

Cnc Designer

Z1100 DC Servo Motor Kontrol ipi

Z1100 Doküman No: 0001



İçindekiler

Z1100 Çipi Hakkında	3
Çip Bağlantıları	4
Pozisyon, Hız ve İvme	5
Hız Profili	6
Hata değeri	7
Seri Haberleşme	8
Seri haberleşmede protokol	8
İşlemciden çipe gönderilen paketin içeriği	9
Çipten işlemciye gönderilen paketin içeriği	9
Hata Kodları	9
Komut Listesi	10
Set tipi komutlar	11
SetKAPH	11
GetKABH	11
SetKAPD	12
GetKABD	12
SetVmax	13
GetVmax	13
SetAcc	14
GetAcc	14
SetStm	15
GetStm	15
SetHvm	16
GetHvm	16
SetHac	17
GetHac	17
SetLimit	18
GetLimit	18

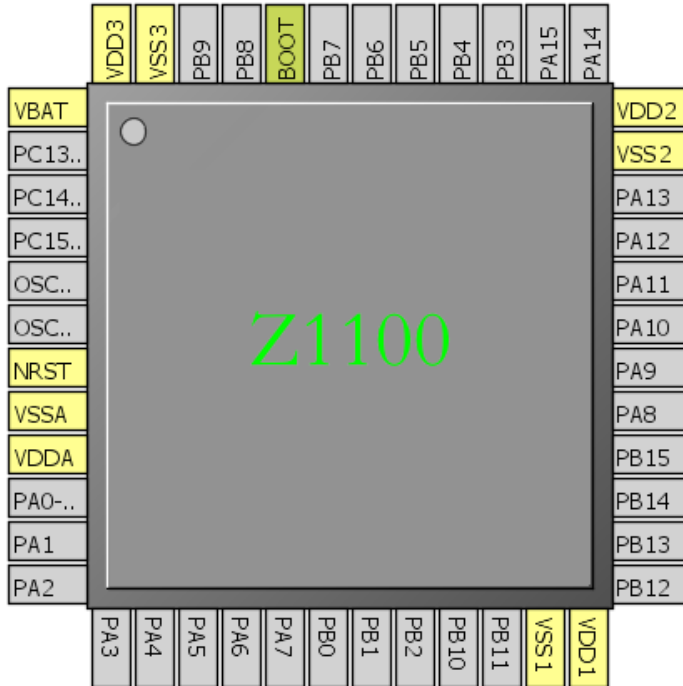


SetCab	19
GetCab.....	19
SetJob	21
GetJob	21
SetBusy	22
GetBusy	22
SetTarget	23
GetTarget.....	23
SetStart.....	24
GetStart	24
SetStop	26
GetStop.....	26
SetEstop.....	27
GetEstop	27
SetClr	28
SetZero	28
SetBaud	28
SetFlash	28
GetEncoder.....	29
GetError.....	29
GetSwitch	29
Z1100 - Elektriksel parametreler	30
Z1100 - Ayar programı.....	31
Z1100 – Çip yazılımını güncelleme.....	32
Z1100 – Yazılımcılar için ipuçları	33
Z1100 – Örnek program.....	34

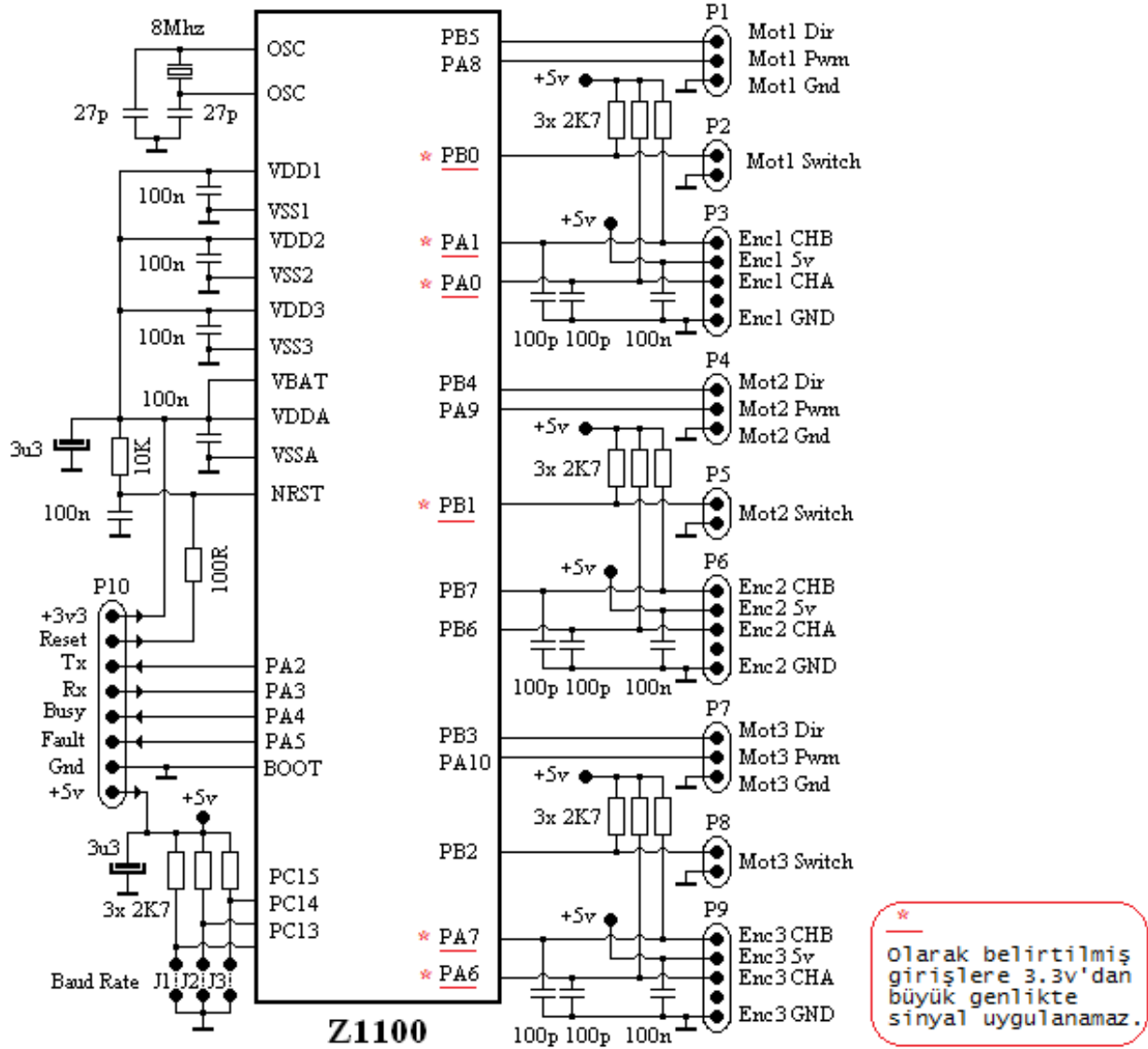


Z1100 Çipi Hakkında

- 3 ayrı DC motoru kontrol edebilir.
- Motor sürücü için Pwm ve Dir sinyalleri üretir.
- Her bir motoru sabit hızda döndürülebilir.
- Her bir motoru noktadan noktaya pozisyonlayabilir.
- Her bir motor için trapez tipi hızlanma yapabilir.
- Motorların durmasını algılama - Zorlanma koruması.
- x4 modunda artımsal encoder kullanır.
- Sıfır konumunu öğrenmek için sıfır noktası switchleri bağlanabilir.
- Motor yada encoder kabloları yazılım ile çaprazlanabilir.
- Sıfır noktası switchlerin aktif durumunun 1 yada 0 oluşu tanımlanabilir.
- Seri haberleşme yaparak ana işlemci ile konuşur.
- 8Mhz saat frekansı ve 3.3v besleme kullanır.
- Çip içindeki yazılım seri haberleşme ile değiştirilebilir.



