

Utilización del Arduino para controlar motores de paso por medio del flasheo de GRBL

Alfredo Ricci, Juan Andrés Urrea

Resumen- Este informe busca mostrar la totalidad del procedimiento realizado para configurar un computador y un arduino como controladores para motores de paso. Se centra especialmente en la instalación del software necesario tanto en su utilización una vez esto se completa. De esta forma, explica el procedimiento de instalación y configuración, sirviendo a la vez como un tutorial para repetir este procedimiento.

1 Introducción

Dada la necesidad de tener un conocimiento previo del manejo tanto manual como automatizado de los motores utilizados para el proyecto **Mechanical Scanner**, se ideó la estrategia de reproducir esta situación a pequeña escala, usando motores de paso para luego conectarlos a un Arduino flasheado con GRBL y aprender, desde cero, los procesos necesarios para transmitir órdenes de trayectorias desde un computador hasta los motores de paso utilizados. De esta forma, la construcción exitosa de este montaje y la capacidad de controlar detalladamente estos dispositivos por medio de un manejo apropiado del software permitirá obtener este conocimiento previo a la llegada y utilización del **Mechanical Scanner**.

2 Montaje Utilizado

DESCRIBIR MONTAJE CON FOTO Y TEXTO.

3 Manejo de Software

3.1 Instalación de Software Utilizado

Antes de la explicación detallada acerca del proceso de instalación, se enuncian a continuación los diferentes programas y software utilizados:

- **GRBL**

– Versión: 0.8

GRBL es un controlador de movimiento disponible para funcionar con Arduino, escrito enteramente en C, que sirve como alternativa al control por puerto paralelo. Este software permite utilizar archivos escritos en *GCode* para diseñar trayectorias a seguir, de manera que estos se convierten en instrucciones para el Arduino, por medio de la instalación de GRBL, que luego son enviadas a la tarjeta controladora CNC.

- **Arduino UNO**

– Versión 1.6.5

Software que funciona como interfaz para controlar el dispositivo arduino desde un computador. Puede instalarse directamente desde la página web de Arduino. Software programado en Java, permite la creación de código que luego es subido directamente al dispositivo Arduino.

- **GCode Universal Sender**

– Versión: 1.0.8

Software escrito en Java que funciona como complemento para GBRL como intérprete de código G que puede ser enviado directamente al Arduino para ser interpretado como trayectoria. Permite, además de esta alternativa automatizada, un control manual por medio de botones que indican movimiento en cada eje, lo que es traducido luego a código G y enviado también al Arduino.

- **Inkscape**

– Versión: 0.91

Software de gráficas vectoriales disponible para uso en Windows, Mac y Linux. De entre sus distintas funcionalidades, permite exportar archivos como **.DXF**, para luego guardarlos como **GCode**, de manera que resulta la herramienta más útil para la creación de trayectorias y su conversión a código g para ser enviado a GCode Universal Sender.

Instalación del Software Requerido

Una vez se sabe cual es el software requerido, se presenta ahora el proceso detallado de instalación de estos. Cabe notar que el proceso mostrado a continuación se realizó tanto en Linux como en Mac. Esto con la finalidad de utilizar los procedimientos más sencillos que ofrece cada sistema operativo. Para este caso, el flasheo de GRBL al Arduino se realizó desde Linux, mientras que la construcción y envío de GCode al Arduino se realizó desde Mac. Estas diferencias se enunciarán en cada paso.

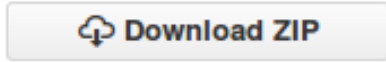
1. Revisión de Librerías de Instalación Linux

Cuando se trabaja con Linux, específicamente la distribución Ubuntu 14.04 "Trusty", se debe realizar una verificación de librerías que permiten la instalación correcta de paquetes. Dicho proceso se puede realizar con los siguientes comandos de instalación en la terminal. En caso de que ya se encuentren actualizadas, en la terminal esto será notificado. En caso de que no, se instalarán.

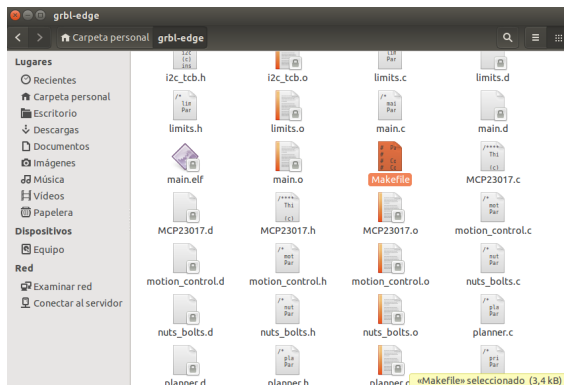
```
sudo apt-get install ruby
sudo apt-get install avrdude
```

2. Descargar GRBL y Flashearlo

Ahora, es necesario descargar todo lo relacionado a GRBL. Esto se puede realizar desde GitHub, en el enlace [1]. Para hacer esto, se recomienda, si es que es la primera vez que se trabaja con esta herramienta, seleccionar la opción **Download Zip** en la parte derecha de la página.



