

Materia: Proyecto Integrador

Docente: Gonzalo Cristian Vera

Alumno: Raúl Jara

ARDUINO IDE

Dado que el Arduino es como un pequeño ordenador que ejecuta una serie de códigos que previamente le hemos introducido, necesitaremos un programa para poder meter estos códigos a la propia placa. Este programa se llama IDE, que significa "Integrated Development Environment" ("Entorno de Desarrollo Integrado"). Este IDE estará instalado en nuestro PC, es un entorno muy sencillo de usar y en él escribiremos el programa que queramos que el Arduino ejecute. Una vez escrito, lo cargaremos a través del USB y Arduino comenzará a trabajar de forma autónoma.



El **entorno de desarrollo integrado (IDE) de Arduino** es una aplicación multiplataforma (para Windows, macOS, Linux) que está escrita en el lenguaje de programación Java. Se utiliza para escribir y cargar programas en placas compatibles con Arduino, pero también, con la ayuda de núcleos de terceros, se puede usar con placas de desarrollo de otros proveedores.

El código fuente para el IDE se publica bajo la Licencia Pública General de GNU, versión 2.

El IDE de Arduino admite los lenguajes C y C++ utilizando reglas especiales de estructuración de códigos. El IDE de Arduino suministra una biblioteca de software del proyecto Wiring, que proporciona muchos procedimientos comunes de E/S.

El código escrito por el usuario solo requiere dos funciones básicas, para iniciar el boceto y el ciclo principal del programa, que se compilan y vinculan con un apéndice de programa main() en un ciclo con el GNU toolchain, que también se incluye. El IDE de Arduino emplea el programa avrdude para convertir el código ejecutable en un archivo de texto en codificación hexadecimal que se carga en la placa Arduino mediante un programa de carga en el firmware de la placa.

PERO QUE ES ARDUINO?

Arduino es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso.

Para poder entender este concepto, primero vas a tener que entender los conceptos de hardware libre y el software libre. El hardware libre son los dispositivos cuyas especificaciones y diagramas son de acceso público, de manera que cualquiera puede replicarlos. Esto quiere decir que Arduino ofrece las bases **para que cualquier otra persona o empresa pueda crear sus propias placas**, pudiendo ser diferentes entre ellas pero igualmente funcionales al partir de la misma base.

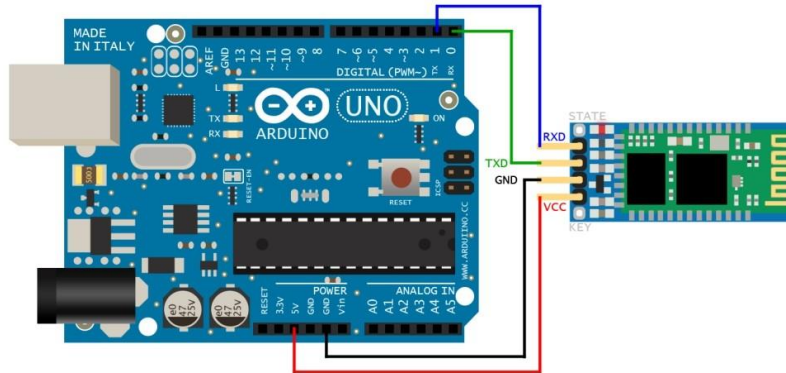
Enlace explicativo: <https://www.youtube.com/watch?v=Zs9MZosVuqo>

El software libre son los programas informáticos **cuyo código es accesible por cualquiera** para que quien quiera pueda utilizarlo y modificarlo. Arduino ofrece la plataforma Arduino IDE (Entorno de Desarrollo Integrado), que es un entorno de programación con el que cualquiera puede crear aplicaciones para las placas Arduino, de manera que se les puede dar todo tipo de utilidades.

El proyecto nació en 2003, cuando varios estudiantes del Instituto de Diseño Interactivo de Ivrea, Italia, con el fin de facilitar el acceso y uso de la electrónica y programación. Lo hicieron para que los estudiantes de electrónica tuviesen una alternativa más económica a las populares BASIC Stamp, unas placas que por aquel entonces valían más de cien dólares, y que no todos se podían permitir.

El resultado fue Arduino, una placa con todos los elementos necesarios

para **conectar periféricos a las entradas y salidas de un microcontrolador**, y que puede ser programada tanto en Windows como macOS y GNU/Linux. Un proyecto que promueve la filosofía 'learning by doing', que viene a querer decir que la mejor manera de aprender es cacharreando.