Çip Bağlantıları



Şema 1

Sekil 2'deki devre mikroişlemci ile haberleşmek üzere çizilmiştir. PC ile haberleşmek için Tx, Rx, Reset ve Busy pinlerine RS232 interface çipi eklenmelidir. Busy pininin durumunu DSR pini ile okuyabilir, DTR pinini de çipi resetlemek için kullanabilirsiniz.

Baud Rate ayarlama

Baud	J3	J2	J1
1200	Yok	Yok	Yok
2400	Yok	Yok	Var
4800	Yok	Var	Yok
9600	Yok	Var	Var
19200	Var	Yok	Yok
38400	Var	Yok	Var
-	-	-	-
-	-	-	-



Pozisyon, Hız ve İvme

Z1100 çipi artımsal çalışan “encoder” dan gelen darbeleri x 4 modunda sayar.

Örneğin devir başına 512 darbe üreten encoder sayesinde Z1100 çipi motorun bir dönüşünde $4 \times 512 = 2048$ sayım (count) yapar.

Pozisyon birimimiz [**sayım**] (count) dır.

Motor sözkonusu encoder ile saniyede N sayma yaptırıyorsa

V hızımız N [**sayım/saniye**] (count/sec) olacaktır.

Motor N sayma hızına 0 saniyeden T saniye sonra ulaşıyorsa

İvmemiz N/T [**sayım/saniye²**] (count/sec²) olacaktır.

Örnek motorumuzun miline bağlı encoder, devir başına 512 darbe üretiyor.

Motor, 32768 sayım/saniye² ivme ile 2 saniye boyunca 0 dan itibaren hızlansın.

2 saniye sonundaki hızımız ve pozisyonumuz ne olacaktır sorusunu cevaplayalım.

Hız = İvme x Zaman = $32768 \times 2 = 65536$ sayım/saniye olarak hesaplanır.

Pozisyon ise $\frac{1}{2} \times \text{İvme} \times \text{Zaman}^2$ bağıntısı ile hesaplanır.

Bu durumda Pozisyon = $\frac{1}{2} \times 32768 \times 2 \times 2 = 65536$ sayım olacaktır.

Z1100 servo kontrol çipine ait hız hesaplamaları ile ilgili parametreler Şekil 3’de verilmiştir.

Posizyon	Sayım
Hız	Sayım/saniye
İvme	Sayım/Saniye ²

Figure 1

Hız Profili

Z1100 çipi, durmakta olan motoru parametre olarak verilmiş ivme değeri ile gene parametre ile bildirilmiş max hız değerine kadar motoru hızlandırır. Hedefe sıfır hızla ulaşabilmek için hesaplanmış olduğu yavaşlama mesafesine kadar motoru bu hızda döndürür. Yavaşlama pozisyonuna gelindiğinde belirlenen ivme ile motoru yavaşlatır ve tam hedefe ulaşıldığında motorun hızı sıfır olmuş olur. (Şekil 4)

Böylece mekanizma silkelenmeden harekete başlar ve silkelenmeden durur.

Hız profili için çipe V_{max} , İvme ve Hedef Pozisyon (Target) değerlerinin verilmesi yeterlidir.

Kısa mesafeli hareketlerde ivme yeterince yüksek verilmiş olsa bile v_{max} değerine çıkmak mümkün olmayabilir. Bu durumda V_{max} 'ın önemi kalmaz ve hesaplamalar gene hedef pozisyona sıfır hızla ulaşmayı sağlayacak şekilde yapılır.

Tüm bu işlemler Z1100 çipi tarafından yapılır.

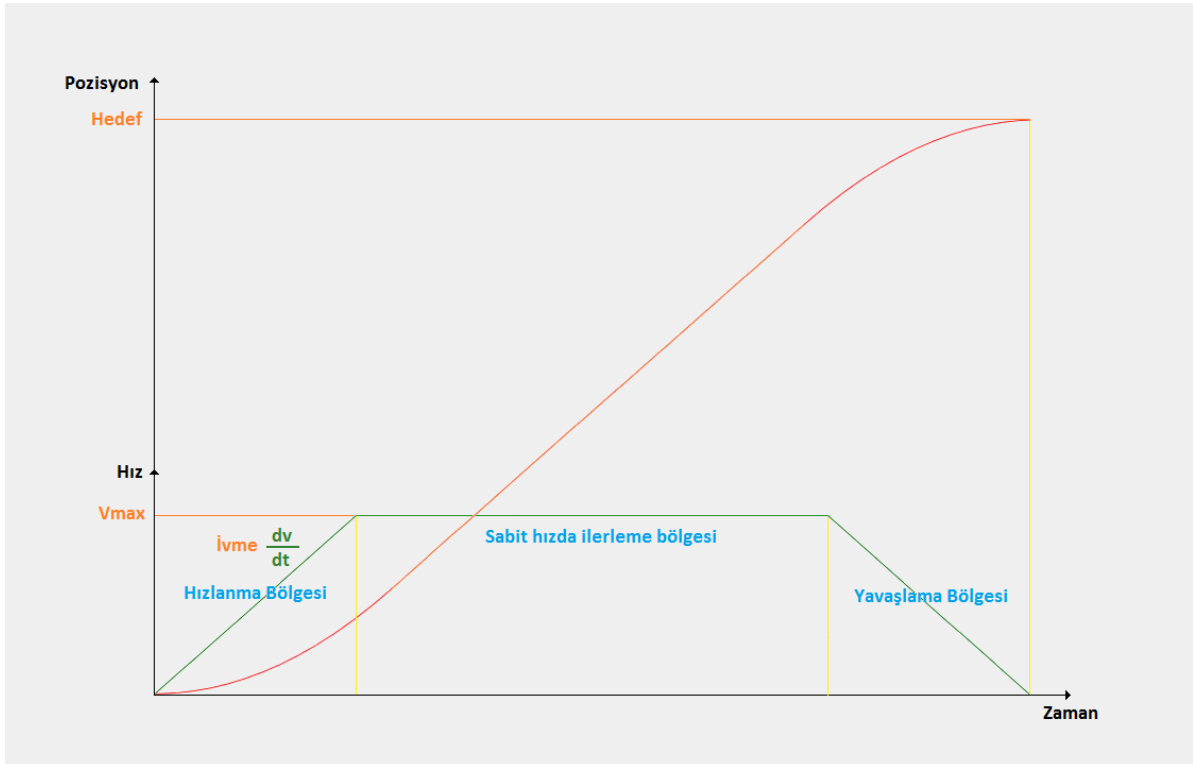


Figure 2

Hesaplamalarda pozisyondan bahsedilirken encoder pozisyonundan bahsedilir ve değeri "sayım" dır. Hızdan bahsedilirken sayım/saniye, ivmeden bahsederken ise sayım/saniye² anlaşılmalıdır.

