

GRBL Konfigürasyonu

İÇİNDEKİLER

İÇİNDEKİLER	i
ÖNSÖZ	1
BAŞLARKEN	1
GRBL AYARLARI	1
\$ - GRBL Ayarlarını Görüntüleme	1
\$x=val - GRBL Ayarını Kaydetme	2
GRBL Ayarları Ve Anlamları	3
\$0 - Step Pulse, microsecond	3
\$1 - Step Idle Delay, milisecond	3
\$2 - Step Port Invert, Mask.....	4
\$3 - Direction Port Invert, Mask	4
\$4 - Step Enable Invert, Boolean	5
\$5 - Limit Pins Invert, Boolean.....	5
\$6 - Probe Pin Invert, Boolean.....	5
\$10 - Status Report, Mask	5
\$11 - Junction Deviation, mm.....	6
\$12 - Arc Tolerance, mm	7
\$13 - Report Inches, Boolean	7
\$20 - Soft Limits, Boolean.....	7
\$21 - Hard Limits, Boolean	8
\$22 - Homing Cycle, Boolean.....	8
\$23 - Homing Dir Invert, Mask	9
\$24 - Homing Feed, mm/min	10
\$25 - Homing Seek, mm/min.....	10
\$26 - Homing Debounce, milisecond.....	10
\$27 - Homing Pull-off, mm	10
\$30 - Max Spindle Speed, RPM.....	10
\$31 - Min Spindle Speed, RPM	11
\$32 - Laser Mode, Boolean.....	11
\$100, \$101, \$102 - [X,Y,Z] Steps/mm	11
\$110, \$111, \$112 - [X,Y,Z] Max Rate, mm/min	12
\$120, \$121, \$122 - [X,Y,Z] Acceleration, mm/sn ²	12
\$130, \$131, \$132 - [X,Y,Z] Max Travel, mm.....	13
MAKİNENİZİ İLK KEZ KURMAK İÇİN HIZLI KILAVUZ	13

ÖNSÖZ¹

Bu metin, dipnotta belirtilen linkte yer alan İngilizce metnin Google Translate aracıyla üretilen çeviri metinden faydalanılarak ve görülen eksikleri giderilmeye çalışılarak Türkçe'ye aktarılmıştır. "Başlarken" kısmından itibaren tamamen çeviri metin yer almaktadır.

Buradaki amaç, hobi amaçlı ve giriş seviyesi CNC kullanıcılarının gerçekleştirmiş oldukları DIY-CNC projelerindeki en temel bileşenlerden GRBL Firmware'in 1.1 versiyonunun konfigürasyonuna dair en detaylı kaynağın Türk kullanıcılar tarafından kullanılabilir olmasını sağlamaktır. Teknik olarak bu metne yansıyan eksikliklerin, başka katkılarla giderileceği temennisiyle.

Hafy

İstanbul-2022

¹ Link: <https://github.com/gnea/grbl/wiki/Grbl-v1.1-Configuration#20---soft-limits-boolean>

BAŞLARKEN

İlk olarak, seçtiğiniz seri terminali kullanarak GRBL'a bağlanın.

Baud Rate'i, 8-N-1 (8 bit, no parity ve 1-stop bit) olarak **115200'e ayarlayın**.

Bağlandıktan sonra, şöyle bir GRBL komutu almalısınız: **GRBL 1.1f ['\$' for help]**

GRBL'nin bir yardım mesajı yazdırması için \$ yazın ve enter tuşuna basın. \$ ve enter'in herhangi bir yerel yankısını görmemelisiniz. GRBL şu yanıtı verecektir:

[HLP:\$\$ \$# \$G \$I \$N \$x=val \$Nx=line \$J=line \$SLP \$C \$X \$H ~ ! ? ctrl-x]

'\$' ile başlayan komutlar, GRBL ayarları değiştirmek, GRBL durum ve çalışma modlarını görüntüleyip/değiştirmek ve Homing Cycle başlatmak için kullanılan sistem komutlarıdır. '\$' bulunmayan son dört komut, GRBL'nin herhangi bir durumunda gönderilebilen gerçek zamanlı kontrol komutlarıdır. Bunlar ya GRBL'in çalışma davranışını hemen değiştirir ya da mevcut konum (diğer adıyla DRO) gibi önemli gerçek zamanlı verilerin raporunu hemen yazdırır.

GRBL AYARLARI

\$\$ - GRBL Ayarlarını Görüntüleme

Ayarları görüntülemek için GRBL'ye bağlandıktan sonra \$\$ yazın ve enter tuşuna basın. GRBL yanıt olarak, aşağıdaki örnekte gösterildiği gibi mevcut sistem ayarlarının bir listesiyle yanıt vermelidir. Tüm bu ayarlar kalıcıdır ve EEPROM'da tutulur, bu nedenle gücü kapatırsanız, Arduino'nuzu bir sonraki açışınızda bunlar tekrar yüklenecektir.

\$x=val ifadesindeki x, düzenlenecek ayarın kimliğini belirtir. GRBL'in önceki sürümlerinde, her ayarın yanındaki parantezler içinde yer alan açıklamalar GRBL v1.1+'da bulunmuyor. Bu, v1.1'de bulunan yeni özellikler için flash memory'de ihtiyaç duyulan alanı elde etmek üzere yapıldı. Pek çok iyi GUI, sizin için bu açıklamaları ekleyerek size yardımcı olacaktır, böylece neye baktığınızı bilebilirsiniz.

