

# High Torque GRBL Bundle



## Safety Statement

The author of this document is not liable or responsible for any accidents, injuries, equipment damage, property damage, loss of money or loss of time resulting from improper use of electrical or mechanical or software products.

Assembling electrical and mechanical CNC machine components like power supplies, motors, drivers or other electrical and mechanical components involves dealing with high voltage AC (alternating current) or DC (direct current) and other hazardous items which can be extremely dangerous and needs high attention to detail, experience, knowledge of software, electricity, electro-mechanics and mechanics.

**BEFORE MAKING ANY CONNECTIONS OR DISCONNECTIONS POWER MUST BE REMOVED FROM THE DEVICE AND THE CONTROLLER. FAILURE TO DO SO WILL VOID ANY AND ALL WARRANTIES.**

**Before starting please read through all the instructions.**

Note : Any Mains power connections must be installed by a Licensed electrician or suitability qualified person.

**Errors and omissions excepted**

# Introduction

The CNC Shield was designed by Protoneer.co.nz to take advantage of the demand for a low-cost controller solution for DIY CNC machines. It was designed to be 100% compatible with Grbl, the Opensource G-Code interpreter, and fit onto the popular Arduino Uno. The CNC Shield can be used to control a number of different types of CNC machines, including CNC milling machines, laser engraving/cutting machines, drawing machines, 3D printers or any project that needs precision control of stepper motors. It uses Pololu and compatible stepper drivers, either the A4988 or the higher current DRV8825.

There are 4 main components needed to get the CNC Shield up and running;

- CNC Shield,
- Stepper Drivers,
- Arduino UNO,
- Stepper Driver Adapter Module.

**Version 3.0 of the CNC Shield is used throughout this guide.**

The CNC Shield is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.

Please note: this document is a guide and not a manual. The CNC Shield and Grbl are Opensource and under constant development and modification. As a result, this document provides guidance but is by no mean comprehensive nor authoritative. It is essential that all users do their own research and find solutions that suit their application and requirements.

# Features

## Arduino UNO Compatible

- 14 digital input/output pins – 6 pins can be used as PWM outputs
- 6 analog inputs
- 16 MHz () crystal oscillator
- Operates at 5V
- Recommended input voltage range: 7V to 12V
- Flash Memory: 32 KB (0.5KB used by bootloader)
- SRAM: 2KB
- EEPROM: 1KB

## CNC Shield

- Version 3.00

- 4-Axis support (X, Y, Z , A-Can duplicate X,Y,Z or do a full 4th axis with custom firmware using pins D12 and D13)
- 2 x End stops for each axis (6 in total - each axis pair shared by same IO pin)
- Spindle enable and direction connection
- Coolant enable connection
- Uses GRBL as control software
- Power supply: DC 12-36V (only the DRV8825 drivers can handle up to 36V so if using A4988 do not exceed 24V)
- Uses removable stepper drivers (DRV8825 or A4988)
- Stepper Motors can be connected with 4 pin dupont/molex connectors
- Jumpers to set Micro-Stepping

### TB6600 Stepper Drivers

- Supports 8 levels of current control
- Supports 7 levels of micro step adjustment
- Interfaces adopt high-speed optocoupler isolation
- Automatic semi-flow to reduce heat
- Large area heat sink
- Anti-high-frequency interference ability
- Input anti-reverse protection
- Overheat, over current and short circuit protection

### Stepper Driver Adapter Module

- Allows the CNC shield to be used with High Current Drivers like the TB6600
- Simple plug and play replacement for A4988 and DRV8825

## Application

Suitable for a variety of small and medium sized automation equipment and instruments, such as: engraving machines, cutting machines, laser typesetting, plotters, drawbots, CNC machine tools, handling the devices. Capable of driving stepper with stepper motors with current requirement of up to 4A.

## Notes for Some Compatible UNOs

Some compatible UNOs uses the CH340 USB to serial chip. In order for your computer to recognize the UNO you may need to download the latest CH340 driver:

[http://www.wch.cn/download/CH341SER\\_ZIP.html](http://www.wch.cn/download/CH341SER_ZIP.html) ([http://www.wch.cn/download/CH341SER\\_ZIP.html](http://www.wch.cn/download/CH341SER_ZIP.html))

Read more here:

<http://www.arduined.eu/ch340-windows-8-driver-download/> (<http://www.arduined.eu/ch340-windows-8-driver-download/>)

<http://www.microcontrols.org/arduino-uno-clone-ch340-ch341-chipset-usb-drivers/> (<http://www.microcontrols.org/arduino-uno-clone-ch340-ch341-chipset-usb-drivers/>)

## Safety and Handling Requirements

**Before wiring it is essential to note a couple of safety issues and handling requirements.**

While the voltages on and around the CNC Shield are low (5V for the Arduino and up to 36V for the CNC Shield and steppers) it is still possible to hurt both yourself and the components if handled incorrectly or without care. **The following points are critical** and cannot be emphasized strongly enough. **Read carefully:**

- **NEVER** connect or disconnect any stepper motor to the CNC Shield while power is on or connected.
- **ALWAYS** disconnect the power before connecting or disconnecting the stepper motors.
- **ALWAYS** connect a stepper motor to the CNC Shield when testing or using the CNC Shield and driver. This is very important because the stepper drivers are designed to ramp up the current until it reaches the current needed to run. Without a stepper motor connected there will be nothing to consume the current and you can end up damaging the stepper driver if it over-heats in the process.

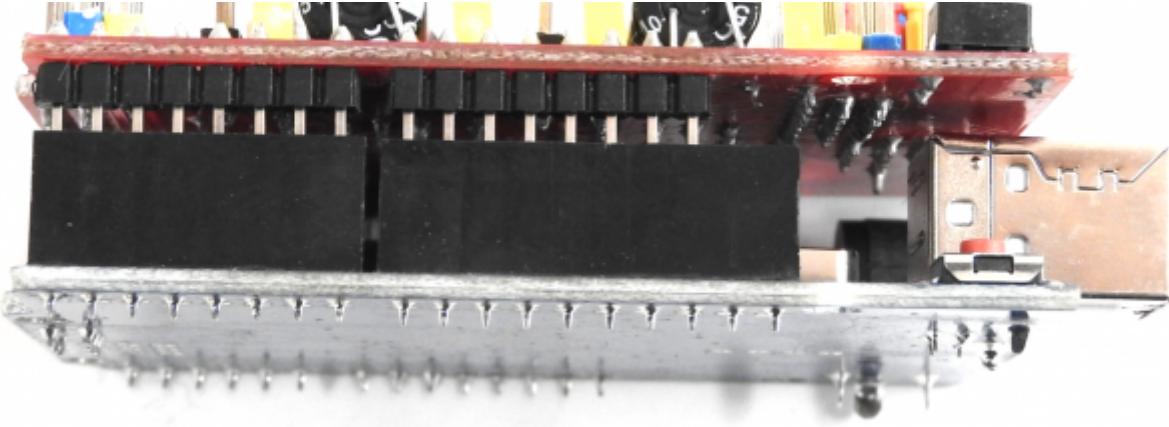
## Component Assembly

Below is a guide on how to put together the controller bundle.