Hata değeri

Z1100 servo kontrol çipi motor pozisyonunu belirli bir hata ile kontrol altında tutmaya çalışır ancak bu hata, kullanılan algoritmalar gereği sıfır olamaz.

Motorla tahrik edilen mekanizmadaki mekanik problemler sorunucu motor zorlanırsa hata değeri artar. Hatanın üst limiti tanımlanabilir ve hata tanımlanan bu değeri aşarsa motor durdurulur. Buna motor mili sıkışması tespiti diyoruz.

Şekil 5'deki basitleştirilmiş blok diyagramda hata sinyali gösterilmiştir.

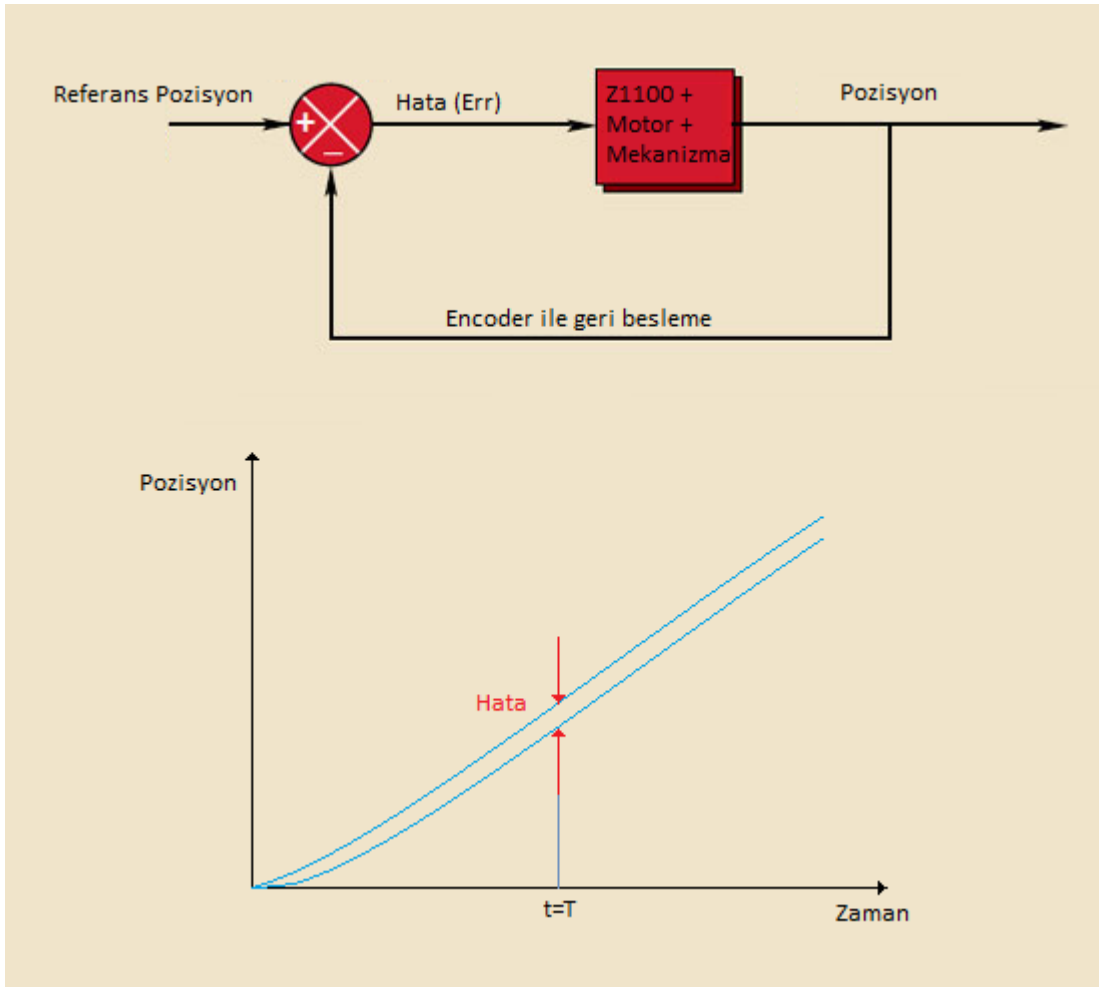


Figure 3

Herhangi bir zamanda sistemin yaptığı hata değeri işlemci tarafından Z1100 çipinden talep edilebilir.

Ya da hata için tanımlanmış üst değer Z1100 çipine bildirilebilir. Z1100 çipi motoru döndürürken mekanizma sıkışıp kalır veya motor ya da encoder kablosu koparsa hata değeri artar ve üst değeri aştığında Z1100 motora enerji vermeyi keser.

Bu sayede sistemin zarar görmesi engellenir.



Seri Haberleşme

Tx ve Rx pinleri ile standart asenkron seri haberleşme yapılır.

İletişim parametreleri 9600 Baud, 8 Data, 1 Stop bit şeklindedir.

Çipin kullanacağı iletişim hızı (Baud Rate) 3 adet jumper ile tanımlanır. Çipe enerji verildiğinde yada Reset pininden resetlendiğinde, çip jumperları okuyarak iletişim hızını öğrenir. Çip çalışırken jumper pozisyonları değiştirilirse iletişim hızı değişmez.

İşlemci ile haberleşebilmek için işlemcinin Tx pini çipimizin Rx pinine, işlemcinin Rx pini de çipimizin Tx pinine bağlanmalıdır. Sayfa 1 de verilen şemada Tx pini Z1100 çipinin Tx yani çıkış pinidir.

Eğer çipi PC ile haberleştirmek isterseniz bu durumda çip ile PC arasına RS232 interface çipi bağlanmalıdır. Çip doğrudan PC RS232 portuna bağlanamaz. Bağlandığı takdirde hasar görebilir.

Ana işlemci, çipte isteklerini protokole sadık kalarak çipe gönderir.

İstekler paketlerle iletilir. Her bir istek bir pakettir ve 1 paket 7 byte uzunluktadır. Çipimiz talep edilmediği sürece işlemciye veri göndermez.

Seri haberleşmede protokol

Kullanıcı işlemcisi çipimize 1 paket yollar ve karşılığında 1 paket bekler.

İşlemcinin çipe göndereceği paket 7 byte'dan oluşur. Çip ilk byte'ı yakaladığında 100ms içinde 7 byte'ın tamamını almak zorundadır. Beklenen süre içinde paketin tamamı alınmazsa bu veriler dikkate alınmaz ancak, işlemciye bu durumda dahi paket gönderir.

İşlemci veri göndermediği halde etraftaki elektriksel gürültü nedeni ile iletişim hattında parazit oluşur ve çipimiz bunu algılasa gene bu durum için de işlemciye paket gönderir.

İşlemci gönderdiği her pakete karşılık 1 paket cevap almak zorundadır. Alacağı paket iletişimin doğru yada yanlış yürüdüğünü, yanlış ise ne tip bir hata yakalandığını, doğru ise çipten talep ettiği verileri içerir.