Ayarlar ve Örnek Değerler	Tanım
\$0=10	Step Pulse, microsecond
\$1=25	Step Idle Delay, milisecond
\$2=0	Step Port Invert, Mask
\$3=0	Direction Port Invert, Mask
\$4=0	Step Enable Invert, Boolean
\$5=0	Limit Pins Invert, Boolean

Ayarlar ve Örnek Değerler	Tanım
\$6=0	Probe Pin Invert, Boolean
\$10=1	Status Report, Mask
\$11=0.010	Junction Deviation, mm
\$12=0,002	Arc Tolerance, mm
\$13=0	Report Inches, Boolean
\$20=0	Soft Limits, Boolean
\$21=0	Hard Limits, Boolean
\$22=1	Homing Cycle, Boolean
\$23=0	Homing dir Invert, Mask
\$24=25.000	Homing Feed, mm/min
\$25=500.000	Homing Seek, mm/ min
\$26=250	Homing Debounce, milisecond
\$27=1.000	Homing Pull-off, mm
\$30=1000.	Max Spindle Speed, RPM
\$31=0.	Min Spindle Speed iş mili hızı, RPM
\$32=0	Laser Mode, Boolean
\$100=250.000	X Steps/mm
\$101=250.000	Y Steps /mm
\$102=250.000	Z Steps /mm
\$110=500.000	X Max Rate, mm/dak
\$111=500.000	Y Max Rate, mm/dak
\$112=500.000	Z Max Rate, mm/dak
\$120=10.000	X Acceleration, mm/ sn ²
\$121=10.000	Y Acceleration, mm/ sn ²
\$122=10.000	Z Acceleration, mm/sn ²
\$130=200.000	X Max Travel, mm
\$131=200.000	Y Max Travel, mm
\$132=200.000	Z Max Travel, mm

\$x=val - GRBL Ayarını Kaydetme

Komut, GRBL'ye **\$x=val** bir seri terminal programı aracılığıyla bağlandığında bu komutu göndererek manuel olarak yapılabilen bir GRBL ayarını kaydeder veya değiştirir, ancak çoğu GRBL GUI'si bunu sizin için kullanıcı dostu bir özellik olarak yapacaktır.

Örneğin, mikro saniye Step Pulse seçeneğini 10us şeklinde manuel olarak değiştirmek için şunu yazmanız ve ardından bir enter girmeniz gerekir: **\$0=10**

Her şey yolunda giderse, GRBL bir 'OK' ile yanıt verir ve bu ayar EEPROM'da saklanır ve sonsuza kadar veya siz onları değiştirene kadar korunur.

Sistem ayarlarını tekrar görüntülemek için \$\$ yazıp enter'e basarak GRBL'nin ayarınızı ne şekilde depolayıp kaydettiğini kontrol edebilirsiniz.

Bazı ayarların (**100\$, 101\$, 102\$** gibi) ayarların düzgün şekilde yüklenmesi için yazılımdan sıfırlama gerektirdiğini unutmayın. Aksi takdirde sonraki herhangi bir hareket komutu, makineyi yanlış koordinatlara hareket ettirerek makineye zarar verebilir.

GRBL Ayarları Ve Anlamları

NOT: GRBL v0.9'dan farklı olarak GRBL v1.1'de, yalnızca **\$10** Status Reports değişti ve yeni olarak **\$30/\$31** Spindle Speed RPM max/min ve **\$32** Laser Mode ayarları eklendi. Geriye kalan her şey aynı.

\$0 - Step Pulse, microsecond

Stepper sürücüler, belirli bir minimum adım için gerekli darbe miktarına göre ayarlanır. Datasheet'i kontrol edin veya sadece birkaç sayı deneyin. Stepper sürücülerin güvenilir bir şekilde tanıyabileceği en kısa pulse'ı istersiniz. Pulse'lar çok uzun süreli ise, sistem çok yüksek feed ve pulse rate ile çalışırken, pulse süreleri uzun olduğu için step pulse'leri üst üste gelmeye başlayabilir ve sorun yaşayabilirsiniz. Varsayılan olarak 10 mikro saniye civarında bir değer öneriyoruz.

\$1 - Step Idle Delay, milisecond

Stepper motorlarınız bir hareketi tamamladığında, GRBL onları devre dışı bırakmayı bu değer süresince geciktirecektir. **VEYA**, bu değeri maksimum 255 milisaniyeye ayarlayarak eksenlerinizi her zaman (konumlarını güçlüce tutacak şekilde) etkin tutabilirsiniz. **\$1=255** koduyla, sadece tekrarlamak için ayarlayarak tüm eksenleri sürekli etkin tutabilirsiniz.

Stepper Idle Lock time, GRBL'nin stepper motorları devre dışı bırakmadan önce kilitli tutacağı süredir. Sisteme bağlı olarak, bu süreyi sıfıra ayarlayarak devre dışı bırakabilirsiniz. Diğer durumlarda, stepper motorları devre dışı bırakmadan önce eksen hareketlerinin tamamen durduğundan emin olmak için 25-50 milisaniyeye ihtiyacınız olabilir. Bu, uzun periyotlarla boştaki açık bırakılamayacak hassasiyetteki makine motorlarının gözetilmesinde yardımcı olacaktır. Bir de bazı stepper sürücüler stepper motorların hangi mikro adımda durduklarını hatırlamazlar. Bu nedenle onları yeniden

etkinleştirdiğinizde, bazı 'kayıp' adımlarla karşılaşabilirsiniz. Bu durumla karşılaşmamak için stepper motorlarınızı **\$1=255** ayarıyla sürekli etkin tutabilirsiniz.

\$2 - Step Port Invert, Mask

Bu ayar Step Pulse sinyalini tersine çevirir. Varsayılan olarak, bir step sinyali normal-LOW seviyesinde başlar ve HIGH durumuna geçmesiyle bir Step Pulse olayı gerçekleşir. **\$0** değeriyle ayarlanan bir Step Pulse süresinden sonra, Pulse Pin'i bir sonraki Step Pulse olayına kadar LOW durumuna sıfırlanır. Tersine çevrildiğinde, Step Pulse davranışı başlangıcı normal-HIGH olarak ve LOW'a geçerken Step Pulse olayı tetiklenip yeniden HIGH konumuna sıfırlanacak şekilde ayarlanmış olur. Çoğu kullanıcının bu ayarı kullanması gerekmeyecektir, ancak özel gereksinimleri olan belirli CNC Stepper Sürücüler için yararlı olabilir. Örneğin, Step Pin'i ters çevrilerek Direction Pin'i ile Step Pulse arasında yapay bir gecikme oluşturulabilir.

