

Arduino

Joman Solorzano C.I 22.515.311

Jonathan Jaramillo C.I 20.955.384

Diciembre 2023

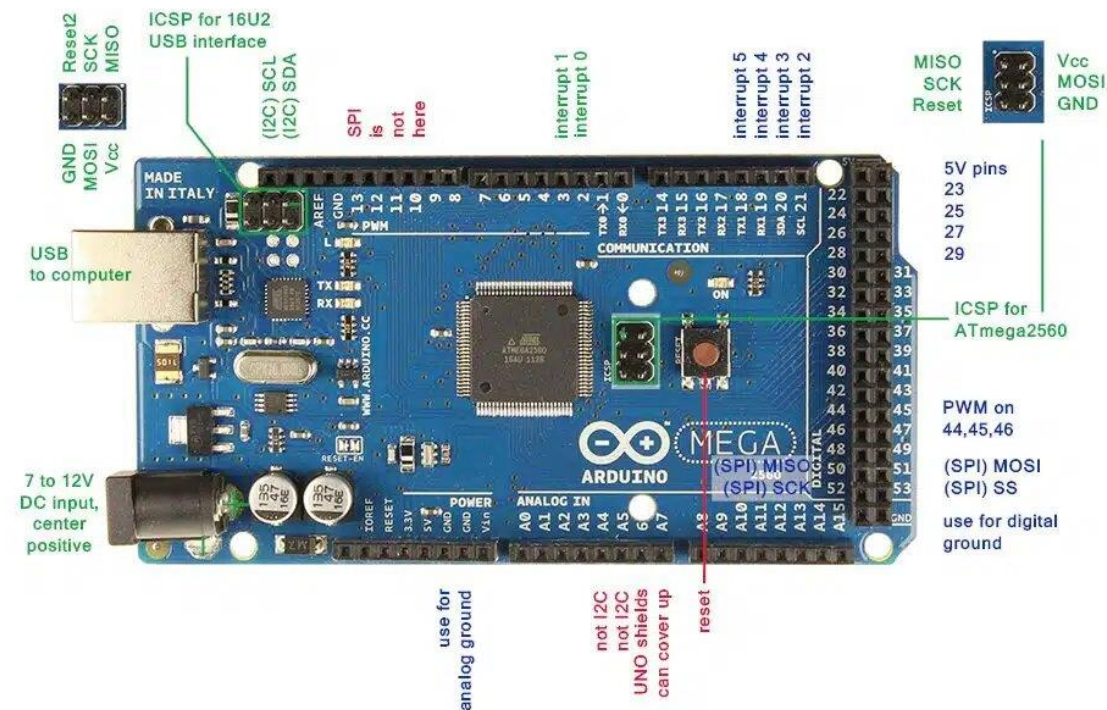
Arduino es una empresa, proyecto y comunidad de usuarios de hardware y software de código abierto que diseña y fabrica microcontroladores de placa única y kits de microcontroladores para construir dispositivos digitales. Sus productos de hardware tienen una licencia CC BY-SA, mientras que el software tiene una licencia GNU Lesser General Public License (LGPL) o GNU General Public License (GPL) lo que permite la fabricación de placas Arduino y la distribución de software por parte de cualquier persona.

Los diseños de placas Arduino utilizan una variedad de microprocesadores y controladores. Las placas están equipadas con conjuntos de pines de entrada/salida (E/S) digitales y analógicas que pueden conectarse a varias placas de expansión o placas de pruebas (para creación de prototipos) y otros circuitos. Las placas cuentan con interfaces de comunicaciones de serie incluido tales como Universal Serial Bus (USB) en algunos modelos, que también se utilizan para cargar programas. Los microcontroladores se pueden programar utilizando los lenguajes de programación C y C++ (Embedded C), utilizando una API estándar que también se conoce como lenguaje de programación Arduino. Además de utilizar cadenas de herramientas de compilación tradicionales, el proyecto Arduino proporciona un entorno de desarrollo integrado (IDE) y una herramienta de línea de comandos desarrollada en Go.