İşlemci de çipten beklediği cevabı (paketi) 100 mili saniye içinde almalıdır. Aksi takdirde komut tekrarı yapılmalıdır.



İşlemciden çipe gönderilen paketin içeriği

İşlemciden gönderilecek paket 0x5A verisi ile başlar. Ardından 8 bitlik komut (Cmd) gönderilir. Bunu takiben 32 bit veri ve son olarak kontrol bilgisi (CS) yollanır.

Gönderilmek istenen 32 bit veri (Data), D3 D2 D1 D0 şeklinde 4 adet 8 bit veriden oluşur ve aşağıdaki gibi parçalanır.

$$D3 = (\text{Data} \gg 24) ;$$

$$D2 = (\text{Data} \gg 16) \& 0xFF;$$

$$D1 = (\text{Data} \gg 8) \& 0xFF;$$

$$D0 = (\text{Data}) \& 0xFF;$$

Bu durumda paketimiz 0x5A, Cmd, D3, D2, D1, D0, CS şeklinde olacaktır.

$$CS = (0x00 - (\text{Cmd} + D3 + D2 + D1 + D0)) \& 0xFF \text{ şeklinde hesaplanır.}$$

Bazı komut parametreleri 32 bitten daha kısa fakat birden fazla parametre içerebilir.

Çipten işlemciye gönderilen paketin içeriği

Çipimiz işlemciden düzgün paket yada protokole uygun olmayan veri aldığına buna muhakkak bir paket ile cevap verir.

Çipin işlemciye yolladığı paket 6 byte veriden oluşur.

Hata Kodu, D3, D2, D1, D0, CS

İşlemci D3, D2, D1, D0 olarak aldığı verileri 32 bite çevirmek için aşağıdaki işlemi yapmalıdır.

$$\text{Data} = (D3 \ll 24) | (D2 \ll 16) | (D1 \ll 8) | D0;$$

$$(\text{Hata Kodu} + D3 + D2 + D1 + D0 + CS) \& 0xFF = 0x00 \text{ olarak kontrol edilmelidir.}$$

Hata Kodları

0..6 bitlerinden herhangi birisi 1 ise hata var demektir ve komutun uygulamaya alınmadığını bildirir. 7 nolu bit hata sınırını aştığı için durdurulan motor olduğunu bildiren flag bitidir.

Bit 7: Zorlandığı için durmuş motor var (GetStop komutuyla hangi motorun durduğu öğrenilebilir)

Bit 6: 3. Motor hareket halinde iken yasaklı komut geldi

Bit 5: 2. Motor hareket halinde iken yasaklı komut

Bit 4: 1. Motor hareket halinde iken yasaklı komut

Bit 3: İletişim Hatası (Overrun, Framing, Noise).

Bit 2: Paketin tamamı 100 mili saniye içinde gelmedi.

Bit 1: Mevcut olmayan komut isteği geldi

Bit 0: CS hesabı yanlış sonuç veriyor.



Komut Listesi

Komutlar		
Çipe veri yollar	Çipten veri ister	
SetKAPH	GetKAPH	Hareketli durumlar için K, A ve B parametreleri.
SetKAPD	GetKAPD	Durağan durumlar için K, A ve B parametreleri.
SetVmax	GetVmax	Max hız değeri.
SetAcc	GetAcc	İvme değeri.
SetHvm	GetHvm	Sıfırlama aşamasında switche doğru ilerleme hızı.
SetHacc	GetHacc	Sıfırlama aşamasında switche doğru ivmelenme değeri.
SetLimit	GetLimit	Milin sıkışma değeri.
SetCab	GetCab	Kablo vs tanımları.
SetJob	GetJob	Görev tanımı.
SetTarget	GetTarget	Ulaşılmak istenen hedef.
SetStart	GetStart	Hareketi başlat.
SetStop	GetStop	Hareketi sonlandır.
SetEstop	GetEstop	Acil durdurma işlemi.
SetClr	----	
SetZero	----	
----	GetEncoder	Encoder pozisyonunu oku.
----	GetError	Anlık pozisyon hatasını oku.

Tablo 1



Set tipi komutlar

SetKAPH

- | | |
|---------------------------------------|--|
| 0x5A 0x00, 0x00, [D2], [D1], [D0], CS | 1. Motor için hareketli durum parametrelerini yükler |
| 0x5A 0x01, 0x00, [D2], [D1], [D0], CS | 2. Motor için hareketli durum parametrelerini yükler |
| 0x5A 0x02, 0x00, [D2], [D1], [D0], CS | 3. Motor için hareketli durum parametrelerini yükler |

[D2] 8 bitlik K parametresi

[D1] 8 bitlik A parametresi

[D0] 8 bitlik B parametresi

Bu komut K, A, B parametrelerini yükler. Bu parametreler servo düzeneğin davranış şeklini belirler. SetKABH komutu ile gönderilen parametreler aynı zamanda SetKABD komutuna ait parametrelerin de üzerine yazılır.

K parametresinin değeri artırılırsa cevap zamanı kısalır ve motorun hız profiline göre hareketi esnasındaki hatası azalır. K parametresinin artması servo sistemi kararsızlığa yaklaştırır. Kararsızlık servo sistemin osilasyona girmesi demektir. Motor bir sağa bir sola hızlıca dönerek yada titreyerek mekanizmanın hasara uğramasına neden olabilir.

K parametresi sıfır verilirse motor asla dönemez. A parametresi servo sistemde ileri yol üzerinde türev etkisi yapar. Değeri artırdıkça kontrol tipi kazançsal kontrol dan uzaklaşıp türevsel kontrole yaklaşır. B parametresi de servo sistem üzerinde A gibi türevsel etkide bulunur. Değeri arttıkça türevsel etki artar. B parametresi takometrik davranışa neden olur.

K, A ve B parametreleri Z1100 - Ayar programı ile deneysel olarak kolayca bulunabilir. Z1100 - Ayar programını www.cncdesigner.com/Z1100/Programs/Tune.zip adresinden indirebilirsiniz.

GetKABH

- | | |
|--|-----------------------------------|
| 0x5A, 0x80, 0x00, 0x00, 0x00, 0x00, 0xC0 | 1. Motorun KABH değerlerini okur. |
| 0x5A, 0x81, 0x00, 0x00, 0x00, 0x00, 0xC0 | 2. Motorun KABH değerlerini okur. |
| 0x5A, 0x82, 0x00, 0x00, 0x00, 0x00, 0xC0 | 3. Motorun KABH değerlerini okur. |

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

K = D2, A = D1 ve B = D0 olarak öğrenilmiş olur.



SetKAPD

0x5A 0x04, 0x00, [D2], [D1], [D0], CS	1. Motor için durağan durum parametrelerini yükler
0x5A 0x05, 0x00, [D2], [D1], [D0], CS	2. Motor için durağan durum parametrelerini yükler
0x5A 0x06, 0x00, [D2], [D1], [D0], CS	3. Motor için durağan durum parametrelerini yükler

[D2] 8 bitlik K parametresi

[D1] 8 bitlik A parametresi

[D0] 8 bitlik B parametresi

Bu komut ile yüklenen K, A, B parametreleri motor hareketsiz iken kullanılır.

Durağan durumdaki parametreler Hareketli durum parametrelerinden küçük seçilmelidir.

