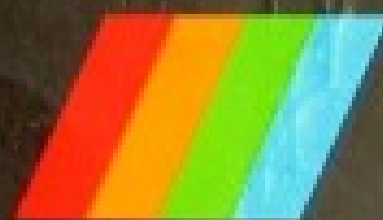


**sinclair**

# **ZX Spectrum+**

## **El Kitabı**



**SPECTRUM YAZILIMLARI**  
Spectrum bilgisayarları için  
kullanılabilen yazılımların tümü yeni  
ZX Spectrum+ 'unuza da uyar.

## ZX SPECTRUM+'u SUNARKEN

Sinclair Research, herkese bilgisayar kullanma olanağı getirmiş olan mikroyonga teknolojisinde uzun zamandır önde gidiyor. Dünyanın ilk ucuz mikrobilgisayan ZX80'i piyasaya sunduktan sonra, bunu giderek gücü ve değeri artan modellerle, ZX81, ZX Spectrum ve QL bilgisayarlarıyla sürdürdük. Bunların tasarımı da, çalıştırılmasında da parolamız, kolay kullanım olmuştur. ZX Spectrum+, bu alanda bizi bir adım daha ileri götürüyor. Bunda Spectrum'un en iyi yanları alınmış, daha da geliştirilmiş ve böylece bilgisayarların en güçlüsü, en popüler ve kullanımı en kolayı yaratılmıştır. Bu yeni bilgisayarın size sunduğu geniş olanakları değerlendireceğinizi umuyoruz.

*Chris Smith*

## İÇİNDEKİLER

### İŞE KOYULALIM 3

### PROGRAMLAMAYA BAŞLAYALIM 17

### ZX SPECTRUM+'U TANIYALIM 41

### SINCLAIR BASIC'I ÖĞRENELİM 49

Yazan Neil Ardley  
Yayınlayan Dorling Kindersley Ltd  
Sinclair Research Ltd'in işbirliğiyle

## BU KİTABIN KULLANIMI

ZX Spectrum+'unuzun kılavuzu, aynı aynı renklerde düzenlenmiş dört bölümden oluşuyor. Bir bölümü bulmak için, o rengi taşıyan sayfaları açmak yeterlidir.

## 1 İŞE KOYULALIM

ZX Spectrum+'unuzun çalışmaya hazırlanışı ■  
Televizyonunuzun ayarlanışı ■ Hazırlamada karşılaşılan sorunlar ■ ZX Spectrum+ neler yapabilir ■ Hazır yazılım nasıl kullanılır ■ Program nasıl yüklenir ■ Yazılım yüklemadaki sorunlar

## 2 PROGRAMLAMAYA BAŞLAYALIM

Klavye – bilgisayarınızın komuta kontrol tablosu ■  
Tuşlar nasıl çalışır ■ Televizyon hesap makinası ■  
Renkler ve kullanımı ■ Basit bazı grafikler ■ Resim defteri olan ekran ■ Desen ve resim çizimleri ■  
Bilgisayar karakterlerinin yarabılışı ■ Hareketli resimler ■ Müzik ve çeşitli sesler ■ Programlarınızı nasıl saklayabilirsiniz ■ Program saklamada karşılaşılan sorunlar

## 3 ZX SPECTRUM+'u TANIYALIM

## 4 SINCLAIR BASIC'I ÖĞRENELİM

Sinclair BASIC komutları için programcının başvuru kılavuzu ■ Spectrum ekranındaki mesajlar ■ BASIC'in ötesi ■ Bilgisayar jargonu – ne anlama geliyor.

## İŞE KOYULALIM

Bu bölüm size, elinizdeki ZX Spectrum+'un geniş olanaklarını araştırmaya nasıl başlayabileceğinizi gösterecek. Bu arada, bilgisayarınızı nasıl istendiği anda çalışmaya hazır durumda tutabileceğinizi de öğreneceksiniz. Burada bir seçim yapma durumundasınız. Ya Spectrum+'unuzun yeteneklerinden yararlanıp, renkli grafiklerini ve çıkarabileceği sesleri kullanarak birkaç program hazırlayabilirsiniz ya da bilgisayar oyunları gibi bazı hazır yazılımları nasıl kullanacağını öğrenebilirsiniz. Seçiminizi ne yönde yaparsanız yapın, kısa zamanda bilgisayarınızı kullanmanın keyfine varacaksınız.



## ZX SPECTRUM+'UN ÇALIŞMAYA HAZIRLANIŞI

Spectrum'unuzu çalıştırmak için, önce aşağıdaki listeyi gözden geçirin, gereksinim duyacağınız bütün gereçlerin bulunup bulunmadığını kontrol edin, sonra da karşı

sayfadaki açıklamaları izleyerek, bağlantıları yapın, bilgisayarınızı da fişe takın. Bu aşamada teypte gereksiniminiz yok.

Bağlantıları iyice yapıp yapmadığınızı kontrol edin. Çünkü Spectrum'unuzu kullanırken, kaza ile bağlantıları yerinden çıkarırsanız, ya da cereyan kesilirse, programınız, Spectrum'daki bilgiler veya sonuçlar silinir.

Bilgisayarla işiniz bitince, fişi prizden çekin.

### Listeniz: Kullanacağınız gereçler tamam mı?

Ambalajı açınca, bulacaklarınız:

1. ZX Spectrum+
2. ZX Adaptör/ Transformator - Spectrum için gerekli 9 voltluk doğrudan akımı sağlar.
3. Anten bağlantısı - Spectrum'unuzu televizyona bağlar.
4. Teyp bağlantısı - Spectrum'unuzu teypte bağlar.
5. Garanti belgesi - Bunu hemen doldurup, postalamana gerekir.
6. Kullanım Kılavuzuna Ek Kaset.
7. Bu elkitabı.

Elinizde olması gereken diğer araçlar:

1. Bir televizyon.
2. Bir kasetli teypt.
3. Bir fiş.



### Hazırlıklara ilişkin sorular ve yanıtlar

**Televizyonumun renkli olması gerekir mi?**  
Hayır. Ancak siyah beyaz televizyonda, Spectrum'un sağlayacağı renkten göremezsiniz.

**Herhangi bir televizyon olur mu?**  
Spectrum'unuzun herhangi bir televizyonda görüntü oluşturabilmesi gerekir. Görüntü oluşmuyorsa, o zaman bilgisayarınız ve televizyonunuzun sistemleri farklı demektir. Televizyonunuz çok eskiyse, ya da televizyon ve bilgisayar farklı ülkelerden satın alınmışsa, böyle bir durumu karşılayabilirsiniz. Bu konuda bir kuşkunuz varsa, televizyon satıcınıza danışın.

**TV alıcısı yerine monitör kullanabilir miyim?**  
Evet. Satıcıdan, size Spectrum için çok daha üstün bir görüntü verecek bir monitör de alabilirsiniz.

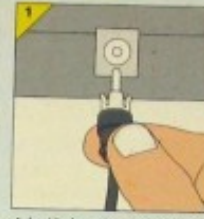
**Nasıl bir elektrik akımı gerekir?**  
Spectrum 240v/50Hz 1.4 amperlik standart akımı kullanır.

**Spectrum parazite yol açar mı?**  
Eğer radyonuz bilgisayara yakınsa, parazite neden olabilir. Ama bu durum radyoya da, bilgisayara da bir zarar vermez.

**Bununla ZX 16K RAM kullanabilir miyim?**  
Hayır. Bu RAM takımı sadece Sinclair ZX81 bilgisayar ile kullanılabilir.

### ZX Spectrum+'un uzun cereyana bağlanması

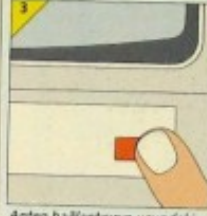
Önce adaptörden çıkan, ucu çıplak kabloyu bir fişe bağlayın. Spectrum'unuzun topraklanmasına gerek yoktur. Bundan sonra, yandaki resimleri izleyerek, Spectrum'u elektrikli ve TV alıcısına bağlayın. Bütün bağlantıları tamamladıktan sonra, nasıl ayar yapacağınızı öğrenmek için sayfayı çevirin.



Adaptörden çıkan kabloların ucundaki küçük fişi, Spectrum'da üzerinde 9VDC yazan prize takın.



Adaptörden çıkan diğer kabloların ucundaki normal fişi elektrik prize takın.



Anten bağlantısının ucundaki fişi Spectrum'da üzerinde TV yazan prize takın. Bu bağlantıdaki fişlerden sadece biri bu prize uyar.



Televizyonunuzun anten kablolarını çıkarın. Anten bağlantısındaki diğer fişi de televizyonun anten girişine takın.



Televizyonunuzu açın, ses düğmesini sonuna kadar kısın. Arka televizyonunuzu Spectrum'dan gelecek sinyalleri almaya ayarlayabilecek durumdasınız.

### Spectrum fişleri ve bağlantıları



## TELEVİZYONUNUZUN AYARLANMASI

Spectrum'unuz, UHF bandının 36. kanalı üzerinden renkli televizyon video sinyalleri gönderir, bu nedenle bilgisayarın sinyallerini alabilmesi için TV alıcınız bu kanala ayarlanmalıdır.

Spectrum'unuzu ceryana ve TV alıcısına bağladığınızda, alıcınızın istasyon alıcı düğmesini, aşağıdaki ilk ekranda gördüğünüz gibi Sinclair'in copyright sinyalini alacak şekilde ayarlayın. Bunu yaptığınızda, Spectrum'un renklerini sinamaya ve aracınızı kullanmaya hazırsınız demektir. Eğer copyright görüntüsünü alamazsanız ya da renkler iyi değilse, o zaman karşı sayfadaki tablodan bağlantıları kontrol edin.



### Spectrum'un renklerinin kontrolü

Spectrum'un renklerini kontrol etmek için, önce B tuşuna sonra da 1 ile 6 arasında bir rakam tuşuna basın. Copyright görüntüsü kaybolacak, önce BORDER kelimeleri sonra da bir rakam görünecektir. ENTER yazılı tuşa basınca, ekranın,

#### BORDER 4



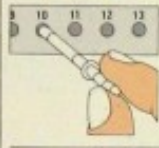
### Ayar düğmeleri

**Döner düğmeli ayar**  
Döner düğme herhangi bir kanala ayarlanabilir. Copyright sinyalini bulana kadar düğmeyi çevirin.



### Tuşlu ayar

Bilgisayar için kullanılacak bir tuş seçin, sonra da copyright sinyalini alacak şekilde ayar yapın. Eğer mümkünse bir tuşa buna ayarın. Böylece Spectrum'u çalıştırmak istediğinizde her seferinde ayar yapma zorunda kalmazsınız.



### Elektronik ayar

Bu sistemde, alıcı kendini istediğiniz kanala ayarlar. Elle ayarlanamayan, sentezle kanal seçimli TV alıcılara bu bilgisayarın kullanımı için elverişli olamaz.

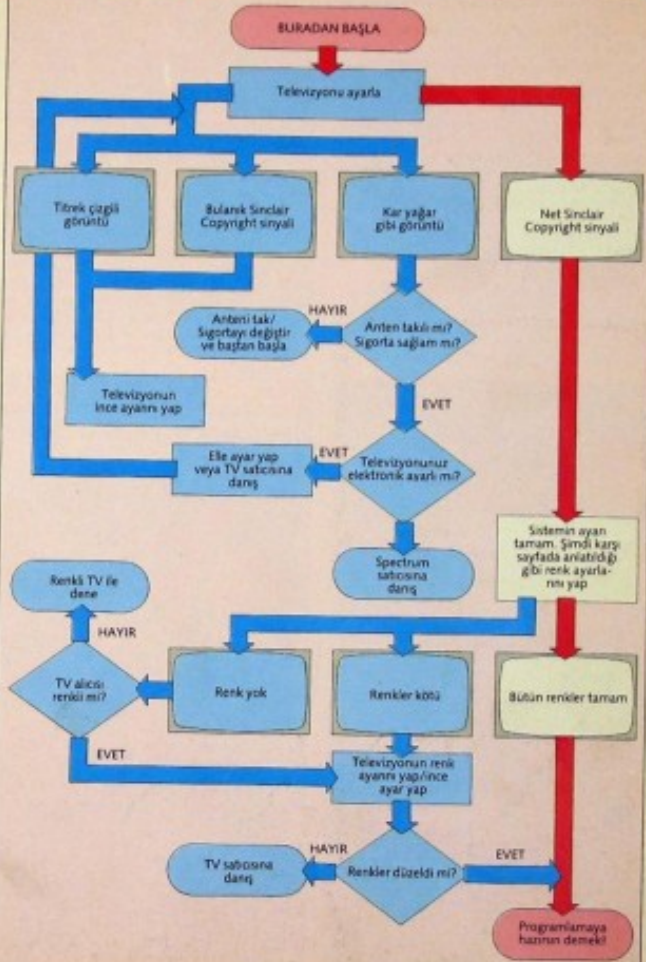


#### BORDER 3



çevresi, rakam tuşunun rengini alacaktır. Aşağıdaki ekranlar önce BORDER 4 ve ENTER, sonra da BORDER 3 ve ENTER tuşlarına basıldığınızda neler olacağını gösteriyor. BORDER 7 ise ekranın çevresini yeniden beyaz yapar.

## Hazırlamada karşılaşılan sorunlar





**Dene**  
(K tuşunu kullanarak) program sıralamasında önce PAPER 7 yazın ve ENTER'e basın. Satır 20'yi yeniden yazın ama FLASH RND komutlarını koymayın. Halkaları yazıp silmesi düzce.

## ÇILGIN MOZAIK

```

10 BORDER 0:CLS
20 LET V=15:LET Y=31
30 PRINT:PRINT (RND*3-1):LET V=
40 FOR G=1 TO 20
50 LET N=V
60 LET V=V+1
70 LET V=V+2
80 IF N=0 THEN LET N=31
90 IF N=31 THEN LET N=0
100 IF V=0 THEN LET V=31
110 IF V=31 THEN LET V=0
120 IF V=0 THEN LET V=31
130 GO TO 20

```



Renkli bir kare ekranda ileri geri hareket ederek, renkli desenler oluşturur. Programı her başlatıldığında başka bir desen çıkarılır.

## Dene

Satır 50'de 143'un yerine 42 yazınca, ortaya yıldızlar çıkar. 33'den 142'ye kadar başka rakamları da deneyebilirsiniz. Ne olacağını görmek için 51. sayfadaki tuş tablosuna bakabilirsiniz.

## Bir program nasıl yeniden başlatılır?

YILDIZLAR ÇUBUKLAR gibi bazı programlar sona erince, ekranda son satır olarak OK yazısı belirir. Bu, bütün programın bittiği anlamına gelir. Yeniden başlamak için RIRUN ve ENTER'e basmak yeterlidir.

Diğer programlar ise, ya ÇILGIN MOZAIK gibi kurup gider ya da DOĞAN GÜNEŞ gibi biter bitmez otomatik olarak yeniden başlar. Bu programları durdurmak için BREAK'e basın.



Program durana kadar ve BREAK yazısı belirene kadar tuşa basmaya devam edin. Yeniden başlatmak için RIRUN ve ENTER tuşlarına basmak yeter.

## Hatalar nasıl düzeltilir?

Yanlış bir tuşa basarsanız ya da SHIFT veya EXTEND MODE tuşlarına düzgün basamıyorsanız, endişelenmeyin. DELETE tuşuna basıldığında, son komut işaret, harf ya da rakam silinir. Daha önceden de silmek isterseniz DELETE'e basmaya devam edin.

## DELETE

Eğer bir satırda hata yapmış ve ENTER tuşuna da basmışsanız, ekranda yanlış silinen bir satır izlenir. Bu, hataya gösterir. DELETE'e basarak hata yaptığınız noktaya kadar gidin, oradan sonra satırı tamamlayıp, ENTER tuşuna basın.

Eğer belleğe hatalı satır sokarsanız, program durur, ekranın en altında hatalı satır gösteren bir rakam belirir. Bu satırın tümünü yeniden yazıp, sırasıyla ENTER, RIRUN ve ENTER'e basın. Artık programın çalışması gerekir.