El Arduino es una placa basada en un microcontrolador ATMELE. Los microcontroladores son **circuitos integrados en los que se pueden grabar instrucciones**, las cuales las escribes con el lenguaje de programación que puedes utilizar en el entorno Arduino IDE. Estas instrucciones permiten crear programas que interactúan con los circuitos de la placa.

El microcontrolador de Arduino posee lo que se llama una **interfaz de entrada**, que es una conexión en la que podemos conectar en la placa diferentes tipos de periféricos. La información de estos periféricos que conectes se trasladará al microcontrolador, el cual se encargará de procesar los datos que le lleguen a través de ellos.

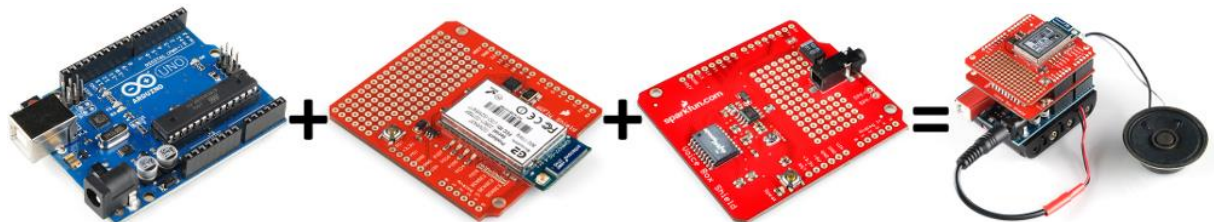
El tipo de periféricos que puedas utilizar para enviar datos al microcontrolador depende en gran medida de qué uso le estés pensando dar. Pueden ser cámaras para obtener imágenes, teclados para introducir datos, o diferentes tipos de sensores.

También cuenta con una **interfaz de salida**, que es la que se encarga de llevar la información que se ha procesado en el Arduino a otros periféricos. Estos periféricos pueden ser pantallas o altavoces en los que reproducir los datos procesados, pero también pueden ser otras placas o controladores.



Arduino es un proyecto y no un modelo concreto de placa, lo que quiere decir que compartiendo su diseño básico te puedes encontrar con **diferentes tipos de placas**. Las hay de varias formas, tamaños y colores para a las necesidades del proyecto en el que estés trabajando, las hay sencillas o con características mejoradas, Arduinos orientados al Internet de las Cosas o la impresión 3D y, por supuesto, dependiendo de estas características te encontrarás con todo tipo de precios.

Además, las placas Arduino también cuentan con otro tipo de componentes llamados Escudos (Shields) o mochilas. Se trata de una especie de placas que se conectan a la placa principal para añadirle una infinidad de funciones, como GPS, relojes en tiempo real, conectividad por radio, pantallas táctiles LCD, placas de desarrollo, y un larguísimo etcétera de elementos.



Qué podemos hacer con uno?

La enorme flexibilidad y el carácter libre y abierto de Arduino hacen que puedas utilizar este tipo de placas prácticamente para cualquier cosa, desde relojes hasta básculas conectadas, pasando por robots, persianas controladas por voz o tu propia vending machine.

Pasos para la Instalación de ARDUINO IDE

En esta sección, aprenderemos en pasos sencillos cómo configurar Arduino IDE en nuestra computadora y preparar la placa para recibir el programa a través del cable USB.

Paso 1: primero debe tener su placa Arduino (puede elegir su placa favorita) y un cable USB. En caso de que use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560 o Diecimila, necesitará un cable USB estándar (enchufe A a enchufe B), del tipo que conectaría a una impresora USB como se muestra en la siguiente imagen.



Cable USB

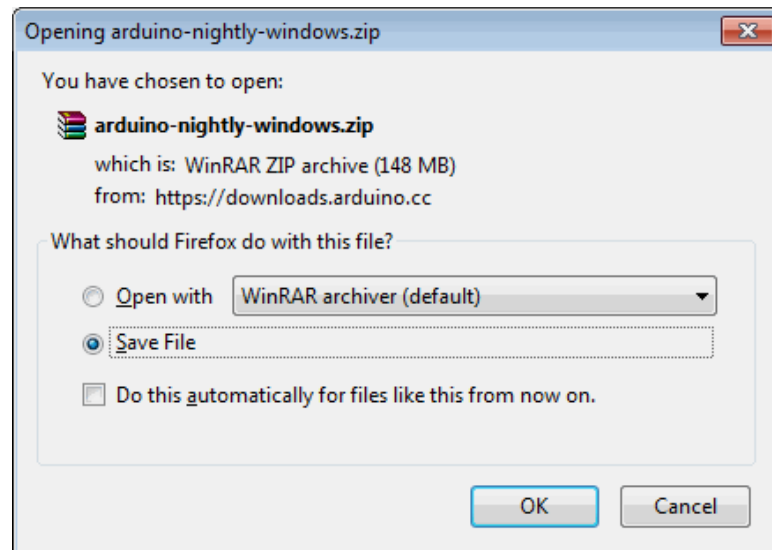
En caso de que use Arduino Nano, necesitará un cable A a Mini-B, como se muestra en la siguiente imagen.