SetKABD komutu daima SetKABH komutundan sonra uygulanmalıdır. Aksi takdirde SetKABH değerleri SetKABD değerlerini ezer.

GetKABD

0x5A, 0x84, 0x00, 0x00, 0x00, 0x00, 0xC0	1. Motorun KABD değerlerini okur.
0x5A, 0x85, 0x00, 0x00, 0x00, 0x00, 0xC0	2. Motorun KABD değerlerini okur.
0x5A, 0x86, 0x00, 0x00, 0x00, 0x00, 0xC0	3. Motorun KABD değerlerini okur.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

K = D2, A = D1 ve B = D0 olarak öğrenilmiş olur.



SetVmax

- | | |
|---------------------------------------|---------------------------------------|
| 0x5A 0x08, [D3], [D2], [D1], [D0], CS | 1. Motor için max hız değerini yükler |
| 0x5A 0x09, [D3], [D2], [D1], [D0], CS | 2. Motor için max hız değerini yükler |
| 0x5A 0x0A, [D3], [D2], [D1], [D0], CS | 3. Motor için max hız değerini yükler |

[D3] = (Vmax >>24)

[D2] = (Vmax >>16) & 0xFF

[D1] = (Vmax >>8) & 0xFF

[D0] = (Vmax & 0xFF)

Vmax, ulaşılabilecek sayım/saniye cinsinden max hız değeridir. Hedef pozisyona ilerleme hızını belirler.

Vmax = 0 yapılırsa motor asla dönemez.

Bu komut yasaklı bir komut olduğu için sadece durmakta olan motorlar için kullanılabilir. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız).

32 bitlik parametre daima pozitif olmalıdır.

GetVmax

- | | |
|--|--------------------------------|
| 0x5A, 0x88, 0x00, 0x00, 0x00, 0x00, 0xBC | 1. Motorun Vmax değerini okur. |
| 0x5A, 0x89, 0x00, 0x00, 0x00, 0x00, 0xBC | 2. Motorun Vmax değerini okur. |
| 0x5A, 0x8A, 0x00, 0x00, 0x00, 0x00, 0xBC | 3. Motorun Vmax değerini okur. |

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, D3, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

Vmax = (D3<<24) | (D2<<16) | (D1<<8) | D0; olarak öğrenilmiş olur.



SetAcc

- 0x5A 0x0C, [D3], [D2], [D1], [D0], CS 1. Motor için hızlanma ve yavaşlama ivmesini yükler
0x5A 0x0D, [D3], [D2], [D1], [D0], CS 2. Motor için hızlanma ve yavaşlama ivmesini yükler
0x5A 0x0E, [D3], [D2], [D1], [D0], CS 3. Motor için hızlanma ve yavaşlama ivmesini yükler

[D3] = (ivme >>24)

[D2] = (ivme >>16) & 0xFF

[D1] = (ivme >>8) & 0xFF

[D0] = (ivme & 0xFF)

Hız profilinde hızlanma değerini belirleyen ve sayım/saniye² cinsinden ivme değeridir.

İvme = 0 yapılırsa motor asla dönemez.

Bu komut yasaklı bir komuttur ve sadece durmakta olan motorlar için kullanılabilir. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız). Değeri artırıldığında motor çok çabuk hızlanır.

GetAcc

- 0x5A, 0x8C, 0x00, 0x00, 0x00, 0x00, 0xBC 1. Motorun ivme değerini okur.
0x5A, 0x8D, 0x00, 0x00, 0x00, 0x00, 0xBC 2. Motorun ivme değerini okur.
0x5A, 0x8E, 0x00, 0x00, 0x00, 0x00, 0xBC 3. Motorun ivme değerini okur.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, D3, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

İvme = (D3<<24) | (D2<<16) | (D1<<8) | D0; olarak öğrenilmiş olur.



SetStm

0x5A 0x10, 0x00, 0x00, 0x00, [D0], CS 1. Motor için örnekleme zamanını yükler
0x5A 0x11, 0x00, 0x00, 0x00, [D0], CS 2. Motor için örnekleme zamanını yükler
0x5A 0x12, 0x00, 0x00, 0x00, [D0], CS 3. Motor için örnekleme zamanını yükler

[D0] örnekleme zamanı ile ilgili katsayı

Örnekleme Zamanı D0 x 32 mikro saniye olarak tanımlanır.

İlk değer olarak D0=1 tanımlıdır.

GetStm

0x5A, 0x90, 0x00, 0x00, 0x00, 0x00, 0xBC 1. Motor için örnekleme zamanını okur
0x5A, 0x91, 0x00, 0x00, 0x00, 0x00, 0xBC 2. Motor için örnekleme zamanını okur
0x5A, 0x92, 0x00, 0x00, 0x00, 0x00, 0xBC 3. Motor için örnekleme zamanını okur



SetHvm

- | | |
|---------------------------------------|--|
| 0x5A 0x14, [D3], [D2], [D1], [D0], CS | 1. Motor için home pozisyonuna hareket hızını yükler |
| 0x5A 0x15, [D3], [D2], [D1], [D0], CS | 2. Motor için home pozisyonuna hareket hızını yükler |
| 0x5A 0x16, [D3], [D2], [D1], [D0], CS | 3. Motor için home pozisyonuna hareket hızını yükler |

[D3] = (HomeVmax >>24)

[D2] = (HomeVmax >>16)

[D1] = (HomeVmax >>8)

[D0] = (HomeVmax & 0xFF)

Home pozisyonuna ilerlerken ulaşılmasına izin verilecek max hız değeridir. Bu komut motor harketsizken verilmelidir.

GetHvm

- | | |
|--|--|
| 0x5A, 0x94, [D3], [D2], [D1], [D0], CS | 1. Motor için home pozisyonuna hareket hızını okur |
| 0x5A, 0x95, [D3], [D2], [D1], [D0], CS | 2. Motor için home pozisyonuna hareket hızını okur |
| 0x5A, 0x96, [D3], [D2], [D1], [D0], CS | 3. Motor için home pozisyonuna hareket hızını okur |

HomeVmax = [D3] <<24 + D2<<16 + [D1] <<8 + [D0]



SetHac

- | | |
|---------------------------------------|--|
| 0x5A 0x18, [D3], [D2], [D1], [D0], CS | 1. Motor için home pozisyonuna hareket ivmesini yükler |
| 0x5A 0x19, [D3], [D2], [D1], [D0], CS | 2. Motor için home pozisyonuna hareket ivmesini yükler |
| 0x5A 0x1A, [D3], [D2], [D1], [D0], CS | 3. Motor için home pozisyonuna hareket ivmesini yükler |

[D3] = (HomeAcc >>24)

[D2] = (HomeAcc >>16)

[D1] = (HomeAcc >>8)

[D0] = (HomeAcc & 0xFF)

Home pozisyonuna ilerlerken ivmlenme değeridir. Bu komut motor harketsizken verilmelidir.

