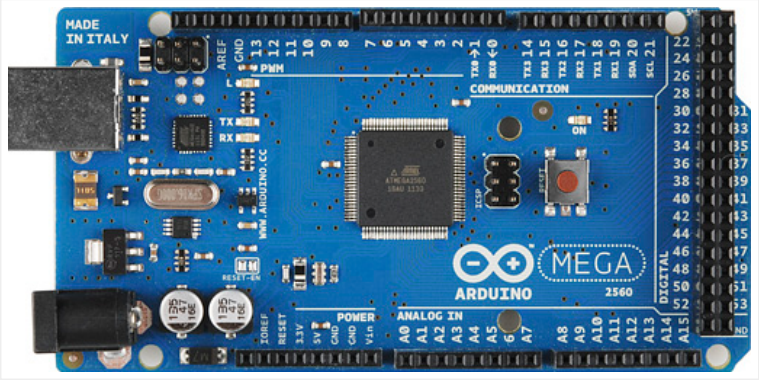
-

Arduino y su documentación en español

Arduino Mega 2560



Descripción general

La Mega 2560 es una placa electronica basada en el Atmega2560. Cuenta con 54 pines digitales de entrada / salida (de los cuales 15 se pueden utilizar como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serie de hardware), un oscilador de 16MHz, una conexión USB, un conector de alimentación, un conector ICSP, y un botón de reset. Contiene todo lo necesario para apoyar el microcontrolador; basta con conectarlo a un ordenador con un cable USB o a la corriente con un adaptador de CA a CC o una batería para empezar. La placa Mega 2560 es compatible con la mayoría de los shield para el Uno y las placas anteriores Duemilanove o Diecimila.

El Mega 2560 es una actualización de la Arduino Mega, al que sustituye. Usted puede encontrar aquí informaciones de garantía de su placa.

Especificaciones técnicas

Microcontrolador	ATmega2560
Tensión de trabajo	5V
Tensión de entrada (recomendada)	7-12V
Tensión de entrada (límite)	6-20V
Pines Digitales I/O	54 (de los cuales 15 proporcionan salida PWM)
Pines de entradas Analógicas	16
DC Corriente por Pin I/O	20 mA
DC Corriente por Pin 3.3V	50 mA
Memoria Flash	256 KB de los cuales 8 KB se usan por el bootloader
SRAM	8 KB
EEPROM	4 KB
Velocidad del reloj	16 MHz
Largo	101.52 mm
Anchu	53.3 mm
Peso	37 g

Programación

La placa Mega 2560 se puede programar con el software de Arduino (IDE). Para más detalles, véase la referencia y tutoriales.
Las Atmega2560 y Mega 2560 vienen preprogramadas con un cargador de arranque (bootloader) que le permite cargar nuevo código en ella sin el uso de un programador de hardware externo. Se comunica utilizando el protocolo original STK500 (referencia, archivos de cabecera C).

También puede pasar por alto el gestor de arranque y programar el microcontrolador a través del conector ICSP (programación seriel en circuito) utilizando Arduino ISP o similar; ver estas instrucciones para más detalles.

En las placas ATmega16U2 (o 8U2 Rev1 y Rev2) el código fuente del firmware está disponible en el repositorio Arduino. El ATmega 16U2 / 8U2 se carga con un cargador de arranque DFU, que puede ser activado por:

En las placas Rev1: el puente de soldadura en la parte posterior de la placa (cerca del mapa de Italia) y luego reiniciar el 8U2.
En las placas de Rev2 o posteriores: existe una resistencia que pone la línea HWB 8U2 / 16U2 a tierra, por lo que es más fácil poner en modo DFU. A continuación, puede utilizar el software FLIP de Atmel (Windows) o el programador DFU (Mac OS X y Linux) para cargar un nuevo firmware. O 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.

Advertencia

El 2560 mega tiene un polyfusible reajustable que protege a los puertos USB de su ordenador desde cortocircuitos y sobrecorriente. Aunque la mayoría de los ordenadores establecen su propia protección interna, el fusible proporciona una capa adicional de protección. Si circulan más de 500 mA por el puerto USB, el fusible interrumpirá automáticamente la conexión hasta que se repara el cortocircuito o se elimina la sobrecarga.