Mini cable

Paso 2: descargue el software Arduino IDE.

Puede obtener diferentes versiones de Arduino IDE desde la página de descargas en el sitio web oficial de Arduino. Debe seleccionar su software, que es compatible con su sistema operativo (Windows, IOS o Linux). Una vez completada la descarga del archivo, descomprima el archivo.



Abrir ventanas nocturnas de Arduino

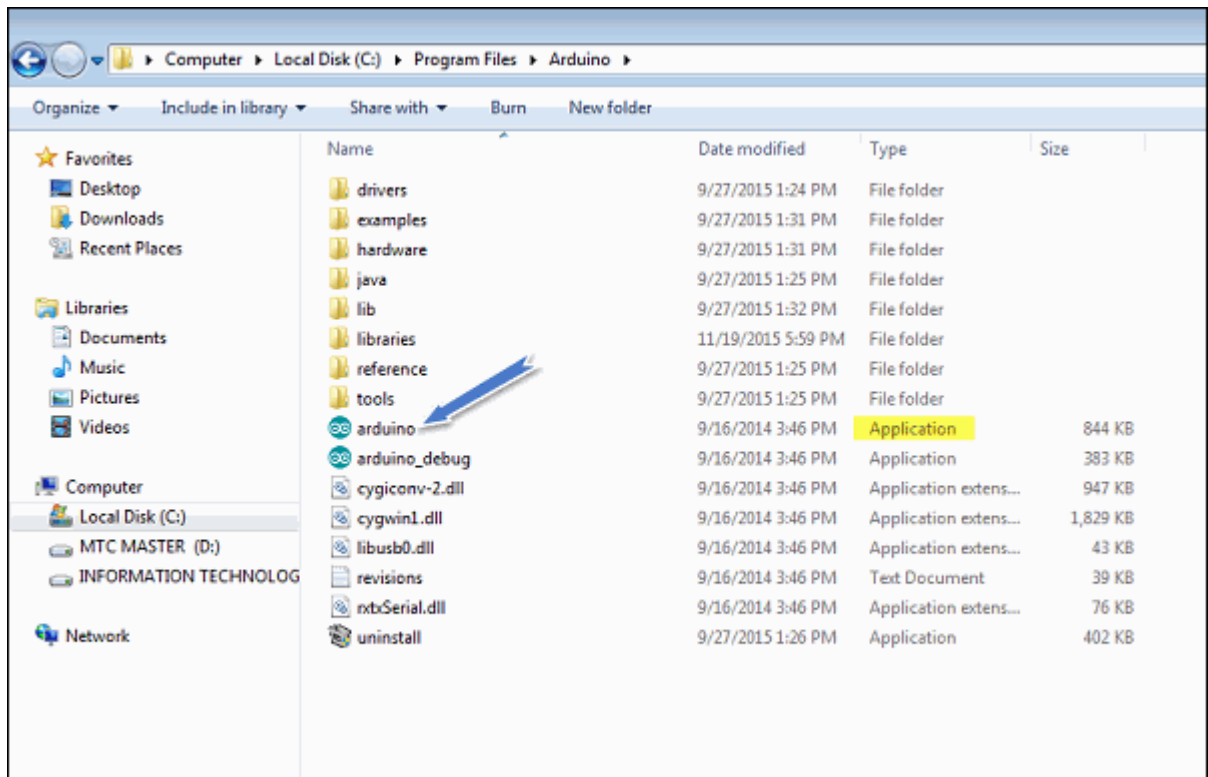
Paso 3 - Enciende tu tablero.

Arduino Uno, Mega, Duemilanove y Arduino Nano obtienen energía automáticamente de la conexión USB a la computadora o de una fuente de alimentación externa. Si está utilizando un Arduino Diecimila, debe asegurarse de que la placa esté configurada para extraer energía de la conexión USB. La fuente de alimentación se selecciona con un puente, una pequeña pieza de plástico que se ajusta a dos de los tres pines entre el USB y las tomas de alimentación. Verifique que esté en los dos pines más cercanos al puerto USB.