GetHac

- | | |
|--|--|
| 0x5A, 0x98, [D3], [D2], [D1], [D0], CS | 1. Motor için home pozisyonuna harehet hızını okur |
| 0x5A, 0x99, [D3], [D2], [D1], [D0], CS | 2. Motor için home pozisyonuna harehet hızını okur |
| 0x5A, 0x9A, [D3], [D2], [D1], [D0], CS | 3. Motor için home pozisyonuna harehet hızını okur |

HomeAcc = [D3] <<24 + D2<<16 + [D1] <<8 + [D0]



SetLimit

0x5A 0x1C, 0x00, 0x00, [D1], [D0], CS 1. Motor için zorlanma değerini yükler
0x5A 0x1D, 0x00, 0x00, [D1], [D0], CS 2. Motor için zorlanma değerini yükler
0x5A 0x1E, 0x00, 0x00, [D1], [D0], CS 3. Motor için zorlanma değerini yükler

[D1] = (Max Hata >>8)

[D0] = (Max Hata & 0xFF)

Motor milinin dönmesi engellenirse encoder sayımları yavaşlar hatta durur ve motorun olması gereken pozisyon ile gerçek pozisyonu arasında hata oluşmaya başlar. Bu hata, Vmax değerine ulaşırsa sistem korumaya geçer. Vmax için 0 ... 16383 arasında değer verilebilir. Çok küçük değerler verilirse sistem sık sık korumaya geçebilir.

Motorlardan hangisinin zorlanıp durduğu GetStop komutu ile öğrenilebilir. Korumaya geçen motora ait hata SetClr komutu ile silinebilir. SetStart ile yeniden başlatılırsa da o motora ait hata silinebilir.

GetLimit

0x5A, 0x9C, 0x00, 0x00, 0x00, 0x00, 0xBC 1. Motor için
0x5A, 0x9D, 0x00, 0x00, 0x00, 0x00, 0xBC 2. Motor için
0x5A, 0x9E, 0x00, 0x00, 0x00, 0x00, 0xBC 3. Motor için

Zorlanma limiti ve sıfır pozisyonuna ilerleme hız parametrelerini okur.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, D3, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

ErrMax = (D2<<8) | D0;

HomeSpeed = (D1<<8) | D0; olarak öğrenilmiş olur.



Z 1100 DC motor servo kontrol çipi

SetCab

0x5A 0x20, 0x00, 0x00, 0x00, [D0], CS 1. Motor için kablo yönlerini tanımlar
0x5A 0x21, 0x00, 0x00, 0x00, [D0], CS 2. Motor için kablo yönlerini tanımlar
0x5A 0x22, 0x00, 0x00, 0x00, [D0], CS 3. Motor için kablo yönlerini tanımlar

D0 0. Bit
0 ise Motor kabloları doğru.
1 ise Motor kabloları ters.

D0 1. Bit
0 ise Encoder1 ChA ve ChB bağlantısı doğru.
1 ise Encoder1 ChA ve ChB bağlantısı ters.

D0 2. Bit
0 ise Home switche basınca çip girişi 0 oluyor.
1 ise Home switche basınca çip girişi 1 oluyor.

D0 Diğer bitler kullanılmıyor.

Mekanizma home switchden uzaklaşacak şekilde itildiğinde GetEncoder ile alınan pozisyon değerleri pozitif yönde artan değerler olmalıdır. Eğer tam tersine negatif değerler alıyorsa encoderin A ve B kabloları ters bağlanmış demektir. Bu durumda kabloların bağlantısına dokunmadan D0.1 biti 1 yapılırsa bu sorun çözülür.

Mekanizma orta pozisyonuna yakın bir noktada beklemekteyken SetKAB, SetVmax, SetAcc komutları ile parametreler yüklenip SetTarget ile 0 pozisyonu gönderilir ardından da SetStart komutu işletilirse mekanizmanın home switche doğru ilerlemesi gerekir. Tam tersine homeswitchden uzaklaşıyorsa motor kabloları sürücüye ters bağlanmış demektir. Kabloları değiştirmeden D0.0 biti 1 yapılırsa bu sorun çözülür.

Z1100 Home Switch pini, switche basıldığında 0, basılmadığında 1 olmalıdır. Eğer switch düzeneğiniz basma durumunda 1, basmama durumunda 0 bilgisi veriyorsa D0.2 bitini 1 yapmalısınız.

Bu komut da yasaklı bir komuttur ve sadece durmakta olan motorlar için kullanılabilir. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız).

GetCab

0x5A 0xA0, 0x00, 0x00, 0x00, 0x00, CS 1. Motor için
0x5A 0xA1, 0x00, 0x00, 0x00, 0x00, CS 2. Motor için
0x5A 0xA2, 0x00, 0x00, 0x00, 0x00, CS 3. Motor için

Motor ve Encoder kablo tanımlamaları ile Switch tipini okur.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.



Z 1100 DC motor servo kontrol ipi

Hata Kodu, 0x00, 0x00, 0x00, D0, CS

- D0 0. Bit
0 ise Motor kabloları doęru.
1 ise Motor kabloları ters.
- D0 1. Bit
0 ise Encoder1 ChA ve ChB baęlantısı doęru.
1 ise Encoder1 ChA ve ChB baęlantısı ters.
- D0 2. Bit
0 ise Home switche basınca ip giriři 0 oluyor.
1 ise Home switche basınca ip giriři 1 oluyor.
- D0 Dięer bitler kullanılmıyor.



SetJob

0x5A 0x24, 0x00, 0x00, 0x00, [D0], CS 1. Motorun görev tanımı
0x5A 0x25, 0x00, 0x00, 0x00, [D0], CS 2. Motorun görev tanımı
0x5A 0x26, 0x00, 0x00, 0x00, [D0], CS 3. Motorun görev tanımı

D0 = 0x00 Motor mili serbest
D0 = 0x01 Tanımsız
D0 = 0x02 Hedef pozisyona git
D0 = 0x03 Sabit hızda sürekli dön

Bu komut uygulandığında daha önceki hatalar silinir.

Sabit hız ile dönme modunda dönüş yönü bilgisi SetStart komutunda tanımlanır.

Bu komut yasaklı bir komuttur ve sadece durmakta olan motorlar için kullanılabilir *. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız).

* Ancak dönmekte olan motora 0x00 görevi (Mili serbest bırak) verilebilir. Bu durumda dönmekte olan motorun enerjisi kesilir.

GetJob

0x5A 0xA4, 0x00, 0x00, 0x00, [D0], CS 1. Motorun görev tanımı
0x5A 0xA5, 0x00, 0x00, 0x00, [D0], CS 2. Motorun görev tanımı
0x5A 0xA6, 0x00, 0x00, 0x00, [D0], CS 3. Motorun görev tanımı

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, 0x00, 0x00, [D0], CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

D0 içinde motor görevi öğrenilir.

D0 = 0x00 Motor mili serbest
D0 = 0x01 Tanımsız
D0 = 0x02 Hedef pozisyona git
D0 = 0x03 Sabit hızda sürekli dön



SetBusy

0x5A 0x28, 0x00, 0x00, 0x00, [D0], CS 1. Motorun busy çıkışına etki tanımı
0x5A 0x29, 0x00, 0x00, 0x00, [D0], CS 2. Motorun busy çıkışına etki tanımı
0x5A 0x2A, 0x00, 0x00, 0x00, [D0], CS 3. Motorun busy çıkışına etki tanımı

D0 0 nolu bit 0 ise Motor busy çıkışına müdahalede bulunmaz.

D0 0 nolu bit 1 ise Motor busy çıkışına müdahalede bulunur.