## ÇOKGEN

```

10 BORDER 1:POWER 6:CLS
20 INPUT N
30 FOR I=1 TO 30 STEP 2
40 LET A=100:LET V=15
50 FOR J=1 TO 30 STEP 360/N
60 LET P=V*FAC(2*PI/N)
70 DRG A+P:FOR N=1 TO 100
80 LET N=V:LET V=V+1
90 NEXT N
100 NEXT I

```



Boş bir ekran göreceksiniz. Önce 6'ya sonra ENTER'e basın. Ekranda bir altıgen belirir. Bitince programı yeniden başlatın ve kaç kenarlı bir şekil istiyorsanız, o rakama basın.

Dene  
Satır 20'de 2'nin yerine başka bir rakam yazın. Bu rakam ne kadar büyük, çokgenlerin arası ne kadar açık olursa, desen o kadar hızlı olur.

## YILDIZLAR VE ÇUBUKLAR

```

10 ZIN 2
20 POWER 7
30 CLS
40 FOR X=28 TO 148 STEP 20
50 FOR Y=28 TO 31
60 PLOT 10,2+Y:DRG 210,0
70 NEXT Y
80 NEXT X
90 PLOT 55,28:DRG 0,131
100 PLOT 210,28:DRG 0,131
110 DRG 0,1
120 FOR X=28 TO 8 STEP 2
130 FOR Y=28 TO 31
140 PRINT AT X,2:Y:PRINT " "
150 NEXT Y
160 NEXT X
170 PRINT AT X,2:Y:PRINT " "

```



Ekranda Amerika Birleşik Devletleri bayrağı belirir.

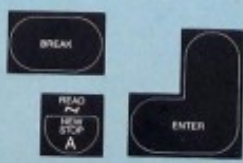
## Dene

Bayrağın renklerini değiştirin. Çubukların rengi satır 10. yıldızların satır 120. yıldızların fonu ise satır 110'dur.

## Yeni bir program nasıl başlatılır?

Bir programla iştin bilince, yerine yepyeni bir program koymak istiyorsanız, programın sona ermesini bekleyin ya da BREAK'e basarak durdurun.

Bir programı bilgisayarın belleğinden silmek için iki yol vardır. Bir A(NEW) ve ardından ENTER tuşlarına basmaktır. Ekran bir an için kararır, sonra yeniden Copyright mesajı belirir.



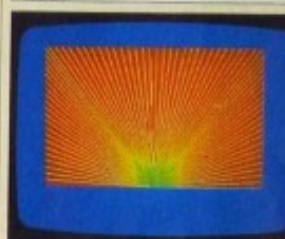
Ya da daha kolay, sadece RESET tuşuna basabilirsiniz. Bu da Spectrum'u ceryanını kesip kapatmakla aynı sonucu verir.

## DOĞAN GÜNEŞ

```

10 BORDER 0:CLS
20 FOR G=1 TO 20
30 FOR X=1 TO 10
40 CLS
50 LET Y=2+RND*10+2
60 FOR N=10 TO 274 STEP 2
70 PLOT 100,0
80 DRG 120,0
90 DRG 0,131
100 DRG 0,131
110 DRG 0,1
120 FOR X=127 TO 127 STEP 2
130 PLOT 127,0
140 DRG 127,0
150 DRG 0,131
160 DRG 0,131
170 FOR X=127 TO 0 STEP -2
180 PLOT 127,0
190 DRG 127,0
200 DRG 0,131
210 DRG 0,131
220 GO TO 10

```



Her birkaç saniyede bir, değişik renklerde doğan güneş andıran bir görüntü belirir. Eğer ekrandaki görüntü kaybolursa bekleyin. Ufuktan yeni bir güneş doğacaktır.

## Dene

Satır 210'daki 200'un yerine bir başka rakam yazarsanız, her gundüğünün ekranda kalış süresini değiştirebilirsiniz. 200, 4 saniyeye eşittir.

## Şimdi ne yapalım?

Çeşitli seçenekler var. Bu programlardan herhangi birini ileride yeniden kullanmak üzere saklamak istiyorsanız, bir kasete kaydedebilirsiniz. Bunu nasıl yapacağınızı öğrenmek için, 38. sayfayı açın. "Programlarınızı nasıl saklayabilirsiniz?" bölümünü okuyun.

Spectrum'unuzla denemeleri sürdürmek istiyorsanız, "Programlamaya başlayın" başlıklı ikinci bölümde programcılık hakkında bilgi bulabilirsiniz. Şu ana kadar, nasıl yapıldığını anlamadan bazı programları denediniz. Spectrum'un programlamasının bazı özellikleri 2. bölümde anlatılacak.

Satır aldığınız bazı bilgisayar oyunlarını denemek isterseniz, o zaman sayfayı çevirip "Hazır yazılımlar nasıl kullanılır"ı okuyun.



## HAZIR YAZILIM NASIL KULLANILIR?

Spectrum'a bir program verirken, tuşlara basarak bir dizi elektronik olarak kodlanmış sinyal üretirsiniz. Bu kodlu sinyaller, Spectrum'un belleğine girer. Bellek bunları, program işleme konusuna bilgisayar kullanabilsin diye saklar. Bu kodlu sinyaller, siz (örneğin NEW ya da RESET tuşuna basarak) bunları silmedikçe ya da Spectrum'u kapatmadıkça, bellekte kalmaya devam eder.

Ancak Spectrum'unuzu her kullanmak istediğinizde yeni bir program vermeniz zorunlu değil. Bunun yerine, bilgisayarınıza doğrudan, otomatik olarak girecek programlar içeren hazır yazılımlar da satın alabilirsiniz. Hazır yazılımlar, sizi Spectrum'u her kullanmak istediğinizde program yazma derdinden kurtarmakla kalmaz, ama aynı zamanda siz kendiniz yazmaya kalksanız günlerinizi, hatta haftalarınızı alacak hazır programlardan oluşan bir kitaplık oluşturmanızı da sağlar. Yazılım imalatçıları, en iyi programlar tarafından yazılmış çok çeşitli programlar sunarlar ve bunların büyük bir kısmı Spectrum'da kullanılabilir. Ne türden programlardan hoşlanıp, kullanabileceğinizi saptamak için Sinclair Spectrum Software Catalogue (Yazılım Kataloğu) na bakabilirsiniz. Bundan sonra da, gereksinim duyduğunuz, istediğiniz programı işleme koyabilirsiniz.

### Program Spectrum'a nasıl yüklenir?

Bir yazılım kasetindeki kodlu sinyaller, saniyede 1500 adet hızıyla kaydedilmiş alçak ve yüksek bazı seslerden oluşur. Bu yazılım kasetini, teypte çalınca teyp programı oluşturan sesleri yansıtır. Teypi Spectrum'a bağladığınızda, bu kodlu sinyaller doğrudan Spectrum'un belleğine gider. Buna program yükleme denir. Bu iki sayfada, teypin nasıl bağlanacağını göreceksiniz, 14 ve 15. sayfalarda ise bunun nasıl kullanılacağını bulacaksınız.

### Yazılım ilişkin sorular ve yanıtlar

#### "Yazılım" ne demektir?

Yazılım (İngilizce Software), bilgisayarları çalıştırmak için verilen programların genel adı. Donanım ise, (İngilizce Hardware) kullanılan araçlar - bilgisayar ve bu işlemlerde yararlanılan diğer gereçlerdir.

#### Yazılım niye kasetli teyple yapılıyor?

Kasetli teypinin kullanımı kolaydır, başkaca bir özel aygıt gerektirmez. Bu tür yazılımın yüklenmesi için gereken tek şey, ucuz bir kasetli teyptir.

#### Kayda alınmış bir program nasıl bir şey?

Birini, teypin Spectrum'a bağlanmadan çalın. Tiz bir düdük sesi duyacaksınız. Bu bilgisayara gidecek yerde oparlöre ulaşan kodlu sinyallerin çıkardığı sesler. Sinyaller teypten Spectrum'a o kadar hızlı aktarılır ki, sesleri teker teker ayırt etmek olanaksızdır.

#### Başka türlü yazılımlar da var mı?

Evet. Kaset yerine ROM kartuşlarda kayıtlı programlar da bulabilirsiniz. Bu kartuşlar, Spectrum'unuzun arkasındaki bir yuvaya girer. ROM kartuşları program, hiç beklemeden anında yüklenir.

Ayrıca Microdrive kartuşlarında da programlar vardır. Bunlarda da programlar, kaset gibi manyetik olarak kaydedilmiştir. Bir kartuşta birkaç program birden olabilir ve kasetin aksine bir programın yüklenmesi dakikalarca değil birkaç saniye sürer. Microdrive kartuşları, Microdrive biriminde çalınabilir. (Bak s.46)

#### En iyi kasetli teyp hangisi?

Spectrum pahalı olmayan herhangi bir portatif kasetli teyple çalışabilir, ama pili değil cırayan bağlantılı teypler yeğlenmelidir. Teypin bir ses ayar düğmesi olmalıdır, ama ton düğmesi zorunlu değildir. Ayrıca özel bilgisayar kasetli teypleri de vardır. Bunlar, programların daha güvenli yüklenmesini ve saklanması sağlamaya özenle özel olarak yapılmıştır.

Bir müzik setinin parçasını oluşturan kasetli teypin bilgisayara bağlanması pek kolay olmayacaktır. Ayrıca bu tür araçların ses çıkışları genellikle Spectrum için yeterince güçlü ses sinyali verememektedir.

#### Teypte kaydedilmiş programların özel bakımı gerekir mi?

Diğer manyetik kayıtlar gibi, kasete kayıtlı program da güçlü mıknatıslardan uzak tutulmalıdır. Bu yüzden kasetleri güçlü bir elektrik akımı kullanan hiçbir şeyin yakınında saklamayın. Yazılım kasetlerinin de, tozdan oldukça uzak tutulması gerekir.

#### Her türden yazılım kullanılabilir mi?

Hayır, sadece ZX Spectrum ya da ZX Spectrum+ için yazılmış yazılımlar kullanılabilir.

### Kasetli teyp nasıl bağlanır?

Spectrum'unuzla beraber gelen, teyp kablosu, bunu teypte bağlamak içindir. Bu bağlantının her iki ucunda da ikiye tane küçük fiş vardır. Kasetli teypinizi Spectrum'un yanına yerleştirin ve kabloyu şekilde gösterildiği gibi bağlayın.

Bu işlemi yaparken Spectrum ya da teypin çalışır durumda olup olmaması önemli değil, ama gene teypin açar ya da kapakının içinde kaset bırakmamak iyi olur. Bu, kasete kayıtlı programların korunması açısından yararlıdır.

### Bağlantıları doğru yapın



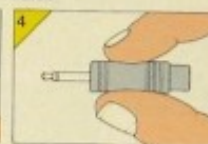
1 Dört fişten birini Spectrum'un arkasındaki EAR girişine sokun



2 Öbür uçtaki aynı renkli fişi teypin, eğer varsa EAR girişine sokun



3 Eğer teypin EAR girişi yoksa, fişi kulaklık girişine, o da yoksa diğer oparlör girişine sokun



4 Eğer bu fişler, teyp girişine uymuyorsa, elektrikli den ya adaptör ya da uygun fişli bir kablo bulmanız gerekir. Spectrum'un EAR girişi 3.5mm'lik jak-fiş ve 1 voltluk bir girdi sinyali için yapılmıştır.

### EAR ve MIC girişleri

Programları yüklerken, burada da gördüğümüz gibi EAR ve MIC girişleri bağlı durabilir. Ama bir program saklamaya hazırlanıyorsanız (Bak s.38) EAR kablosunu sökmeniz gerekir.

### Yazılım için öğütler

■ Spectrum'un teyp kablosunda, bilgisayarla teyp arasındaki bağlantılar karışmasın diye, farklı renklerde fişler kullanılmıştır. Teypinizi Spectrum'la kullandığınızda, düzeni değiştirmeyin, bir rengi EAR, bir rengi MIC girişleri için kullanın.

■ Bazı kasetli teypler çevrelerindeki elektrikli araçlardan etkilenirler. Bundan bilgisayar ile teyp arasında gidecek sinyaller de etkilenir ve bu yüzden programlar düzgün yüklenemeyebilir. Eğer teypiniz düzgün çalışmazsa o zaman teypin televizyon ya da bilgisayarın yanından uzaklaştırın.





## PROGRAM NASIL YÜKLENİR?

Kasetli teypi Spectrum'unuza bağladığınızda göre, program yükleyip, işleme koymaya hazırsınız demektir. Ya bir hazır yazılım kaseti ya da kendi programlarınızı içeren bir kaset kullanabilirsiniz. Her iki durumda da yapacağınız aynı işlemdir.

1 Kaseti takın ve en başa sarın.

2 Kasetli teypin ses ve ton düğmelerini gerekli düzeye getirin. Ses düğmesi üçte iki oranında açık olmalı, diğer ton düğmesi varsa, tize getirilmelidir.

3 J'ye basın, ekranda LOAD yazısı gözükmelidir. Sonra programın adını tırmak işaretleri içinde yazın. Örneğin LOAD "Prog 1"



4 ENTER'e basın. Ekran silinecektir.

5 Teypi çalıştırın. Ekranın çevresi kırmızı veya mavi olacak ya da kırmızı ve mavi yarıp sönecektir. Bu Spectrum'un program aradığını gösterir.



6 Birkaç saniye sonra, ekranın çevresindeki kırmızı mavi çizgiler aşağı-yukarı hareket etmeye başlar. Bu Spectrum'un sinyali almaya başladığını belirtir.

7 Ekranda Program sözcüğü, ardından programın adı veya Bytes yazısı, bundan sonra da bir harf ya da rakam görünür. Bu bilgisayarın istenen programı başıyla bulduğunu işaretidir.



8 Bilgisayar programı yüklemek üzere beklerken, kırmızı ve mavi çizgiler yeniden belirir.

9 Ekranın çevresinde san ve mavi çizgiler belirir. Bu da Spectrum'un programı yüklemekte olduğunu gösterir. Eğer program çok uzunsa, yükleme birkaç dakika sürebilir.



10 Eğer program bölümlere ayrılmışsa, 7, 8 ve 9. adımlar bir veya birkaç kez daha tekrarlanabilir.

11 Program, yüklenir yüklenmez otomatik olarak başlayabilir. Teypi durdurmayı unutmayın.

12 Eğer program yüklenir yüklenmez otomatik olarak başlamazsa, ekran silinir, OK, 0-1 yazısı belirir. Teypi durdurun.



13 R(RUN) ve ENTER'e basın, program başlar.



### Yazılım yükleme öğütler

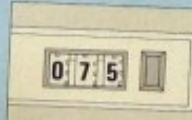
İşte size yükleme zamanı kazandıracak bazı öğütler:

1 Bütün kasetlere, aradığınız programı kolayca bulabileceğiniz biçimde etiket yapıştırın. Eğer bir kasette birden fazla program varsa, etikete sırasıyla bunların hepsini yazın. Programların adını tam bilgisayarın tanıyacağı biçimde yazmayı unutmayın.

Program Adı	Adres	Boyut
Program 1	0000	1000
Program 2	0000	1000
Program 3	0000	1000
Program 4	0000	1000
Program 5	0000	1000

2 Kasetli teypin sayacı varsa, bunu üzerinde birden fazla program bulunan kasetlerde aradığınızı hızla

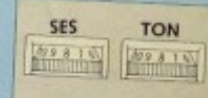
bulmak için kullanın. Kasetin başlangıcında sayacı sıfıra getirin. LOAD komutu verin, ardından o kasetle bulunmayan bir programın adını tırmak işaretleri içinde yazın. Kaseti çalıştırınızda, Spectrum kasetteki bütün programların adını teker teker ekrana yazacak, ama yüklemeyecektir. Programın adının yanına sayacıdaki rakamı yazın. İleride hangi programın nerede olduğunu kolayca bulabilirsiniz.