## Arduino and CNC Shield connection

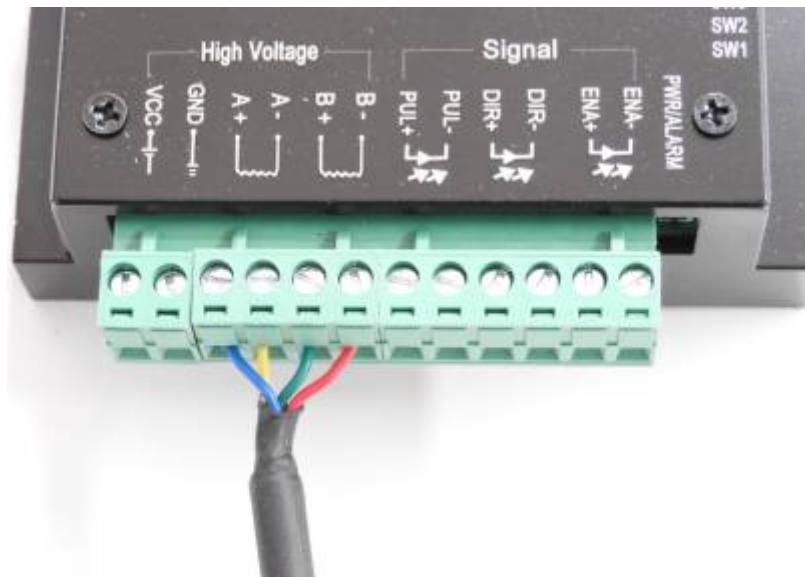
- Taking normal static electricity precautions, insert the CNC Shield into the Arduino Uno making sure the correct pins of the CNC Shield are inserted into the correct UNO headers.
- Press fit the CNC Shield on the arduino gently and carefully after lining the pins. Please refer to the image below on how to line up the pins. Please ensure that the black headers of the Arduino and the CNC shield make contact.





## Stepper Motor and Driver Wiring

- The wiring convention for the stepper motor is **Blue-Yellow-Green-Red** for the pins **A+, A-, B+, B-** respectively. Please use the appropriate cable for the axis you are wiring. Wire one end of the extension wire to the terminal block of the stepper motor driver and the other in the female EDG-4P connector.



- Please follow the same convention when wiring the stepper motors to the EDG connectors and match up the wire colors.





## Stepper Driver Adapter wiring

- There are 6 wires on the Stepper Driver Adapter module. These correspond to the terminal block on the TB6600 stepper motor driver.
- The table below shows the pinout of the Stepper Driver Adapter module that you need to follow to interface with the TB6600's pinout.

**Adapter Module TB6600 Terminal**

EN+	ENA+
EN-	ENA-
PUL+	PUL+
PUL-	PUL-
DIR+	DIR+
DIR-	DIR-

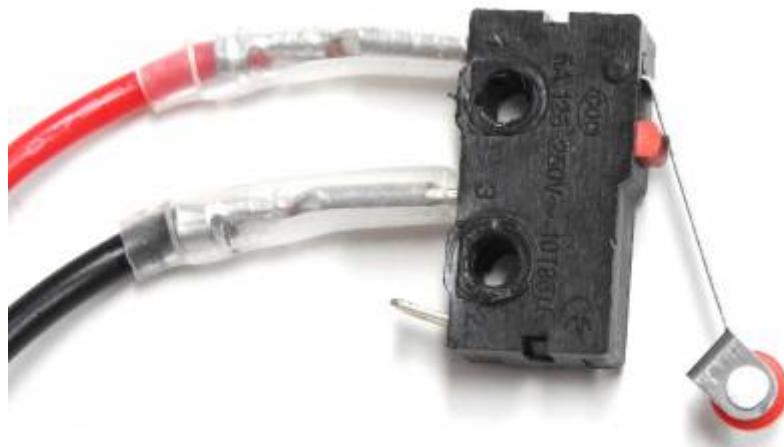
- Strip the ends of the adapter module's wiring harness and insert the pins accordingly as per the table above as shown in the image below



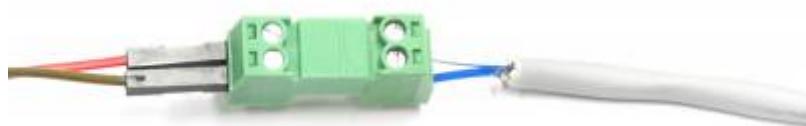


## Limit Switch Wiring

- The limit switch kits are used as hard limits for your machine. The limit switch kit includes a set of spade connectors which will be used to connect onto the terminals of the limit switch.
- Crimp the spade connectors onto the appropriate limit switch cable. Connect the red wire to the first terminal and the black wire to the middle terminal on the limit switch.
- The end result is shown in the image below.



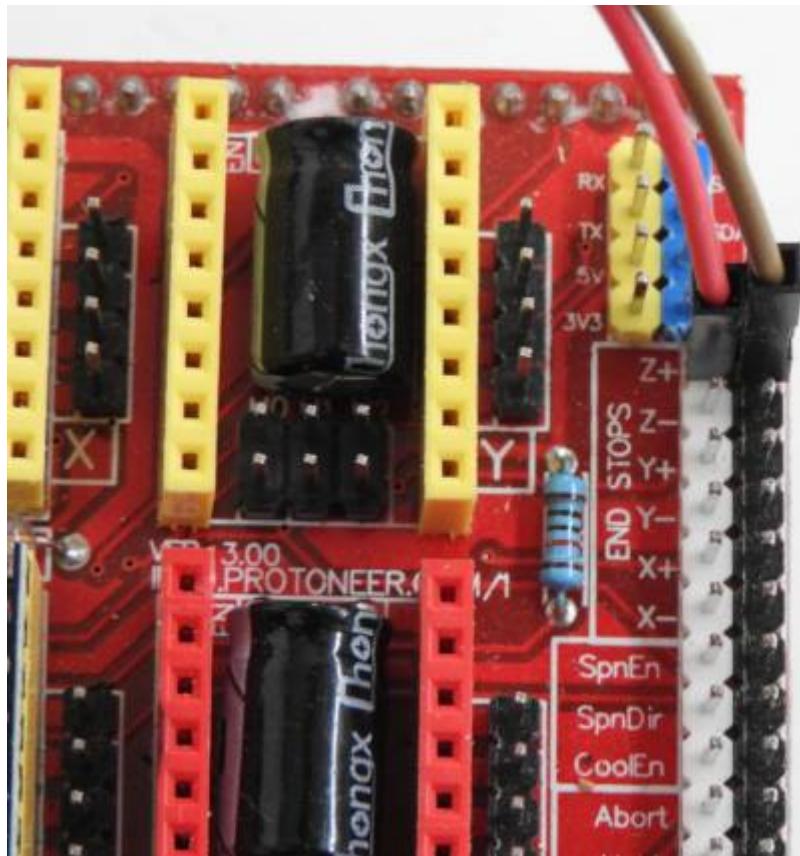
- Use the provided EDG-2P connector to terminate one end of the limit switch into the male end of the EDG connector and terminate the provided dupont cables to the female end as shown below. (Please note that the cable color may vary)