0 nolu bit 1 yapıldığı takdirde motor harekete başladığında Busy pini 0 olur. Motor hareketini bitirdiğinde Busy pinini 1 yapmak ister. Ancak bu esnada diğer motorların duruyor olması ya da Busy tanımlamalarının yapılmamış olması gerekir.

Tüm motorlar için de Busy biti tanımlandı ise bu durumda tüm motorlar durmadan Busy pini 1 olamaz.

GetBusy

0x5A 0xA8, 0x00, 0x00, 0x00, [D0], CS 1. Motorun busy bit tanımı
0x5A 0xA9, 0x00, 0x00, 0x00, [D0], CS 2. Motorun busy bit tanımı
0x5A 0xAA, 0x00, 0x00, 0x00, [D0], CS 3. Motorun busy bit tanımı

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, 0x00, 0x00, [D0], CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

D0 içinde motor görevi öğrenilir.

D0 da 0. Bit bu motor için tanımlanmış busy değerini bildirir.

Busy pini işlemci ile doğrudan 1 yada 0 olarak donanımsal olarak okunur. Eğer herhangi bir motorun dönüp dönmediğini sorgulayarak öğrenmek isterseniz GetStart komutu gönderip motorun Start bitinin 1 olup olmadığına da bakabilirsiniz. Start edilmiş motor meşgul anlamındadır.



SetTarget

0x5A 0x2C, [D3], [D2], [D1], [D0], CS 1. Motor için hedef
0x5A 0x2D, [D3], [D2], [D1], [D0], CS 2. Motor için hedef
0x5A 0x2E, [D3], [D2], [D1], [D0], CS 3. Motor için hedef

[D3] = (Hedef >>24)

[D2] = (Hedef >>16) & 0xFF

[D1] = (Hedef >>8) & 0xFF

[D0] = (Hedef & 0xFF)

Motorun ulaşması istenen sayım cinsinden pozisyon değeridir. Sistem SetVel komutu ile hız servosuna dönüştürülürse bu durumda SetTarget komutu bir anlam ifade etmez.

Bu komut yasaklı bir komuttur ve sadece durmakta olan motorlar için kullanılabilir. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız).

GetTarget

0x5A 0xAC, 0x00, 0x00, 0x00, 0x00, CS 1. Motor için
0x5A 0xAD, 0x00, 0x00, 0x00, 0x00, CS 2. Motor için
0x5A 0xAE, 0x00, 0x00, 0x00, 0x00, CS 3. Motor için

Bu komut, daha önce SetTarget komutu ile bildirilmiş hedef değerini ister.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, D3, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

Target = (D3<<24) | (D2<<16) | (D1<<8) | D0; olarak öğrenilmiş olur.



SetStart

Bu komut her 3 motorun start parametrelerini içerir. İlgili bitlerden 0 olanlar anlam ifade etmez.

0x5A 0x30, 0x00, 0x00, 0x00, [D0], CS

D0 0. Bit 1 yapılırsa 1. Motor çalıştır.

D0 1. Bit 1 yapılırsa 2. Motor çalıştır.

D0 2. Bit 1 yapılırsa 3. Motor çalıştır.

D0 3. Bit Kullanılmıyor.

D0 4. Bit Hız servo modunda 2. Motorun dönüş yönü: 1 ise bir yöne, 0 ise aksi yöne.

D0 5. Bit Hız servo modunda 3. Motorun dönüş yönü: 1 ise bir yöne, 0 ise aksi yöne.

D0 6. Bit Hız servo modunda 3. Motorun dönüş yönü: 1 ise bir yöne, 0 ise aksi yöne.

D0 7. Bit Kullanılmıyor.

Çalışması istenen motorlar için bu komut yollanmadan önce K, A, B, Vmax, İvme, MaxErr, Home Speed parametreleri sıfırdan farklı olacak şekilde yüklenmiş olmalıdır.

0..3 nolu bitlere sadece 1 yazılabilir. Motor hareketi tamamlandığında ilgili bit otomatik olarak silinir.

4..7 nolu bitler sadece hız servo modunda motor dönüş yönü için kullanılır. 0 ya da 1 olarak değer verilebilir.

Bu komut yasaklı bir komuttur ve sadece durmakta olan motorlar için kullanılabilir. Bir motor dönerken bu komut yollanırsa hata cevabı alınır (Sayfa 10'a bakınız).

GetStart

0x5A 0xB0, 0x00, 0x00, 0x00, 0x00, CS

Bu komut, hangi motorların dönmekte olduğunu öğrenir.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, 0x00, 0x00, D0, CS

Hata Kodu 0x00 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

D0 0. Bit 1 ise 2. Motor dönüyor.

D0 1. Bit 1 ise 3. Motor dönüyor.

D0 2. Bit 1 ise 3. Motor dönüyor.



Z 1100 DC motor servo kontrol ipi

D0 3. Bit kullanılmıyor.

D0 4. Bit Sabit hız modunda 2. Motorun dönüş yönü.

D0 5. Bit Sabit hız modunda 3. Motorun dönüş yönü.

D0 6. Bit Sabit hız modunda 2. Motorun dönüş yönü.

D0 7. Bit kullanılmıyor.



SetStop

0x5A 0x34, 0x00, 0x00, 0x00, [D0], CS

Bu komut her 3 motorun stop parametrelerini içerir. İlgili bitlerden 0 olanlar anlam ifade etmez.

D0 0. Bit 1 yapılırsa 1 nolu motor yavaşlayarak durur.

D0 1. Bit 1 yapılırsa 2 nolu motor yavaşlayarak durur.

D0 2. Bit 1 yapılırsa 3 nolu motor yavaşlayarak durur.

D0 3..7 Bitler kullanılmıyor.

Bu komut sadece hız servo modunda dönen motorları durdurmak için kullanılır. Diğer görevlerde çalışan motorlar için etkisi yoktur.

GetStop

0x5A 0xB4, 0x00, 0x00, 0x00, [D0], CS

Sıkışma nedeniyle korumaya alınan motorun kimlik bilgisi bu komut ile öğrenilir. Ayrıca durmakta olan motorlar da gene bu komut ile öğrenilebilir. Gelen veriler aşağıdaki gibidir.

Hata Kodu, 0x00, 0x00, 0x00, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

D0 0.Bit 1 ise komut ile 1 nolu motor durdu.

D0 1.Bit 1 ise komut ile 2 nolu motor durdu.

D0 2.Bit 1 ise komut ile 3 nolu motor durdu.

D0 3.Bit Kullanılmıyor

D0 4.Bit 1 ise 2. Motor sıkıştığı için durduruldu.

D0 5.Bit 1 ise hata nedeniyle 2 nolu motor durduruldu.

D0 6.Bit 1 ise hata nedeniyle 3 nolu motor durduruldu.

D0 7.Bit Kullanılmıyor

Bir motor hata limitini aştığında durur ve bir komut gönderildiğinde alınacak cevaba ait Hata kodunda bu rapor edilir. Bu hata silinmediği sürece kalır. SetClr komutu ile ya da SetStart komutu ile hata vermiş motor başlatılırsa da bu hata silinir.



