





342097

Uno Starter Kit (Contents)

1x LED - RGB

5x LED - Blue

5x LED - Red

5x LED - Yellow

1x 7-seg LED 1x module

1x 7-seg LED 4x module

1x 8x8 dot LED array

1x 2x16 LCD display

4x Large button switch

1x Buzzer (active)

1x Buzzer (passive)

2x Ball tilt sensor

1x Relay 5v

1x Flame sensor

1x IR receiver

1x Temp & Humidity

1x LM35 Temp Sensor

1x RTC module

1x Servo Motor

1x Stepper module

1x Stepper Motor

 $1x\ 10K\ \Omega$ Pot

5x 10K Ω resistor

5x 1K Ω resistor

 $8x 220 \Omega$ resistor

3x Photo Resistor

1x IC 74HC595N 16-pin DIP

1x IR remote control

1x 6-cell AA Battery pack

1x Acrylic mounting plate

1x Mounting hardware

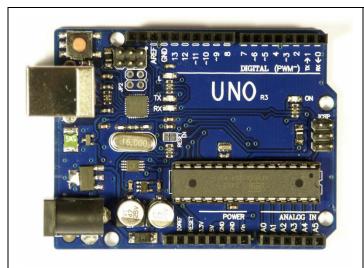
Arduino Uno compatible board

1x USB cable

Dupont connector wires

400,170-pin Breadboard

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.
- Connect the UNO board to one of your computer's USB port.
- 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_dec12a | Arduino 1.0 File Edit Sketch Tools Help Auto Format Archive Sketch sketch_dec12 Fix Encoding & Reload Ctrl+Shift+M Serial Monitor Arduino Uno Serial Port Arduino Duemilanove w/ ATmega328 Arduino Diecimila or Duemilanove w/ ATmega168 Programmer Arduino Nano w/ ATmega328 Burn Bootloade Arduino Nano w/ ATmega168 Arduino Mega 2560 or Mega ADK Arduino Mega (ATmega1280) Arduino Leonardo Arduino Esplora Arduino Micro Arduino Mini w/ ATmega328 Arduino Mini w/ ATmega168 Arduino Ethernet Arduino BT w/ ATmega328 Arduino BT w/ ATmega168 LilvPad Arduino w/ ATmega328 LilyPad Arduino w/ ATmega168 Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328 Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168 Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328 Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168 Arduino NG or older w/ ATmega168 Arduino NG or older w/ ATmega8 Arduino Robot Control Arduino Robot Motor

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:

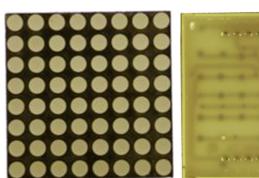
LEDs

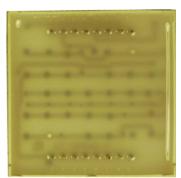


LED - Light Emitting Diodes

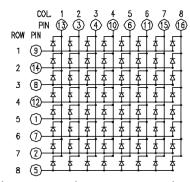
- Connect a current-limiting resistor (220 ohm) between the LED's positive pin and the 5v pin. Connect the LED's negative pin directly to your Arduino output pin. -OR-
- Connect a current-limiting resistor (220 ohm) between the Arduino output pin and the LED's positive pin. Connect the LED's negative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

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



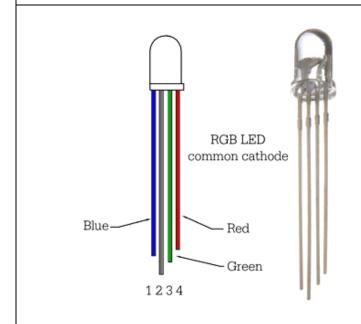


LED - 8x8 Matrix (1588BS or similar)



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

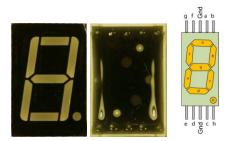


LED - RGB LED

Connect the common pin (2) to your ground, and the Red (4), Green (3), and Blue (1) pins to your Arduino output pins.

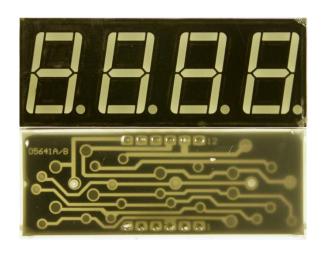
If using PWM (Pulse Width Modulation) capable outputs, you can effectively mix the RGB primary colors to produce thousands or different output colors in the single LED. (See Examples, 01.Basic, **Fade** sketch example in the IDE)

Using the IDE command "digitalWrite(led, HIGH)" and digitalWrite(led, LOW) you can turn the individual color LEDs on and off. (See Examples, 01.Basic, Blink in the IDE) Combine Red+Green = Yellow, Green+Blue=Cyan, Red+Blue=Purple, and Red+Blue+Green=White.

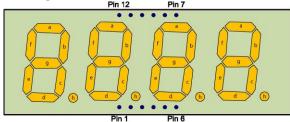


(1-bit) **7-segment LED** (TOS5121AS or similar)

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)



Viewed from top

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



Large button switch - momentary contact, NO

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



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.