**Z limit switch wiring (Method 1).**

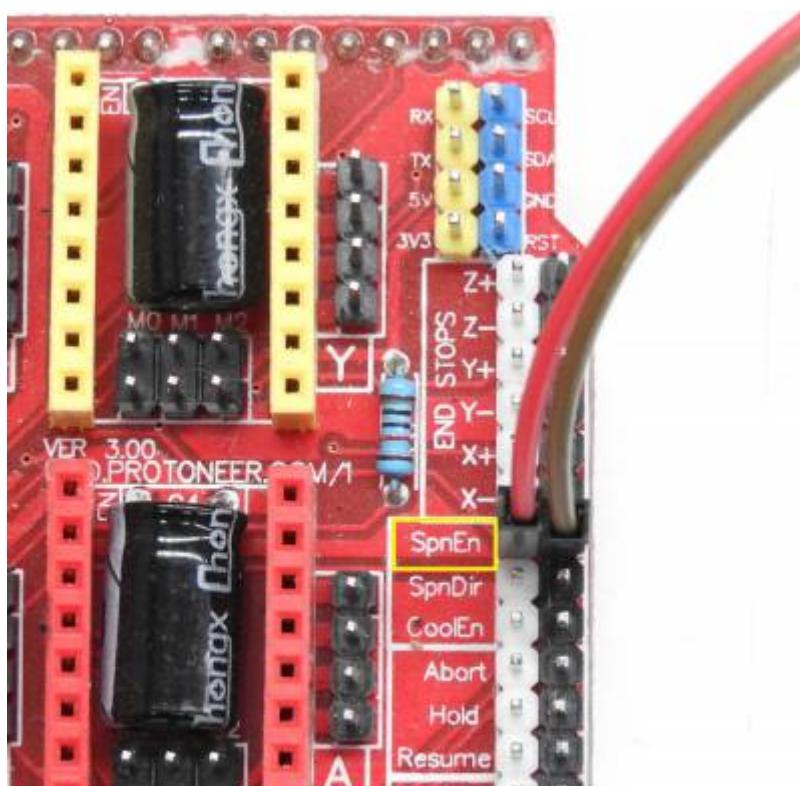
- The dupont cable will then plug into the CNC shield as shown below. Take note of the polarity, plug the **red** dupont cable into the '**White**' terminal and the **brown** dupont cable into the the '**Black**' terminal next to the Z+ row as shown in the image below. Save this step for the final assembly. Repeat this step for the X and Y axes, you will need to put the X axis limit switch in the X+ Row and Y axis limit switch in the Y+ row similar to what we did for the the Z axis limit switch above. **Please note**

**that you will need Version A of the GRBL 1.1 firmware (In the software and programming section) for this limit switch position to be recognised.** Version A of the GRBL 1.1 software has been modified to allow you to retain the functionality of the original location of the Z axis limit switch.



#### Z limit switch wiring (Method 2).

This wiring method uses Version B of the firmware. If you choose this firmware please put the Z limits in the SpnEn location as indicated below. This is because Version B of the firmware changes the pin layout slightly in the firmware.





## Emergency Stop Switch Wiring

Note : Any Mains power connections must be installed by a Licensed electrician or suitability qualified person. Different countries have different regulations that should be followed when doing any mains wiring. The emergency stop switch MUST be wired in by an electrician or similarly qualified individual.

- The emergency stop switch cuts off power to the power supply and therefore the stepper motor drivers. This is a must have safety feature should you require to stop your machine in as little time as possible during an emergency. Wiring this switch requires some disassembly. Open the switch housing by unscrewing the 4 screws with a Phillips head screwdriver to access the terminals
- Because this emergency stop is being wired on the mains voltage section of your power supply, a licensed electrician or similarly qualified individual is required to wire this step. The emergency stop switch must be placed in line with the live wire. Please make sure to drill holes for the Live cable to pass through the housing.

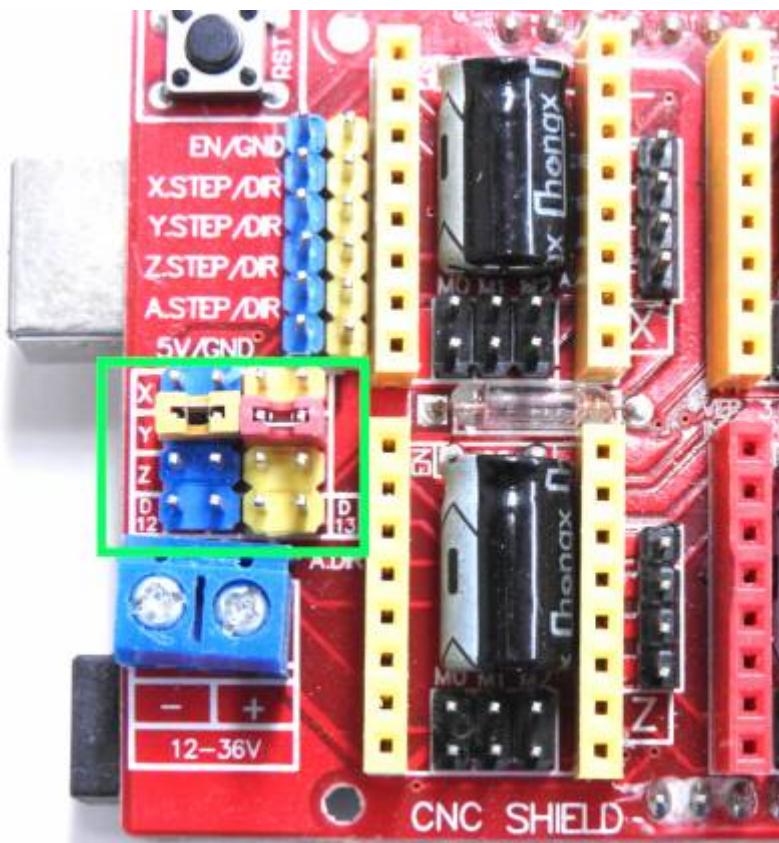
Please click the image to enlarge



## Cloning the Y-Axis.

- If you have a machine with dual Y-Axis actuators, you will need to clone the Y-axis so that the second

Y-axis gantry also known as the A-axis can move at the same rate and direction as the Y-axis. The Y axis can be cloned by putting the provided jumpers in across the 2 sets of headers of the Y-axis as highlighted in the Green square below. (Please note that the jumper colour may vary).



## Power Supply Wiring

Note : Any Mains power connections must be installed by a Licensed electrician or suitability qualified person. Different countries have different regulations that should be followed when doing any mains wiring. Note: All GRBL High Torque bundles Pre-October 1st 2020 use 12V Power supplies. Current bundles use 24V power supplies. Functionality is still the same.

