<http://arduino-info.wikispaces.com/IR-RemoteControl>



830-pin Breadboard

Power rails run the length of each side and are color coded blue for negative and red for positive. Inside rows of 5 pins each are connected together, but not to each other, and not to the power rails.

Additional Resources:

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

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

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

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

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

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

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

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

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

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

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

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

Earthshine Electronics Beginners Guide to Arduino:

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

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

Suggested Projects

Arduino Tutorials and Examples	http://arduino.cc/en/Tutorial/HomePage
Osepp Starter Kit Tutorials	http://osepp.com/learning-centre/start-here/101-basic-starter-kit/
LED Game	http://osepp.com/learning-centre/start-here/101-basic-starter-kit/tutorial-4-an-led-game/
Voltage Meter	http://osepp.com/learning-centre/start-here/101-basic-starter-kit/tutorial-5-building-voltage-meter/
Tone Generation	http://osepp.com/learning-centre/start-here/101-basic-starter-kit/tutorial-6-using-buzzer-to-play-a-melody/
7-Segment LED Countdown Display	http://osepp.com/learning-centre/start-here/101-basic-starter-kit/tutorial-7-counting-down-with-a-7-segment-led/
Arduino Melody	http://www.arduino.cc/en/Tutorial/melody
Analog Read Voltage (example)	http://arduino.cc/en/Tutorial/ReadAnalogVoltage
4-Channel Volt Meter	http://startingelectronics.com/projects/arduino-projects/arduino-4-channel-LCD-voltmeter/
Measuring Voltage (serial display via IDE)	http://startingelectronics.com/articles/arduino/measuring-voltage-with-arduino/
Arduino Digital Voltmeter	http://www.electroschematics.com/9351/arduino-digital-voltmeter/