

Arduino UNO R3

Visión de conjunto

Arduino UNO es una placa electrónica basada en el ATmega328 (ficha técnica). Cuenta con 14 pines de entradas/salidas digitales (de las cuales 6 se puede utilizar como salidas PWM), 6 entradas analógicas, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, una cabecera ICSP, y un botón de reset. Contiene todo lo necesario para utilizar el microcontrolador, basta con conectarlo a un ordenador con un cable USB, a un adaptador de CA a CC o a una batería para empezar.

El UNO se diferencia de todas las placas anteriores en que no utiliza el chip controlador FTDI USB-to-serial. En su lugar, ofrece el Atmega16U2 programado como un convertidor de USB a serie.

La Revisión 3 de la placa tiene las siguientes características nuevas:

- Añadidos pines SDA y SCL colocados cerca del pin AREF y otros dos nuevos colocados cerca del pin RESET, el primero es IOREF que permite a los escudos adaptarse a la tensión de la placa. En el futuro, los escudos serán compatibles tanto con la placa que utilizan el AVR, que funcionan con 5V y con el Arduino Due que trabaja a 3.3V. El segundo es un pin no conectado, que está reservado para usos futuros.
- Circuito RESET más fuerte.
- Un Atmega 16U2 reemplazar el 8U2 de las versiones anteriores.

El UNO es el último de una serie de placas Arduino USB y el modelo de referencia para la plataforma Arduino.

Resumen

Microcontrolador	ATmega328
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Pines Digitales de I/O	14 (de los cuales 6 proporcionan salida PWM)
Pines de entrada analógica	6
Corriente DC por Pin I/O	40 mA
Corriente DC por Pin 3.3V	50 mA
Memoria Flash	32 KB (ATmega328) de los cuales 0,5 KB utilizado por gestor de arranque
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad del reloj	16 MHz

Esquema y Diseño de Referencia

Archivos de EAGLE: [arduino-uno-Rev3-reference-design.zip](#)

Esquema:[arduino-uno-Rev3-schematic.pdf](#)

Nota: El diseño de referencia de Arduino puede utilizar un Atmega8, 168, o 328, los modelos actuales utilizan un ATmega328, pero un Atmega8 se muestra en el esquema de referencia. La configuración de pines es idéntica en todos los tres procesadores.

Alimentación

El Arduino UNO puede ser alimentado a través de la conexión USB o con una fuente de alimentación externa. El origen de la alimentación se selecciona automáticamente.

Las fuentes de alimentación externas (no-USB) pueden ser tanto un transformador o una batería. El transformador se puede conectar usando un conector macho de 2.1mm con centro positivo en el conector hembra de la placa. Los cables de la batería puede conectarse a los pines Gnd y Vin en los conectores de alimentación.

La placa puede operar con un suministro externo de 6 a 20 voltios. Sin embargo, si se proporcionan menos de 7V, el pin de 5V puede proporcionar menos de cinco voltios y la placa puede ser inestable. Si se utilizan más de 12V, el regulador de voltaje se puede sobrecalentar y dañar la placa. El rango recomendado es de 7 a 12 voltios.

Los pines de alimentación son como sigue:

- **VIN.** La tensión de entrada a la placa Arduino cuando se utiliza una fuente de alimentación externa (en lugar de 5 voltios de la conexión USB o de otra fuente de alimentación regulada). Se puede suministrar tensión a través de este pin, o, si se está alimentado a través de la clavija de alimentación, acceder a ella a través de este pin.
- **5V.** Este pin saca 5V regulados por el regulador de la placa. La fuente de alimentación regulada utilizada para alimentar el microcontrolador y los otros componentes en la placa. Esta puede provenir de la clavija de alimentación, suministrada a través de USB o de otra fuente de 5V regulada o de VIN a través del regulador integrado en la placa. Suministrar tensión por medio de las conexiones de 5V o de 3.3V se salta el regulador y puede dañar tu placa. No se recomienda.
- **3V3.** Una tensión de alimentación 3,3 voltios generados por el regulador incorporado. El consumo de corriente máxima es de 50 mA.
- **GND.** Pines de toma de tierra.
- **IOREF.** Este pin en la placa Arduino proporciona la referencia de tensión con la que opera el microcontrolador. Un escudo configurado correctamente puede leer el voltaje del pin IOREF y seleccionar la fuente de alimentación adecuada o habilitar a adaptadores de nivel de tensión en las salidas para trabajar con 5V o 3.3V.