- Each of your TB6600 Stepper motor drivers will need to be powered from an external source. The provided power supply is rated at the correct voltage. There are 2 power supplies that are compatible with this controller bundle. In this guide there is a 24V 350W Mean Well power supply. There is another variant of this kit that uses a 12V power supply. Wiring will still be the same. We highly recommend that all mains connection be terminated by a qualified electrician or a suitably qualified person.
- Use the included 2 core 3mm cable to wire your TB6600 stepper drivers to your power supply. The Red and Black wires correspond to the positive and negative terminals of your power supply and

motor drivers respectively, **Failure** to follow this convention **will cause damage** to your electrical components.

- As mentioned earlier we recommend that the mains connections be terminated by a qualified electrician or similar.
- Please note that the power lead from the wall socket to the power supply must be sourced separately.
- As we are using external drivers as opposed to the DRV8825/A4988 drivers, the CNC shield does not require any power input as the external drivers are externally powered.
- The Arduino Micro-Controller does not require any external power. Its power is provided via the USB connection with your computer.

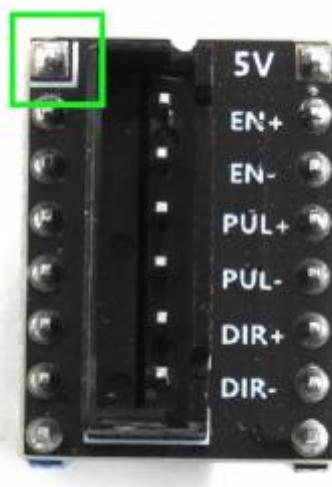
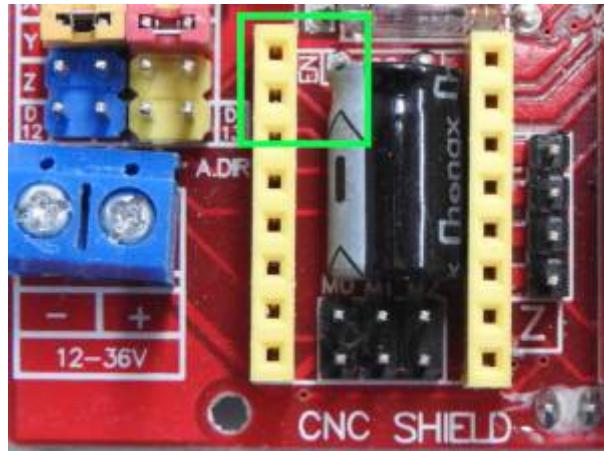


- Terminate the red and black cables to your Motor drivers. Please note the polarity

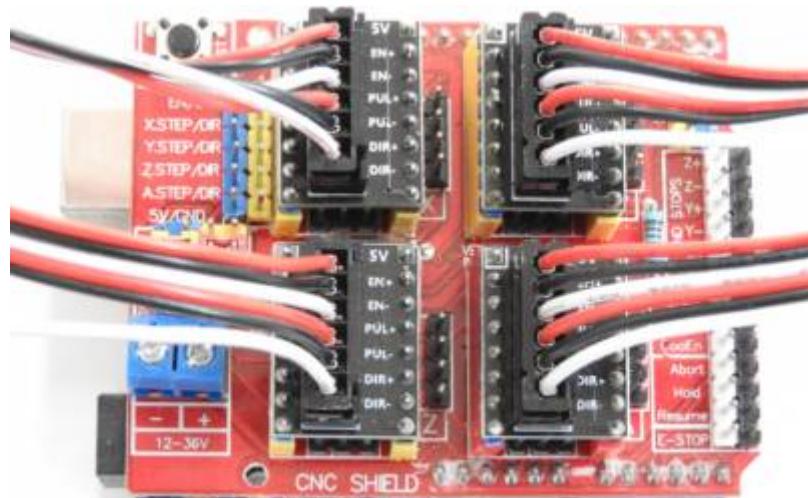


## Final Assembly

- Insert the adapter modules into the sockets while noting the correct orientation. On the adapter module is a square outlined in white on the top left corner. Line it up with the pin labelled **EN** on the CNC Shield. This is the correct orientation of the adapter module. Press fit gently. The images below can be used as a guide.



- Insert the adapter module wiring harness into the socket.



- Connect your limit switches and emergency stop to the appropriate headers.
- Connect the remaining connectors to the stepper motor drivers.

- Double check your connections.

## XYZ Probe Wiring

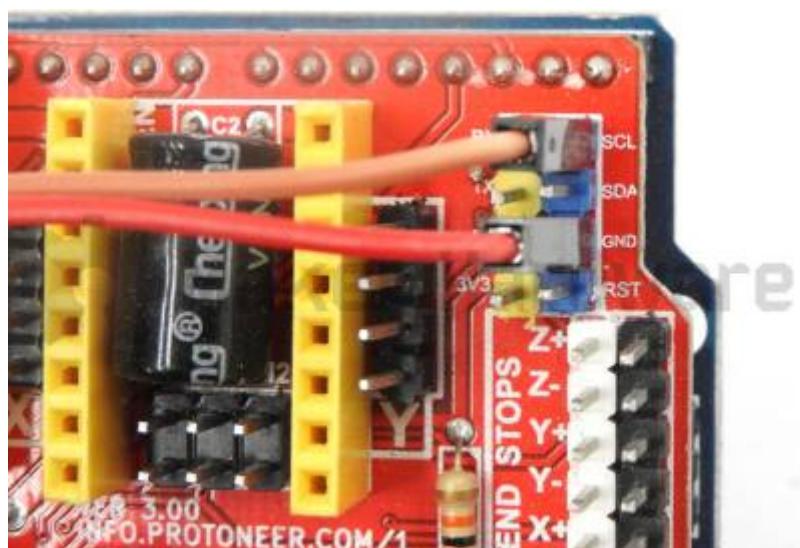
Connect the male part of the dupont jumper cable into the male half of the 2 pin EDG connector as shown.



Connect the red and black wires of the touch probe to the female half of the touch probe connector and plug it into the EDG connector with the dupont leads.



Finally, put the brown jumper in the SCL header and the red jumper in the GND header as shown below.



To configure the XYZ touch probe in software, follow the guide here. ([http://makerhardware.net/wiki/doku.php?id=add\\_on\\_packs:xyz\\_touch\\_probe](http://makerhardware.net/wiki/doku.php?id=add_on_packs:xyz_touch_probe))

# TB6600 Stepper Motor Driver Settings

Before activating your machine, you must configure your settings to ensure that your stepper motors are running in sync. The TB6600 has a series of dip switches that can be adjusted to run your motors to your required specification. This dip switch array is located on the front face of the driver.

The dip switch is considered **ON** when it is pointing down wards and **OFF** when pointing upwards. Dip switches 1-3 adjust the stepping rate of the motor while switches 4-6 adjust the current settings of the motor. Please see our recommended current settings for the 2 variants of the Nema23 stepper motors as well as the recommended stepping rate in the tables below.