Alimentación eléctrica

El Mega 2560 puede ser alimentado a través de la conexión USB o con una fuente de alimentación externa. La fuente de alimentación se selecciona automáticamente.
La alimentación externa (no USB) puede venir de un adaptador de CA a CC o de una batería. El adaptador se puede conectar al enchufe de 2,1 mm de centro-positivo en la clavija de alimentación de la placa. Los cables desde una batería pueden ser insertados en GND y en el pin Vin del conector de alimentación.
La tarjeta puede funcionar con un suministro externo de 6 a 20 voltios. Si se alimenta con menos de 7 V, sin embargo, el pin de 5V puede suministrar menos de cinco voltios y la placa se puede volver inestable. Si se utiliza más de 12 V, 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 los siguientes:

- Vin. La tensión de entrada a la placa cuando se utiliza una fuente de alimentación externa (en contraposición a 5 voltios de la conexión USB o de otra fuente de alimentación regulada). Se puede suministrar tensión a través de este pin, o, si el suministro de tensión es a través de la toma de alimentación, acceder a él a través de este pin.
- 5V. Este pin es una salida de 5V regulada del regulador de la placa. La placa puede ser alimentada vía [illegible] alimentación de CC (7 - 12 V), por el conector USB (5 V), o por el pin VIN de la placa(7-12V). El suministro de tensión a través de los pines de 5 V o 3,3 V no pasa por el regulador, y puede dañar la placa. No es aconsejable.
- 3V3. Un suministro de 3,3 voltios generado por el regulador de la placa. El consumo de corriente máximo es de 50 mA.
  GND. Ips pines de tierra.
- IOREF. Este pin en la placa proporciona la referencia de tensión con la que opera el microcontrolador. Un escudo bien configurado puede leer la tensión del pin IOREF y seleccionar la fuente de alimentación adecuada o habilitar traductores de tensión en las salidas para trabajar con el 5 V o 3.3 V.

Memoria

Ver el mapeo entre los pines de Arduino y los puertos Atmega2560:

Cada uno de los 54 pines digitales de la Mega se puede utilizar como una entrada o como una salida, utilizando las funciones pinMode(), digitalWrite() y digitalRead(). Operan a 5 voltios. Cada pin puede proporcionar o recibir 20 mA como condición de funcionamiento recomendada y tiene una resistencia de pull-up (desconectada por defecto) de 20-50 k ohmios. Un máximo de 40 mA es el valor que no debe superarse para evitar daños permanentes en el microcontrolador.

Además, algunos pines tienen funciones especializadas:

- **Serie**: 0 (RX) y 1 (TX); Serie 1: 19 (RX) y 18 (TX); Serie 2: 17 (RX) y 16 (TX); Serie 3: 15 (RX) y 14 (TX). Se utiliza para recibir (RX) y transmitir datos serie (TX) TTL. Los pines 0 y 1 también están conectados a los pines correspondientes del chip serie ATmega16U2 USB-a-TTL.

- **Interrupciones externas**: 2 (interrupción 0), 3 (interrupción 1), 18 (interrupción 5), 19 (interrupción 4), 20 (interrupción 3), y 21 (interrupción 2). Estos pines pueden configurarse para activar una interrupción en un nivel bajo, un flanco ascendente o descendente, o un cambio en el nivel. Véase la función attachInterrupt() para más detalles.

- **PWM**: 2 a 13 y 44 a 46. proporcionan una salida PWM de 8 bits con la función analogWrite().

- **SPI**: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). Estos pines soportan la comunicación SPI utilizando la biblioteca SPI. Los pines SPI también se repiten en el conector ICSP, que es físicamente compatible con el Arduino / Genuino Uno y las antiguas placas Duemilanove y Diecimila Arduino.

- **LED**: 13. Hay un LED incorporado conectado al pin digital 13. Cuando el pin está a nivel HIGH, el LED está encendido, cuando el pin está a nivel LOW, está apagado.

- **TWI**: 20 (SDA) y 21 (SCL). TWI soporte de comunicación utilizando la biblioteca Wire. Tenga en cuenta que estos pines no están en la misma ubicación que los pines TWI de las antiguas placas Duemilanove o Diecimila Arduino.
  Ver también el mapeado de los pines Arduino Mega 2560.

El Mega 2560 tiene 16 entradas analógicas, cada una de las cuales proporcionan 10 bits de resolución (es decir, 1024 valores diferentes). Por defecto se miden de masa a 5 voltios, aunque es posible cambiar el extremo superior de su rango usando la función analogReference () y el pin AREF.
Hay un par de pines en la placa:

- **AREF**. Tensión de referencia para las entradas analógicas. Se utiliza con analogReference().