El proyecto Arduino comenzó en 2005 como una herramienta para estudiantes del Interaction Design Institute Ivrea en Italia, con el objetivo de proporcionar una forma sencilla y de bajo costo para que principiantes y profesionales crearan dispositivos que interactuaran con su entorno utilizando sensores. Ejemplos comunes de dispositivos de este tipo destinados a aficionados principiantes incluyen robots simples, termostatos y detectores de movimiento.

Puertos

Para este ejemplo usamos el modelo Arduino Mega 2560



30	RESET	ATMEGA2560-16AU	71
33	XTAL2		PA7 (AD7)
34	XTAL1		PA6 (AD6)
98	AREF		PA5 (AD5)
00	A/VCC		PA4 (AD4)
99	AGND		PA3 (AD3)
10	VCC	(OC0A/OC1C/PCINT7)	PA2 (AD2)
31	VCC	(OC1B/PCINT6)	PA1 (AD1)
61	VCC	(OC1A/PCINT5)	PA0 (AD0)
80	VCC	(OC2A/PCINT4)	
11	VCC	(MISO/PCINT3)	
32	GND	(MOSI/PCINT2)	PB7
33	GND	(SS/PCINT1)	PB6 (OC1B/PCINT6)
62	GND		PB5 (OC1A/PCINT5)
81	GND		PB4 (OC2A/PCINT4)
42	PL7		PB3 (MISO/PCINT3)
41	PL6		PB2 (MOSI/PCINT2)
40	PL5	(OC5C)	PB1 (SS/PCINT0)
39	PL4	(OC5B)	PB0
38	PL3	(OC5A)	(A15) PC7
37	PL2	(T5)	(A14) PC6
36	PL1	(ICP5)	(A13) PC5
35	PL0	(ICP4)	(A12) PC4
82	PK7	(ADC15/PCINT23)	(A11) PC3
83	PK6	(ADC14/PCINT22)	(A10) PC2
84	PK5	(ADC13/PCINT21)	(A9) PC1
85	PK4	(ADC12/PCINT20)	(A8) PC0
86	PK3	(ADC11/PCINT19)	
87	PK2	(ADC10/PCINT18)	(T0) PD7
88	PK1	(ADC9/PCINT17)	(T1) PD6
89	PK0	(ADC8/PCINT16)	(XCK1) PD5
90	PJ7	(PCINT15)	(ICP1) PD4
99	PJ6	(PCINT14)	(TXD1/INT3) PD3
68	PJ5	(PCINT13)	(RXD1/INT2) PD2
67	PJ4	(PCINT13)	(SDA/INT1) PD1
66	PJ3	(PCINT12)	(SCL/INT0) PD0
65	PJ2	(XCK3/PCINT11)	
64	PJ1	(TXD3/PCINT10)	(CLK0/ICP3/INT7) PE7
63	PJ0	(RXD3/PCINT9)	(T3/INT6) PE6
79	PH7	(T4)	(OC3C/INT5) PE5
99	PH6	(OC2B)	(OC3B/INT4) PE4
18	PH5	(OC4C)	(OC3A/AIN1) PE3
17	PH4	(OC4B)	(XCK0/AIN0) PE2
16	PH3	(OC4A)	(TXD0) PE1
15	PH2	(XCK2)	(RXD0/PCIN8) PE0
14	PH1	(TXD2)	
13	PH0	(RXD2)	(ADCT7/TD1) PF7
27			(ADC6/TD0) PF6
18			(ADC5/TMS) PF5
17			(ADC4/TCK) PF4
16			(ADC3) PF3
15			(ADC2) PF2
14			(ADC1) PF1
13			(ADC0) PF0
12			
29			(OC0B) PG5
28			(TOSC1) PG4
70			(TOSC2) PG3
63			(ALE) PG2
52			(RD) PG1
51			(WR) PG0