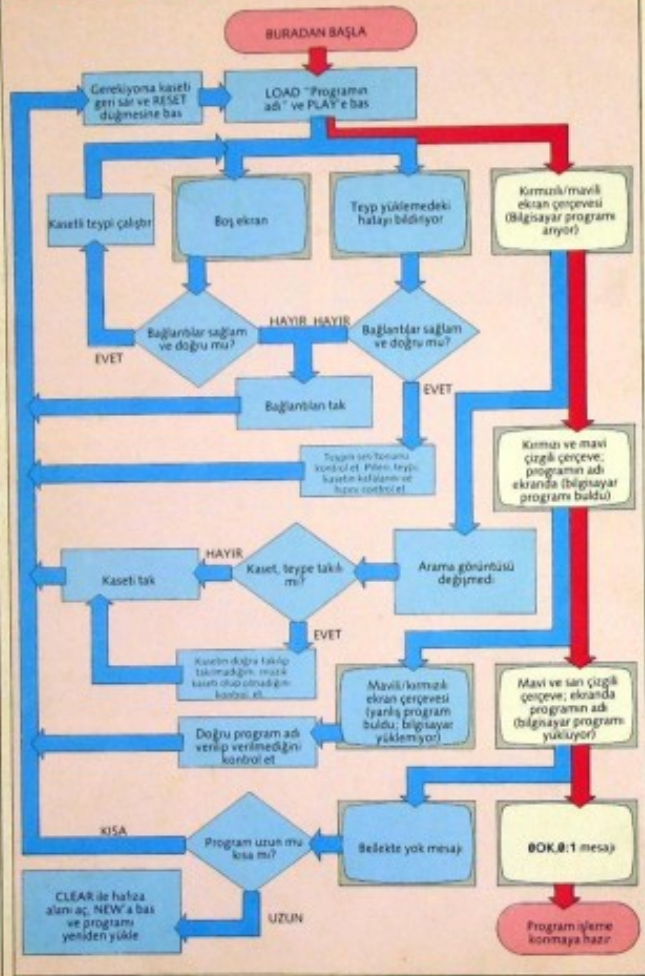
3 Eğer kaset doğru programı bulmuşsa ya da programın adını bilmiyorsanız, LOAD komutundan sonra tırmak içinde

programın adını yazacak yerde, LOAD " " yazın. İki tırmak arasında hiç boşluk bırakmayın. Bu durumda Spectrum'unuz bulduğu ilk programı yükleyecektir. Eğer gördüğünüz program adı, istediğiniz değilse, BREAK'e basın, kaseti sanp gene deneyin.

4 Spectrum'unuzun yüklenmesine yardımcı olacak ses ve ton düzeyine



dikkat edin. Bunları, teypinizi yüklemeye önce yukarıda gösterildiği gibi ayarlayın.



## PROGRAMLAMAYA BAŞLAYALIM

Bu bölüm, ZX Spectrum+'da program yazmaya bir giriştir. Klavyeyi tanıtarak, Spectrum'unuzu nasıl kullanabileceğinizi anlatmaktadır. Böylece Spectrum'unuzu nasıl çalıştırmaya başlayabileceğinizi göreceksiniz. Bu bölümde deneyeceğiniz kısa programlar, kendi programlarınızı yazarken bilgisayarınızın kapasitesini sonuna kadar kullanabilmeniz için Spectrum'un kendine özgü özellikleri üzerinde yoğunlaştırılmıştır.





# KLAVYE – BİLGİSAYARINIZIN KOMUTA KONTROL TABLOSU

ZX Spectrum'un kendine özgü bir dili vardır – BASIC diye bilinen bilgisayar dili. Talimatlarınıza uyması için, Spectrum'u BASIC dili konuşarak programlamamız gerekir. Bunu, Spectrum'un klavyesini kullanarak yapacaksınız. Ayrıca bu klavye, bilgisayar sizin programınızı işlerken, bilgisayarın kontrol etmesini de sağlar.

**GRAPH**  
Bu tuş, 1 ile 8 arasındaki şekilleri ve grafik karakterleri seçmede kullanılır. Bu tuşa basıp, sonra da CAPS SHIFT tuşuna basarak veya basmadan bir rakam tuşuna basarsanız, ekranda bir grafik karakter belirecektir.

**NEW**  
Bu tuş, bilgisayarın BASIC dilindeki belleğini temizler, bellekteki bütün programları siler.

**DELETE**  
Yanlış bir tuşa basmış, bir komut, harf, rakam veya işaretli silmek istiyorsanız, bu tuşa basın – bak sayfa 10.

**EDIT**  
Bu tuş, bütün bir programı yeniden yazmak için bunun bir satırının düzeltilmesini sağlar – bak sayfa 21.

**EXTEND MODE**  
Bu tuşun yardımıyla, tuşların üstündeki komutları verebilirsiniz. Bunun ardından SYMBOL SHIFT'i kullanırsanız, üstündeki işaret ya da komutu elde edersiniz – bak sayfa 20-21.



**CAPS SHIFT**  
Bu tuşa basarken, bir harf tuşuna basarsanız, büyük harf elde edersiniz. Yan yana birçok büyük harf kullanacaksanız önce CAPS LOCK'a basın.

**CAPS LOCK**  
Eğer sürekli büyük harf yazmak istiyorsanız, harf tuşlarına basmaya başlamadan buna basarsınız. Büyük harflerle yazdıktan sonra, tekrar buna basıp küçük harflere dönebilirsiniz.

**BEEP**  
Bu tuş, Spectrum'un ses vermesini denetleyen komutları elde etmede kullanılır.

**TRUE VIDEO ve INV VIDEO**  
Bu tuşlar, program satırlarına, normal ya da ters (dişi) renkler elde etmek üzere kontrol kodlarının yerleştirilmesi için kullanılır.

## Tuşlar ve Komutlar

Komutlar, BASIC dilinde bilgisayara belli bir şey yapması talimatı veren özel sözcüklerdir – PRINT, INPUT gibi. Bilgisayarların çoğunda bu komutları, tipki daktiloda yazarak gibi harfleri teker teker yazmak zorundasınız ve her sözcüğün imlasının da tam doğru olması gerekir. Ancak Spectrum'da bir tek tuşa basarak, bütün komutun ekranda

görünmesini sağlayabilirsiniz.

Sinclair BASIC'te 80 komut vardır. Bunları 36 tuşa (26'sı harf, 10'u rakam) basarak verebilirsiniz. Spectrum bu kadar çok çeşitlilik gösteren BASIC komutları kullandığından, birçok tuşa sadece bir değil, bilgisayarın tanıyacağı çok çeşitli komutlar verebilirsiniz. Aslında birçok tuş size komutların yanısıra harf, rakam, işaret, hatta şekil (grafik karakterler) sağlar. Bütün bunları programlarınızda kullanabilirsiniz.

**Renk tuşları**  
Bu altı tuş, ekrandaki renkleri kontrol eden komutları verir.

## Komutların ve İşaretlerin seçimi

Spectrum'un klavyesinde çok kullanacağınız iki tuş var: EXTEND MODE ve SYMBOL SHIFT. Bunlarla, diğer tuşlardaki komut ya da işaretlerden hangisinin ekranda görünmesini istediğinizi seçebilirsiniz. Bu tuşları nasıl kullanabileceğinizi size daha önce 8. sayfada kısaca anlattık. Şimdi klavyeyi biraz daha tanıdıktan sonra, önümüzdeki iki sayfada, bilgisayarın klavyesinde gördüğünüz herhangi bir şeyi nasıl seçebileceğinizi anlatacağız. Bir defa bunu öğrendiğiniz mi, programlarınızı yazmaya başlayabilirsiniz.

**Rakam tuşları**  
Bunlar, rakamları yazmanın yanısıra, gösterilen renklerin programlara yerleştirilmesi için kontrol kodlarını da verir. (Bak s. 33). 8'de 4'ten 0'a kadarki tuşların hemen üstünde yer alan bu komutlar ancak ZX Microdrive'larla kullanılabilir.

**BREAK**  
Bu tuş, işleme konan bir programı durdurur. Program bilgisayarın belleğinden silinir.

**ENTER**  
Bu tuşa basınca, bir program satırını Spectrum'un belleğine girer. Bu tuş, ayrıca çok sık, bir program sırasında bilgisayara bilgi beslemeye de kullanılır.

**SYMBOL SHIFT**  
Bir tuşun üstündeki ikinci bölümden işaret ya da alt komutu kullanmak isterseniz, bu tuşa SYMBOL SHIFT'i birlikte basarsınız gerekir. Eğer SYMBOL SHIFT'e, EXTEND MODE'dan sonra basarsanız, o zaman bilgisayar, ikinci bölümün hemen üstündeki komuta uyar – bak sayfa 20-21.

**Boşluk tuşu**  
Tipki daktiloda olduğu gibi bu tuşa dokununca bir ara elde edilir.

**Göstergeç kontrolü**  
Bu tuşlara basarak göstergecin (İngilizceyi çıkarır) oklarını yönünde hareket ettirmesi sağlanabilir. Bu tuşlar, programlarda genellikle şekillerin ekrandaki hareketlerini denetlemek için kullanılır. Ayrıca program üzerindeki düzeltmelerde de yararlı olur.

## TUŞLAR NASIL ÇALIŞIR?

ZK Spectrum'un tuşlarından herhangi biriyle altı kadar değişik komut, harf, rakam ya da işaret yazabilirsiniz. Spectrum'un özelliklerinden birini iyice öğrendiğiniz takdirde klavyedeki herhangi bir harf ya da komutu seçmek pek o kadar karışık gelmeyecektir. Bir tuşa bastığınızda, ekranda görünecek olan, o anda bilgisayarınızın hangi modda olduğuna bağlıdır. Farklı modlar, sizin komut, harf ya da grafik karakterleri gibi çok farklı türde bilgileri iletmenizi sağlar. Bunun yaranı şudur: Siz klavyeyi kullanırken, aslında Spectrum, talimat ve bilgileri sırasıyla verebilmeniz için size klavyedeki modları seçmenize yardımcı olur. Bu iki sayfada bu modların ne işe yaradığını göreceksiniz.

### Komut, simge ya da karakter nasıl seçilir?

Aşağıda, bir harf ya da rakam tuşundaki herhangi bir komut, işaret ya da karakterin nasıl seçilebileceğini göreceksiniz. Bir

tuştaki herhangi bir işlevi seçerken, bunun tuşun neresinde olduğuna dikkat edin ve aşağıdaki iki örnekten yararlanarak, hangi tuşlara basarak

neyi, nasıl bulabileceğinizi belirleyin. Daha önce ekrandaki göstergeci bakıp bilgisayarın hangi modda olduğunu saptayın.

#### Harf tuşu

BIN BRIGHT BORDER B*	Komut (K) modu
Sadece tuş	BORDER
ve tuş	
Genişletilmiş (E) mod	
EXTEND MODE	sonra sadece tuş
EXTEND MODE	sonra
EXTEND MODE	sonra
EXTEND MODE	sonra
Harf (L) modu	
Sadece tuş	b
CAPS LOCK	ve tuş
CAPS LOCK	ve tuş
Büyük harf (C) modu	
CAPS LOCK	sonra sadece tuş
CAPS LOCK	sonra
CAPS LOCK	sonra
Grafik (G) modu	
GRAPH	sonra A ile U arasındaki tuşlar kullanılarak belirlenecek grafik

#### Sayı tuşu

MGNTA LINE 3 #	Komut (K) modu
Sadece tuş	3
ve tuş	#
Genişletilmiş (E) mod	
EXTEND MODE	sonra sadece tuş
EXTEND MODE	sonra
EXTEND MODE	sonra
Harf (L) modu	
Sadece tuş	3
ve tuş	#
Büyük harf (C) modu	
CAPS LOCK	sonra sadece tuş
CAPS LOCK	sonra
CAPS LOCK	sonra
Grafik (G) modu	
GRAPH	sonra sadece tuş
GRAPH	sonra
GRAPH	sonra

### Komut modu

Spectrum'unuzu çalıştırın ya da Copyright mesajını göreceğiniz biçimde düzenleyin. ENTER'e basın. Sol alt köşede yanıp sönen bir K harfi belirecektir. Bu yanıp sönen kareye göstergeci (İngilizcesi cursor) denir. Bu size basacağınız ilk tuşun neresiye yansıyacağını gösterir. K ise bilgisayarın komut modunda olduğunu belirtir. Herhangi bir tuşa basın. Tuşun çıkıntılı bölümündeki üst komutu ekranda belirdeğini göreceksiniz. Örneğin Q'a basarsanız PLOT sözcüğü belir. DELETE tuşuna basarak komutu silin ve diğer tuşları deneyin. Rakam tuşları size rakam verecek, ancak bir harf tuşuna basarsanız, çıkıntılı bölümdeki üst komutu göreceksiniz.

Gene DELETE'e basarak, K göstergecinin belirmesini sağlayın. Şimdi de SYMBOL SHIFT tuşlarından birine basın, elinizi orada tutarken, bir başka tuşa daha dokununuz. Bu defa çıkıntılı bölümdeki harfin hemen üstündeki komut ya da işaret belirecektir. Rakam tuşunda ise, çıkıntılı bölümün sağ üst köşesindeki işaret belirir. Yani, Komut modu tuşun çıkıntılı bölümü ile ilişkilidir.

### Harf ve büyük harf modları

Bilgisayar komut modundayken, bir komut yada işaret yazdıktan sonra otomatik olarak göstergeci L'ye çevirir. Artık harf moduna geçmiştir. Herhangi bir harf tuşuna basın, bir küçük harf belirecektir. Rakam tuşuna basın, rakam görürsünüz. Büyük harf yazmak isterseniz, CAPS SHIFT'e basıp bunu tutarken harfe basın.

Eğer hep büyük harf yazacaksanız, önce CAPS LOCK'a basın. Göstergeci C'ye döndürür. Spectrum'unuz büyük harf moduna geçti, artık hangi harfe basarsanız büyük harf yazarsınız. Rakamlar büyük harf modunda da yazılabilir. Harf (L) moduna dönmek için CAPS LOCK'a bir kez daha basın.

### Genişletilmiş mod

Bu modun adı genişletilmiş mod ve bunu EXTEND MODE'a basarak elde edersiniz. Göstergeci E'ye döndürür. Herhangi bir harf tuşuna bastığınızda, çıkıntılı bölümün üzerinde bulunan iki komuttan üsttekini

elde edersiniz. Örneğin B'ye basarsanız BIN çıkar. Altındaki komutu ya da işareti elde etmek için ise SYMBOL SHIFT tuşlarından birine basıp tutun ve harf tuşuna dokununuz. B tuşunda örneğin BRIGHT yazısı çıkar. Yani genişletilmiş modda bir tuşa (ya da EXTEND MODE'a) basarsanız, bilgisayar otomatik olarak harf ya da büyük harf moduna geçer.

### Grafik Modu

Beşinci mod, grafik modudur ve GRAPHIC tuşuna basarak geçer. Göstergeci G'ye döndürür. 1'den 8'e kadar rakamlara basıp bunların üzerindeki grafik karakterlere bakın. Şimdi ise CAPS SHIFT'e basarak sonra 1 ile 8 arasında herhangi bir sayıya basın. Aynı şekli yeniden çıkacaktır ama bu defa siyah ile beyaz yer değiştirmiştir. Grafik modundan çıkmak için mutlaka GRAPH tuşuna yeniden basmanız gerekir, çünkü bilgisayar bu moddan otomatik olarak çıkmaz.

### Spectrum'da düzeltme

Spectrum'a talimat verdiğinizde ya da kendi programınızı yazmaya başladığınızda, komutlarda ya da program satırlarında düzeltmeler yapmak veya bunları değiştirmek isteyebilirsiniz. Bu oldukça kolaydır.

#### Hatanın düzeltilmesi

BASIC'de hata bir satır ya da komut vermeye kalkarsanız, Spectrum bu hatanın önünde bir ? işaretini yakıp söndürmeye başlar. Hatayı düzeltmek için, göstergeci hatanın sağına götürebilmek üzere, göstergeci kontrol tuşlarından sağdakine ya da soldakine sürekli basın. Ardından ya DELETE'e basarak hatayı yok edin ya da gereken komut, harf, rakam ve simgeyi yazın. Ardından ENTER'e basın.