- **Reset**. Llevar esta línea a nivel LOW para reiniciar el microcontrolador. Normalmente se utiliza para añadir un botón de reinicio para escudos que bloquean la la placa.

Comunicación

La placa Mega 2560 tiene una serie de facilidades para la comunicación con un ordenador, otra placa, u otros microcontroladores. El Atmega2560 ofrece cuatro UART hardware para TTL (5 V) para la comunicación serie. Una ATmega16U2 (ATmega 8U2 revisión 1 y 2) tiene canales que uno de ellos a través de USB y proporciona un puerto com virtual para el software en el equipo (en las máquinas Windows necesitará un archivo .inf, pero las máquinas OSX y Linux reconocen la placa como un puerto COM automáticamente. El software de Arduino (IDE) incluye un monitor serie que permite que los datos de texto simples puedan ser enviados hacia y desde la placa. Los LEDs RX y TX de la placa parpadean cuando se están transmitiendo datos a través de la ATmega8U2 / ATmega16U2 chip y conexión USB al ordenador (pero no para la comunicación serie en los pines 0 y 1).

Una biblioteca SoftwareSerial permite la comunicación serie en cualquiera de los pines digitales del Mega 2560.

El Mega 2560 también es compatible con la comunicación TWI y SPI. El software de Arduino (IDE) incluye una biblioteca Wire para simplificar el uso del bus TWI; Ver la documentación para más detalles. Para la comunicación SPI, utilice la biblioteca SPI.

Características físicas y compatibilidad con Escudos

analógicas de 0 a 5, el conector de alimentación, y conector de ICSP están todos en ubicaciones equivalentes. Por otra parte, la UART principal (puerto serie) se encuentra en los mismos pines (0 y 1), al igual que las interrupciones externas 0 y 1 (pines 2 y 3, respectivamente). SPI está disponible a través del conector ICSP en ambas placas Mega 2560 y Duemilanove / Diecimila. Tenga en cuenta que I2C no se encuentra en los mismos pines en la placa Mega 2560 (20 y 21) como en las placas Duemilanove / Diecimila (entradas analógicas 4 y 5).

Reset (Software) automático

En lugar de necesitar una activación del pulsador de reset antes de un proceso de carga, la Mega 2560 está diseñada de manera que permite que pueda ser reseteada por el software que se ejecuta en un ordenador conectado. Una de las líneas de control de flujo por hardware (DTR) de la ATmega8U2 está conectada a la línea de reset de la Atmega2560 a través de un condensador de 100 nanofaradios. Cuando esta línea se afirma (pasa a nivel LOW), la línea de reset pasa a nivel HIGH el tiempo suficiente para restablecer el chip. El software de Arduino (IDE) utiliza esta capacidad que le permite subir 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 descenso de DTR puede ser bien coordinado con el inicio de la carga del programa.
Esta configuración tiene otras implicaciones. Cuando la placa Mega 2560 está conectada ya sea a un ordenador con Mac OS X o Linux, se resetea cada vez que se realiza una conexión a la misma desde el software (a través de USB). Durante el siguiente medio segundo o así, el gestor de arranque se ejecuta en los Atmega2560. Mientras que está programado para ignorar los datos con formato incorrecto (por ejemplo, cualquier cosa además de un proceso de carga del nuevo código), esta interceptará los primeros bytes de datos enviados a la placa después de abrir una conexión. Si un programa en ejecución en el placa recibe la configuración de una sola vez u otro tipo de datos cuando se inicia por primera vez, asegúrese de que el software con el que se comunica espera un segundo después de abrir la conexión y antes de enviar estos datos.
La placa Mega 2560 contiene una pista del circuito impreso que se puede cortar para desactivar el reinicio automático. Los puntos de soldadura a ambos lados de la pista se pueden soldar juntos para volver a habilitarla. Este está etiquetado como "RESET-EN". También se puede desactivar el reinicio automático mediante la conexión de una resistencia de 110 ohmios de 5 V a la línea de reset; ver este este hilo del foro para más detalles.