Bu Step Port Invert ayarı, ters çevrilecek eksenleri bit bayrakları olarak saklayan bir değerdir. Gerçekten nasıl çalıştığını tam olarak anlamanız gerekmez. Tersine çevirmek istediğiniz eksenler için ayar değerini girmeniz yeterlidir. Örneğin, X ve Z eksenlerini ters çevirmek istiyorsanız, **\$2=5** komutunu GRBL'ye gönderdiğinizde ayarlanmış değer şu şekilde olmalıdır:

\$2=5 (step port invert mask:00000101)

Değer	Mask	X'i Ters Çevir	Y'yi Ters Çevir	Z'yi Ters Çevir
0	00000000	N	N	N
1	00000001	Y	N	N
2	00000010	N	Y	N
3	00000011	Y	Y	N
4	00000100	N	N	Y
5	00000101	Y	N	Y
6	00000110	N	Y	Y
7	00000111	Y	Y	Y

\$3 - Direction Port Invert, Mask

Bu ayar, her eksen için yön sinyalini tersine çevirir. GRBL varsayılan olarak, Direction Pin sinyali LOW iken eksenlerin pozitif yönde ve Pin HIGH iken negatif yönde hareket ettiğini varsayar. Eksenler bazı makinelerde bu şekilde hareket etmez. Bu ayar, ters yönde hareket eden eksenler için Direction Pin sinyalini tersine çevirecektir.

Bu ters çevirme maskesi ayarı, tam olarak Step Port Invert Mask gibi çalışır ve hangi eksenin ters çevrileceğini bit bayrakları olarak saklar. Bu ayarı yapılandırmak için, tersine çevirmek istediğiniz eksenlerin değerini göndermeniz yeterlidir. Bunun için de yukarıdaki tablo kullanılabilir. Örneğin, yalnızca Y eksenini yönünü tersine çevirmek istiyorsanız, GRBL'ye **\$3=2** komutu verildiğinde şu çıktı okunmalıdır:

\$3=2 (dir port invert mask:00000010)

\$4 - Step Enable Invert, Boolean

Stepper Enable Pin'i, varsayılan ayarda devre dışı bırakmak için HIGH ve etkinleştirmek için LOW'dur. Kurulumunuzun tersi gerekiyorsa, **\$4=1** komutunu yazarak Step Enable Pin'ini ters çevirebilirsiniz. **\$4=0** ile devre dışı bırakın. (Değişikliği yüklemek için bir güç döngüsü gerekebilir.)

\$5 - Limit Pins Invert, Boolean

Limit Pinleri, varsayılan ayarda, Arduino'nun dahili pull-up direnci ile normal-HIGH tutulur. Bir limit pini LOW durumuna geçtiğinde, GRBL bunu tetiklenmiş olarak yorumlar. Ters davranış için, **\$5=1** komutuyla limit pinlerinin çalışmasını ters çevirmeniz yeterlidir. **\$5=0** ile devre dışı bırakabilirsiniz. Değişikliği yüklemek için bir güç döngüsüne ihtiyacınız olabilir.

NOT: Daha gelişmiş kullanım için, limit pinlerindeki dahili pull-up direnci config.h'de devre dışı bırakılabilir.

\$6 - Probe Pin Invert, Boolean

Probe Pini, varsayılan olarak, Arduino'nun dahili pull-up direnci ile normal-HIGH tutulur. Probe Pini LOW durumuna geçtiğinde, GRBL bunu tetiklenmiş olarak yorumlar. Ters davranış için, **\$6=1** komutuyla Probe Pini davranışını ters çevirmeniz yeterlidir. **\$6=0** ile de devre dışı bırakabilirsiniz. (Değişikliği yüklemek için bir güç döngüsüne ihtiyacınız olabilir.)

\$10 - Status Report, Mask

'?' komutu ile istendiğinde, GRBL gerçek zamanlı verilerini içeren bir durum raporunun kullanıcıya geri döndürülmesi bu ayara bağlı olarak gerçekleşir. Bu veriler, mevcut çalışma durumunu, gerçek zamanlı konumu, gerçek zamanlı Feed Rate'i, Pin durumlarını, mevcut geçersiz kılma değerlerini, arabellek durumlarını ve o anda yürütülen G-code satır numarasını (derleme zamanı seçenekleri aracılığıyla etkinleştirildiyse) içerir.

Varsayılan olarak, GRBL v1.1+'daki yeni rapor uygulaması, standart durum raporundaki hemen her şeyi içerecektir. Verilerin çoğu gizlidir ve yalnızca değişir

görünür. Bu, eski rapor stiline göre verimliliği önemli ölçüde artırır ve daha hızlı güncellemeler ve makineniz hakkında daha fazla veri almanıza olanak tanır. Bunun nasıl çalıştığına dair ara yüz dokümanlarında ana hatlarda açıklamalar vardır, ancak çoğu yalnızca GUI geliştiricileri veya meraklıları için uygulanabilir.

İşleri basit ve tutarlı kılmak için GRBL v1.1'in yalnızca iki raporlama seçeneği vardır. Bunlar, öncelikle yalnızca kullanıcıların ve geliştiricilerin işleri ayarlamasına yardımcı olmak için buradadır.

- Konum tipi, makine konumunu (MPos:) veya iş konumunu (WPos:) gösterecek şekilde belirtilebilir, ancak her ikisini aynı anda göstermez. İş konumunu etkinleştirmek, GRBL ile bir seri terminal aracılığıyla doğrudan etkileşimde bulunulduğunda belirli senaryolarda yararlıdır; ancak **varsayılan olarak makine konumu raporlaması kullanılmalıdır.**
- GRBL planlayıcısının ve seri RX tamponlarının kullanım verileri etkinleştirilebilir. Bu, ilgili arabelleklerde bulunan blok veya bayt sayısını gösterir. Bu genellikle, bir akış arabirimini test ederken GRBL'nin nasıl performans gösterdiğini belirlemeye yardımcı olmak için kullanılır. *Bu, varsayılan olarak devre dışı bırakılmalıdır.*

Raporlama seçeneklerini etkinleştirmek veya devre dışı bırakmak için aşağıdaki tabloyu kullanabilirsiniz. Etkinleştirmek istediklerinizin listelenen değerlerini eklemeniz ve ardından ayar değerinizi GRBL'ye göndererek kaydetmeniz yeterlidir. Örneğin, makine konumu(Mpos:) ve arabellek verisi raporları olmayan varsayılan rapor ayarı **\$10=1**'dir. İş konumu(Wpos:) ve arabellek verileri isteniyorsa, ayar **\$10=2** olacaktır.