Memoria

El ATmega328 tiene 32 KB (con 0,5 KB utilizado para el gestor de arranque). También tiene 2 KB de SRAM y 1 KB de memoria EEPROM (que puede ser leída y escrita por la librería EEPROM).

Entradas y salidas

Cada uno de los 14 pins digitales en la UNO se puede usar como entrada o salida, utilizando las funciones `pinMode()`, `digitalWrite()` y `digitalRead()`. Funcionan a 5 voltios. Cada pin puede proporcionar o recibir un máximo de 40 mA y tiene una resistencia interna de pull-up (desconectada por defecto) de 20-50 kOhmios. Además, algunos pines tienen funciones especializadas:

- **Serie: 0 (RX) y 1 (TX).** Se utiliza para recibir (RX) y transmitir (TX) datos serie TTL. Estos pines están conectados a los pines correspondientes del chip ATmega16U2 USB-to-Serial TTL.
- **Interrupciones Externas: 2 y 3.** Estos pines pueden ser configurados para activar una interrupción en un valor bajo, un flanco ascendente o descendente, o un cambio en el valor. Ver la función `attachInterrupt()` para más detalles.
- **PWM: 3, 5, 6, 9, 10, y 11.** Proporcionan 8-bit de salida PWM con la función `analogWrite()`.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** Estos pines proporcionan comunicación SPI mediante la librería SPI.
- **LED: 13** Hay un LED integrado en la placa conectado al pin digital 13. Cuando el pin esta a valor alto, el LED está encendido, cuando el pin está bajo, está apagado.

El UNO tiene 6 entradas analógicas, etiquetadas de A0 a A5, cada uno de los cuales proporcionan 10 bits de resolución (es decir, 1024 valores diferentes). Por defecto se mide de tierra a 5 voltios, aunque es posible cambiar el extremo superior del rango usando el pin AREF y la función `analogReference()`. Además, algunos pines tienen funciones especializadas:

- **TWI: A4 o pin SDA y A5 o pin SCL.** Soporte para comunicación TWI usando la librería Wire.

Hay otro par de pines en la placa:

- **AREF.** Voltaje de referencia para las entradas analógicas. Se utiliza con `analogReference()`.
- **Reset.** Poner esta línea BAJA para reiniciar el microcontrolador. Típicamente usado para añadir un botón de reset a los shields que no dejan acceso a este botón en la placa. Vea también la asignación entre pines de Arduino y ATmega328. La asignación para el Atmega8, 168, y 328 es idéntica.

Comunicación

El Arduino UNO facilita en varios aspectos la comunicación con el ordenador, otro Arduino u otros microcontroladores. El ATmega328 ofrece comunicación serie con la UART TTL (5V) que está disponible en los pines digitales 0 (RX) y 1 (TX). Un ATmega16U2 en la placa envía esta comunicación serie a través de USB y aparece como un puerto COM virtual para el software en el ordenador. El firmware del '16U2 utiliza los controladores estándar USB COM, y no es necesario ningún controlador externo. Sin embargo, en Windows se requiere un archivo .inf. El software de Arduino incluye un monitor de puerto serie que permite

enviar y recibir información textual de la placa Arduino. Los LEDs RX y TX de la placa parpadean cuando los datos se transmiten a través del chip USB a serie y la conexión USB al ordenador (pero no para la comunicación en serie en los pines 0 y 1).

La librería SoftwareSerial permite la comunicación serie en cualquiera de los pines digitales de la UNO.

El ATmega328 también soporta la comunicación I2C(TWI) y SPI. El software de Arduino incluye una librería Wire para simplificar el uso del bus I2C, vea la documentación para más detalles. Para la comunicación SPI, utilice la librería de SPI.