Örneğin bilgisayarın 7 ile 8'i çarpmasını istiyorsanız ama + elde etmek için SYMBOL SHIFT'e basmayı unuttunuz. Bunun yerine

#### PRINT 7\*8

çıkır. Spectrum bu komuta uyamaz, bu nedenle siz ENTER'e basar basmaz hata yaptığınız ?'nin önünde bir soru işareti yanıp sönmeye başlar. Bütün

yapacağınız, göstergeci hatanın sağına götürüp, ardından DELETE'e basıp b'yi yoketmek. Bundan sonra SYMBOL SHIFT ve B'ye basarak + elde edin ve ENTER'e basarak bilgisayarın doğru talimata uymasını sağlayın. Göstergeci yensiden satırın sonuna kadar götürmeniz gerekmez. Spectrum talimatı uygulay ve sonucu verir.

#### Bir program satırını nasıl düzeltilir?

Bir program yazarken, sıralama



denilen, birbirini izleyen numaralarla bir dizi satır ardında koyarsınız. Eğer program yazdıktan sonra, buna K (LIST) ve ENTER tuşlarına basarak "sıralarsanız", programın satırlarından birinin karşısına > işaretini göreceksiniz. Eğer görüyorsanız, göstergeci aşağı yukarı kontrol eden tuşlara basın. EDIT tuşuna basarsanız, bu işaretin karşısındaki satırın eşi ekranın altında bir daha çıkar. Bunu daha önce anlattığımız gibi göstergeci kontrol ve DELETE tuşlarını kullanarak "teğşterebilirsiniz. ENTER'e basınca da o satır eski yerine döner. Başka bir satırı daha düzeltmek istiyorsanız, > işaretini istediğiniz satırın karşısına gelecek şekilde yukarı aşağı oynatın ve EDIT'e basın. Eğer bu uzun sürecektir, sırasıyla LIST, istenen satırın numarası ve EDIT'e basın. Her iki halde de istediğiniz satır ekranın en altında belirecektir. Bunda düzeltme yapabilirsiniz. Programın bütün bu satırını silmek istiyorsanız, sadece satırın numarasını yazıp ENTER'e basın. Eğer bir hata içerir bir programı işleme koyduysanız, ekranda bunun haberi görürsünüz. Bu 74. sayfada açıklanacaktır.



# TELEVİZYON HESAP MAKİNASI

ZX Spectrum+ son derece hızlı ve doğru hesap yapabilir. Bütün gereken bir kaç rakam ve bu rakamlarla ne yapılmasını istediğinizi gösteren +, - gibi bazı işaretlerdir.

Önce şu talimatı verin (+ işaretini K tuşunda bulacaksınız):

## PRINT 6+2

Bu bir komuttur. ENTER'e basınca komut kaybolur, ekranda cevap, 8 görünür.

Spectrum'unuz, hesaplamada, aritmetik işlemler denilen beş işaret kullanır. Bunların her birinin ne yaptığını aşağıdaki tablo da görebilirsiniz. Bunların hepsini PRINT ile aynı şekilde kullanabilirsiniz.

PRINT 6+2 gibi komutlar vererek, Spectrum'unuzu hesap makinasına dönüştürebilirsiniz. Ayrıca sıradan bir hesap makinasının yapmadığı birçok şey yapabilir. Herşeyden önce, hem hesabı hem sonucunu aynı anda gösterir. Şu komutu verin:

## PRINT "6+2="; 6+2

Bilgisayar bunu şöyle yanıtlar:

6+2=8

Olan şu: PRINT, tırnak işaretleri ("") arasındaki herşeyin ekranda görünmesini sağlar, bu nedenle 6+2= görünür. Tırnak işaretleri arasındaki işaretlere dizi denir. Noktalı virgül Spectrum'a, eşit işaretinden hemen sonra sonucu iletmesi talimatını verir.

### Spectrum'un hesaplama işaretleri

Aşağıdaki işaretler ya da "aritmetik işlemler", Spectrum tarafından matematik işlemler için kullanılır. Dikkat edin bilgisayar x ya + işaretlerini kullanmaz.

İşaret	Tuş	İşlevi	Örnek
+	K	İki rakamı toplar	8+2=10
-	J	Bir rakamı diğerinden çıkarır	8-2=6
*	B	İki rakamı çarpır	8*2=16
/	V	Bir rakamı diğerine böler	8/2=4
↑	H	Birinci rakamı diğerinin üstüne çıkarır	8↑2=64

### İlk programınız

Bir komut yerine getirilince, Spectrum'unuz bunu unutur. Bilgisayarın, hesabı tekrarlamasını istiyorsanız, bunu program olarak yazabilirsiniz. Şu talimatı yazıp ENTER'e basın:

## 10 PRINT 6+2

Bu defa talimata derhal uymaz. Sadece talimatı ekrana yansıtır. R(RUN) ve ENTER'e basın. O zaman sonuç, 8, ekranda belirir.

Bütün talimat, şimdi bir bilgisayar talimatı haline gelmiştir. Başına rakam yazarak, bu talimatı Spectrum'unuzun belleğine aktardınız, ama emir alana kadar işleme koymamasını söylediniz. Bu talimata artık komut değil cümle denir ve bir programın numaralanmış bir satırını oluşturur. Programdaki cümleler daima satır sırasına göre işleme konur ve bunlar gerektiğinde aralarına satır eklenebilir diye daima onlu sayılarla başlar.

Artık Spectrum'unuzu gerçekten çalıştırabilirsiniz. Şu programı verin. Her satır yazdıktan sonra ENTER'e basmayı unutmayın. Bitirince, R(RUN) ve ENTER'e basın. Programı işleme koyunca şu sonuç göreceksiniz:

### SAYI TABLOSU

10 LET A=1	20 PRINT A; " ";
20 LET A=A+1	40 GO TO 20

1	2	3	4	5	6	7	8	9	0
10	20	30	40	50	60	70	80	90	00
100	200	300	400	500	600	700	800	900	000
1000	2000	3000	4000	5000	6000	7000	8000	9000	0000
10000	20000	30000	40000	50000	60000	70000	80000	90000	00000
100000	200000	300000	400000	500000	600000	700000	800000	900000	000000
1000000	2000000	3000000	4000000	5000000	6000000	7000000	8000000	9000000	0000000
10000000	20000000	30000000	40000000	50000000	60000000	70000000	80000000	90000000	00000000
100000000	200000000	300000000	400000000	500000000	600000000	700000000	800000000	900000000	000000000
1000000000	2000000000	3000000000	4000000000	5000000000	6000000000	7000000000	8000000000	9000000000	0000000000

1'den 203'e kadar bütün rakamlar görülüyor. N, ara tuşu, STOP veya BREAK dışındaki herhangi bir tuşa basın. Bir dizi sayı daha göreceksiniz.

Bu program bir değişken kullanıyor. Bu olayda değişken, n. Değişken olarak

herhangi bir harf ya da sözcük de kullanılabilir. Burada n sayı anlamına geliyor. Değişkene program boyunca değişen bir değer verilir. 10. satırda LET komutu kullanarak bu değer 1 olarak verilmiş. 20. satırda bir boşluk bırakılmış. 30. satırda yine LET kullanılmış, ama bu kez değeri 1 arttırılmış. Bu nedenle n 2 olur. 40. satırda (tek) bir komut GOTO kullanılarak program 20. satıra döndürülmüş. Artık 20. satırda 2 rakamı vardır. Bu, bütün ekran rakamla dolana kadar döne döne tekrarlanır.

### Programın sayı istemesi nasıl sağlanır

BREAK'e basarak programı durdurun. Yeni satır yazın:

## 10 INPUT n

Bu satır eski programdaki 10. satırın yerini alır. İşleme koyduğunuzda, bilgisayar bir rakam vermenizi bekler. Herhangi bir rakama, sonra ENTER'e basın. Bundan sonra görünecek rakamlar, sizin yazdığınız rakam ile başlar çünkü INPUT n, n'nin değerini sizin yazacağınız rakama eşitler. INPUT bilgisayara bir program sırasında bilgi alma talimatı verir.

### Bir çarpım cetveli programlama

Eski programı temizlemek için RESET düğmesine basın ve yeni programı verin. Bu program Spectrum'a çarpım yaptırır. Herhangi bir rakama bastığınızda, ekranda bu rakam için çarpım cetveli verilir. N, BREAK, ara tuşu dışında herhangi bir tuşa basınca tablo devam eder. BREAK'e basıp, yeni bir çarpım cetveli daha yapın. İşte 3 ve 146 ile yapılabilecek çarpım cetvellennin programı

### ÇARPIM CETVELİ

10 LET A=1	20 PRINT A; " ";
20 LET A=A+1	40 GO TO 20

1	2	3	4	5	6	7	8	9	0
10	20	30	40	50	60	70	80	90	00
100	200	300	400	500	600	700	800	900	000
1000	2000	3000	4000	5000	6000	7000	8000	9000	0000
10000	20000	30000	40000	50000	60000	70000	80000	90000	00000
100000	200000	300000	400000	500000	600000	700000	800000	900000	000000
1000000	2000000	3000000	4000000	5000000	6000000	7000000	8000000	9000000	0000000
10000000	20000000	30000000	40000000	50000000	60000000	70000000	80000000	90000000	00000000
100000000	200000000	300000000	400000000	500000000	600000000	700000000	800000000	900000000	000000000
1000000000	2000000000	3000000000	4000000000	5000000000	6000000000	7000000000	8000000000	9000000000	0000000000



### Parantezlerin kullanımı

Zaman zaman hesaplarda parantez kullanmanız gerekecek. Şu iki komutu verin ve sonuçları karşılaştırın:

## PRINT 6+2/4

## PRINT (6+2)/4

İlkinin sonucu 6.5, ikincisinin ise 2. Bu iki farklı sonucun nedeni, bilgisayarın hesaplarda kullandığı bir öncelikler sistemine sahip olması. Önce / , sonra - veya + , en son olarak da \* veya = uygular. Ama daima parantez içindeki işleme öncelik verir. Bu nedenle, yukarıdaki ilk komutta, önce 2'yi 4'e böler, sonra sonuçla (6.5) 6'yi toplar. İkinci komutta ise, 6 ile 2'yi toplayıp, sonra 4'e böler.

### Spectrum'da noktalama işaretleri

Spectrum çok çeşitli noktalama işaretleri kullanır. Bunlar çok önemlidir, çünkü, bunların çoğu aynı zamanda bilgisayarın bir program satırını anlamasını ya da bir sonuç sağlamanın etkileyecek biçimde bilgisayara talimat vermeye de yarar.

- **Noktalı virgül** PRINT ile birlikte kullanınca, bilgisayara iki yanındaki öğelerin ekranda yan yana görünmesi talimatı verir.
- **İki nokta** Program satırındaki cümlelerden birinin sona erdiğini, bir diğerinin başlayacağını haber verir.
- **Tırnak** Tırnak işaretleri arasında yazılanlar, rakam ya da değişken olarak değil, bir ifade olarak değerlendirilir. Dört tırnak işaretleri ile başlayıp biter.
- **Virgül** PRINT ile birlikte kullanınca, bilgisayara izleyen öğeyi ya bir satırın ortasında, ya da yeni satırın başında ekrana yansıtmasını bildirir. Bunu birer ya da milyonlar hanesini ayırmak için kullanmayın.
- **Nokta** Ya bir ondalık işareti ya da noktadır.





## BASİT BAZI GRAFİKLER

ZX Spectrum+'ınuzunla alçak ya da yüksek tanımlı bazı çizimler yapabilirsiniz. Her ikisi de aynı anda ekrana yansıtılabilir. Alçak tanımlı çizimler renkli bazı karelerden oluşur. Bu iki sayfada bu karelerin klavyeden nasıl elde edilebileceğini ve bunların ekrana nasıl yansıtılabileceğini göreceksiniz.

### Alçak tanımlı çizimlerde ekran

Alçak tanımlı çizimlerde ekrana enlemesine 32 karakter yerleştirilebilir. Boylamasına ise 22 karakter alır. Ekrandaki her pozisyonu iki ayrı rakam belirler. Önce dikey satır rakamı verilir. Bu, aşağıda kaç satır kaldığını gösterir. En üstteki satırın rakamı 0, en alttakini ise 21'dir. Bunu yatay sütun rakamı izler. Sol baştaki sütunun rakamı 0, en sağdakini ise 31'dir. (80. sayfada alçak tanımlı çizim çizilmesinin tümünü görebilirsiniz). Şimdi göreceğiniz program, bu pozisyonların renklerle nasıl doldurulacağını gösteriyor. R tuşundaki RND komutu rastsal bir mürekkep rengi seçer.

### RASTSAI KARFLER

```
10 BORDER 1: INK RND 47
20 DO UNTIL
30 DO TO 30
```



Bütün ekrani kareler kaplar. Herhangi bir karakterin belli bir pozisyonda belirmesi için, PRINT ile birlikte AT komutunu kullanmanız gerekir. PRINT'in ardından AT, bundan sonra da önce satır rakamı, virgül, sütun rakamı ve noktali virgül tuşlarına basın. Örneğin, şu komutta

```
PRINT AT 11,16: "*" "
```

11. satır, 16. sütunda bir yıldız çıkacaktır.

### Gökkuşağı deseni nasıl çizilir?

Renkli desenler elde etmenin iyi bir yolu, çizim programınızda FOR NEXT döngüsünü kullanmaktır. FOR NEXT döngüsü, kendisini belirli kez tekrarlayan bir program parçasıdır. Bunu yaparken, ekrana belli karakterleri yerleştirmesi de istenebilir.

### Grafik karakterlerin seçimi

ZX Spectrum+'ınuzun klavyesinde bazı grafiklerin programlanmasını kolaylaştıran bir dizi grafik karakter vardır. Bunları 1-8 arasındaki tuşlarda görebilirsiniz.

Grafik karakterleri ekrana aktarmak için, GRAPH tuşu ardından 1'den 8'e kadar olan tuşlara, aralarında ara tuşuna da dokunarak, basın. Grafik karakterler ekranın altında belirecektir. Bu karakterlerin tuşlarını üzerinde beyaz olan bölümü ekrana mürekkep renginde, siyah bölümü de kâğıt renginde yansıtır. Bu defa CAPS SHIFT'e basarak, bu tuşlara bir kez daha dokunur. Mürekkep ve kâğıt renkleri yer değiştirir.

Program satırlarına grafik karakterleri aynen böyle yerleştirirsiniz. Grafik karakterlerden rakamlara dönmek isterseniz, GRAPH'a bir kez daha basın.

8 tuşu  
Bu tuş, genellikle GRAPH ve CAPS SHIFT ile birlikte tam bir renkli kare elde etmede kullanılır.

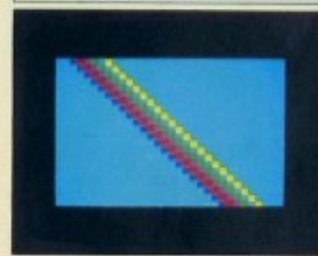
GRAPH  
Bu tuş Spectrum'u grafik moduna geçirmede kullanılır.



Her defasında sadece bir döngü kullanmakla sınırlı değilsiniz. Bir döngüyü, bir diğerinin içine yerleştirebilirsiniz ve bu çoğu kez yararlı sonuçlar verir. Aşağıda (biri diğerinin içine "yuvalanmış") iki FOR NEXT döngüsünde INK ve AT'in sağlayacağı pozisyonların ve renklerin değişebileceğini göreceksiniz. Bu döngülerin nasıl hazırlanacağını bu sayfanın sonundaki karede bulabilirsiniz.

### GÖKKUŞAĞI

```
10 BORDER 0: PAPER 5: CLS
20 LET X=1
30 FOR I=0 TO 21
40 FOR J=0 TO 8
50 PRINT INK C: AT I, J: " "
60 NEXT J
70 NEXT I
```



### Resimlerin programlanması

Alçak tanımlı çizimlerde, grafik karakterlerin pozisyonları ve renkleri ile oynayarak resimleri "boyayabilirsiniz". 80. sayfadaki basit çizim çizilmesinden yararlanarak resimlerinizi programlayabilirsiniz. Bundan sonra, karşı sayfada gösterildiği gibi grafik karakterleri seçerek, resmi oluşturacak program satırlarını birer birer verebilirsiniz.

