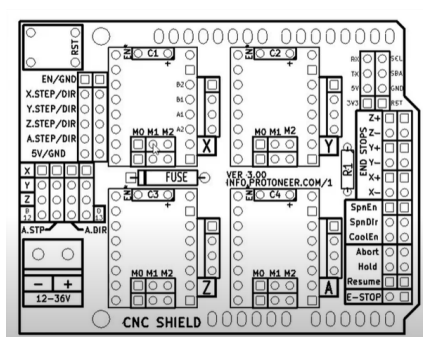


Configuración de parámetros para el driver CNC -GRBL

Se configuró los driver a 1/8 de paso y los husillos del x y y tienen un paso de 8mm por vuelta Y el del eje z 2mm por vuelta.

En la siguiente imagen se muestran los driver para los motores en nuestro caso se colocaron jumpers en M0 y M1, es por ello que se configuró a 1/8.



En la siguiente tabla se muestra los datos sobre la configuración de los pasos al nosotros tener configurado los driver en 1/8 necesitaremos de 8 veces la cantidad de pasos para poder cubrir una distancia.

A4988			
	MS1 Jumper	MS2 Jumper	MS3 Jumper
Single Step	0	0	0
1/2 Step	1	0	0
1/4 Step	0	1	0
1/8 Step	1	1	0
1/16 Step	1	1	1

El driver GRBL necesita una configuración previa para funcionar y tener la mejor precisión posible, para esto se configurarán los parámetros del \$0 -\$22, cabe mencionar que algunos parámetros se colocaron de default.

Dependiendo del motor a pasos que se compró, se buscó el datasheet para visualizar los grados por paso que da el motor, este motor cuenta con 1.8°, pero necesitamos saber cuantos pasos requiere para dar una vuelta, por lo tanto, se tiene lo siguiente:

$$\frac{(un\ giro\ completo)360}{1.8} = Se\ requieren\ de\ 200\ pasos\ para\ una\ vuelta\ completa$$

Lo que sigue después es saber dependiendo de la varilla, cuantos milímetros se recorre por cada vuelta, por lo tanto, para las varillas que se compraron se tiene que por cada vuelta recorre 8 milímetros, con estos datos sabemos que necesitamos 200 pasos para cubrir una distancia pero al estar configurado en 1/8 esto quiere decir que necesitaremos 8 veces esa cantidad de pasos, por lo que se hace una multiplicación:

$$200\ pasos * 8 = 1600$$

Esta cantidad de pasos será dividida entre la cantidad de mm que tiene de avance por vuelta la varilla que será utilizada en el proyecto, en nuestro caso el avance es de 8mm por vuelta, por lo tanto:

$$\frac{1600}{8} = 200\ pasos.$$

De esta manera se colocan los parámetros del \$100 al \$101 que corresponden a los pasos por milímetros de los motores x, y ; para el caso del motor z el dato cambia pues en este se recorren 2 mm en lugar de 8 por lo tanto:

$$200\ pasos * 8 = 1600$$

$$\frac{1600}{2} = 800\ pasos.$$

Después configurar el ancho de pulso que manda el Arduino, se colocó en 10

```
$0=10 (step pulse, usec)
```

Se colocaron los valores de los pasos en los diferentes valores según los datos que se habían calculado

```
$100=200.000 (x, step/mm)
$101=200.000 (y, step/mm)
$102=800.000 (z, step/mm)
```

Los puntos del 110 al 112 por otro lado son los datos que nosotros indicamos como la velocidad máxima a la que nosotros configuramos a los motores para alcanzar como una condición fija.

```
$110=400.000 (x max rate, mm/min)
$111=400.000 (y max rate, mm/min)
$112=400.000 (z max rate, mm/min)
```

Los datos del 120 al 122 muestran los datos de aceleración que nosotros designamos a los motores y también el límite de aceleración que estos tienen. Se configuró a 999 para desactivar la aceleración.

```
$120=999.000 (x accel, mm/sec^2)
$121=999.000 (y accel, mm/sec^2)
$122=999.000 (z accel, mm/sec^2)
```

En el caso de los datos del 130 al 132 son los parametros que nosotros establecimos como la distancia máxima a la que el motor va a viajar es decir la delimitacion que nosotros le estamos otorgando para desplazar.

```
$130=200.000 (x max travel, mm)
$131=200.000 (y max travel, mm)
$132=200.000 (z max travel, mm)
```

La lista de configuraciones queda de la siguiente manera:

```
$0=10 (step pulse, usec)
$1=25 (step idle delay, msec)
$2=0 (step port invert mask:00000000)
$3=0 (dir port invert mask:00000000)
$4=0 (step enable invert, bool)
$5=0 (limit pins invert, bool)
$6=0 (probe pin invert, bool)
$10=3 (status report mask:00000011)
$11=0.010 (junction deviation, mm)
$12=0.002 (arc tolerance, mm)
$13=0 (report inches, bool)
$20=0 (soft limits, bool)
$21=0 (hard limits, bool)
$22=0 (homing cycle, bool)
$23=0 (homing dir invert mask:00000000)
$24=25.000 (homing feed, mm/min)
$25=500.000 (homing seek, mm/min)
$26=250 (homing debounce, msec)
$27=1.000 (homing pull-off, mm)
$100=200.000 (x, step/mm)
$101=200.000 (y, step/mm)
$102=800.000 (z, step/mm)
$110=400.000 (x max rate, mm/min)
$111=400.000 (y max rate, mm/min)
$112=400.000 (z max rate, mm/min)
$120=999.000 (x accel, mm/sec^2)
$121=999.000 (y accel, mm/sec^2)
$122=999.000 (z accel, mm/sec^2)
$130=200.000 (x max travel, mm)
$131=200.000 (y max travel, mm)
$132=200.000 (z max travel, mm)
```