Rapor türü	Değer	Tanım
Pozisyon Türü	0	Etkinleştir WPos: . Devre Dışı Bırak MPos: .
Pozisyon Türü	1	Etkinleştir MPos: . Devre Dışı Bırak WPos: .
Tampon Verileri	2	Etkin Buf: alanı ile planlayıcı ve Serial RX kullanılabilir arabelleği görünür.

\$11 - Junction Deviation, mm

Junction Deviation (Kavşak Sapması), hızlandırma yöneticisi tarafından bir G-code program yolunun çizgi segmenti kavşaklarında ne kadar hızlı hareket edebileceğini belirlemek için kullanılır. Örneğin, G-code yolunda 10 derecelik keskin bir dönüş var ve makine hareketi tam hızda iken köşeyi Step kaybetmeden güvenli bir şekilde geçmek için makinenin ne kadar yavaşlaması gerektiğini belirlemeye bu ayar yardımcı olur.

Bunu nasıl hesapladığımız biraz karmaşıktır, ancak genel olarak daha yüksek değerler, Step ve konum kaybı riskini artırırken, köşelerde daha hızlı hareket sağlar. Daha düşük değerler, hızlanma yöneticisini daha dikkatli yapar ve daha yavaş viraj alınmasını sağlar. Bu nedenle, makineniz çok hızlı viraj almaya çalışırken bir sorunla karşılaşırsanız, virajlara girerken daha da yavaşlatmak için bu değeri azaltın. Makinenizin kavşaklarda daha hızlı hareket etmesini istiyorsanız, hızlandırmak için bu değeri artırın.

Meraklılar için, çok basit, verimli ve sağlam bir yöntemle hem hız hem de kavşak açısını hesaplayan GRBL viraj alma algoritması hakkında [buradan](#) bilgi alabilirsiniz.

\$12 - Arc Tolerance, mm

GRBL render, G2/G3 dairelerini, yayları ve sarmalları, onları küçük küçük çizgilere bölerek oluşturur, böylece yay oluşturma keskinliği asla bu değerin altında kalmaz. 0.002mm ayarı çoğu CNC makinesinin hassasiyetinin çok altında olduğu için, muhtemelen asla bu ayarı yapmanız gerekmeyecektir. Ancak çemberlerinizin çok kaba olduğunu veya ark izlemenin yavaş çalıştığını fark ederseniz, bu ayarı yapabilirsiniz. Daha düşük değerler daha yüksek hassasiyet sağlarken GRBL'yi çok fazla küçük çizgiyle aşırı yükleyerek performans sorunlarına yol açabilir. Alternatif olarak, daha yüksek değerler daha düşük bir hassasiyete yol açar, ancak GRBL'nin uğraşacak daha az satırı olduğundan arc(yay) performansını hızlandırabilir.

Meraklılar için, Arc Tolerance, bitiş noktaları yay üzerinde bulunan bir çizgi parçasından yani bir kirişten maksimum dikey mesafe olarak tanımlanır. Bazı temel geometrilerle, bu ayarı karşılayan yayı izlemek için çizgi parçalarının uzunluğunu çözümleriz. Yayları bu şekilde modellemek harikadır, çünkü yay çizgisi segmentleri, doğruluğu asla kaybetmeden optimum arc izleme performansını sağlamak için otomatik olarak ayarlanır ve uzunlukla ölçeklenir.

\$13 - Report Inches, Boolean

GRBL, makinenin o anda tam olarak nerede olduğu, koordinat ofsetleri ve probleme parametreleri konularında kullanıcıya geri bildirim sağlamak için gerçek zamanlı bir konum raporlama özelliğine sahiptir. Varsayılan olarak, mm cinsinden rapor verecek şekilde ayarlanmıştır, ancak **\$13=1** komutuyla bu boolean bayrağını true olarak gönderdiğinizde inç cinsinden raporlamaya ayarlanacaktır. **\$13=0** komutu ise tekrar mm olarak ayarlamak için kullanılır.

\$20 - Soft Limits, Boolean

Soft Limits, makinenizin seyahat sınırlarının çok ötesine geçmesini, çarpmasını veya pahalı bir şeyi kırmasını önlemeye yardımcı olan bir güvenlik özelliğidir. Makineniz, her eksen için maksimum hareket sınırlarını ve GRBL'nin makine koordinatlarına göre nerede bulunduğunu bilerek çalışır. GRBL'ye her yeni G-code hareketi gönderildiğinde, yanlışlıkla makine alanınızı aşmış olduğunuz kontrol edilir. Bunu yaparsanız, GRBL nerede olursa olsun anında Feed Hold yayınlayacak, iş mili ile soğutmayı kapatacak ve ardından bu probleme dair sistem alarımını kuracaktır. Hard Limits gibi ani bir zorunlu duruştan kaynaklanmadığından, makine konumu da daha sonra korunacaktır.

NOT: Soft Limits, Home Cycle etkinleştirilmesini ve her eksen için doğru Max Travel ayarlarını gerektirir. Çünkü GRBL'nin nerede olduğunu bilmesi gerekir. **\$20=1** etkinleştirmek ve **\$20=0** devre dışı bırakmak için kullanılacak komutlardır.

