INICIANDO CON ARDUINO

ARDUINO UNO r3

Por William David Galvis

Octubre 28 del 2013

¿Qué es el Arduino Uno?

El Arduino Uno es una tarjeta con un microcontrolador ATmega328 tiene 14 pines para entradas (I) y salidas (O) digitales que van desde la número 0 hasta la I/O número 13. (Enmarcados con rojo en la figura 1).

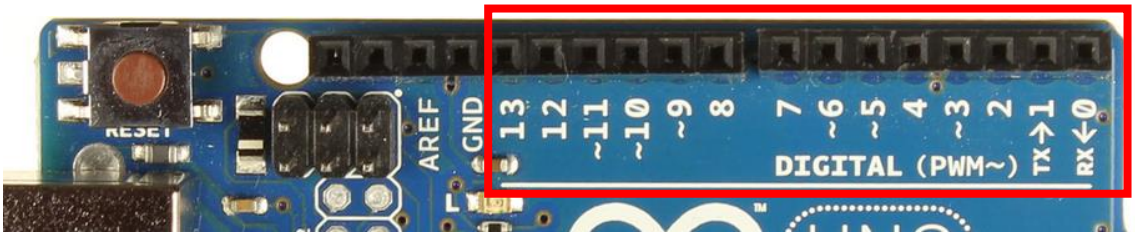

Figura 1.

E-Mail: info@microkitselectronica.com - www.microkitselectronica.com
Calle: 41 No. 51 15, local 141, Teléfono: 232 77 84,
Centro Comercial Paseo Bolívar, Carabobo con los Huesos, Medellín, Antioquia, Colombia.

De esos 14 pines 6 pueden ser usados como salidas PWM (también llamadas en algunos casos como salidas análogas) estos pines están marcados con la tilde de la eñe o virgulilla (~) en la imagen anterior.

Tiene 6 entradas análogas marcadas como ANALOG IN usadas para la lectura de sensores que entregan una variación de voltaje dependiendo de la variable que miden, estas entradas análogas se pueden usar también como salidas digitales. Los otros pines auxiliares de voltaje están marcados en la tarjeta como POWER y son voltajes que pueden ser usados para conectar dispositivos directamente a la tarjeta y para que se acoplen correctamente a los circuitos de complemento para el Arduino (Shields).

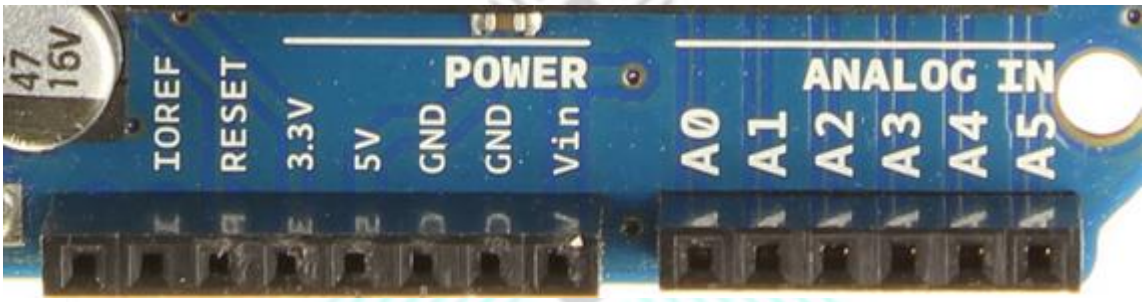

Figura 2.

Pines	Función
IOREF	Este pin en el Arduino se usa para proveer el voltaje de referencia con el que el microcontrolador va a operar.
RESET	Este pin está conectado al botón de reset y facilita el reset del microcontrolador desde una tarjeta que se conecte al Arduino por ejemplo un Shield o escudo.
3.3V	Es un voltaje entregado por el regulador en la tarjeta, la máxima corriente que puede entregar es de 50mA
5V	Este pin entrega 5 voltios regulados directamente del circuito instalado en la tarjeta, el Arduino puede ser alimentado directamente desde la entrada USB o usando el Jack de potencia ingresando un voltaje DC entre 7 y 12 voltios.
GND	Son los pines de conexión a tierra o referencia.
AREF	Este pin se usa para configurar el voltaje de referencia usado por el módulo de conversión análogo a digital del microcontrolador, se usa para casos específicos de ADC, si no se cambia su voltaje el Microcontrolador queda por defecto a 5V internamente, al cambiarlo, se debe especificar por software que el voltaje de referencia para el ADC estará en ese pin.