5 Volt Relay module

Three input pins: +, -, S ...connect - to ground, connect + to 5V, Connect S to your Arduino "signal" pin to trip the relay.

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

LCD



2x16 LCD Display - 2-line, 16-character LCD display with backlight.

The LiquidCrystal library allows you to control LCD displays that are compatible with the Hitachi HD44780 driver. There are many of them out there and you can usually tell them by the 16-pin interface.

See:

http://www.arduino.cc/en/Reference/LiquidCrystal http://www.arduino.cc/en/Tutorial/LiquidCrystal http://www.arduino.cc/en/Tutorial/LiquidCrystalDisplay

Hitachi HD44780 LCD controller information: https://en.wikipedia.org/wiki/Hitachi HD44780 LCD controller

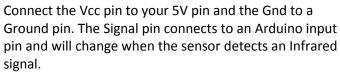
Sensors and modules



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 $^{\sim}$ 1100nm light. Response time is 15us, supply voltage is 3.3-5V; output is analog.

IR Receiver (VS1838B or similar)



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.

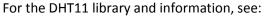


Temperature and Humidity sensor

Three connections:

(-) = ground (-) Note the square solder pad. (center pin) = +5V

(S)= Signal (digital, serial output)



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

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

https://tkkrlab.nl/wiki/Arduino KY-015 Temperature and humidity sensor module





LM35 data sheet: http://www.ece.usu.edu/ece store/spec/Im35dt-3p.pdf

LM35 Temperature Sensor

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



Play a tone or melody using the passive buzzer:

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

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

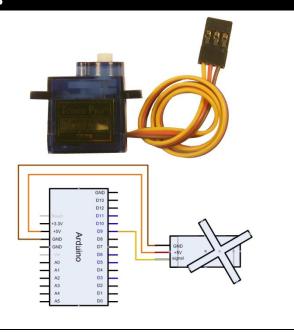
Passive & Active buzzers

Use as a speaker, buzzer or other audible indicator.

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

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.

Motors



Servo motor

Note that Servo motor color schemes may vary:

Brown / Black = ground

Orange / Red = +5V

Yellow / White = signal (use digital PWM connection.)

Servo examples are included with the IDE, see:

http://arduino.cc/en/Tutorial/Knob http://arduino.cc/en/Tutorial/Sweep



Stepper motor and controller

Stepper IC = ULN2003AN (or similar)

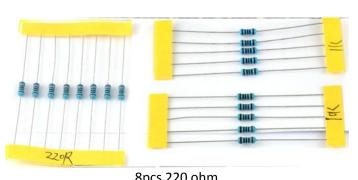
The stepper motor included in the kit connects to the controller through a white connector socket. Four inputs connect to your Arduino IN1, IN2, IN3, and IN4. Power for the motor and controller is provided through the ground (-) and 5-to-12V (+) pins.

Use an external power supply for the motor to avoid damaging the Arduino. Connect the ground of your external supply to the ground of the Arduino and the signal IN# pins to digital outputs.

For sketch examples, see:

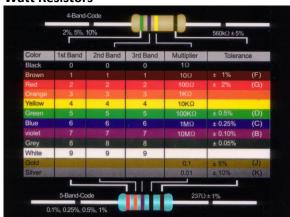
http://arduino.cc/en/Tutorial/MotorKnob

Resistors



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

1/4 Watt Resistors



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

10K Potentiometer

Resistance between outer pins is 10K ohms. Resistance between one outer pin and the center (wiper) pin is 0-10K ohms based on position.

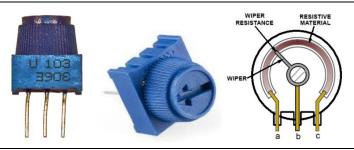
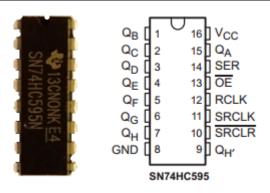


Photo Resistor

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



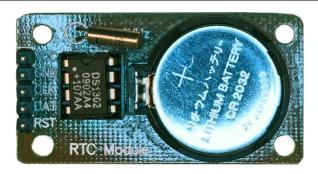
other



Tri-state 8-bit shift register IC SN74HC595N (or similar) The datasheet refers to the 74HC595 as an "8-bit serialin, 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



Real Time Clock Module

The DS1302 trickle-charge timekeeping chip contains a real-time clock/calendar and 31 bytes of static RAM. It communicates with a microprocessor via a simple serial interface. The real-time clock/calendar provides seconds, minutes, hours, day, date, month, and year information.

For library and code example, see:

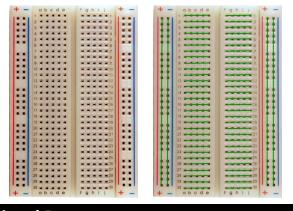
http://playground.arduino.cc/Main/DS1302RTC



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



400-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

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

Earthshine Electronics Beginners Guide to Arduino:

https://docs.google.com/file/d/0Bw ruMOtRDDgNXI3OTFGZXhIZ2c/edit?usp=sharing

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