\$21 - Hard Limits, Boolean

Hard Limits, temelde Soft Limits ile aynı şekilde çalışır, ancak bunda fiziksel anahtarlar kullanılır. Temel olarak bazı anahtarlar (mekanik, manyetik veya optik) her eksenin hareketinin sonuna yakın olarak veya programınız gittiğinde sorun olabileceğini düşündüğünüz kadar uzaklıktaki bir yere bağlanır. Anahtar tetiklendiğinde makine, tüm hareketi hemen durduracak, soğutmayı ve iş milini (bağlıysa) kapatacak ve sizi makinenizi kontrol ederek her şeyi sıfırlamaya zorlayan alarm moduna geçecektir.

GRBL ile Hard Limits kullanmak için Limit Pinleri dahili bir pull-up direnci ile normal-HIGH tutulur. Bu nedenle tek yapmanız gereken Pin ve GROUND arasında NO(Normally Open) bir anahtara kablo bağlamak ve **\$21=1** komutu ile Hard Limits'i etkinleştirmektir. **\$21=0** komutu ise devre dışı bırakmak için kullanılır. Elektrik parazitine önlem alınmasını şiddetle tavsiye ederiz. Bir eksen hareketinin her iki ucu için bir sınır istiyorsanız, Pine ve GROUND'a paralel olarak iki anahtar bağlamanız yeterlidir, böylece bunlardan biri devreye girerse, Hard Limits tetiklenir.

Bir Hard Limit olayının, stepper motorlarının hemen durduğu ve muhtemelen steplerini kaybetmiş olacağı kritik bir olay olarak kabul edildiğini unutmayın. GRBL'nin konumla ilgili herhangi bir geri bildirimi olmaz, bu nedenle nerede olduğu konusunda bir bilgiyi garanti edemez. Bu nedenle, bir Hard Limit tetiklenirse, GRBL sonsuz döngü ALARM moduna girerek sizi makinenizi kontrol ederek GRBL'yi sıfırlamaya zorlar. Bunun tamamen bir güvenlik özelliği olduğunu unutmayın!

\$22 - Homing Cycle, Boolean

Ahh, Homing. CNC'ye yeni başlayanlar için söyleyelim ki GRBL'nizi oturumlar arasında her başlattığınızda makinede bilinen ve tutarlı bir konumu doğru ve hassas bir şekilde bulmak için Homing Cycle kullanılır. Başka bir deyişle, herhangi bir anda nerede olduğunuzu, her defasında tam olarak bilirsiniz. Diyelim ki bir şeyi işlemeye başlıyorsunuz veya bir işte bir sonraki adıma başlamak üzeresiniz ve güç kesiliyor, GRBL'yi yeniden başlatıyorsunuz ve GRBL'nin, onları Open-loop Control etmesi nedeniyle stepper motorların nerede olduğu hakkında hiçbir fikri yok. Nerede olduğunuzu bulma görevi size kaldı. Homing varsa, yerini belirlemek için her zaman makine sıfır referans noktasına sahip olursunuz, böylece tek yapmanız gereken Homing Cycle çalıştırıp kaldığınız yerden devam etmektir.

GRBL için Homing Cycle ayarlamak için, limit anahtarlarının temas edilemeyecek veya hareket etmeyecek sabit bir konumda olması gerekir, aksi takdirde referans

noktanız bozulur. Bu anahtarlar genellikle her eksenin +x, +y, +z'deki en uzak noktasına kurulurlar. Limit anahtarlarınızı limit pinleriyle bağlayın, elektrik parazitini azaltmaya yardımcı olması için önerilen bir RC filtresi ekleyin ve Homing'i etkinleştirin. Merak ediyorsanız söyleyelim, limit anahtarlarınızı hem Hard Limits hem de Homing için kullanabilirsiniz. Birbirleriyle güzel çalışıyorlar.

Homing Cycle'ı ilk kez denemeden önce, her şeyi doğru ayarladığınızdan emin olun, aksi takdirde Homing garip davranabilir. İlk olarak, makine eksenlerinizin Kartezyen koordinatlarına göre doğru yönlerde hareket ettiğinden emin olun (sağ el kuralı). Değilse, **\$3** komutu üzerinden Direction Invert ayarı ile düzeltin. İkinci olarak, limit anahtarı pinlerinizin GRBL'nin durum raporlarında 'tetiklendi' olarak gösterilmediğinden emin olun. Varsa, kablolanızı ve ayarlarınızı kontrol edin. Son olarak, **\$13** komutuyla ayarlanan Max Travel ayarlarınızın doğruluğundan biraz emin olun (%20 gibi), çünkü GRBL bu değerleri, Homing anahtarlarını ne mesafede araması gerektiğini belirlemek için kullanır.

Varsayılan olarak, GRBL'nin Homing Cycle'ı, çalışma alanını netleştirmek için önce Z eksenini pozitif yönde hareket ettirir ve ardından hem X hem de Y eksenlerini aynı anda pozitif yönde hareket ettirir. Homing Cycle'nızın nasıl davranacağını ayarlamak için, Açıklamalar Sayfasında, GRBL ayarlamalarının ne yaptığına (hatta derleme zamanı seçeneklerine) dair yer alan bilgilerden faydalanın.

Ayrıca, Homing etkinleştirildiğinde dikkat edilmesi gereken bir şey daha var. GRBL, siz bir Homing Cycle gerçekleştirene kadar tüm G-code komutlarını kilitleyecektir. Bu, **\$X** komutuyla kilit devre dışı bırakılmadıkça eksen hareketi olmayacağı, ancak kilit devre dışı bırakıldıktan sonra olacağı anlamına gelir. Tüm CNC kontrol yazılımları olmasa da çoğu benzer bir şey yapar. Bu, kullanıcıların bir konumlandırma hatası yapmasını önlemek için bir güvenlik özelliğidir ki böyle bir hatanın yapılması çok kolaydır ve bu hata bir parçayı mahvettiğinde üzücü olur. Bunu can sıkıcı bulur veya bir garip hata ile karşılaşırsanız, lütfen bize bildirin ve herkesin mutlu olması için üzerinde çalışmaya çalışalım. :)

NOT: İleri düzey kullanıcılar, daha fazla Homing seçeneği için config.h'ye bakabilir. Başlangıçta Homing kilidinin devre dışı bırakılmasını, bir Homing Cycle sırasında hangi eksenlerin önce ve hangi sırayla hareket edeceğini yapılandırabilir ve daha fazlasını yapabilirsiniz.