Tabla 1.

¿Qué es un Shield?

Una característica de los Arduinos es que son tarjetas modulares y permiten la integración con otros circuitos electrónicos que facilitan su uso y extienden sus capacidades agregando características y funciones, estos circuitos hechos a la medida para empotrarse en el Arduino y ser compatibles pin a pin con ellos, son llamados Shield, en español significa escudo o complemento. Por ejemplo está el MP3 Shield (figura 3a), el cual es un complemento para que se pueda controlar una lista de reproducción desde el Arduino, pausar adelantar, y copiar música son unas de las funciones que facilita este módulo, el joystick Shield es usado para hacer del Arduino todo un control de juegos, por su lado el LCD Keypad Shield agrega un periférico de salida al Arduino, permitiendo la visualización de datos en un display LCD 2X16 además de facilitar el ingreso de datos con seis pulsadores.

a b c

Figura 3.

¿Cómo se programa el microcontrolador en la placa Arduino?

Elementos físicos necesarios (Hardware):

Una facilidad de los Arduinos es que permiten ser programados por comunicación serial desde el computador, simplemente se manda el programa por el cable USB y el microcontrolador lo lee por su puerto serial y lo escribe en su memoria interna para luego ejecutarlo, esto es posible porque a las placas Arduino antes de ser vendidas se les programa el microcontrolador con un programa llamado bootloader para no usar un circuito de programación, en caso de que el microcontrolador no tuviera programado el bootloader se podría adquirir uno con él preinstalado (ejemplo el Atmega328) o programarlos con un circuito de programación que es externo al Arduino y sería el intermediario entre la

comunicación USB del computador y el protocolo de programación del microcontrolador el cual necesita velocidades y voltajes diferentes a los que entrega el computador, para que el programa se pueda almacenar en la memoria flash interna, estos circuitos de programación son llamados quemadores o programadores y en algunos casos depuradores, estos suelen ser costosos y complicados de conectar, esta particularidad hace que la programación de Arduino sea más fácil y económica.

Para que un microcontrolador con comunicación serial pueda “entenderse” con un computador con comunicación USB se necesita un chip conversor serial a USB, en el mercado existen muchos fabricantes que ofrecen circuitos integrados de este tipo, los Arduinos Duemilanove en particular, usan los FT232RL de la empresa Ftdichip, para hacer la conversión, el Arduino UNO usa un microcontolador Atmel, programado como un conversor serial a USB en vez de un FTD232 este micro es el MEGA16U2, otras tarjetas Arduino al incorporar microcontroladores con módulos internos de comunicación USB se les programa bootloaders para cargar programas directamente desde el puerto USB y no necesitarían más que el cable de comunicación con el computador.

Software necesario para la programación:

El Software usado para la programación de estas tarjetas es el Arduino IDE, estas siglas significan ambiente de desarrollo integrado (en inglés: Integrated Development Environment) que hace referencia a un conjunto de herramientas necesarias para la edición, compilación, programación del microcontrolador entre otras funciones en un solo programa, su uso es gratuito y libre, se puede descargar la última versión disponible en: http://arduino.cc/en/main/software