Conecte la placa Arduino a su computadora con el cable USB. El LED de alimentación verde (etiquetado PWR) debería iluminarse.

Paso 4: inicia Arduino IDE.

Después de descargar su software Arduino IDE, debe descomprimir la carpeta. Dentro de la carpeta, puede encontrar el icono de la aplicación con una etiqueta de infinito (application.exe). Haga doble clic en el icono para iniciar el IDE.



Inicie Arduino IDE

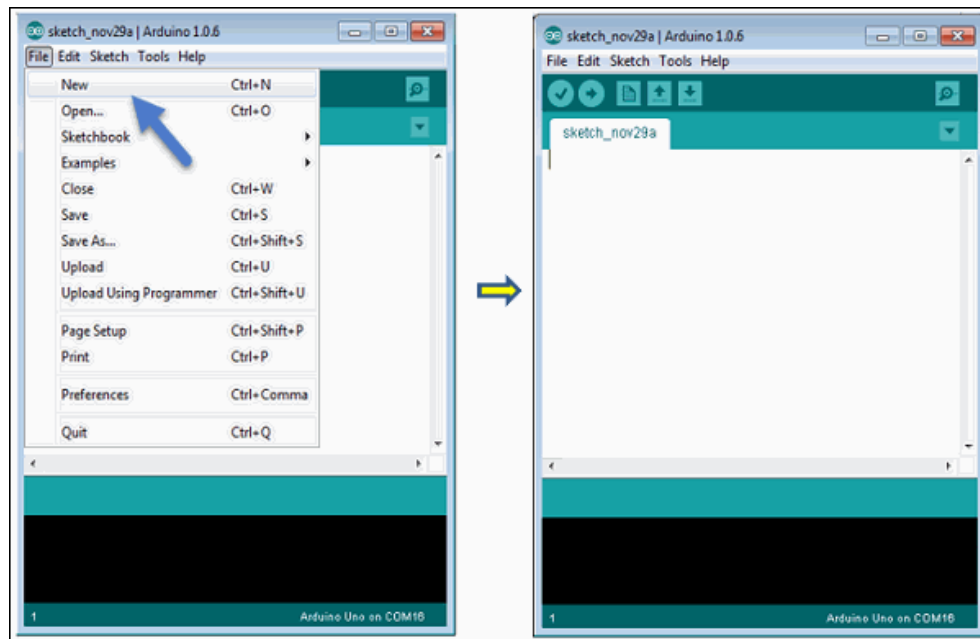
Paso 5: abre tu primer proyecto.

Una vez que se inicia el software, tiene dos opciones:

Crea un nuevo proyecto.

Abra un ejemplo de proyecto existente.

Para crear un nuevo proyecto, seleccione Archivo → Nuevo.