\$23 - Homing Dir Invert, Mask

GRBL, varsayılan olarak, limit anahtarlarınızın pozitif yönde olduğunu kabul eder; anahtar etrafında yavaşça ileri geri giderek makine sıfırına karar vermeden önce, ilk olarak Z eksenini ardından da X ve Y eksenlerini pozitif yönde hareket ettirir. Makinenizde negatif yönde bir limit anahtarı varsa, Homing Dir maskesi eksenlerin yönünü tersine

çevirebilir. Tıpkı Step Port Invert ve Direction Port Invert maskeleri gibi çalışır. Yapılması gereken tek şey, ters çevirmek istediğiniz eksen(lere) dair tablodaki değeri **\$23** komutu ile ayarlamaktır.

\$24 - Homing Feed, mm/min

Homing Cycle, önce limit anahtarlarını daha yüksek bir Homing Seek Rate ile arar ve onları bulduktan sonra daha yavaş bir Feed Rate ile makine sıfır noktasının kesin konumuna hareket eder. Homing Feed, daha yavaş bir Feed hızıdır. Bunu, tekrarlanabilir ve hassas makine sıfır konumu sağlayan bir değere ayarlamak için **\$24** komutu kullanabilirsiniz.

\$25 - Homing Seek, mm/min

Homing Seek, Homing Cycle arama hızı veya limit anahtarlarını bulmaya çalıştığı ilk hızdır. Yeterince kısa bir süre içinde, limit anahtarlarınıza çok hızlı geldiğinde çarpmaya engel olacak bir hız ayarlayın. Bu ayarı set etmek için **\$25** komutu kullanılır.

\$26 - Homing Debounce, milisecond

Bazı anahtarlar, tetiklendiğinde, sinyal kendi düzeyine(HIGH/LOW) tamamen yerleşmeden önce birkaç milisaniye boyunca sinyali HIGH ve LOW arasında "sektiren" elektrik/mekanik bir gürültüye sahip olabilir. Bunu çözmek için, ya bir tür donanım ile sinyali geri döndürmeye ya bir sinyal koşullandırıcıya veya yazılım tarafından sinyalin sıçramayı bitirmesine izin verecek kısa bir gecikme ayarlamaya gerek vardır. GRBL yalnızca makine sıfırını bulurken kısa bir gecikme gerçekleştirerek Homing yapar. Bu gecikme değerini, tekrarlanabilir bir Homing elde etmek için anahtarınızın ihtiyaç duyduğu değere ayarlayın. Çoğu durumda, 5-25 milisaniye iyidir. Bu ayarı **\$26** komutuyla set edebilirsiniz.

\$27 - Homing Pull-off, mm

Homing aynı limit anahtarlarını paylaşabildiği Hard Limits ile uyuşmak için, Homing Cycle tamamlandıktan sonra bu çekme(pull-off) hareketiyle tüm limit anahtarlarını devre dışı bırakır. Başka bir deyişle, bir Homing Cycle'dan sonra Hard Limit'in kazara tetiklenmesini önlemeye yardımcı olur. Bu değer limit anahtarını serbest bırakacak büyüklükte olduğundan emin olun. Değilse, GRBL onu temizleyemediği için bir alarm hatası verecektir. Bu değeri **\$27** komutu ile set etmek gerekir.

\$30 - Max Spindle Speed, RPM

Bu, iş mili hızını belirlemek için maksimum 5V'luk bir PWM Pin çıkışını ayarlar. Örneğin, 10000rpm'yi 5V olarak ayarlamak istiyorsanız, **\$30=10000** komutunu kullanın. 5V'ta 255RPM için **\$30=255**... Bir program, \$30maksimum iş mili hızından daha yüksek bir

iş mili devrini ayarlamaya çalışırsa, GRBL daha hızlı gidemediği için yalnızca maksimum 5V çıkışı verir. GRBL, varsayılan olarak, maks-min RPM'leri 255 eşit aralıklı artışlarla 0.02-5V PWM Pin çıkışıyla doğru orantılı biçimde ilişkilendirir. PWM Pini 0V ise bu, iş milinin kapalı olduğunu gösterir. Bunun çalışmasını ayarlamak için config.h'de ek yapılandırma seçenekleri bulunduğunu unutmayın.

\$31 - Min Spindle Speed, RPM

Bu, minimum 0,02V PWM Pin çıkışı (0V devre dışı) için iş mili hızını ayarlar. Daha düşük RPM değerleri GRBL tarafından kabul edilir, ancak RPM'nin sıfır olduğu durumlar dışında PWM çıkışı 0,02V'nin altına düşmez. Sıfırsa, iş mili devre dışıdır ve PWM çıkışı 0V'tur. Ayarı değiştirmek için **\$31** komutu kullanılır.

\$32 - Laser Mode, Boolean

Etkinleştirildiğinde, GRBL bir iş mili hızıyla (**S**) (lazer gücü) programlandığında ardışık **G1**, **G2** veya **G3** hareket komutlarıyla sürekli olarak hareket edecektir. İş mili PWM Pini, her harekette durmadan anında güncellenecektir. Bu modu kullanmadan önce lütfen [GRBL lazer belgelerini](#) ve lazer cihazınızın belgelerini okuyun. Lazerler çok tehlikelidir. Görüşünüze anında kalıcı olarak zarar verebilir ve yangınlara neden olabilirler. GRBL, GPL lisansı ile tanımlandığı şekilde, Firmware'in neden olabileceği herhangi bir sorun için herhangi bir sorumluluk kabul etmez.

