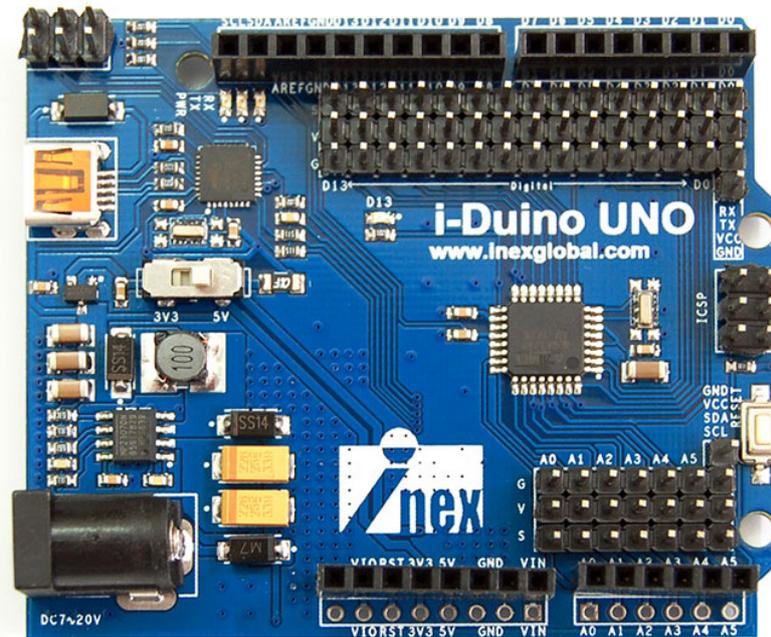


i-Duino UNO



Overview

The i-Duino UNO is a microcontroller board based on the Arduino UNO. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

Basic features

- New external IO pin map
- Efficient DC-DC Power supply with wide range input.
- 3.3V/5V Operating Voltage selection
- All pins out for Sensor and Servo
- UART/IIC interface breakout

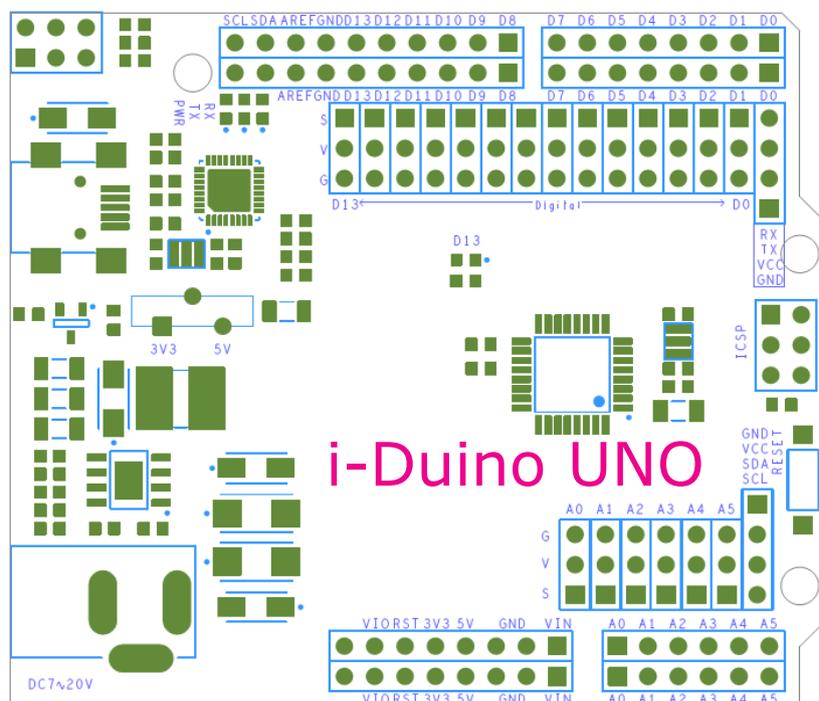
Specifications

Microprocessor	ATmega328,ATmega16U2
PCB size	68.58mm X 58.42mm X 1.6mm
Indicators	Power,TX,RX,D13
Power supply(recommended)	7-23V DC
Power supply(limits)	23 VDC (max)
Communication Protocol	UART,SPI,IIC
Clock Speed	16MHz

Electrical Characteristics

Specification	Min	Type	Max	Unit
Input voltage	7	-	23	VDC
Operating Voltage	-	3.3/5	-	VDC
DC Current per I/O Pin	-	40	-	mA

Hardware



Digital/Servo Interface: D0~D13 pin

Sensor Interface: A0 A1 A2 A3 A4 A5 pin

Pin Map

Index	Name of Arduino	Alternate Function	Pin of Atmega328
1	D0	UART DIN	PD0
2	D1	UART DOUT	PD1
3	D2	External Interrupt 0	PD2
4	D3	External Interrupt 1/PWM	PD3
5	D4	-	PD4
6	D5	PWM	PD5
7	D6	PWM	PD6
8	D7	-	PD7
9	D8	-	PB0
10	D9	PWM	PB1
11	D10	SPI_NSS/PWM	PB2
12	D11	SPI_MOSI/PWM	PB3
13	D12	SPI_MISO	PB4
14	D13	SPI_SCK	PB5
15	A0		PC0
16	A1		PC1
17	A2		PC2
18	A3		PC3
19	A4	IIC_SDA	PC4
20	A5	IIC_SCL	PC5

Power

The i-Duino UNO can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 7 to 23 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable.

Power supply select switch:

On i-Duino UNO board, there is a supply power select switch for selecting the 3.3/5V power supply. Much compatible shields can be plug in i-Duino UNO board.

The power pins are as follows:

- **VIN.** The input voltage to the i-Duino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 500 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 3.3/5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.

- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the `analogWrite()` function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the SPI library.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 3.3/5 volts, though is it possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The i-Duino UNO has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.

The ATmega328 provides UART CMOS/TTL (3.3/5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Iteaduno board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins.

The ATmega16U2 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Programming

The i-Duino UNO can be programmed with the Arduino software (download). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega16U2 on the i-Duino UNO comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user-contributed tutorial for more information.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Iteaduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega16U2 is connected to the reset line of the ATmega328 via a 100nF capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is

programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from VCC to the reset line; see this forum thread for details.

USB Over-current Protection

The i-Duino UNO has a RTC that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

i-Duino UNO Pinout Diagram