Programación

Arduino UNO se puede programar con el software Arduino (descarga). Seleccione "Arduino UNO de los menú**Herramientas> Tarjeta** (de acuerdo con el microcontrolador en la placa). Para obtener más información, consulte la referencia y tutoriales.

El ATmega328 en la Arduino UNO viene precargado con un gestor de arranque que le permite cargar nuevo código a la misma sin el uso de un programador de hardware externo. Se comunica utilizando el protocolo STK500 original, (archivos de cabecera C).

También puede pasar por alto el gestor de arranque y programar el microcontrolador a través del conector ICSP (In-Circuit Serial Programming), ver las instrucciones para más detalles.

El código fuente del firmware ATmega16U2 disponible. El ATmega16U2 se carga con un cargador de arranque DFU, que puede ser activado En el 16U2 hay una resistencia a la línea HWB y poniéndola a tierra hace más fácil de poner en modo DFU.

Se puede utilizar el software FLIP de Atmel (Windows) o el programador DFU (Mac OS X y Linux) para cargar un nuevo firmware. O bien, puede utilizar el conector ISP con un programador externo (sobrescribir el gestor de arranque DFU). Ver este tutorial aportado por el usuario para obtener más información.

Reinicio Automático por Software

En vez de necesitar reiniciar presionando físicamente el botón de reset antes de cargar, el Arduino UNO está diseñado de manera que permite que se restablezca por software desde el ordenador al que está conectado. Una de las líneas de control de flujo por hardware (DTR) del ATmega16U2 está conectada a la línea de reinicio del ATmega328 mediante un condensador de 100 nanofaradios. Cuando esta línea se pone a nivel bajo (0V), la línea de reinicio baja el tiempo suficiente para reiniciar el chip. El software de Arduino utiliza esta característica para permitir cargar el código con sólo pulsar el botón de subida en el entorno Arduino. Esto significa que el gestor de arranque puede tener un tiempo de espera más corto, ya que el bajar DTR puede estar bien coordinado con el inicio de la carga.

Esta configuración tiene otras implicaciones. Cuando el UNO se conecta a un ordenador con Mac OS X o Linux, se reinicia cada vez que se realiza una conexión a la misma desde el software (a través de USB). El medio segundo aproximadamente posterior, el gestor de

arranque se está ejecutando en el ONU. A pesar de estar programado para ignorar datos mal formateados (es decir, nada, además de una carga de nuevo código), se interceptan los primeros bytes de datos enviados a la placa justo después de abrir una conexión. Si un sketch ejecutándose en la placa recibe algún tipo de configuración inicial o otro tipo de información al inicio del programa, asegúrate de que el software con el que se comunica espera un segundo después de abrir la conexión y antes de enviar los datos.

El UNO contiene una pista que se puede cortar para desactivar el auto-reset. Las terminaciones a ambos lados de la pista se pueden soldar de nuevo para volver a habilitarlo. Están etiquetadas con "RESET-EN". También se puede desactivar el reinicio automático mediante la conexión de una resistencia de 110 ohm de 5V a la línea de reset, ver este hilo del foro para más detalles.

Protección contra sobretensiones en USB

Arduino UNO tiene un multifusible reinicializable que protege los puertos USB de tu ordenador de cortocircuitos y sobrecorrientes. Aunque la mayoría de los ordenadores ofrecen su protección interna, el fusible proporciona una capa adicional de protección. Si más de 500 mA son detectados en el puerto USB, el fusible automáticamente corta la conexión hasta que el cortocircuito o la sobrecarga se desaparece.

Características físicas

La longitud máxima y la anchura de la placa UNO son de 2,7 y 2,1 pulgadas respectivamente, con el conector USB y el conector eléctrico que se extienden más allá de la dimensión anterior. Cuatro agujeros para fijación con tornillos permiten colocar la placa en superficies y cajas. Ten en cuenta que la distancia entre los pines digitales 7 y 8 es de 160 milésimas de pulgada (0,16"), no un múltiplo de la separación de 100 milésimas de pulgada de los otros pines.