Yandaki program nasıl bir sonuç elde edeceğinizi gösterecektir. Buradaki bütün şekiller rakam tuşlarındadır. Programı işleme koymak için, tümünü verene kadar bekleyebileceğiniz gibi, her satırı verdikçe işleme koyacak olursanız, robotun çeşitli bölümlerinin nasıl oluştuğunu görebilirsiniz. (Unutmayın eğer yanlış grafik karakter verecek olursanız, bunları da tipki yanlış rakam veya harfleri düzelttiğiniz gibi düzeltebilirsiniz.)

### ZX ROBOT

```
10 BORDER 2: PAPER 1: CLS
20 LET X=1
30 FOR I=0 TO 21
40 FOR J=0 TO 8
50 PRINT INK C: AT I, J: " "
60 NEXT J
70 NEXT I
```



70. satırda PRINT'ten sonra görülen TAB komutu, bir karakteri bilgisayarı o anda üzerinde çalıştığı satırda belli bir yere yerleştirmek için kullanılır. TAB'ı izleyen 0 ile 31 arasındaki rakam, sütunu gösterir.

### FOR NEXT döngüsünün kullanımı

FOR NEXT döngüsü daima FOR ve TO komutlarıyla, bir değişken ve bunun başlangıç ve bitiş değerlerini içeren bir satırla başlar. Örnek:

```
30 FOR C=1 TO 6
```

Burada değişken C. Bununla başlayacak döngü, bilgisayarı bu programı tekrarlamasını sağlayacak satırları içerecektir. Bunlarda da C değişkeni kullanılabilir. FOR NEXT döngüsü, daima NEXT komutu ve değişkenle sona erer. Örnek

```
50 NEXT C
```

Program işleme konunca, FOR'dan NEXT'e kadar ki bölüm kendisi defalarca tekrarlar. Değişken TO'dan önceki değeriyle başlar ve TO'dan sonraki limite kadar her seferinde 1 artar. Bu durumda kendini altı kez tekrarlayacak, C başlangıçta 1, sonra 2,3,4,5 ve son olarak da 6 olacaktır.

25. sayfadaki ilk programda, bir "yuva" da üç döngü kullanılmıştır. Bu demektir ki, "dış" döngünün her turunda, "orta"daki bir tur yaparken de "iç" döngü bütün devrelerini yapmaktadır.

## RESİM DEFTERİ OLAN EKRA

ZX Spectrum + ile yapılabilecek çizimler, iri parçalardan oluşan alçak tanımlı çizimlerle sınırlı değildir. Karmaşık yetenekleri nedeniyle Spectrum'u düz veya eğri hatlı bazı şekillerin ayrıntılı görüntülerini çizmek için de kullanabilirsiniz.

Yüksek tanımlı çizimler, bir çizgi oluşturmak ya da düz renkli bir şekli doldurmak için yanyana dizilmiş noktalarla oluşur. Her nokta, alçak tanımlı çizimlerde kullandığımız karelerin 64'te biri büyüklüğündedir. Şu komutu verirsiniz

**PLOT 128,87**

bunlardan biri ekranın tam ortasında görülebilir.

Yüksek tanımlı çizimlerde kullanılan noktalara pixel denir. Bu İngilizce picture cell'in, resim hücrelerinin kısaltılmışıdır. Tipik alçak tanımlı çizimde olduğu gibi pixel'in yerini de iki rakam belirler. Ancak bunlar alçak tanımlı çizimlerdeki rakamlar değildir.

### Yüksek tanımlı çizim çizelgesi

Yüksek tanımlı çizim çizelgesinde yatay 256, dikey 176 pixel vardır. Ancak basit çizimin tersine, ilk rakam yatay koordinatı, yani ekranda enlemesine yerini gösterir. Bu rakamlar sol köşede 0, sağ köşede 255'tir. İkinci rakam olan dikey koordinatta ise rakamlar aşağıdan yukarıya doğru büyür. Aşağıda 0, yukarıda 175'tir. 0,0 pozisyonu alçak tanımlı çizimde olduğu gibi sol üst değil, sol alt köşedir. Yüksek tanımlı çizim çizelgesi için sayfa 80'e bakın.

### Yer belirleme ve çizme

Yüksek tanımlı çizimler için üç komut yeterlidir: PLOT, DRAW ve CIRCLE. PLOT'u virgülle ayrılmış yatay ve dikey koordinatlar izler ve komut buraya bir pixel kondurur. DRAW'u da virgülle ayrılmış iki rakam izler ama bunlar koordinat değildir. Bunlar pixel olarak, iki noktanın arasındaki yatay, dikey ya da çarpazlama mesafesidir. DRAW komutu iki nokta arasında bir çizgi çeker.

Programda eğer daha PLOT ve DRAW kullanılmamışsa ilk pozisyon 0,0'dir.

Kullanılmışsa, hangisi en son kullanılmışsa, PLOT'daki son nokta ya da DRAW ile ulaşılan son noktadır. Bundan sonraki DRAW komutu yeni noktaya bir çizgi çeker. Eğer çizgi sola ya da aşağı gidecekse, yatay veya dikey mesafeler eksi (-) değerle verilir. Şu programı deneyin:

### YILDIZ

```
10 INK 0
20 PLOT 0,0
30 DRAW 70,140
40 DRAW 180,0
50 DRAW 140,180
60 DRAW 0,140
70,140
```



PLOT başlangıç noktasını ekranın tepesine götürür, ardından DRAW komutları beş kırmızı çizgi çizer.

Şimdi programa şu satırları ekleyin:

```
4 BORDER 1: PAPER 6: INK 1: CLS
5 CIRCLE 128,87,87
```

Programı yeniden işleme koyun, renkli zeminli bir daire içinde kırmızı çizilmiş bir yıldız görürsünüz.



CIRCLE için üç değer gerekir. İlk ikisi dairenin merkezinin konumunu, üçüncüsü ise yarıçapı belirler. DRAW komutlarına da bir üçüncü değer ekleyebilirsiniz. Programda 2

ile -2 arasındaki değerleri deneyin ve bakın ne oluyor.

### Şekiller nasıl doldurulur?

Alçak tanımlı çizimde, şekillerin içini, birçok çizgiyi yanyana çizerek kolayca doldurabilirsiniz. Bunu DRAW pozisyonlarını her defasında 1 arttıran FOR NEXT döngüsü ile elde edebilirsiniz.

### DOLU ÜÇGEN

```
10 BORDER 1: PAPER 6: INK 0: CLS
20 FOR X=100 TO 100
30 PLOT 100,100
40 DRAW X,100
50 NEXT X
```



Eğer çizgileri biraz aralıklı çizerseniz ilginç bir sonuç elde edersiniz. Bunu FOR satırına STEP komutu ve bir rakam da ekleyerek yaparsınız. Bu teknik 11. sayfadaki doğan güneş programında kullanılmaktadır. Üçgen programında da aynı işi yapacaktır. Satır 20'yi değiştirip yeniden işleme koyun.

```
20 FOR X=100 TO 100 STEP 4
```

Bu defa aşağıda gördüğünüz yelpaze benzeri şekil oluşur. Bunun nedeni STEP'in her çizgi çizildikçe x'i 1 ilerletecek yerde 4 sıçratmasıdır.



### Ekrandaki resim defteriniz

Spectrum'unuz çok yeteneklidir. Bir resim ya da şekil çizdiğiniz zaman her defasında bir program yazmanız gerekmez. Bunun yerine ekrana doğrudan resim çizmenizi mümkün kılacak bir program yazabilirsiniz. İşte buna olanak sağlayacak basit bir program.

Program size mürekkep rengi soracak olan INPUT komutu ile başlıyor. Ardından (bu defa bir diziyi belirlemek üzere \$ işaretini kullanarak) INPUT komutu ile dört tuşa - u, d, l ve r - basılığa bilgisayarın kısa bir çizgi çizmesini sağlıyor.

### RESİM DEFTERİ VE ÖRNEK

```
10 INPUT "INK: "; I
20 BORDER 1: PAPER 6: INK I: CLS
30 PLOT 85,25
40 LET X=0
50 INPUT "X: "; X
60 IF X=0 THEN DRAW 0,X
70 IF X=1 THEN DRAW 0,100
80 IF X=2 THEN DRAW 100,100
90 IF X=3 THEN DRAW 100,0
100 GOTO 50
```



### IF ve THEN ile karar verme

Resim defteri programının 60-90 satırları IF THEN cümlesi içeriyor. Bu Spectrum'unuzun karar almasını sağlar. Bu durumda bilgisayar u, d, l ve r tuşlarına mı basıldığını bakar. IF (eğer) bu tuşlara basılmışsa, THEN (o zaman) bilgisayara bir çizgi çizmesi söylenmiş demektir. Başlığınız büyük harfle çizgi çizmez.

IF her zaman, Spectrum'un yapıp yapmadığını sınavacağı bir olgu ider - belli bir tuşa basılıp basılmadığı gibi. Gerçekse ya da oluyorsa, bu durumda THEN'de belirtilen iş yapılır. Değilse, program bir sonraki satıra geçer.

THEN'i izleyen satırdaki her şey bir karara bağlıdır. Şu satırda

```
110 IF B=5 THEN PRINT "+"; GOTO 200
```

b ancak 5 ise bilgisayar 200. satıra gidecektir.



Bu değişiklikler her programdaki bütün görüntüyü etkiler. FLASH, BRIGHT ve INVERSE'i bir PRINT cümlesinde kullanırsanız, bu üç komutun etkisi o sabirli yazılarla sınırlı kalır.



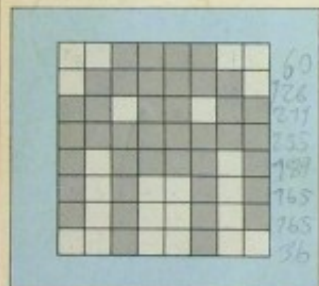
## BİLGİSAYAR KARAKTERLERİNİN YARATILIŞI

ZX Spectrum +1'unuz, sadece klavyedeki tuşlarla yapılan grafik karakterlerle sınırlı değildir. Belleğin özel bir bölümünde sizin oluşturacağınız karakterleri de saklayabilir. Bunlara kullananın belirleyeceği grafik karakterler denir ve her programda bunlardan en çok 21 tane kullanılabilir.

Her karakter mürekkep rengindeki 64 küçük nokta ya da pixel'den oluşur. Bunlar her bir sırada 8 tane olmak üzere sekiz sıra halinde ve her karakter alçak tanımlı çizim çizelgesindeki bir karakterin yerini işgal eder - tipki klavyedeki standart grafik karakterler gibi.

### Bir grafik karakterin hazırlanışı

Önce aşağıda gördüğünüz gibi 8x8'lik bir çizelge hazırlayın. Sonra karakteri bazı kareleri doldurarak biçimlendirin. Bu kareler, mürekkep rengindeki pixel'lerdir. Sonra ya yazarak ya da kafanızdan, dolu kareleri 1, boşları da 0 olarak alın. Şu bir örnek çizimi:



Kullanının belirlediği grafik karakterlerden her biri a ile u (ya da A ile U - hiç fark etmez) arasında bir harfle tanımlanır. Karakteri programlamak için sekiz POKE USR cümlesi vermeniz gerekir. Bunlar BIN ile başlar, ardından da çizelgenin her satırına karşılık 8 tane 1 veya 0'dan oluşan bir ikil rakam ile sona erer. Örümcek karakteri s olsun.

```
POKE USR "A" 11111111
POKE USR "B" 11111111
POKE USR "C" 11111111
POKE USR "D" 11111111
POKE USR "E" 11111111
POKE USR "F" 11111111
POKE USR "G" 11111111
POKE USR "H" 11111111
POKE USR "I" 11111111
POKE USR "J" 11111111
POKE USR "K" 11111111
POKE USR "L" 11111111
POKE USR "M" 11111111
POKE USR "N" 11111111
POKE USR "O" 11111111
POKE USR "P" 11111111
POKE USR "Q" 11111111
POKE USR "R" 11111111
POKE USR "S" 11111111
POKE USR "T" 11111111
POKE USR "U" 11111111
```

Şimdi programı işleme koyun, ardından GRAPH ve s'ye basın. Ekranda s yerine bir örümcek belirir. Bir de şu satırları ekleyin. 30. satırdaki örümceği de anlattığımız yoldan elde edin. Programı işleme koyun. Bütün ekranı örümcekler kaplar.

### ÖRÜMCEKLER

```
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
POKE USR "S" 11111111
```



Kendi karakterlerinizi oluştururken, unutmayın, bunları tanımlayan programı işleme koymadıkça bunları ekranda göremeyeceksiniz. O zamana kadar bunlar harflerden oluşan sıralamalar halinde kalacaktır.

### Renklerin benekli karelerle "kanştırılması"

Spectrum'unuzda renkleri kolayca kanştırabilirsiniz. Bunu yapmak için bir kareyi yüzde 50'si mürekkep, yüzde 50'si kağıt renginde doldurarak bir karakter oluşturmak gerekir.

```
POKE USR "A" 11111111
POKE USR "B" 11111111
POKE USR "C" 11111111
POKE USR "D" 11111111
POKE USR "E" 11111111
POKE USR "F" 11111111
POKE USR "G" 11111111
POKE USR "H" 11111111
POKE USR "I" 11111111
POKE USR "J" 11111111
POKE USR "K" 11111111
POKE USR "L" 11111111
POKE USR "M" 11111111
POKE USR "N" 11111111
POKE USR "O" 11111111
POKE USR "P" 11111111
POKE USR "Q" 11111111
POKE USR "R" 11111111
POKE USR "S" 11111111
POKE USR "T" 11111111
POKE USR "U" 11111111
```

Bütün yapmanız gereken, iki pixel satırı belirleyip bilgisayara, bunları karakterde dönüşü kullanma talimatı vermektir. Programı işleme koyunca benekli bir kare göreceksiniz. Eğer aynı tekniği renk komutları içeren bir programda kullanırsanız, benekli kare, programın mürekkep ve kağıt renklerinin karışımını yansıttacaktır.

### READ ve DATA ile çizimleri basitleştirme

Bilgisayar grafiklerinizi oluşturmanın daha kolay bir yolu vardır. O da READ ve DATA ile ondalık sayılar kullanmaktır. Önce 0 ile 1'lerden oluşan sekiz ikil sayıyı ondalık sayıya çevirir. Bunu PRINT BIN ardından sayıyı vererek yapar. Örneğin

### PRINT BIN 00111100

Spectrum'da 60 görünür, bu 00111100'in ondalık halidir. Örümceğin satırlarının ondalık sayıları 60, 126, 219, 255, 189, 165, 165 ve 36'dır.

Artık READ ve DATA'yı kullanabilirsiniz. Bu iki komut, bir programa değişken olarak rakam gibi birçok değeri kolayca verebilmeyi sağlar. READ'i bir değişken izler. Bu, eğer rakamsa, bir veya birkaç harf, bir dizi ise \$'i izleyen tek bir harftir. Burada ondalık sayıları okumak (READ) istediğiniz göre, rakamsal bir değişken gerekir. Buna y dıyelim.

Spectrum'unuz READ ile karıştırsanız, programdaki ilk DATA (veri) cümlesine bakar. Bu cümle, virgüllerle ayrılmış bir dizi değere sahiptir. Bilgisayar buradaki ilk değeri alır, READ'e geldiğinde ise ikinci değeri verir ve bu sırayla böyle gider.

İşte size örümcek yapacak yeni bir program.

```
POKE USR "A" 11111111
POKE USR "B" 11111111
POKE USR "C" 11111111
POKE USR "D" 11111111
POKE USR "E" 11111111
POKE USR "F" 11111111
POKE USR "G" 11111111
POKE USR "H" 11111111
POKE USR "I" 11111111
POKE USR "J" 11111111
POKE USR "K" 11111111
POKE USR "L" 11111111
POKE USR "M" 11111111
POKE USR "N" 11111111
POKE USR "O" 11111111
POKE USR "P" 11111111
POKE USR "Q" 11111111
POKE USR "R" 11111111
POKE USR "S" 11111111
POKE USR "T" 11111111
POKE USR "U" 11111111
```

Aslında bu program bir karakter oluşturmak üzere belleğe herhangi sekiz ondalık sayıyı yerleştirir. 30. satırdaki \$'yi istediğiniz harfe çevirir. 50. satırdaki DATA dan sonra herbin virgüliyle ayrılmış sekiz sayı yazılır. Programı işleme koyduktan sonra GRAPH ile seçtiğiniz harfe basın.