Please refer to the highlighted row in the tables below

**Stepping Rate**

Micro Step	Pulse/Rev	S1	S2	S3
NC	NC	ON	ON	ON
1	200	ON	ON	OFF
2/A	400	ON	OFF	ON
2/B	400	ON	OFF	OFF
4	800	OFF	ON	ON
<b>8</b>	<b>1600</b>	<b>OFF</b>	<b>ON</b>	<b>OFF</b>
16	3200	OFF	OFF	ON
NC	NC	OFF	OFF	OFF

## Current Control Settings

**Standard Motor Current Settings**

If you have the standard Nema23 Stepper motor (1.26Nm), please follow the settings in the table below.

Current (A)	S4	S5	S6
0.5	ON	ON	ON
1	OFF	ON	ON
1.5	ON	OFF	ON
2	OFF	OFF	ON
2.5	ON	ON	OFF

Current (A)	S4	S5	S6
3	OFF	ON	OFF
3.5	ON	OFF	OFF
4	OFF	OFF	OFF

### High Torque Motor Current Control Settings

If you have the High Torque Stepper motor, please follow the settings in the table below:

Current (A)	S4	S5	S6
0.5	ON	ON	ON
1	OFF	ON	ON
1.5	ON	OFF	ON
2	OFF	OFF	ON
2.5	ON	ON	OFF
3	OFF	ON	OFF
<b>3.5</b>	<b>ON</b>	<b>OFF</b>	<b>OFF</b>
4	OFF	OFF	OFF

# Software and Programming

Your Arduino compatible microcontroller comes with the boot loader pre-flashed and is ready to be flashed with GRBL. Flashing GRBL on the Arduino is pretty straight forward. Please adhere to the following steps:

Download and install the Arduino IDE

- Download the Arduino IDE here ([https://www.arduino.cc/download\\_handler.php](https://www.arduino.cc/download_handler.php))
- Locate the downloaded file. By default it should be in your **Downloads** folder.
- Double click to run the installation process. Follow the prompts to install.

Download and install the GRBL libraries.

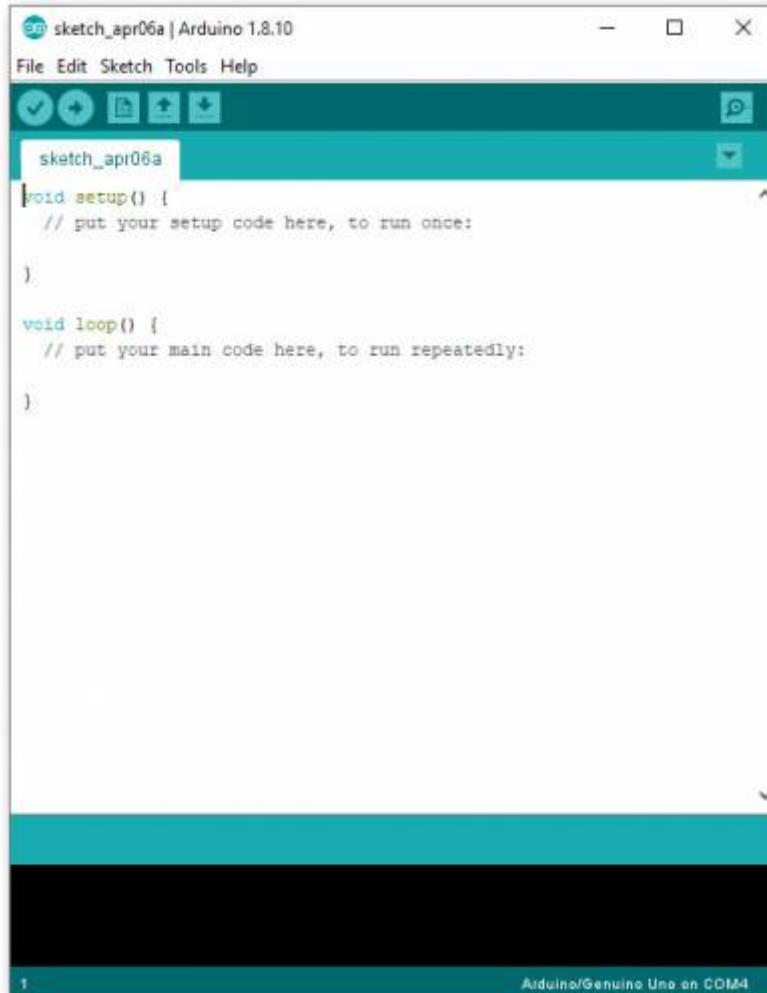
- Download GRBL V1.1 **Version A** here (<https://www.makerstore.com.au/download/software/grbl1.1h-V2.zip>)
- Note where the file has been downloaded and extract the .zip file titled **grbl**.
- After the extracting, copy the extracted folder to your Arduino Libraries folder. (By Default, your libraries folder should be in: **(C:\Program Files (x86)\Arduino\libraries)**)

Download GRBL V1.1 **Version B** here (<https://www.makerstore.com.au/download/software/grbl-1.1-master.zip>)

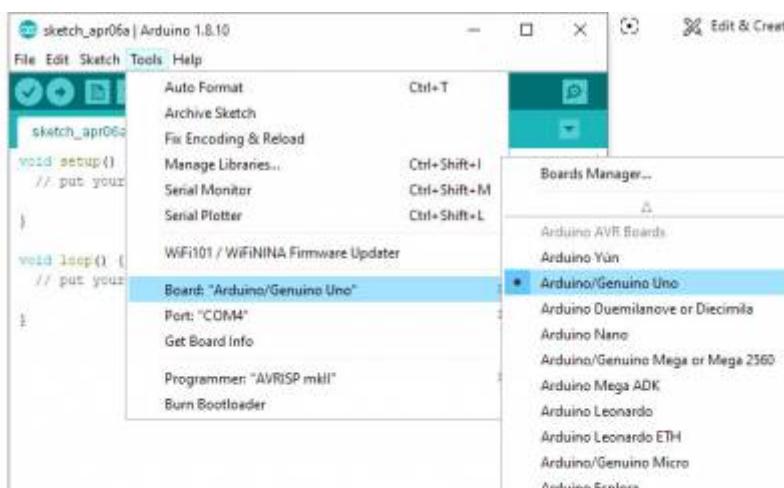
- Note where the file has been downloaded and extract the .zip file titled **grbl**.
- After the extracting, copy the extracted folder to your Arduino Libraries folder. (By Default, your libraries folder should be in: **(C:\Program Files (x86)\Arduino\libraries)**)

## Configuring your Arduino

- Before uploading any code, you must configure your board in 2 steps; Selecting a board and selecting the port that the board is connected to.
- Open the Arduino IDE and you will see a blank sketch such as below:

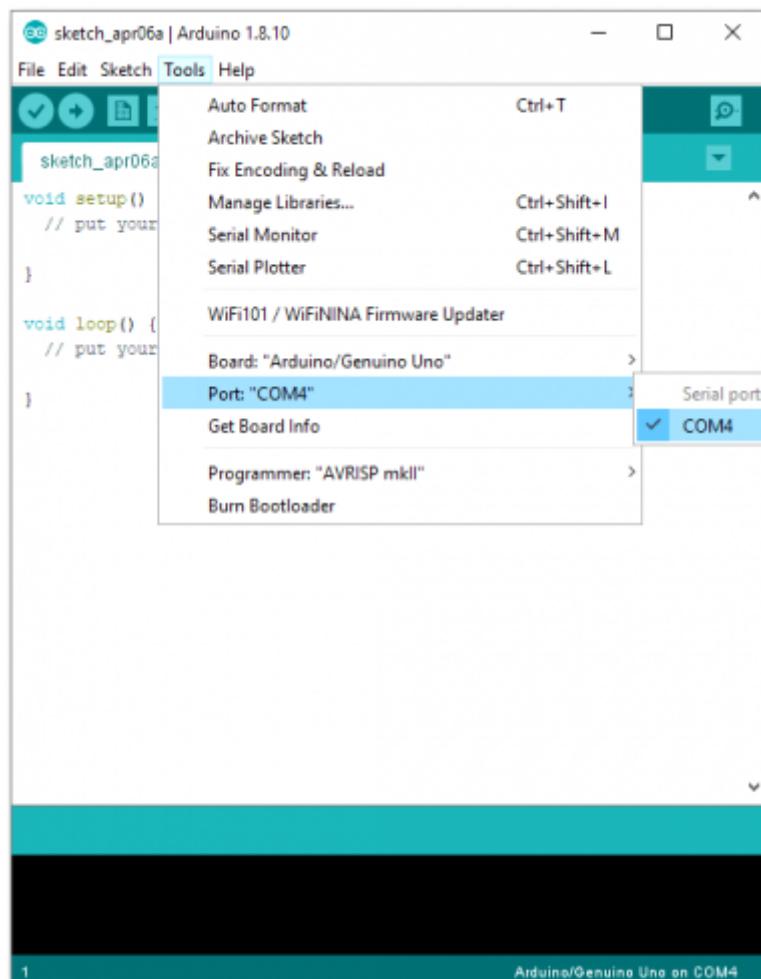


- To select your board, click on **Tools - Board - Arduino Uno/Genuino Uno**.





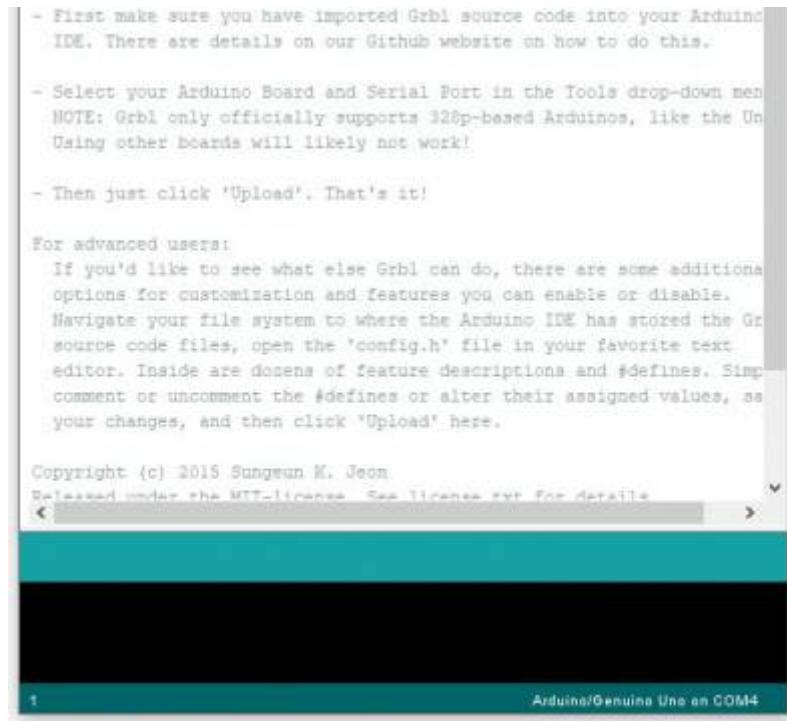
- To select the port, click on **Tools - Port** and choose that port that your board is connected to.



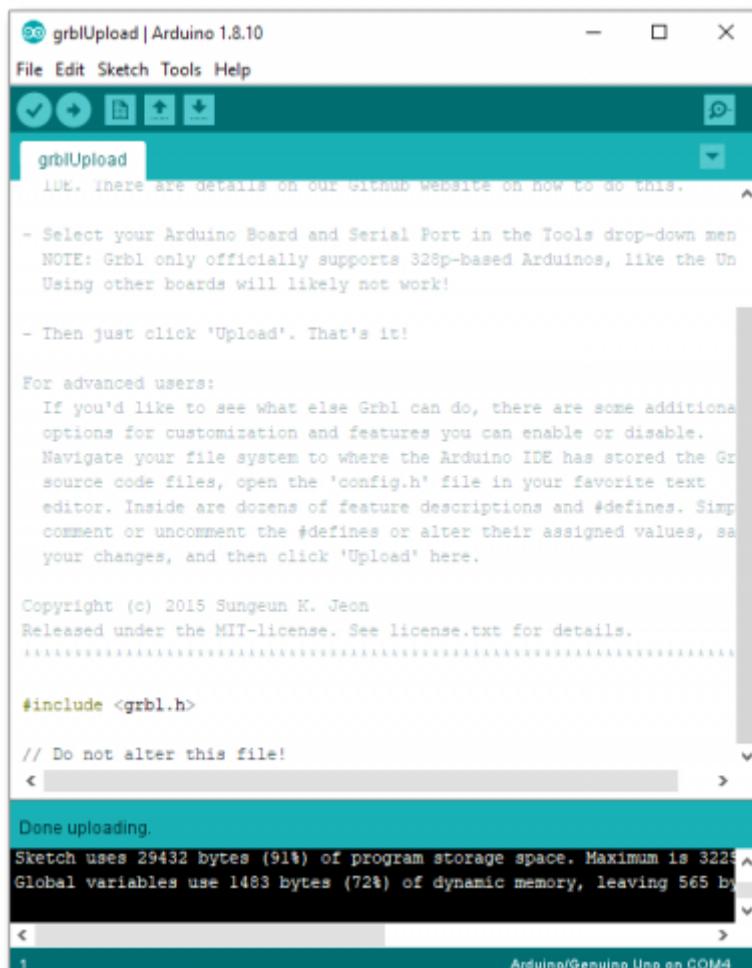
## Flashing GRBL

- To flash GRBL click on File- **Examples - grbl - grblUpload** and a grbl sketch will be shown. Upload the sketch to the arduino by clicking on the arrow button highlighted by the yellow square.