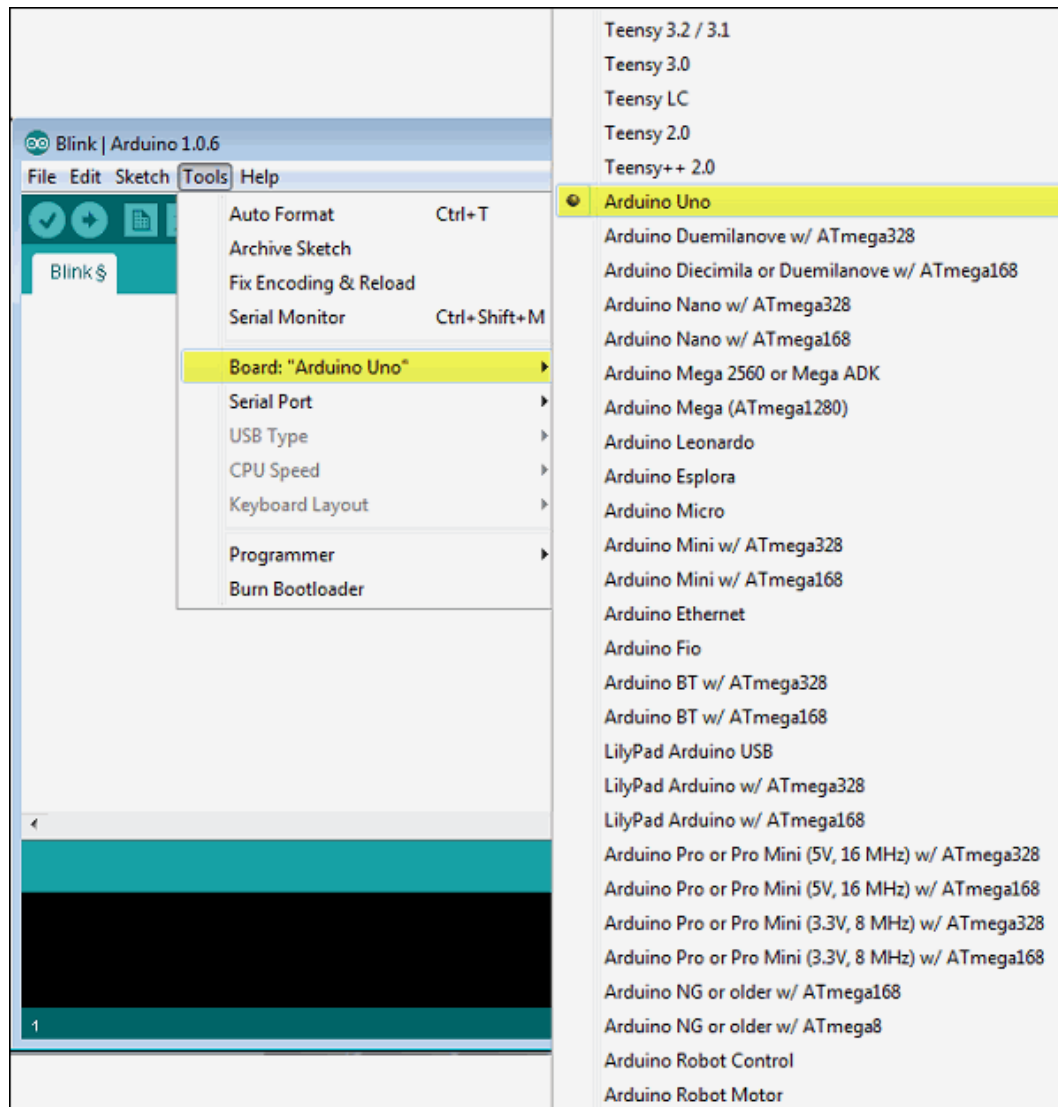
Editar archivo

Para abrir un ejemplo de proyecto existente, seleccione Archivo → Ejemplo → Conceptos básicos → Parpadeo.

Aquí, estamos seleccionando solo uno de los ejemplos con el nombre Blink. Enciende y apaga el LED con cierto retraso de tiempo. Puede seleccionar cualquier otro ejemplo de la lista.

Paso 6: selecciona tu placa Arduino.

Para evitar cualquier error al cargar su programa en la placa, debe seleccionar el nombre correcto de la placa Arduino, que coincide con la placa conectada a su computadora. Vaya a Herramientas → Tablero y seleccione su tablero.