Devre dışı bırakıldığında, GRBL her zaman olduğu gibi çalışacak ve her iş mili hızı (**S**) komutuyla hareketi durduracaktır. Bu, iş milinin hızlarını değiştirmesini imkân veren bir duraklamayı sağlamak üzere herhangi bir freze makinesinin varsayılan işlemidir.

\$100, \$101, \$102 - [X,Y,Z] Steps/mm

GRBL, her stepper motorların her bir adımının, aracı gerçekte ne kadar mesafe hareket ettireceğini bilmelidir. Makinenizin bir eksen için steps/mm hesaplamak üzere bilmeniz gerekenler:

- Stepper motorunuzun devri başına kat edilen mm. Bu, kayış tahrik dişlilerinize veya vidalı mil hatvesine bağlıdır.
- Stepper motorlarınızın devir başına toplam adımları (tipik olarak 200)
- Denetleyicinizin adım başına Micro Step sayısı (tipik olarak 1, 2, 4, 8 veya 16). İpucu: Yüksek Mikro Step değerleri (ör. 16) kullanmak, stepper motor torkunu azaltabilir; bu nedenle istenilen eksen hareketi çözünürlüğünü ve rahat çalışma özelliklerini sağlayan en düşük değer kullanılmalıdır.

Steps/mm artık şöyle hesaplanabilir:

$$\text{steps_per_mm} = (\text{steps_per_revolution} * \text{microsteps}) / \text{mm_per_rev}$$

Bu değeri her eksen için hesaplayın ve bu ayarları GRBL'ye yazın. X eksen için **\$100**, Y eksen için **\$101** ve Z eksen için **\$102** komutlarını kullanabilirsiniz.

\$110, \$111, \$112 - [X,Y,Z] Max Rate, mm/min

Bu değer, her eksenindeki hareketin gerçekleştirilebileceği maksimum hızı ayarlar. GRBL bir hareket planladığında, hareketin eksenlerden herhangi birinin maksimum hızının aşılmasına neden olup olmadığını kontrol eder. Eğer öyleyse, eksenlerin hiçbirinin maksimum hız limitlerini aşmamasını sağlamak için hareketi yavaşlatacaktır. Bu, her eksenin -özellikle de tipik olarak daha yavaş olan Z ekseninin- gayet kullanışlı biçimde sınırlandırılması gereken bağımsız bir hıza sahip olduğu anlamına gelir.

Bu değerleri belirlemenin en basit yolu, maksimum hız ayarlarını yavaşça artırarak ve hareket ettirerek her bir eksen tek tek test etmektir. Örneğin, X eksenini test etmek için GRBL'ye **G0 X50** gibi eksenin maksimum hızına ulaşması için yeterli hareket mesafesine sahip bir G-code gönderin. Stepper motorlarınız hareketi kestiğinde maksimum hız eşiğine ulaştığı anlaşılacaktır. Biraz gürültü yapacak, ancak motorlara zarar vermeyecektir. Bu şekilde ulaşılan en yüksek değer %10-20 altında bir değer girilmelidir. Böylece iş parçanızın/aletinizin aşınma, sürtünme ve kütle ataletini hesaplayabilirsiniz. Ardından, diğer eksenleriniz için tekrarlayın.

NOT: Bu maksimum hız ayarı aynı zamanda **G0** G-code'una ait arama hızlarını da belirler.

\$120, \$121, \$122 - [X,Y,Z] Acceleration, mm/sn²

Bu, eksen ivmelenme parametrelerini mm/(saniye*saniye) olarak ayarlar. Basitçe söylemek gerekirse, daha düşük bir değer GRBL'nin makinede ürettiği hareketi hantallaştırır, daha yüksek bir değer ise daha seri hareketler sağlar ve istenen Feed hızlarına çok daha kısa sürede ulaşılır. Maksimum hız ayarına çok benzer şekilde, her eksenin kendi hızlanma değeri vardır ve birbirinden bağımsızdır. Bu, çok eksenli bir hareketin yalnızca en düşük katkı sağlayan eksenin yapabileceği kadar çabuk ivmeleneyeceği anlamına gelir.

Yine, maksimum hız ayarı gibi, bu ayar için değerleri belirlemenin en basit yolu, stepper motor hareketini kesene kadar her bir eksen yavaş artan ivmelenme değerleriyle ayrı ayrı test etmektir. Ardından, ulaşılan maksimum değer %10-20 altında bir değerle ivmelenme değerini X eksen için **\$120**, Y eksen için **\$121** ve Z eksen için **\$122** komutlarıyla ayarlayabilirsiniz.

Bu değerlerin hesabında aşınma, sürtünme ve kütle ataleti hesaba katılmalıdır. Bazı G-code uygulamalarının öncesinde, yeni ayarlarla boş olarak(?) test edilmesini önemle

tavsiye ederiz. Bazen tüm eksenlerde birlikte hareket ederken makinenizdeki yüklenme farklıdır.

\$130, \$131, \$132 - [X,Y,Z] Max Travel, mm

Bu ayar, mm cinsinden her eksen için uçtan uca maksimum hareket mesafesini belirler. Bu, yalnızca Soft Limits (ve Homing) etkinse kullanışlıdır, çünkü bu sadece GRBL'nin Soft Limits özelliği tarafından bir hareket komutuyla makine limitlerinizi aşp aşmadığınızı kontrol etmek için kullanılır. X eksen için **\$130**, Y eksen için **\$131** ve Z eksen için **\$132** komutlarını kullanarak her eksenin maksimum hareket mesafesini GRBL'ye kaydedebilirsiniz.

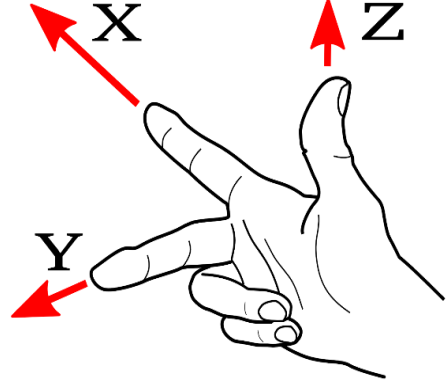
MAKİNEİNİZİ İLK KEZ KURMAK İÇİN HIZLI KILAVUZ

GRBL'nin varsayılan yapılandırması, kullanıcıların ayarları değiştirmek zorunda kalmadan başarılı hareketi görebilmelerini sağlamak için kasıtlı olarak gayet genel biçimde düzenlemiştir.