### Renk kontrol kodlarının kullanımı

INK (mürekkep) ve PAPER (kağıt) gibi komutları kullanırsanız, PRINT cümlelerinde ilk tırnaktan sonra kontrol kodlarını koyabilirsiniz.

Ekrana yansıtacak karakterler, bu durumda, sıralamada ekranda alacakları renkte görünürler. Kodları elde etmek için önce

EXTEND MODE'a sonra da CAP SHIFT ile birlikte veya tek başına bir rakam tuşuna basın. Bu çizelge herhangi bir rengi nasıl seçeceğinizi gösteriyor:

BAŞ	1	2	3	4	5	6	7	8	9	0
MAVİ	KIRMIZI	MOR	YEŞİL	A. MAVİ	SARİ	BEYAZ	FLAŞ	FLAŞ	FLAŞ	FLAŞ
MUR	MUR	MUR	MUR	MUR	MUR	MUR	MUR	MUR	MUR	MUR
MAVİ	KIRMIZI	MOR	YEŞİL	A. MAVİ	SARİ	BEYAZ	BRIGHT	BRIGHT	BRIGHT	BRIGHT
KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT	KAGIT

### Satranç tahtası çizimi

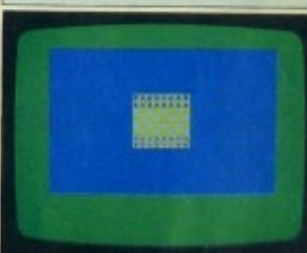
Bu, ekrana satranç tahtası çizip, taşlarını oynamaya hazır yerleştiren bir programdır. Aşağıdaki çerçevede gibi, renk kontrol kodlarını kullanarak, renklerin bir sıralamada yer almasını sağlayarak bunu yapabilirsiniz.

### SATRANÇ

```
POKE USR "A" 11111111
POKE USR "B" 11111111
POKE USR "C" 11111111
POKE USR "D" 11111111
POKE USR "E" 11111111
POKE USR "F" 11111111
POKE USR "G" 11111111
POKE USR "H" 11111111
POKE USR "I" 11111111
POKE USR "J" 11111111
POKE USR "K" 11111111
POKE USR "L" 11111111
POKE USR "M" 11111111
POKE USR "N" 11111111
POKE USR "O" 11111111
POKE USR "P" 11111111
POKE USR "Q" 11111111
POKE USR "R" 11111111
POKE USR "S" 11111111
POKE USR "T" 11111111
POKE USR "U" 11111111
```

500. satırda başlayan alt program oyun taşlarını belirler.

```
POKE USR "A" 11111111
POKE USR "B" 11111111
POKE USR "C" 11111111
POKE USR "D" 11111111
POKE USR "E" 11111111
POKE USR "F" 11111111
POKE USR "G" 11111111
POKE USR "H" 11111111
POKE USR "I" 11111111
POKE USR "J" 11111111
POKE USR "K" 11111111
POKE USR "L" 11111111
POKE USR "M" 11111111
POKE USR "N" 11111111
POKE USR "O" 11111111
POKE USR "P" 11111111
POKE USR "Q" 11111111
POKE USR "R" 11111111
POKE USR "S" 11111111
POKE USR "T" 11111111
POKE USR "U" 11111111
```





## HAREKETLİ GÖRÜNTÜLER

Bilgisayar çizimleri, eğer karakterler ve çizimler ekranda hareket edebilirse çok daha hoş görünür. Spectrum'unuzda bu hareketliliği yaratmak çok zor değildir. Bütün yapmanız gereken, bir karakterin ya da çizimin bulunduğu yeni sürekli değiştirmektir. Bunu yapmanın en iyi yolu da bir ya da daha fazla FOR NEXT döngüsü kullanmaktır.

### Dikey ve yatay hareket

Bu programı yazıp, işleme koyun. Eğer 32. sayfadaki örümcek gibi grafik karakteri yaptıktan sonra, Spectrum'unuzda kapatmadınızsa ya da reset'e getirmedinizse 10 ile 50 arasındaki satırları yazmanız gerekmez. Bu grafik karakterin hâlâ bellekte "s" harfinde bulunması gerekir.