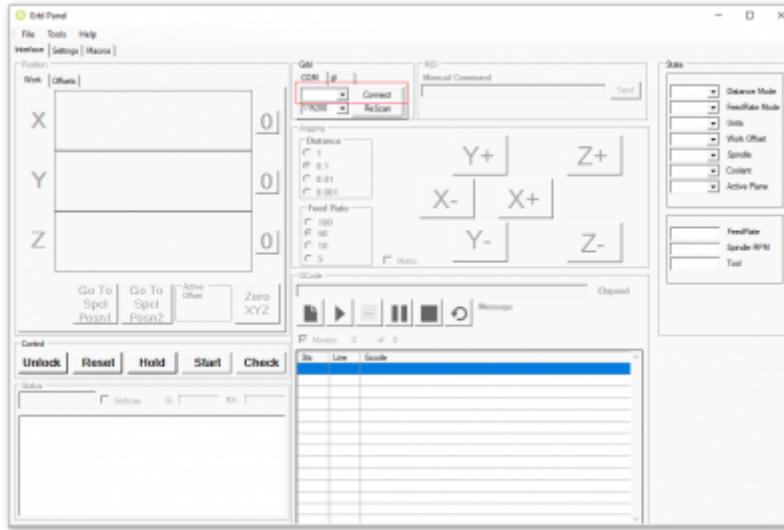
- Upon a successful upload, the IDE will report “Done Uploading.”



# Configuring GRBL Settings in GRBL PANEL

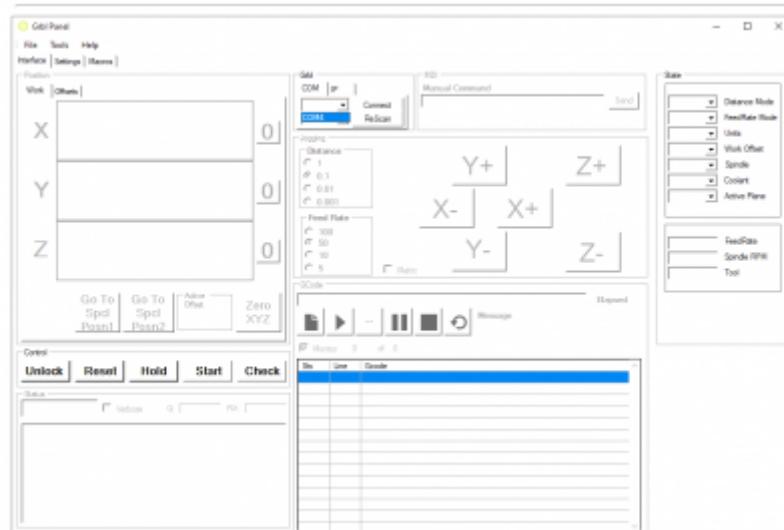
There are many G-Code senders that can be used. We recommend the GRBL Panel G-Code sender because of its ease of use and features..

- You can download GRBL Panel here (<https://github.com/gerritv/Grbl-Panel/releases/download/1.0.9.18/GrblPanel-Release-1.0.9.18A.zip>)
- Locate the downloaded file. By default, it should be in your Downloads folder.
- Unzip the downloaded file and copy the folder to your desktop.
- Open the folder and locate the file titled **GRBLPanel** and double click to open it. You will be greeted with the user interface as seen below. Note that there is no board connected as highlighted in the red square.



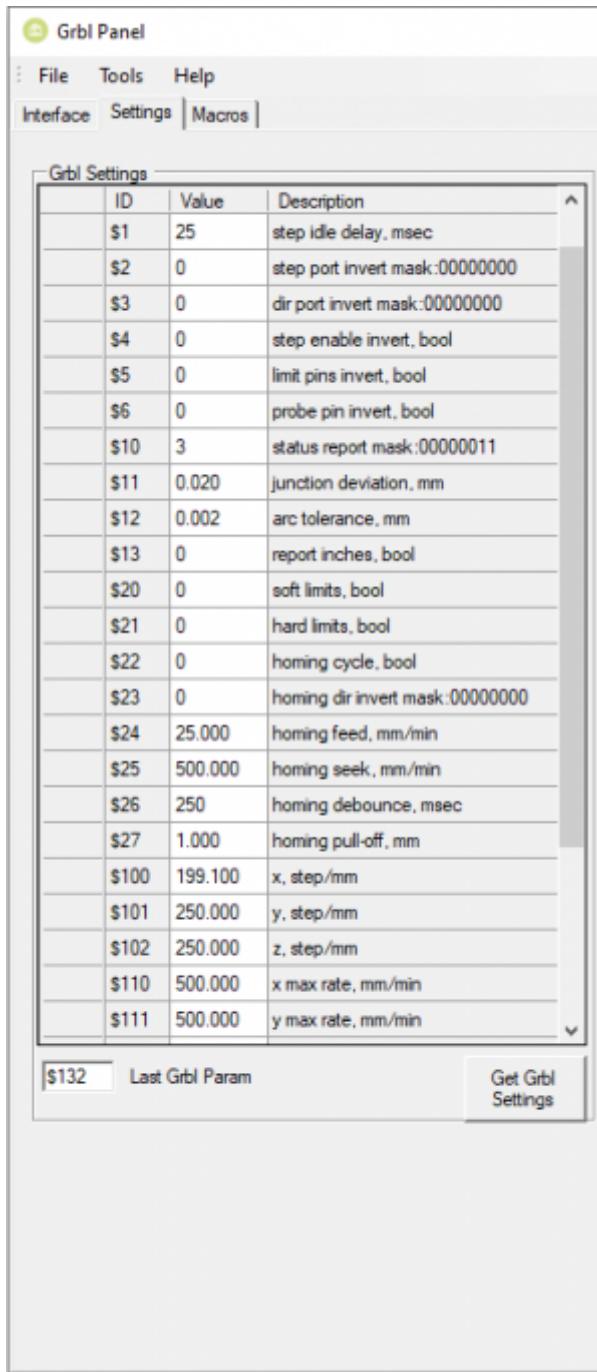
Click to expand image

- Connect the Arduino to the computer and press the **Rescan** button and select correct port in the drop down menu as indicated below. Click **Connect** and your arduino should be connected.



Click to expand image

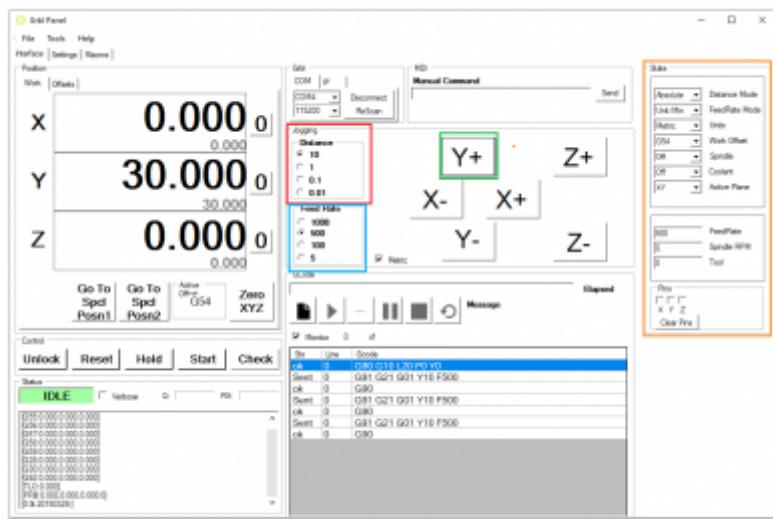
- Before jogging your machine, it is recommended to change the settings to ensure that your motors will run in sync and therefore accurately. You can configure the settings of your machine by accessing the **Settings** tab and changing the values in the **Value** column. Press the **Enter** key to save the changes for each value.