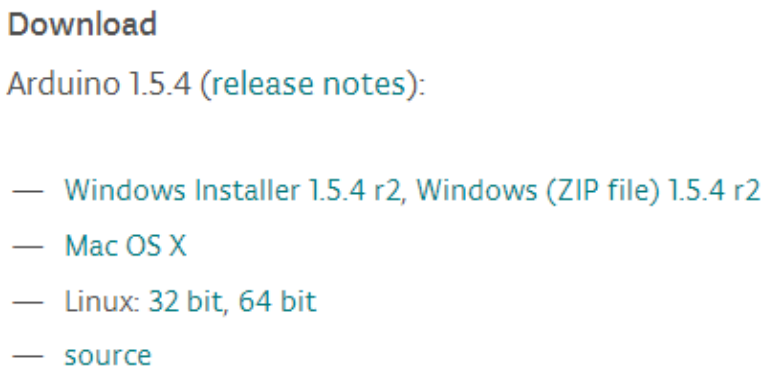

Figura 4.

Se elige, la versión compatible con el sistema operativo que se esté trabajando, se descarga, y se descomprime o extraen los archivos, una vez extraídos en Windows se ven como en la figura 5a.

a b

Figura 5

Se hace clic en el icono de Arduino y ya se tiene listo el entorno de desarrollo (figura 5b).

Cargar un programa de ejemplo Encender y apagar un led (Blink)

1. El entorno de arduino trae muchos programas de ejemplo ya instalados y listos para usar, uno de ellos es el “hola mundo” de los microcontroladores poner a titilar un led. Para usar los programas de ejemplo que hay en en entorno hay que navegar hasta blink (figura 6) asi: hacer clic en Archivos>Ejemplos>Basic>Blink al hacer clic en Blink se abre otra ventana de Arduino con el codigo respectivo.

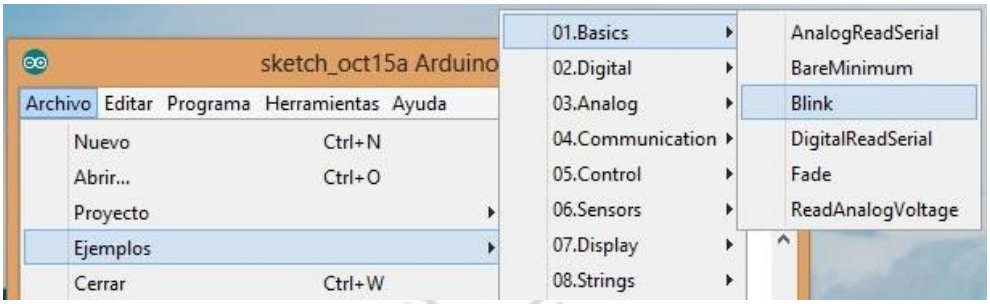

Figura 6

2. Se conecta el Arduino Uno al computador usando el cable USB, luego en Herramientas se busca el puerto COM virtual (Puerto serial virtual) que se le asignó a la tarjeta y se selecciona.

Figura 7

3. De la misma manera se busca el Arduino Uno en la pestaña Board en el menú de herramientas.

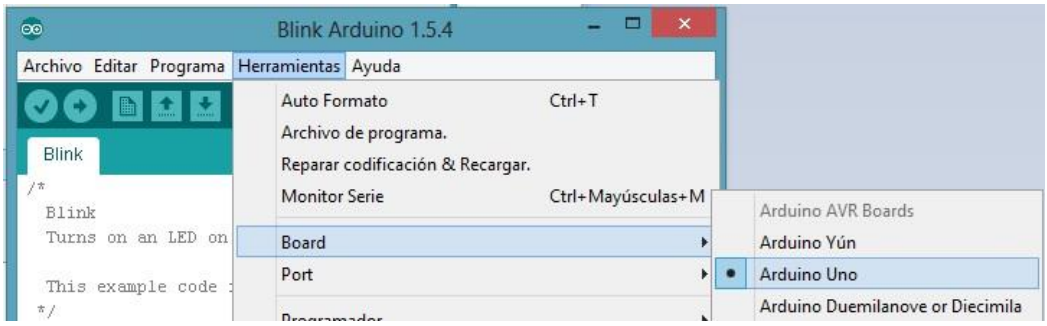

Figura 8

4. Teniendo el programa listo se procede a cargarlo al microcontrolador del Arduino, haciendo clic en la flecha horizontal “Subir” (figura 9), se observará en la tarjeta que los leds de comunicación serial TX y RX empiezan a parpadear indicando que hay comunicación entre el PC y el Arduino durante la programación, cuando esto termine el led conectado al pin 13 del Arduino iniciará una secuencia de encendido y apagado como lo indica el programa, y ya se tendrá el microcontrolador programado con un ejemplo como el siguiente:

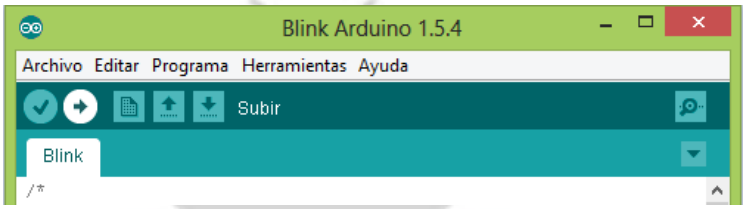

Figura 9

```
 /* Blink Enciende un led por un segundo y luego lo apaga por un
segundo, repetidamente. Este código de ejemplo es de dominio
público. */

// El pin 13 tiene un led conectado en muchas tarjetas Arduino.

// Se le da un nombre al pin 13, para este caso “led”:

int led = 13;

void setup() { // La rutina setup corre una vez se presione reset:

pinMode(led, OUTPUT); // Inicializa el pin 13 como una salida

// La rutina loop una y otra vez infinitas veces hasta que se
//apague el microcontrolador:

void loop() {

digitalWrite(led,HIGH);     //enciende el led (HIGH es alto)

delay(1000);                // Espera un segundo

digitalWrite(led, LOW);     // apaga el led con bajo voltaje

delay(1000);                // Espera un segundo

  }
```

Errores

Así como se hizo con este ejemplo, se pueden cargar muchos otros de prueba para familiarizarse con la programación en Arduino antes de proceder a crear un programa propio, en esta fase se experimenta con posibles errores o problemas a la hora de probar el Arduino y pueden ser:

Error de conexión del cable USB, cuando este se conecta mal puede alimentar el Arduino pero no comunicar bien y al intentar programar en puede aparecer un mensaje de error en la consola de Arduino (Caja negra en la parte inferior) diciendo: `avrdude: stk500_getsync(): not in sync: resp=0x00`

Se soluciona conectando nuevamente el cable USB al PC y al Arduino.

También es posible que no reconozca el Arduino conectado al computador y en ese caso la opción port de la figura 7, no mostrará ningún puerto COM disponible es posible que el computador no tenga el driver de comunicación instalado para ello hay que descargar e instalar este controlador aquí.

Si esto ocurre en Linux no se necesita instalar el driver sino montar el puerto serial para que el IDE de Arduino tenga los permisos de usarlo, para ello se usa el comando:

```
sudo chown [Nombre de usuario] /dev/ttyUSB[x]
```

Donde [Nombre de usuario] será el usuario Linux al que se le da el permiso para que el IDE de Arduino use el puerto serial. Y [x] será el número del puerto donde se conecta el Arduino, por ejemplo para un usuario llamado Bill con el Arduino conectado al puerto ttyUSB1 el comando sería:

```
sudo chown Bill /dev/ttyUSB1.
```

De esa forma se habilita el puerto serial para Linux en caso de que no se otorguen los permisos por defecto.

Hasta aquí una breve introducción al Arduino Uno r3 y lo necesario para programar algo senillo y aprender algunos conceptos que se van a necesitar a lo largo de proyectos electrónicos en esta plataforma de desarrollo.

REFERENCIAS

Arduino uno: http://arduino.cc/en/Main/arduinoBoardUno

E-Mail: info@microkitselectronica.com - www.microkitselectronica.com
Calle: 41 No. 51 15, local 141, Teléfono: 232 77 84,
Centro Comercial Paseo Bolívar, Carabobo con los Huesos, Medellín, Antioquia, Colombia.