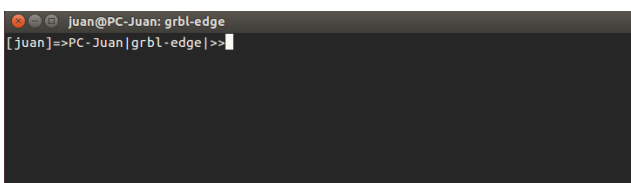
El archivo zip descargado debe luego ser descomprimido en una carpeta cuya localización resulte fácil de recordar. Debido a como está en GitHub, el archivo descargado se llamará **grbl-edge**. Para continuar este procedimiento, se abre ahora la carpeta recién descargada, buscando el archivo llamada **Makefile**, el cual debe ser después abierto con un editor de texto de preferencia personal, en este caso **Gedit**, editor de texto predeterminado de Ubuntu.



Una vez abierto este archivo, se busca la línea que empieza por la palabra *PROGRAMMER*, para reemplazarla por lo siguiente:

```
PROGRAMMER = -c stk500v1 -P /dev/  
ttyACM0 -b 115200
```

El archivo luego debe guardarse para actualizar y aplicar los cambios realizados. Una vez se hace esto, se debe abrir una terminal nueva. Esto para aprovechar las ventajas de control que esta aplicación ofrece. Desde esta se debe acceder a la carpeta **grbl-edge** creada anteriormente. Aunque la apariencia de la terminal depende sustancialmente de la configuración en cada equipo, para este caso específico se obtiene la siguiente ventana:



Una vez dentro de este entorno, se utiliza las siguientes líneas de código, para inicializar el archivo **make** dentro de la carpeta.

```
sudo make clean  
sudo make
```

Realizar esto causará que aparezcan varias notificaciones en la terminal, las cuales sirven únicamente para informar de los procesos completados. Para finalizar este procedimiento, solo resta flashear GRBL al Arduino. Para hacer esto, se debe conectar el Arduino al computador en uso. Luego de hacer esto, se ejecuta en la terminal la siguiente línea de código.

```
sudo make flash
```

Esta línea instalará GRBL en el arduino, tras de lo cual se puede desconectar del computador y usar como controlador de movimiento de ahora en adelante.

3. Descargar GCode Universal Sender

Para descargar este programa, se recurre de nuevo a GitHub, donde está guardado todo lo relacionado a este software, disponible en [2]. Para comenzar, se descarga la última versión estable disponible. Para Linux y Windows, esta es la versión 1.0.8, mientras que para Mac es la versión 1.0.7. Esto se hace por medio de los links de descarga dados en la sección de información de la página.

Stable builds

1.0.8 - The last stable build before 2.0 work.

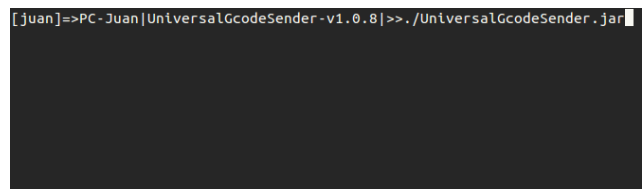
1.0.7 - Requires Java 7 or higher.

1.0.6 - Requires Java 6 or higher.

Older releases can be found on the [downloads page](#)

Dependiendo de en qué sistema operativo se esté trabajando, la manera de ejecutar este programa será diferente. Para este caso, con Ubuntu, se debe utilizar la terminal para ejecutar el archivo **.jar** dentro de la carpeta descargada, usnado la siguiente línea de código:

```
./UniversalGcodeSender.jar
```



4. Descargar Inkscape

Existen distintas formas de instalar **Inkscape**, las cuales dependen del sistema operativo en el cual se trabaje. Lo más recomendable es realizarlo desde la página oficial [3], sin embargo para este caso se realizó desde el **Centro de Software de Ubuntu**. Trabajando desde Mac, la sola instalación permite ya las funcionalidades completas de Inkscape, entre las

cuales se incluye la exportación a **Gcode**. Para el caso de la instalación desde Ubuntu, se deben seguir unas instrucciones adicionales para obtener la extensión

GCodeTools. Para esto, se debe descargar el archivo zip de [4] y seguir las instrucciones iniciales de [5].