SetEstop

0x5A 0x38, 0x00, 0x00, 0x00, [D0], CS

Bu komut her 3 motorun stop parametrelerini içerir. İlgili bitlerden 0 olanlar anlam ifade etmez.

D0 0. Bit 1 yapılırsa 1 nolu motor derhal durur.

D0 1. Bit 1 yapılırsa 2 nolu motor derhal durur.

D0 2. Bit 1 yapılırsa 3 nolu motor derhal durur.

D0 3..7 Bitler kullanılmıyor.

Bu komut dönmekte olan motoru en kısa sürede durdurmaya çalışır ve mekanizmada büyük sarsıntıya neden olur.

GetEstop

0x5A 0xB8, 0x00, 0x00, [D1], [D0], CS 1. Motor pozisyon hatası

Hata Kodu, 0x00, 0x00, 0x00, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

D0 0.Bit 1 ise komut ile 1 nolu motor durdu.

D0 1.Bit 1 ise komut ile 2 nolu motor durdu.

D0 2.Bit 1 ise komut ile 3 nolu motor durdu.

D0 3.Bit Kullanılmıyor

D0 4.Bit 1 ise 2. Motor sıkıştığı için durduruldu.

D0 5.Bit 1 ise hata nedeniyle 2 nolu motor durduruldu.

D0 6.Bit 1 ise hata nedeniyle 3 nolu motor durduruldu.

D0 7.Bit Kullanılmıyor

Bir motor hata limitini aştığında durur ve bir komut gönderildiğinde alınacak cevaba ait Hata kodunda bu rapor edilir. Bu hata silinmediği sürece kalır. SetClr komutu ile ya da SetStart komutu ile hata vermiş motor başlatılırsa da bu hata silinir.



Z 1100 DC motor servo kontrol çipi

SetClr

0x5A 0x3C, 0x00, 0x00, 0x00, 0x00, CS

Zorlandığı için durdurulan motorun üretmiş olduğu hatayı siler (Sayfa 10).

SetZero

0x5A 0x40, 0x00, 0x00, 0x00, 0x00, CS 1. Motoru Home pozisyonuna yollar

0x5A 0x41, 0x00, 0x00, 0x00, 0x00, CS 2. Motoru Home pozisyonuna yollar

0x5A 0x42, 0x00, 0x00, 0x00, 0x00, CS 3. Motoru Home pozisyonuna yollar

SetBaud

0x5A 0x44, [D3], [D2], [D1], [D0], CS

Haberleşme hızını değiştirir.

SetFlash

0x5A 0x48, 0x00, 0x00, 0x00, 0x00, CS

Set komutları ile tanımlanmış parametreleri Flash Roma yazar. Çip çalıştırıldığında Flash Roma yüklenmiş bu parametreleri okur.



GetEncoder

0x5A 0xBC, 0x00, 0x00, 0x00, [D0], CS 0. Encoder pozisyonu
0x5A 0xBD, 0x00, 0x00, 0x00, [D0], CS 1. Encoder pozisyonu
0x5A 0xBE, 0x00, 0x00, 0x00, [D0], CS 2. Encoder pozisyonu

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, D3, D2, D1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

Encoder Pozisyonu = (D3<<24) | (D2<<16) | (D1<<8) | D0; olarak öğrenilmiş olur.

GetError

0x5A 0xC0, 0x00, 0x00, [D1], [D0], CS 1. Motor pozisyon hatası
0x5A 0xC1, 0x00, 0x00, [D1], [D0], CS 2. Motor pozisyon hatası
0x5A 0xC2, 0x00, 0x00, [D1], [D0], CS 3. Motor pozisyon hatası

Bu komut, anlık pozisyon hatasını ister. (Figure 3'e bakabilirsiniz)

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, 0x00, 0xD1, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

Error = (D1 << 8) + D0; olarak öğrenilmiş olur.

GetSwitch

0x5A 0xC0, 0x00, 0x00, 0x00, [D0], CS 1. Motor pozisyon home switch durumu
0x5A 0xC1, 0x00, 0x00, 0x00, [D0], CS 2. Motor pozisyon home switch durumu
0x5A 0xC2, 0x00, 0x00, 0x00, [D0], CS 3. Motor pozisyon home switch durumu

Bu komut, home switch konumunu okur.

Cevap olarak ilk önce Hata kodu olmak üzere aşağıdaki paket alınır.

Hata Kodu, 0x00, 0x00, 0x00, D0, CS

Hata Kodu 0x00 yada 0x80 olarak alındıysa ve gelen tüm verilerin toplamı 0x00 ise

Switch = D0, 0 yada 1 olarak öğrenilmiş olur.



Z1100 - Elektriksel parametreler

Z1100 ipi STM32F103C8 iřlemcisine servo uygulamasına zel program yklenerek elde edilmiřtir.

Besleme amalı kullanılacak 3.3v reglatrn 100mA akımı verebilmesi yeterlidir.

Elektriksel parametreler iin STM32F103C8 dokmanlarına mrecat edebilirsiniz.



Z1100 - Ayar programı

Z1100 ipi ile kararlı ve hızlı bir servo dzeneęi yapmak iin K, A ve B parametrelerinin dzgn ayarlanması gerekir. Z1100 - Tune bu parametrelerin kolayca deęiřtirilmesini ve motor tepkisinin gzlenmesini saęlar.

Programı www.cncdesigner.com/Z1100/Programs/Tune.zip adresinden indirebilirsiniz.



Z1100 – ip yazılımını gncelleme

Z1100 ASIC iptir. oęu ASIC ipten farklı olarak seri port zerinden yeniden programlanabilir.

Z1100 ipi iindeki kontrol yazılımında hata tespit eder ve rapor ederseniz kısa bir sre sonra dzeltilmiř yazılımı www.cncdesigner.com/Z1100/Updates adresinden indirip ipin iine atabilirsiniz.

Bunun iin interface portuna RS232 modl baęlayıp PC'ye baęlamanız yeterlidir.

Bu amala www.cncdesigner.com/Z1100/Programs/Load.zip programını kullanabilirsiniz.

Uyarı:

Program yklemek iin Load programı yerine JTAG cihazı ile ipe baęlanıp yazma ya da okuma korumalarını kaldırırsanız ip silinir ve bir daha program ykleyemezsiniz.



Z1100 – Yazılımcılar için ipuları

ipin resetlenip resetlenmedięini nasıl anlarım?

Baskı devre izim kurallarına uyulmadan izilmiş baskı devrelerde endüktif etkiye sahip motorların alışıp durmaları esnasında oluşan EMI baskı devredeki Z1100 ipinin resetlenmesine neden olabilir.

ipin beklenen davranış sergilememesi halinde ipin istem dışı resetlenip parametrelerini silmesinden Őüphelenmelisiniz.

Herhangi bir motorun dönebilmesi için K, Vmax, İvme gibi parametrelerinin sıfırdan farklı olarak yüklenmesi gerekir. Bu durumda bu parametreleri yükledięiniz halde GetKAB, GetVmax, GetAcc komutları sıfır deęerini döndürüyorsa ip sizin isteęiniz dışında resetlenmiş demektir.

Not: Z1100 – Tune ve Z1100 – Load programlarının kullanım açıklamaları. Bu dokümana eklenecektir.



Z 1100 DC motor servo kontrol ipi

Z1100 – rnek program

Daha sonra eklenecektir.