Yapmak isteyeceğiniz ilk şey, step motorlarınızı genellikle CNC'ye bağlı olmadan çalıştırmaktır. Üretici yönergelerinize göre stepper sürücüleriniz ve stepper motorlarınızla GRBL'yi bağlayın. Bir seri terminal veya birçok GRBL GUI'sinden biri aracılığıyla GRBL'ye bağlanın. GRBL'ye bazı **G1** veya **G0** komutları gönderin. Stepper motorunuzun döndüğünü görmelisiniz. Stepper motorlarınızla ilgili sorun yaşıyorsanız, aşağıdakileri deneyin:

- Step sürücü üreticinizin yönergelerine göre her şeyin doğru şekilde bağlandığından ve güç verildiğinden emin olun.
- Stepper motorlarınız CNC'ye zaten monte edilmişse, eksenlerin serbestçe hareket ettiğinden ve sıkışmadığından emin olun. Kolayca anlayamıyorsanız, stepper motorları çıkarmayı deneyin ve yüksüz çalışıp çalışmadıklarını kontrol edin.
- Stepper motorların ve eksen lineer hareket mekanizmalarının tamamen sıkı ve güvenli olduğundan emin olun. Aktarma organları bileşenlerindeki küçük ayar vidalarının gevşemesi çok yaygın bir sorundur. Tekrar sıkın ve sürekli olarak gevşerse kalıcı olmayan bir iplik tutucu (Loctite Blue) uygulamayı deneyin.
- Daha zor sorunlar için, sorunu hızla yalıtmak için problemleri elemeyi geçirin. Arduino'dan her şeyin bağlantısını keserek başlayın. GRBL'nin kendi kendine iyi çalışıp çalışmadığını test edin. Ardından, her seferinde bir şey ekleyerek test edin.
- Stepper motorlar çalıştırılıyorsa ve hareket etmeye çalışırken gıcırdama sesi çıkarıyorsa, Acceleration ve Max Rate değerlerini düşürmeyi deneyin. Bu ses, stepper motorun steplerini kaybettiğini ve çok fazla tork yükü veya çok hızlı gitmesi nedeniyle yetişemediğinin bir işaretidir.
- GRBL'nin varsayılan Step Pulse ayarları, piyasadaki stepper sürücülerinin büyük çoğunluğu tarafından desteklenir. Çok yaygın olmamakla birlikte, hala sorun yaşıyorsanız veya olağandışı bir kurulumunuz varsa bu ayarları kontrol edin.

Ardından, makinenizin Kartezyen(XYZ) koordinat çerçevesine göre doğru yönlerde hareket ettiğinden ve gösterildiği gibi sağ el kuralını karşıladığından emin olmanız gerekir:



Henüz yapmadıysanız, stepper motorları CNC'nize monte edin. GRBL'ye **G91 G0 X1** veya **G91 G0 X-1** gibi X eksenini sırasıyla +1 mm ve -1 mm hareket ettirecek bazı hareket komutları gönderin. Tüm eksenleri kontrol edin. Bir eksen doğru hareket etmiyorsa, **\$3** komutuyla yönü tersine çevirmek için Direction Port Invert maskesi ayarını değiştirin.

CNC makinelerinde koordinat çerçevelerinin nasıl kurulduğuna aşina değilseniz, [LinuxCNC'nin](#) bu harika şemasına bakın. Hareketlerin yalnızca alete özgü olduğunu unutmayın. Bu nedenle, tipik bir CNC portal yönlendiricide, sabit masa yerine takım hareket edecektir. X eksenini pozitif olarak sağa hizalanırsa, pozitif bir hareket komutu takımı sağa hareket ettirecektir. Oysa sabit bir araçla hareket eden bir tablo, aynı komut için tabloyu sola hareket ettirecektir, çünkü araç tabloya göre sağa hareket etmektedir.

Son olarak, istediğiniz veya maksimum performansa yaklaşmak için ayarlarınızı yapın. **\$100**, **\$101** ve **\$102** eksen Steps/mm ayarlarınızın kurulumunuz için doğru olduğundan emin olarak başlayın. Bu safha, adım adım artışlarınıza, sürücünüzdeki Micro Step'lere ve mekanik parametrelere bağlı olarak ilerler. Makine üreticiniz bunu sizin için sağlamadıysa, bilinen makineniz için bunu nasıl hesaplayacağınızı gösteren çevrimiçi birçok kaynak vardır. Performansı artırmak için **\$120**, **\$121** ve **\$122** ivmelenme ve **\$110**, **\$111** ve **\$112** Max Rate değerlerini değiştirin. Atalet, kesme kuvvetleri ve hızdaki motor torku düşüşlerini hesaba katmak için mutlaka bu değerleri maksimum değerinin %80'inden fazla olmayacak şekilde ayarlayın. **\$130**, **\$131** ve **\$132** Homing veya Soft Limits kullanmayı planlıyorsanız Max Travel değerlerini, ileride sorun yaşamamak için gerçek değerlere yakın bir şekilde ayarlamanız önerilir. **\$130**, **\$131** ve **\$132** Max Travel parametrelerinin grblControl'de görünen 'Machine Coordinates' parametrelerine karşılık geldiğini unutmayın.

Bu noktada, ilerlemeye hemen hemen hazırsınız! GRBL artık CNC makinenizi hareket ettirebilir ve G-code işlerini çalıştırabilir. Homing veya Soft Limits veya iş mili/lazer kontrolü için limit anahtarları gibi daha fazla özellik eklemeniz gerekiyorsa; size bu konuda yardımcı olacak başka Wiki sayfaları var.

İyi şanslar ve iyi eğlenceler!