- Microcontrolador ATmega2560: El microcontrolador que equipa la placa es el Atmel ATmega2560 de 8 bits. Este procesador posee 256KB de memoria flash, 8KB de RAM y 4KB de EEPROM.
- Pines digitales y analógicos: Tiene un total de 54 pines digitales para el manejo de entradas/salidas digitales, entre estos se encuentran los 16 pines analógicos para lectura de sensores, los 14 pines PWM para control de motores, etc.
- Puertos seriales: Posee 4 puertos seriales, UART0, UART1, UART2 y UART3 por lo que ofrece una capacidad extra para la conexión al ordenador y la comunicación con otros dispositivos.
- Reloj de tiempo real: El reloj de tiempo real incorporado permite a la placa realizar tareas de forma permanente sin depender de la computadora principal.
- Cuenta con 54 pines digitales, 15 pines PWM, 16 entradas analógicas, 4 puertos UART (uno para el puerto serie), un cristal oscilador resonante de 16MHz, compresión/expansión SPI y tiene la capacidad de conectarse a Internet y a otros dispositivos externos a través de Bluetooth, Wifi y Ethernet.

Temporizador

Se programa en el Arduino para que se active en un tiempo predeterminado y en cada momento que este temporizador se activa el Arduino ejecuta una función de Interrupción. Por ejemplo, se puede programar un temporizador que se dispare a cada segundo, o a cada 3 segundos y así sucesivamente, para que el Arduino cumpla una tarea específica.

Tipos de temporizadores:

Dado que el temporizador es un recurso del microcontrolador del Arduino podremos realizar el conteo de tiempo sin la necesidad de ningún hardware externo.

Temporizador Arduino 0: Es de 8bits, usado para las funciones `delay()`, `millis()`, `micros()`. Es conveniente NO modificar este temporizador para evitar alterar estas funciones que son muy comunes en nuestros códigos.

Temporizador Arduino 1: Es de 16 bits, usado por la librería Servo en Arduino Uno. Se usa principalmente para el control de Servos, sin embargo si no estás controlando servos, puedes usar este temporizador en Arduino 1 libremente.

Temporizador Arduino 2: Es de 8 bits, usado en la función `tone()`. Si no usas este módulo, puedes emplear el temporizador Arduino 2 libremente para temporizar otra aplicación.

Temporizador Arduino 3, 4, y 5: Solo disponibles en los Mega; todos de 16bits, muy utilizados principalmente para el control de Servos, sin embargo también se puede usarlos como mejor a uno le convenga.

¿Cuánto cuestan?

El precio varía mucho según el modelo pero podemos decir que van desde los 6\$ USD hasta los 100\$ USD

¿Dónde se comprarían?

Estando en Venezuela nuestra mejor opción sería www.arduinoe.com o MercadoLibre; con suerte en alguna tienda local, aunque también podríamos adquirirlo mediante la página oficial de Arduino u otras páginas como EBay, Amazon, Craigslist entre otras.

¿Qué se necesitan para programarlos?

Estos son algunos de los entornos de desarrollo integrado (IDE) para programar en Arduino

- Arduino IDE 2.2.1
- Nightly Builds
- Arduino with Chromebook
- MicroPython With Arduino Boards
- Arduino PLC IDE 1.0.3
- Legacy IDE (1.8.X)/ Arduino IDE 1.8.19

Ejemplo de código/programa para Arduino

Entrada digital

Este ejemplo lee el valor de un interruptor conectado en el pin 2. Cuando el interruptor está cerrado en el pin de entrada habrá un estado alto (HIGH) y se encenderá el LED.

```
int ledPin = 13; // Pin de salida para el LED

int inPin = 2; // Pin de entrada (donde está conectado el interruptor)

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(inPin, INPUT);
}

void loop() {
  if (digitalRead(inPin) == HIGH){      // Si se activa interruptor
    digitalWrite(ledPin, HIGH); // Enciende el LED
    delay(1000);                  // Pause de 1 segundo
    digitalWrite(ledPin, LOW);  // Apaga el LED
    delay(1000);                  // Pausa de 1 segundo
  }
}
```