### İNEN ÖRÜMCEK

```
10 BORDER 3: PAPER 8: CLS
20 FOR C=0 TO 7
30 GOTO 40
40 FOR X=0 TO 10
50 GOTO 60
60 FOR Y=0 TO 10
70 GOTO 80
80 FOR Z=0 TO 10
90 GOTO 90
100 GOTO 110
110 GOTO 120
120 GOTO 130
130 GOTO 140
140 GOTO 150
150 GOTO 160
160 GOTO 170
170 GOTO 180
180 GOTO 190
190 GOTO 200
200 GOTO 210
210 GOTO 220
220 GOTO 230
230 GOTO 240
240 GOTO 250
250 GOTO 260
260 GOTO 270
270 GOTO 280
280 GOTO 290
290 GOTO 300
300 GOTO 310
310 GOTO 320
320 GOTO 330
330 GOTO 340
340 GOTO 350
350 GOTO 360
360 GOTO 370
370 GOTO 380
380 GOTO 390
390 GOTO 400
400 GOTO 410
410 GOTO 420
420 GOTO 430
430 GOTO 440
440 GOTO 450
450 GOTO 460
460 GOTO 470
470 GOTO 480
480 GOTO 490
490 GOTO 500
500 GOTO 510
510 GOTO 520
520 GOTO 530
530 GOTO 540
540 GOTO 550
550 GOTO 560
560 GOTO 570
570 GOTO 580
580 GOTO 590
590 GOTO 600
600 GOTO 610
610 GOTO 620
620 GOTO 630
630 GOTO 640
640 GOTO 650
650 GOTO 660
660 GOTO 670
670 GOTO 680
680 GOTO 690
690 GOTO 700
700 GOTO 710
710 GOTO 720
720 GOTO 730
730 GOTO 740
740 GOTO 750
750 GOTO 760
760 GOTO 770
770 GOTO 780
780 GOTO 790
790 GOTO 800
800 GOTO 810
810 GOTO 820
820 GOTO 830
830 GOTO 840
840 GOTO 850
850 GOTO 860
860 GOTO 870
870 GOTO 880
880 GOTO 890
890 GOTO 900
900 GOTO 910
910 GOTO 920
920 GOTO 930
930 GOTO 940
940 GOTO 950
950 GOTO 960
960 GOTO 970
970 GOTO 980
980 GOTO 990
990 GOTO 1000
1000 GOTO 1010
1010 GOTO 1020
1020 GOTO 1030
1030 GOTO 1040
1040 GOTO 1050
1050 GOTO 1060
1060 GOTO 1070
1070 GOTO 1080
1080 GOTO 1090
1090 GOTO 1100
1100 GOTO 1110
1110 GOTO 1120
1120 GOTO 1130
1130 GOTO 1140
1140 GOTO 1150
1150 GOTO 1160
1160 GOTO 1170
1170 GOTO 1180
1180 GOTO 1190
1190 GOTO 1200
1200 GOTO 1210
1210 GOTO 1220
1220 GOTO 1230
1230 GOTO 1240
1240 GOTO 1250
1250 GOTO 1260
1260 GOTO 1270
1270 GOTO 1280
1280 GOTO 1290
1290 GOTO 1300
1300 GOTO 1310
1310 GOTO 1320
1320 GOTO 1330
1330 GOTO 1340
1340 GOTO 1350
1350 GOTO 1360
1360 GOTO 1370
1370 GOTO 1380
1380 GOTO 1390
1390 GOTO 1400
1400 GOTO 1410
1410 GOTO 1420
1420 GOTO 1430
1430 GOTO 1440
1440 GOTO 1450
1450 GOTO 1460
1460 GOTO 1470
1470 GOTO 1480
1480 GOTO 1490
1490 GOTO 1500
1500 GOTO 1510
1510 GOTO 1520
1520 GOTO 1530
1530 GOTO 1540
1540 GOTO 1550
1550 GOTO 1560
1560 GOTO 1570
1570 GOTO 1580
1580 GOTO 1590
1590 GOTO 1600
1600 GOTO 1610
1610 GOTO 1620
1620 GOTO 1630
1630 GOTO 1640
1640 GOTO 1650
1650 GOTO 1660
1660 GOTO 1670
1670 GOTO 1680
1680 GOTO 1690
1690 GOTO 1700
1700 GOTO 1710
1710 GOTO 1720
1720 GOTO 1730
1730 GOTO 1740
1740 GOTO 1750
1750 GOTO 1760
1760 GOTO 1770
1770 GOTO 1780
1780 GOTO 1790
1790 GOTO 1800
1800 GOTO 1810
1810 GOTO 1820
1820 GOTO 1830
1830 GOTO 1840
1840 GOTO 1850
1850 GOTO 1860
1860 GOTO 1870
1870 GOTO 1880
1880 GOTO 1890
1890 GOTO 1900
1900 GOTO 1910
1910 GOTO 1920
1920 GOTO 1930
1930 GOTO 1940
1940 GOTO 1950
1950 GOTO 1960
1960 GOTO 1970
1970 GOTO 1980
1980 GOTO 1990
1990 GOTO 2000
2000 GOTO 2010
2010 GOTO 2020
2020 GOTO 2030
2030 GOTO 2040
2040 GOTO 2050
2050 GOTO 2060
2060 GOTO 2070
2070 GOTO 2080
2080 GOTO 2090
2090 GOTO 2100
2100 GOTO 2110
2110 GOTO 2120
2120 GOTO 2130
2130 GOTO 2140
2140 GOTO 2150
2150 GOTO 2160
2160 GOTO 2170
2170 GOTO 2180
2180 GOTO 2190
2190 GOTO 2200
2200 GOTO 2210
2210 GOTO 2220
2220 GOTO 2230
2230 GOTO 2240
2240 GOTO 2250
2250 GOTO 2260
2260 GOTO 2270
2270 GOTO 2280
2280 GOTO 2290
2290 GOTO 2300
2300 GOTO 2310
2310 GOTO 2320
2320 GOTO 2330
2330 GOTO 2340
2340 GOTO 2350
2350 GOTO 2360
2360 GOTO 2370
2370 GOTO 2380
2380 GOTO 2390
2390 GOTO 2400
2400 GOTO 2410
2410 GOTO 2420
2420 GOTO 2430
2430 GOTO 2440
2440 GOTO 2450
2450 GOTO 2460
2460 GOTO 2470
2470 GOTO 2480
2480 GOTO 2490
2490 GOTO 2500
2500 GOTO 2510
2510 GOTO 2520
2520 GOTO 2530
2530 GOTO 2540
2540 GOTO 2550
2550 GOTO 2560
2560 GOTO 2570
2570 GOTO 2580
2580 GOTO 2590
2590 GOTO 2600
2600 GOTO 2610
2610 GOTO 2620
2620 GOTO 2630
2630 GOTO 2640
2640 GOTO 2650
2650 GOTO 2660
2660 GOTO 2670
2670 GOTO 2680
2680 GOTO 2690
2690 GOTO 2700
2700 GOTO 2710
2710 GOTO 2720
2720 GOTO 2730
2730 GOTO 2740
2740 GOTO 2750
2750 GOTO 2760
2760 GOTO 2770
2770 GOTO 2780
2780 GOTO 2790
2790 GOTO 2800
2800 GOTO 2810
2810 GOTO 2820
2820 GOTO 2830
2830 GOTO 2840
2840 GOTO 2850
2850 GOTO 2860
2860 GOTO 2870
2870 GOTO 2880
2880 GOTO 2890
2890 GOTO 2900
2900 GOTO 2910
2910 GOTO 2920
2920 GOTO 2930
2930 GOTO 2940
2940 GOTO 2950
2950 GOTO 2960
2960 GOTO 2970
2970 GOTO 2980
2980 GOTO 2990
2990 GOTO 3000
3000 GOTO 3010
3010 GOTO 3020
3020 GOTO 3030
3030 GOTO 3040
3040 GOTO 3050
3050 GOTO 3060
3060 GOTO 3070
3070 GOTO 3080
3080 GOTO 3090
3090 GOTO 3100
3100 GOTO 3110
3110 GOTO 3120
3120 GOTO 3130
3130 GOTO 3140
3140 GOTO 3150
3150 GOTO 3160
3160 GOTO 3170
3170 GOTO 3180
3180 GOTO 3190
3190 GOTO 3200
3200 GOTO 3210
3210 GOTO 3220
3220 GOTO 3230
3230 GOTO 3240
3240 GOTO 3250
3250 GOTO 3260
3260 GOTO 3270
3270 GOTO 3280
3280 GOTO 3290
3290 GOTO 3300
3300 GOTO 3310
3310 GOTO 3320
3320 GOTO 3330
3330 GOTO 3340
3340 GOTO 3350
3350 GOTO 3360
3360 GOTO 3370
3370 GOTO 3380
3380 GOTO 3390
3390 GOTO 3400
3400 GOTO 3410
3410 GOTO 3420
3420 GOTO 3430
3430 GOTO 3440
3440 GOTO 3450
3450 GOTO 3460
3460 GOTO 3470
3470 GOTO 3480
3480 GOTO 3490
3490 GOTO 3500
3500 GOTO 3510
3510 GOTO 3520
3520 GOTO 3530
3530 GOTO 3540
3540 GOTO 3550
3550 GOTO 3560
3560 GOTO 3570
3570 GOTO 3580
3580 GOTO 3590
3590 GOTO 3600
3600 GOTO 3610
3610 GOTO 3620
3620 GOTO 3630
3630 GOTO 3640
3640 GOTO 3650
3650 GOTO 3660
3660 GOTO 3670
3670 GOTO 3680
3680 GOTO 3690
3690 GOTO 3700
3700 GOTO 3710
3710 GOTO 3720
3720 GOTO 3730
3730 GOTO 3740
3740 GOTO 3750
3750 GOTO 3760
3760 GOTO 3770
3770 GOTO 3780
3780 GOTO 3790
3790 GOTO 3800
3800 GOTO 3810
3810 GOTO 3820
3820 GOTO 3830
3830 GOTO 3840
3840 GOTO 3850
3850 GOTO 3860
3860 GOTO 3870
3870 GOTO 3880
3880 GOTO 3890
3890 GOTO 3900
3900 GOTO 3910
3910 GOTO 3920
3920 GOTO 3930
3930 GOTO 3940
3940 GOTO 3950
3950 GOTO 3960
3960 GOTO 3970
3970 GOTO 3980
3980 GOTO 3990
3990 GOTO 4000
4000 GOTO 4010
4010 GOTO 4020
4020 GOTO 4030
4030 GOTO 4040
4040 GOTO 4050
4050 GOTO 4060
4060 GOTO 4070
4070 GOTO 4080
4080 GOTO 4090
4090 GOTO 4100
4100 GOTO 4110
4110 GOTO 4120
4120 GOTO 4130
4130 GOTO 4140
4140 GOTO 4150
4150 GOTO 4160
4160 GOTO 4170
4170 GOTO 4180
4180 GOTO 4190
4190 GOTO 4200
4200 GOTO 4210
4210 GOTO 4220
4220 GOTO 4230
4230 GOTO 4240
4240 GOTO 4250
4250 GOTO 4260
4260 GOTO 4270
4270 GOTO 4280
4280 GOTO 4290
4290 GOTO 4300
4300 GOTO 4310
4310 GOTO 4320
4320 GOTO 4330
4330 GOTO 4340
4340 GOTO 4350
4350 GOTO 4360
4360 GOTO 4370
4370 GOTO 4380
4380 GOTO 4390
4390 GOTO 4400
4400 GOTO 4410
4410 GOTO 4420
4420 GOTO 4430
4430 GOTO 4440
4440 GOTO 4450
4450 GOTO 4460
4460 GOTO 4470
4470 GOTO 4480
4480 GOTO 4490
4490 GOTO 4500
4500 GOTO 4510
4510 GOTO 4520
4520 GOTO 4530
4530 GOTO 4540
4540 GOTO 4550
4550 GOTO 4560
4560 GOTO 4570
4570 GOTO 4580
4580 GOTO 4590
4590 GOTO 4600
4600 GOTO 4610
4610 GOTO 4620
4620 GOTO 4630
4630 GOTO 4640
4640 GOTO 4650
4650 GOTO 4660
4660 GOTO 4670
4670 GOTO 4680
4680 GOTO 4690
4690 GOTO 4700
4700 GOTO 4710
4710 GOTO 4720
4720 GOTO 4730
4730 GOTO 4740
4740 GOTO 4750
4750 GOTO 4760
4760 GOTO 4770
4770 GOTO 4780
4780 GOTO 4790
4790 GOTO 4800
4800 GOTO 4810
4810 GOTO 4820
4820 GOTO 4830
4830 GOTO 4840
4840 GOTO 4850
4850 GOTO 4860
4860 GOTO 4870
4870 GOTO 4880
4880 GOTO 4890
4890 GOTO 4900
4900 GOTO 4910
4910 GOTO 4920
4920 GOTO 4930
4930 GOTO 4940
4940 GOTO 4950
4950 GOTO 4960
4960 GOTO 4970
4970 GOTO 4980
4980 GOTO 4990
4990 GOTO 5000
5000 GOTO 5010
5010 GOTO 5020
5020 GOTO 5030
5030 GOTO 5040
5040 GOTO 5050
5050 GOTO 5060
5060 GOTO 5070
5070 GOTO 5080
5080 GOTO 5090
5090 GOTO 5100
5100 GOTO 5110
5110 GOTO 5120
5120 GOTO 5130
5130 GOTO 5140
5140 GOTO 5150
5150 GOTO 5160
5160 GOTO 5170
5170 GOTO 5180
5180 GOTO 5190
5190 GOTO 5200
5200 GOTO 5210
5210 GOTO 5220
5220 GOTO 5230
5230 GOTO 5240
5240 GOTO 5250
5250 GOTO 5260
5260 GOTO 5270
5270 GOTO 5280
5280 GOTO 5290
5290 GOTO 5300
5300 GOTO 5310
5310 GOTO 5320
5320 GOTO 5330
5330 GOTO 5340
5340 GOTO 5350
5350 GOTO 5360
5360 GOTO 5370
5370 GOTO 5380
5380 GOTO 5390
5390 GOTO 5400
5400 GOTO 5410
5410 GOTO 5420
5420 GOTO 5430
5430 GOTO 5440
5440 GOTO 5450
5450 GOTO 5460
5460 GOTO 5470
5470 GOTO 5480
5480 GOTO 5490
5490 GOTO 5500
5500 GOTO 5510
5510 GOTO 5520
5520 GOTO 5530
5530 GOTO 5540
5540 GOTO 5550
5550 GOTO 5560
5560 GOTO 5570
5570 GOTO 5580
5580 GOTO 5590
5590 GOTO 5600
5600 GOTO 5610
5610 GOTO 5620
5620 GOTO 5630
5630 GOTO 5640
5640 GOTO 5650
5650 GOTO 5660
5660 GOTO 5670
5670 GOTO 5680
5680 GOTO 5690
5690 GOTO 5700
5700 GOTO 5710
5710 GOTO 5720
5720 GOTO 5730
5730 GOTO 5740
5740 GOTO 5750
5750 GOTO 5760
5760 GOTO 5770
5770 GOTO 5780
5780 GOTO 5790
5790 GOTO 5800
5800 GOTO 5810
5810 GOTO 5820
5820 GOTO 5830
5830 GOTO 5840
5840 GOTO 5850
5850 GOTO 5860
5860 GOTO 5870
5870 GOTO 5880
5880 GOTO 5890
5890 GOTO 5900
5900 GOTO 5910
5910 GOTO 5920
5920 GOTO 5930
5930 GOTO 5940
5940 GOTO 5950
5950 GOTO 5960
5960 GOTO 5970
5970 GOTO 5980
5980 GOTO 5990
5990 GOTO 6000
6000 GOTO 6010
6010 GOTO 6020
6020 GOTO 6030
6030 GOTO 6040
6040 GOTO 6050
6050 GOTO 6060
6060 GOTO 6070
6070 GOTO 6080
6080 GOTO 6090
6090 GOTO 6100
6100 GOTO 6110
6110 GOTO 6120
6120 GOTO 6130
6130 GOTO 6140
6140 GOTO 6150
6150 GOTO 6160
6160 GOTO 6170
6170 GOTO 6180
6180 GOTO 6190
6190 GOTO 6200
6200 GOTO 6210
6210 GOTO 6220
6220 GOTO 6230
6230 GOTO 6240
6240 GOTO 6250
6250 GOTO 6260
6260 GOTO 6270
6270 GOTO 6280
6280 GOTO 6290
6290 GOTO 6300
6300 GOTO 6310
6310 GOTO 6320
6320 GOTO 6330
6330 GOTO 6340
6340 GOTO 6350
6350 GOTO 6360
6360 GOTO 6370
6370 GOTO 6380
6380 GOTO 6390
6390 GOTO 6400
6400 GOTO 6410
6410 GOTO 6420
6420 GOTO 6430
6430 GOTO 6440
6440 GOTO 6450
6450 GOTO 6460
6460 GOTO 6470
6470 GOTO 6480
6480 GOTO 6490
6490 GOTO 6500
6500 GOTO 6510
6510 GOTO 6520
6520 GOTO 6530
6530 GOTO 6540
6540 GOTO 6550
6550 GOTO 6560
6560 GOTO 6570
6570 GOTO 6580
6580 GOTO 6590
6590 GOTO 6600
6600 GOTO 6610
6610 GOTO 6620
6620 GOTO 6630
6630 GOTO 6640
6640 GOTO 6650
6650 GOTO 6660
6660 GOTO 6670
6670 GOTO 6680
6680 GOTO 6690
6690 GOTO 6700
6700 GOTO 6710
6710 GOTO 6720
6720 GOTO 6730
6730 GOTO 6740
6740 GOTO 6750
6750 GOTO 6760
6760 GOTO 6770
6770 GOTO 6780
6780 GOTO 6790
6790 GOTO 6800
6800 GOTO 6810
6810 GOTO 6820
6820 GOTO 6830
6830 GOTO 6840
6840 GOTO 6850
6850 GOTO 6860
6860 GOTO 6870
6870 GOTO 6880
6880 GOTO 6890
6890 GOTO 6900
6900 GOTO 6910
6910 GOTO 6920
6920 GOTO 6930
6930 GOTO 6940
6940 GOTO 6950
6950 GOTO 6960
6960 GOTO 6970
6970 GOTO 6980
6980 GOTO 6990
6990 GOTO 7000
7000 GOTO 7010
7010 GOTO 7020
7020 GOTO 7030
7030 GOTO 7040
7040 GOTO 7050
7050 GOTO 7060
7060 GOTO 7070
7070 GOTO 7080
7080 GOTO 7090
7090 GOTO 7100
7100 GOTO 7110
7110 GOTO 7120
7120 GOTO 7130
7130 GOTO 7140
7140 GOTO 7150
7150 GOTO 7160
7160 GOTO 7170
7170 GOTO 7180
7180 GOTO 7190
7190 GOTO 7200
7200 GOTO 7210
7210 GOTO 7220
7220 GOTO 7230
7230 GOTO 7240
7240 GOTO 7250
7250 GOTO 7260
7260 GOTO 7270
7270 GOTO 7280
7280 GOTO 7290
7290 GOTO 7300
7300 GOTO 7310
7310 GOTO 7320
7320 GOTO 7330
7330 GOTO 7340
7340 GOTO 7350
7350 GOTO 7360
7360 GOTO 7370
7370 GOTO 7380
7380 GOTO 7390
7390 GOTO 7400
7400 GOTO 7410
7410 GOTO 7420
7420 GOTO 7430
7430 GOTO 7440
7440 GOTO 7450
7450 GOTO 7460
7460 GOTO 7470
7470 GOTO 7480
7480 GOTO 7490
7490 GOTO 7500
7500 GOTO 7510
7510 GOTO 7520
7520 GOTO 7530
7530 GOTO 7540
7540 GOTO 7550
7550 GOTO 7560
7560 GOTO 7570
7570 GOTO 7580
7580 GOTO 7590
7590 GOTO 7600
7600 GOTO 7610
7610 GOTO 7620
7620 GOTO 7630
7630 GOTO 7640
7640 GOTO 7650
7650 GOTO 7660
7660 GOTO 7670
7670 GOTO 7680
7680 GOTO 7690
7690 GOTO 7700
7700 GOTO 7710
7710 GOTO 7720
7720 GOTO 7730
7730 GOTO 7740
7740 GOTO 7750
7750 GOTO 7760
7760 GOTO 7770
7770 GOTO 7780
7780 GOTO 7790
7790 GOTO 7800
7800 GOTO 7810
7810 GOTO 7820
7820 GOTO 7830
7830 GOTO 7840
7840 GOTO 7850
7850 GOTO 7860
7860 GOTO 7870
7870 GOTO 7880
7880 GOTO 7890
7890 GOTO 7900
7900 GOTO 7910
7910 GOTO 7920
7920 GOTO 7930
7930 GOTO 7940
7940 GOTO 7950
7950 GOTO 7960
7960 GOTO 7970
7970 GOTO 7980
7980 GOTO 7990
7990 GOTO 8000
8000 GOTO 8010
8010 GOTO 8020
8020 GOTO 8030
8030 GOTO 8040
8040 GOTO 8050
8050 GOTO 8060
8060 GOTO 8070
8070 GOTO 8080
8080 GOTO 8090
8090 GOTO 8100
8100 GOTO 8110
8110 GOTO 8120
8120 GOTO 8130
8130 GOTO 8140
8140 GOTO 8150
8150 GOTO 8160
8160 GOTO 8170
8170 GOTO 8180
8180 GOTO 8190
8190 GOTO 8200
8200 GOTO 8210
8210 GOTO 8220
8220 GOTO 8230
8230 GOTO 8240
8240 GOTO 8250
8250 GOTO 8260
8260 GOTO 8270
8270 GOTO 8280
8280 GOTO 8290
8290 GOTO 8300
8300 GOTO 8310
8310 GOTO 8320
8320 GOTO 8330
8330 GOTO 8340
8340 GOTO 8350
8350 GOTO 8360
8360 GOTO 8370
8370 GOTO 8380
8380 GOTO 8390
8390 GOTO 8400
8400 GOTO 8410
8410 GOTO 8420
8420 GOTO 8430
8430 GOTO 8440
8440 GOTO 8450
8450 GOTO 8460
8460 GOTO 8470
8470 GOTO 8480
8480 GOTO 8490
8490 GOTO 8500
8500 GOTO 8510
8510 GOTO 8520
8520 GOTO 8530
8530 GOTO 8540
8540 GOTO 8550
8550 GOTO 8560
8560 GOTO 8570
8570 GOTO 8580
8580 GOTO 8590
8590 GOTO 8600
8600 GOTO 8610
8610 GOTO 8620
8620 GOTO 8630
8630 GOTO 8640
8640 GOTO 8650
8650 GOTO 8660
8660 GOTO 8670
8670 GOTO 8680
8680 GOTO 8690
8690 GOTO 8700
8700 GOTO 8710
8710 GOTO 8720
8720 GOTO 8730
8730 GOTO 8740
8740 GOTO 8750
8750 GOTO 8760
8760 GOTO 8770
8770 GOTO 8780
8780 GOTO 8790
8790 GOTO 8800
8800 GOTO 8810
8810 GOTO 8820
8820 GOTO 8830
8830 GOTO 8840
8840 GOTO 8850
8850 GOTO 8860
8860 GOTO 8870
8870 GOTO 8880
8880 GOTO 8890
8890 GOTO 8900
8900 GOTO 8910
8910 GOTO 8920
8920 GOTO 8930
8930 GOTO 8940
8940 GOTO 8950
8950 GOTO 8960
8960 GOTO 8970
8970 GOTO 8980
8980 GOTO 8990
8990 GOTO 9000
9000 GOTO 9010
9010 GOTO 9020
9020 GOTO 9030
9030 GOTO 9040
9040 GOTO 9050
9050 GOTO 9060
9060 GOTO 9070
9070 GOTO 9080
9080 GOTO 9090
9090 GOTO 9100
9100 GOTO 9110
9110 GOTO 9120
9120 GOTO 9130
9130 GOTO 9140
9140 GOTO 9150
9150 GOTO 9160
9160 GOTO 9170
9170 GOTO 9180
9180 GOTO 9190
9190 GOTO 9200
9200 GOTO 9210
9210 GOTO 9220
9220 GOTO 9230
9230 GOTO 9240
9240 GOTO 9250
9250 GOTO 9260
9260 GOTO 9270
9270 GOTO 9280
9280 GOTO 9290
9290 GOTO 9300
9300 GOTO 9310
9310 GOTO 9320
9320 GOTO 9330
9330 GOTO 9340
9340 GOTO 9350
9350 GOTO 9360
9360 GOTO 9370
9370 GOTO 9380
9380 GOTO 9390
9390 GOTO 9400
9400 GOTO 9410
9410 GOTO 9420
9420 GOTO 9430
9430 GOTO 9440
9440 GOTO 9450
9450 GOTO 9460
9460 GOTO 9470
9470 GOTO 9480
9480 GOTO 9490
9490 GOTO 9500
9500 GOTO 9510
9510 GOTO 9520
9520 GOTO 9530
9530 GOTO 9540
9540 GOTO 9550
9550 GOTO 9560
9560 GOTO 9570
9570 GOTO 9580
9580 GOTO 9590
9590 GOTO 9600
9600 GOTO 9610
9610 GOTO 9620
9620 GOTO 9630
9630 GOTO 9640
9640 GOTO 9650
9650 GOTO 9660
9660 GOTO 9670
9670 GOTO 9680
9680 GOTO 9690
9690 GOTO 9700
9700 GOTO 9710
9710 GOTO 9720
9720 GOTO 9730
9730 GOTO 9740
9740 GOTO 9750
9750 GOTO 9760
9760 GOTO 9770
9770 GOTO 9780
9780 GOTO 9790
9790 GOTO 9800
9800 GOTO 9810
9810 GOTO 9820
9820 GOTO 9830
9830 GOTO 9840
9840 GOTO 9850
9850 GOTO 9860
9860 GOTO 9870
9870 GOTO 9880
9880 GOTO 9890
9890 GOTO 9900
9900 GOTO 9910
9910 GOTO 9920
9920 GOTO 9930
9930 GOTO 9940
9940 GOTO 9950
9950 GOTO 9960
9960 GOTO 9970
9970 GOTO 9980
9980 GOTO 9990
9990 GOTO 10000
10000 GOTO 10001
10001 GOTO 10002
10002 GOTO 10003
10003 GOTO 10004
10004 GOTO 10005
10005 GOTO 10006
10006 GOTO 10007
10007 GOTO 10008
10008 GOTO 10009
10009 GOTO 10010
10010 GOTO 10011
100
```



## MÜZİK VE ÇEŞİTLİ SESLERİN YAPIMI

ZX Spectrum+'un, programlarınıza çok çeşitli müzikal sesler ve özel ses efektleri ekleyip, canlılık kazandıracak bir ses sentesizeri vardır. Müzik bilginiz az da olsa, hiç olmasa da, kullanımı çok kolaydır. Sentesizerin çıkardığı sesler size Spectrum'un iç oparöründen ulaşır.

### Seslerin programlanması

Spectrum'unuzdan ses çıkartabilmek için tek bir komut yeterlidir: BEEP. Bunu iki sayı veya bunları temsil eden değişkenler izler. İlki bilgisayara bu sesin (saniye olarak) ne kadar süreceğini bildirir, ikincisi ise sesin tınısının ne kadar alçak ya da yüksek olduğunu gösterir. Ortadaki C(do)nin değeri 0, C(do) diyeyizinki 1, B(si) bemolunki -1'dir.