Revisiones

El Mega 2560 no utiliza el chip controlador FTDI USB-a-serie utilizados en los diseños anteriores. En lugar de ello, se cuenta con el ATmega16U2 (placas Arduino ATmega8U2 revisión 1 y 2) programado como un convertidor de USB a serie.
La revisión 2 de la placa Mega 2560 tiene una resistencia que lleva la línea 8U2 HWB a tierra, por lo que es más fácil de poner en modo DFU.
La revisión 3 de la placa Arduino y la actual Genuino Mega 2560 tienen las siguientes características mejoradas:

- Pines1.0 : Pines SDA y SCL - cerca del pin AREF - y otros dos nuevos pines colocados cerca del pin de RESET,el IOREF que permite a los escudos adaptarse a la tensión suministrada desde la placa. En el futuro, los escudos serán compatibles tanto con la placa que utilice el AVR, que operan con 5 V como con la placa que utiliza ATSAM3X8E, que opera con 3.3 V. El segundo es un pin no está conectado, que está reservado para usos futuros.
- Circuito de RESET más vigoroso.
- ATmega 16U2 sustituye al 8U2.

No hay comentarios:

Publicar un comentario

Comentar como: Javier Martín A ▾ Cerrar sesión

Publicar Vista previa Avisarme

Página principal

Suscribirse a: Entradas (Atom)

- Arduino Due
- Uso de la SPI extendida con la Due
- Biblioteca Wire
- Keyboard Controller
- Mouse Controller
- getOemKey()
- getKey()
- keyReleased()
- keyPressed()
- KeyboardController
- getButton()
- getYChange()
- mouseReleased ()
- getXChange()
- mousePressed ()
- MouseMoved()
- mouseDragged()
- MouseControler
- Task()
- Instalación de bibliotecas adicionales
- Bibliotecas
- usbhost
- getModifiers()
- Audio.begin()
- Biblioteca de Audio
- Audio.prepare()
- Audio.write()
- Sencillo reproductor de audio
- Stream
- Biblioteca EEPROM
- EEPROM Update
- EEPROM Iteración
- EEPROM Put
- EEPROM Get
- EEPROM CRC
- EEPROM Escritura
- EEPROM Borrado
- EEPROM Lectura
- Clase GPRS
- Clase GSM_SMS
- Clase GSMVoiceCall
- Clase GSM
- Biblioteca GSM
- Clase GSMClient
- Clase GSMServer
- Clase GMSModem
- Clase GSMScanner
- Clase GSMPIN
- Clase GSMBand
- GSM Web Client
- GMS Web Server
- Hacer llamada de voz
- Enviar un SMS
- Recibir una llamada de voz
- Recibir un SMS
- Gestion de banda
- Scan de redes GSM
- Gestión del PIN GSM
- Test GPRS en GSM
- Test Modem GSM
- Servidor Web GSM
- Biblioteca Lyquid Crystal
- Biblioteca SD

- Archivos SD
- Lista de archivos SD
- Lectura/Escritura en tarjeta SD
- Control de posición Servo
- Biblioteca Servo
- Barrido de Servo
- Funciones SPI
- Ejemplo Software Serial
- Biblioteca Software Serial
- Biblioteca SPI
- Dos puertos de recepción
- Motores Paso a Paso
- Biblioteca Stepper
- Control motor Paso a Paso con un potenciómetro
- Motor Paso a Paso una revolución
- Motor Paso a Paso Un paso cada vez
- Motor Paso a Paso con potenciómetro
- Arduino Wifi 101
- Funciones Wifi - Clase IPAddress
- Funciones Wifi - Clase Wifi
- Comienzo con Arduino Wifi
- Funciones Wifi - Clase Client
- Funciones Wifi - Clase Server
- Funciones Wifi - Clase UDP
- Wifi conexión WAP cifrada
- Wifi conexión WEP cifrada
- Wifi Conexión no cifrada
- Wifi Servidor de chat sencillo
- Wifi Escaneo de redes
- Wifi Repetidas llamadas
- Wifi Cliente web
- Wifi Servidor Web
- Consulta a un servidor
- UDP Envio/Recepción de cadenas
- Bridge - Clase Server
- Bridge - Clase Client
- Bridge - HttpClient
- Bridge - Clase Mailbox
- Bridge - Clase FileIO
- Bridge - Clase Console
- Bridge - Clase Process
- Bridge - Clase Bridge
- Biblioteca Bridge
- Bridge - Pixel en la Consola
- Bridge - Tabla ASCII a Consola
- Bridge - Ejemplo Bridge
- Bridge - Datalogger
- Bridge - Lectura de Console

Archivo del blog

▼ 2016 (1)

▼ marzo (1)

¿QUE ES ARDUINO? MODELOS DE PLACAS ARDUINO LENGU...