Click to expand image

## Configuring and tuning your machine

- After changing the settings to suit your machine, manually jog your machine and check to see if the distance traveled in the software matches the distance physically traveled by the machine. The image below shows the different sections of the panel that will need to be adjusted to your preference.



Click to expand image

- The **Red** square sets the stepping distance from 0.01mm to 10mm
- The Feed rate shown in the **Blue** square but can be overridden by the Feed rate in the **Orange** square
- The Orange square has settings for milling.
- As seen in the image, the Y axis shown in **Green** has been moved in the positive direction by 30mm. Your machine should move to 30mm. If it does not, please change the appropriate settings \$101 which is the number of steps per mm traveled by the motor to suit your machine.
- Repeat for all your axes until the desired accuracy has been reached.
- More information on using the GRBL Panel can be found here: <https://github.com/gerrity/Grbl-Panel/wiki/Using-Grbl-Panel> (<https://github.com/gerrity/Grbl-Panel/wiki/Using-Grbl-Panel>)

## Recommended GRBL Settings

Below are the recommended GRBL settings. The highlighted settings must be modified to suit your machine.

### Recommended Lead Screw Driven Machine Settings

This set of settings are for the lead screw driven Lead CNC 1000 x 1000mm. Please adjust the work area of your lead screw driven machine accordingly.

Setting	Parameter	Function
\$0	10	step pulse, usec
\$1	255	step idle delay, msec
\$2	0	step port invert mask:00000000
\$3	0	dir port invert mask:00000000

Setting	Parameter	Function
\$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	0.02	junction deviation, mm
\$12	0	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	100	homing feed, mm/min
\$25	1000	homing seek, mm/min
\$26	250	homing debounce, msec
\$27	5	homing pull-off, mm
\$100	200	x, step/mm
\$101	200	y, step/mm
\$102	200	z, step/mm
\$110	5000	x max rate, mm/min
\$111	5000	y max rate, mm/min
\$112	2500	z max rate, mm/min
\$120	150	x accel, mm/sec
\$121	150	y accel, mm/sec
\$122	150	z accel, mm/sec
<b>\$130</b>	<b>824</b>	<b>x max travel, mm</b>
<b>\$131</b>	<b>780</b>	<b>y max travel, mm</b>
<b>\$132</b>	<b>50</b>	<b>z max travel, mm</b>

## Recommended Belt Driven Machine Settings

Below are the recommended settings for an Ox 1000 x 1500mm machine. Please adjust your X, Y and Z max travel settings to suit your machine accordingly.

Setting	Parameter	Function
\$0	10	step pulse, usec
\$1	255	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	0.02	junction deviation, mm
\$12	0	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	100	homing feed, mm/min
\$25	1000	homing seek, mm/min
\$26	250	homing debounce, msec
\$27	5	homing pull-off, mm
\$100	26.670	x, step/mm
\$101	26.670	y, step/mm
\$102	200	z, step/mm
\$110	5000	x max rate, mm/min
\$111	5000	y max rate, mm/min
\$112	2500	z max rate, mm/min
\$120	150	x accel, mm/sec
\$121	150	y accel, mm/sec
\$122	150	z accel, mm/sec
<b>\$130</b>	<b>820</b>	<b>x max travel, mm</b>
<b>\$131</b>	<b>1275</b>	<b>y max travel, mm</b>
<b>\$132</b>	<b>57</b>	<b>z max travel, mm</b>

## Recommended Gear Rack Driven Machine Settings

Below are the recommended settings for an Ox 1000 x 1500mm machine with our gear rack transmission in the Y axis. Adjust your X, Y and Z max travel settings to suit your machine accordingly.

Setting	Parameter	Function
\$0	10	step pulse, usec
\$1	255	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	0.02	junction deviation, mm
\$12	0	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	100	homing feed, mm/min
\$25	1000	homing seek, mm/min
\$26	250	homing debounce, msec
\$27	5	homing pull-off, mm
\$100	26.670	x, step/mm
\$101	31.900	y, step/mm
\$102	200	z, step/mm
\$110	5000	x max rate, mm/min
\$111	5000	y max rate, mm/min
\$112	2500	z max rate, mm/min
\$120	150	x accel, mm/sec
\$121	150	y accel, mm/sec
\$122	150	z accel, mm/sec

Setting	Parameter	Function
\$130	820	x max travel, mm
\$131	1275	y max travel, mm
\$132	57	z max travel, mm

## Links and Credits

Designer of the CNC Shield:	<a href="https://blog.protoneer.co.nz/arduino-cnc-shield/">https://blog.protoneer.co.nz/arduino-cnc-shield/</a> ( <a href="https://blog.protoneer.co.nz/arduino-cnc-shield/">https://blog.protoneer.co.nz/arduino-cnc-shield/</a> )
Grbl to v0.9:	<a href="https://github.com/grbl/grbl/wiki">https://github.com/grbl/grbl/wiki</a> ( <a href="https://github.com/grbl/grbl/wiki">https://github.com/grbl/grbl/wiki</a> )
Grbl from v1.1:	<a href="https://github.com/gnea/grbl/wiki">https://github.com/gnea/grbl/wiki</a> ( <a href="https://github.com/gnea/grbl/wiki">https://github.com/gnea/grbl/wiki</a> )
Grbl GUIs:	<a href="https://github.com/gnea/grbl/wiki/Using-Grbl">https://github.com/gnea/grbl/wiki/Using-Grbl</a> ( <a href="https://github.com/gnea/grbl/wiki/Using-Grbl">https://github.com/gnea/grbl/wiki/Using-Grbl</a> )
Driver Details:	<a href="https://www.makerstore.com.au/product/stepper-driver-tb6600-4a-40v/">https://www.makerstore.com.au/product/stepper-driver-tb6600-4a-40v/</a> ( <a href="https://www.makerstore.com.au/product/stepper-driver-tb6600-4a-40v/">https://www.makerstore.com.au/product/stepper-driver-tb6600-4a-40v/</a> )
Wikipedia Arduino entry:	<a href="https://en.wikipedia.org/wiki/Arduino">https://en.wikipedia.org/wiki/Arduino</a> ( <a href="https://en.wikipedia.org/wiki/Arduino">https://en.wikipedia.org/wiki/Arduino</a> )
Arduino IDE:	<a href="https://www.arduino.cc/en/Main/Software">https://www.arduino.cc/en/Main/Software</a> ( <a href="https://www.arduino.cc/en/Main/Software">https://www.arduino.cc/en/Main/Software</a> )

The High Torque GRBL Control System was designed and published by Tawanda under the Open Source Licence by Maker Store.

electronics/grbl\_high\_torque\_bundle.txt Last modified: 2021/08/26 23:27  
by Maker Hardware Team (/wiki/doku.php?id=user:sab)

## Maker Hardware

By Makers - For Makers



Except where otherwise noted, content on this wiki is licensed under the following license:  
CC Attribution-Noncommercial-Share Alike 4.0 International