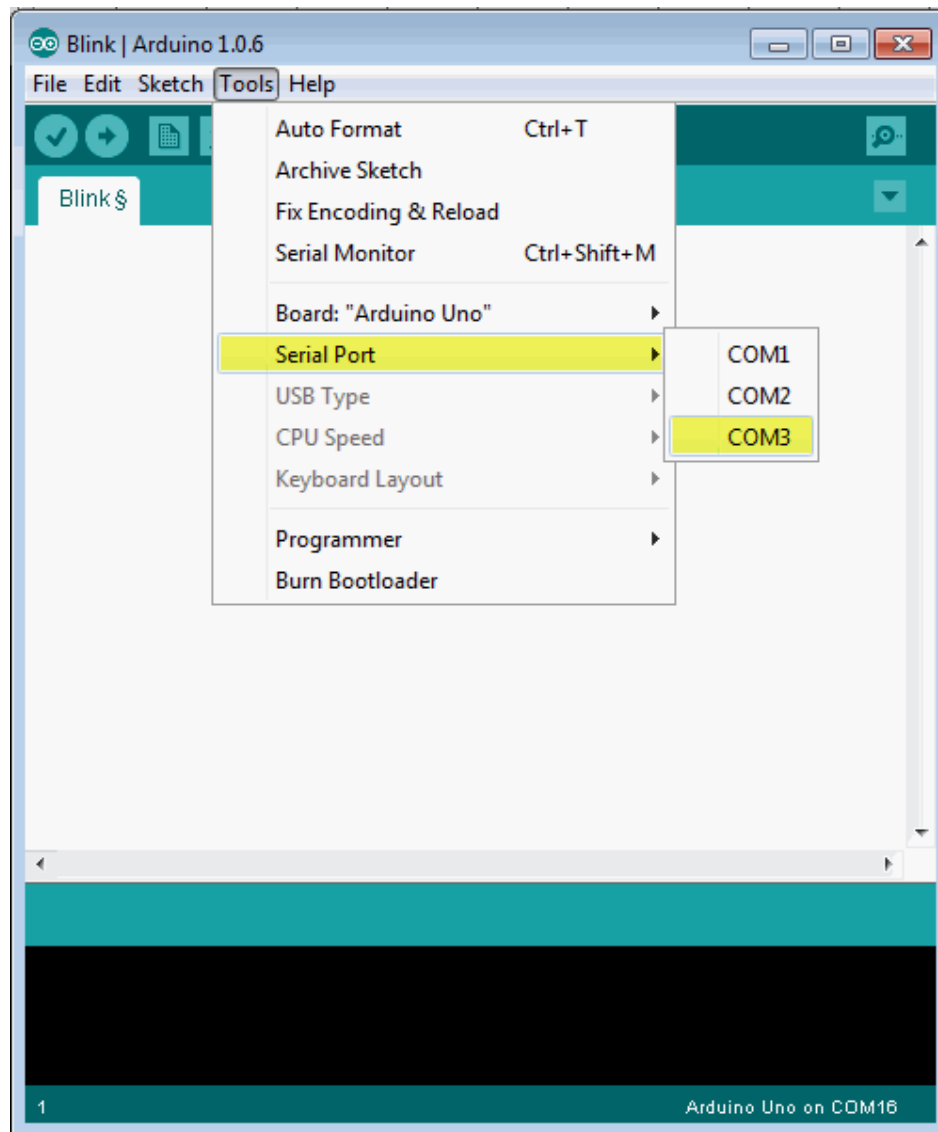


Seleccionar tablero

Aquí, hemos seleccionado la placa Arduino Uno de acuerdo con nuestro tutorial, pero debe seleccionar el nombre que coincida con la placa que está utilizando.

Paso 7: selecciona tu puerto serie.

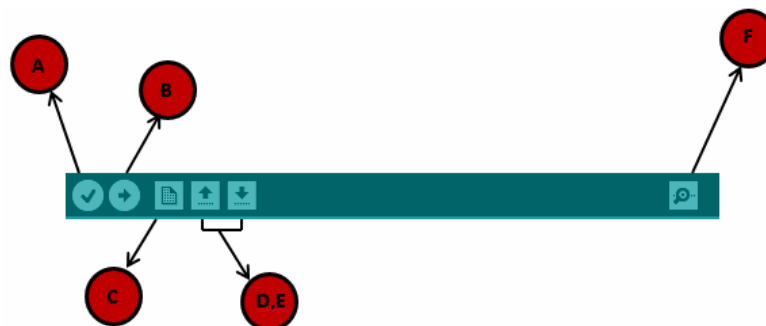
Seleccione el dispositivo en serie de la placa Arduino. Vaya al menú Herramientas → Puerto serie. Es probable que sea COM3 o superior (COM1 y COM2 generalmente están reservados para puertos seriales de hardware). Para averiguarlo, puede desconectar su placa Arduino y volver a abrir el menú, la entrada que desaparece debe ser de la placa Arduino. Vuelva a conectar la placa y seleccione ese puerto serie.



Puerto serial

Paso 8: sube el programa a tu tablero.

Antes de explicar cómo podemos cargar nuestro programa en el tablero, debemos demostrar la función de cada símbolo que aparece en la barra de herramientas Arduino IDE.



Subir programa

1. A - Se usa para verificar si hay algún error de compilación.
2. B: se utiliza para cargar un programa en la placa Arduino.
3. C: acceso directo utilizado para crear un nuevo boceto.
4. D: se utiliza para abrir directamente uno de los bocetos de ejemplo.
5. E: se utiliza para guardar su boceto.
6. F: monitor en serie utilizado para recibir datos en serie de la placa y enviar los datos en serie a la placa.

Ahora, simplemente haga clic en el botón "Cargar" en el entorno. Espere unos segundos; verá los LED RX y TX en el tablero, parpadeando. Si la carga se realiza correctamente, aparecerá el mensaje "Carga finalizada" en la barra de estado.

Nota: si tiene un Arduino Mini, NG u otra placa, debe presionar el botón de reinicio físicamente en la placa, inmediatamente antes de hacer clic en el botón de carga en el software Arduino.