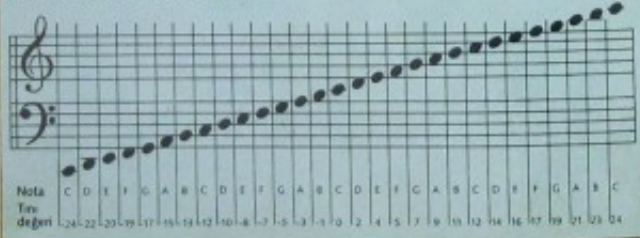
Spectrum'unuzun çıkaracağı bütün sesleri duyabilmek için şu programı verin:

```
10 FOR P=0 TO 60 STEP 1
20 BEEP 0.2:P
30 PRINT AT 0.0: "TİNE 0.010"
40 NEXT P
```

Spectrum en yüksekinden (69) en alçağına (-60) bütün notaları çaldı. En yüksek notaları duyamadığınızı, en alçakların ise

### Müzik yapmada tınısların değeri

İşte Spectrum'un en baştan en tize kadar bütün tınısları. Diyiz için 1 ekleyip, bemol için 1 çıkartın.



tıkırtı gibi geldiğini farkedebilirsiniz. Çünkü bu sesler insan kulağının duyabilme sınırlarının ötesindeki seslerdir. Bu sayfanın altındaki çizelge, notaların tını değerini gösteriyor. Böylece yazılı bir müziği, Spectrum'un programına dönüştürebilirsiniz.

### Ses efektleri

BEEP'i tını değerini hızla değiştirecek bir döngüye yerleştirerek, Spectrum'unuzdan çok çeşitli sesler, efektler elde edebilirsiniz. Bu programları deneyin ve bunlarla kendi seslerinizi geliştirmeye çalışın. Dikkat ederseniz, ses süreleri, saniyenin yüzde biri gibi çok kısa. Döngülü programları durdurmak için BREAK'e basın.

### ÇOŞKULU

```
10 LET P=INT (1000*(RND-0.5))
20 BEEP P/100:P
30 GOTO 10
```

Bu program, üç notalık bir gruba rastsal tınılarda üst üste geliyor. Tını alanı hayli geniş, ama 10. satırdaki değerlerle oynayarak, bu alanı da değiştirebilirsiniz.

### MAKİNA

```
10 FOR K=10 TO 30
20 BEEP -0.1:P
30 BEEP -0.1:24-P
40 NEXT K
```

Bu program, bir giderek yükselen, diğeri alçalan tınıda iki ses çıkıyor. Bunun nedeni iki BEEP cümlesinin, sadece saniyenin yüzde biri kadar aralarla, farklı tınılarda iki ses üst üste çıkması.

### KALKIŞ

```
10 FOR P=10 TO 40 STEP 0.2
20 BEEP P/10:P
30 NEXT P
```

Bu, makina programına benzer, ancak iki nota da birbirine 6 yarımlı nota uzaklıkta birlikte yükselir. Bu arada tını değeri her defasında 0,2 - yarımlı notanın beşte biri - kadar değişir. Bu da sesin tınısının daha yavaş yükselmesine yol açar. STEP değerini değiştirerek, diğer tını değişikliklerini de deneyin.

### KLAVYE DEĞİŞİKLİĞİ

```
10 LET P=CODE INKEY$
20 IF P=0 THEN GOTO 10
30 BEEP P/10:10-30/P/2
40 GOTO 10
```

Bu program sizin herhangi bir tuşa basmanızı bekler. Basınca her biri değişik bir ses çıkarır. Bir tuşa basarken, CAPS SHIFT'e basarsanız, sesin başlangıcını göreceksiniz. Bu program, her tuşa basıldığında CODE INKEY\$, P'nin değerini değiştirdiğinden bu sonucu verir. İkinci satır ise, bir başka tuşa basılmazsa bilgisayarı susturur. 51. sayfadaki karakterler tablosunda, CODE'un vereceği değerleri görebilirsiniz.

### Spectrum'un sesini nasıl yükseltebilirsiniz?

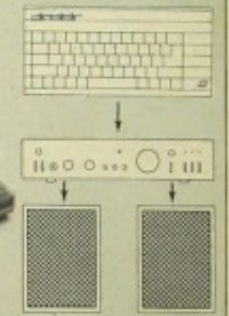
Spectrum'unuzun sesini daha yükseltmek için EAR girişine kulaklık ya da amplifikatör ve oparör bağlayabilirsiniz. Amplifikatörün ses ayar düğmesini oynatarak sesi yükseltip kısımlırsınız.

Bunu yapmanın en kolay yolu, Spectrum'un kaset bağlantısı ile EAR girişini kasetli teypin AMIC girişine bağlamaktır. Gerekliyse kaseti çıkarıp, sonra da teypin PLAY, REWIND (geri sarma) ve FAST FORWARD (ileri

sarma) düğmelerine basın. Kasetli teypin ses düğmesini ayarlarken, bilgisayarı sesinin teypin oparöründen geldiğini duyacaksınız. Ya da kulaklığınızı da kasetli teype takabilirsiniz.

Eğer sesleri tam olarak duymak isterseniz, Spectrum'u bir müzik seline de bağlayabilirsiniz. Bunun için bir ucunda Spectrum'a girecek 3.5mm'lik bir jak-fiş, diğer ucunda müzik setinin girişine takılacak özel bir bağlantı

gerekir. Spectrum, kasetli teypier ya da diğer teypier gibi düz sinyaller ürettiğinden, amplifikatördeki REPLAY ya da LINE IN girişleri kullanılabilir. Eğer bir sorunuz olursa, bu tür malzeme satan bir mağazaya danışın.



### Ses ve Görüntü

Spectrum'un çıkardığı seslerin ekrandaki görüntü ile uyumu çok hoş olur. Programlara nasıl ses ekleyebileceğinizi görmek için 34. sayfadaki Kaçan Örümcek programına dönün.

Unutmayın 135. satıra hareketi yavaşlatmak için bir PAUSE cümlesi eklemişsiniz. Programı böyle yavaşlatacak yerde, ses çıkaracak bir duraklama da programlayabilirsiniz. 135. satırı şöyle değiştirin:

### 135 GOSUB 500

Şimdi de programa şu satırları ekleyin:

```
200 STOP
300 FOR P=40-1 TO 30-1 STEP -1
410 BEEP P/10:P
500 NEXT P
550 RETURN
```

Programı işleme koyun, örümcek aşağı inerken bir vizitli çıkaracaktır. Alt program, örümceğin her bir alttaki pozisyona geçişinde çok hızlı üç nota çalar. 500. satırı ve 510. satırdaki 0.02'yi değiştirip notaları hızlandırıp yavaşlatarak yeni notaları ekleyin.



## PROGRAMLARINIZI NASIL SAKLAYABİLİRSİNİZ?

Bir süre sonra, kendi programlarınızı kasette saklamak isteyeceksiniz. Bunu yapmak için, Spectrum'unuza bir kasetli teyp bağlayın ve

### Kendi programlarınızı kaydedin

- 1 Önce Spectrum'unuzu 14. sayfada anlatıldığı gibi uygun bir kasetli teyp, kaset bağlantısı ile bağlayın. Ancak Spectrum'un sadece MIC girişinin kasetli teyp'e bağlı olmasına dikkat edin.
- 2 Eğer kasetli teypin bir kayıt ses kontrol düğmesi varsa, bunu üçte iki oranında açık tutun. Yoksa, merak etmeyin, ses kayıt düzeyi otomatik olarak ayarlanacaktır.
- 3 SAVE'e basın, ardından da programın adını tırnak içinde yazın-örneğin:

SAVE "prog2"

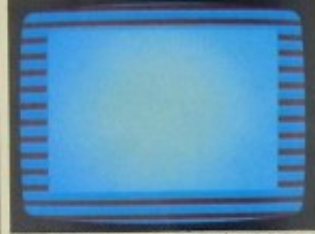


Ad için, on harf ya da rakamdan oluşan herhangi bir bölüm kullanılabilir. ENTER'e basın. SAVE satırı kaybolacak ve ekranda Spectrum'dan gelen kasetli teyp kullanma talimatlarını göreceksiniz.



bilgisayardaki programı burada saklayın. Spectrum, programı teyp'e, kasete kaydedilebilecek biçimde yollar. Bundan sonra programı her kullanmak istediğinizde, bunu 14 ve 15. sayfalarda anlatıldığı gibi bilgisayara yükleyerek kullanabilirsiniz. Bu iki sayfada, programların nasıl saklanabileceğini ve programın doğru olarak kaydedilip edilmediğini nasıl kontrol edebileceğinizi göreceksiniz.

- 4 Teypi, kayıt durumuna getirin. Bu, genellikle RECORD ve PLAY'e aynı anda basarak yapılır.
- 5 Siz beklerken, Spectrum programı aktarılır. Önce ekranın çevresinde aşağı yukarı oynayan mavi-kırmızı çizgiler göreceksiniz.



Sonra kısa süreyle mavi-sarı çizgiler görünür. Bu, Spectrum programın adını teyp'e aktarıldığını gösterir.

- 6 Bunu kısa bir ara, sonra da tekrar mavi-kırmızı çizgiler izler. Ardından, Spectrum programı teyp'e yollarken yeniden mavi-sarı çizgiler belirir. Uzun bir programın aktarımı epey zaman olabilir.



- 7 Program teyp'e aktarıldığında, OK, 0:1 mesajı görünür. Teypi durdurun. Program artık saklanmıştır. Eğer isterseniz bunu kontrol edebilirsiniz veya "doğrultabilirsiniz."

### Programınızı nasıl doğrultabilirsiniz?

Bilgisayarın programı kasetli teyp'e aktarmış olmasına rağmen, programın kasete başıyla kaydedildiğinden emin olamazsınız. Ancak, Spectrum'unuz bunu sizin için kontrol edebilir.

Bu süreç, doğrultma denir. Önce kaseti en başa sarın. Sonra Spectrum'un EAR girişini teypin EAR girişine bağlayın. (İsterseniz MIC girişine de bağlı kalabilir.) Ardından VERIFY tuşuna basıp, tırnak içinde programın adını yazın. ENTER'e basın ve teypi çalışsın. Aynı sırayla mavi-kırmızı ve mavi-sarı çizgilerin görünmesi gerekir. Programın adı da ekranda belirecek ve doğrultma işlemi sonuna kadar ekranda kalacaktır.



İkinci mavi-sarı bölümün sonunda, şu mesaj belirecektir.

OK, 0:1

Bu, Spectrum'unuzun belleğinde olan programı, teyp'teki program ile karşılaştırıp, tastaman aynı bulduğunu gösterir.



Artık programınızın, kasete geçtiğinden kesinlikle emin olabilirsiniz.

### Yazılım saklamada öğütler

1. Saklayacağınız programın adını kasetin etiketine ya da bir karta yazın. Büyük ya da küçük harfleri aynı ekranda görüldüğü gibi kullanın. Eğer teypinizin sayacı varsa, bundan yararlanıp programın yerini saytayı, adının yanına sayıyı yazın.
2. Programı saklamadan önce, adını bir REM cümlesi ile programa ekleyin - örneğin

S REM SPIDER program Version 3

Program işleme konunca, bilgisayar bütün REM cümlelerini gözard eder. Programın istediğiniz yerlerine REM'i kullanarak, görüş ve düğüncelerinizi ekleyebilirsiniz.

Eğer bu mesajı alamazsanız, bir aksilik var demektir. Önce 16. sayfadaki **Yazılım yüklemesindeki sorunlar** tablosuna bakın. Çünkü program kasete aktarılmış ama bir hatadan ötürü, doğrultma için bilgisayara yüklenemiyor olabilir. Eğer burada bir hata varsa, hatayı düzeltin, kaseti başa sarın ve programı yeniden doğrultun. Bilgisayar programı hâlâ doğrulamıyorsa, o zaman bir sonraki sayfadaki **Yazılım saklamadaki sorunlar** tablosuna bakın. Bu arada NEW ya da reset'e basmayın veya bilgisayarı kapatmayın, aksi halde elinizde güvenilir bir kopya olmadan bellekteki programı kaybedersiniz.

### Programın otomatik başlaması

SAVE'in ardından, programın adını ve LINE 1 yazabilirsiniz - örnek:

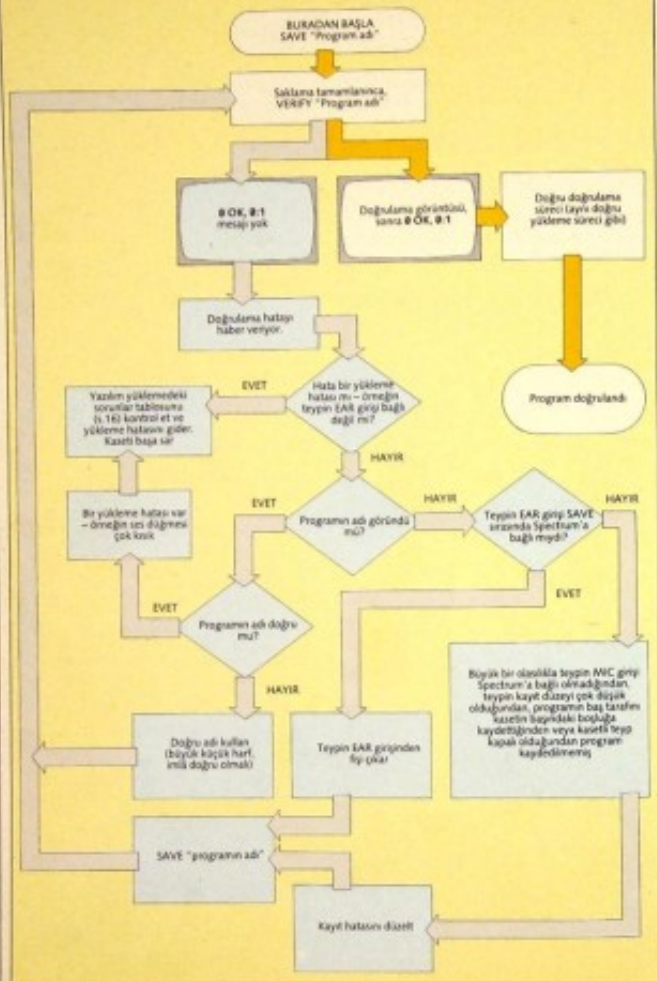
SAVE "SPIDER" LINE 1

Saklama süreci ilkinden farklı değildir. Ancak bu defa doğrulatırken, VERIFY ve programın adından sonra LINE 1'i dahil etmeyin. LINE 1 ile saklanan programlar, yüklendiğinde otomatik olarak başlar. RUN'i kullanmaya gerek yoktur, (ama program başlatmaya teypi durdurmayı unutmayın).

Olan şu: Program 1. satırda başlıyor, ama 1. satır yoksa, bilgisayar hemen programın ilk satırına geçiyor. 1'i bir başka rakama çevirirseniz, program otomatik olarak bu rakamı taşıyan satırdan başlar.

### CODE, SCREEN\$ ve DATA'nın saklanması

SAVE, CODE ve SCREEN\$ ile Spectrum'un belleğinin bir bölümünü ve DATA ile de bir düzünü saklamak için kullanılır. Bunlar için Programının Başvuru Kılavuzuna bakın.



## ZX SPECTRUM+'U TANIYALIM

Bu bölümde, ZX Spectrum + 'un içine girecek, klavyenin altındaki çeşitli öğelerin neye yaradığını ve bilgisayarın çalışabilmesi için bunların birbirine nasıl bağlandığını öğreneceğiz. Ayrıca yanı donanımlar, yani Spectrum'ünüzü tam bir bilgisayar sistemi haline getirecek, eklenebilir parçaları nasıl kullanacağınızı göreceksiniz. Nihayet, burada bilgisayarınızın teknik yönleri –örneğin belleğin düzenleniş, Spectrum'ün teknik özellikleri – hakkında daha çok bilgi bulacaksınız.



## İÇİNDE NELER VAR?

ZX Spectrum + 'unuzu nasıl çalıştırdığını öğrenmek için, içini açıp bakmaya kalkmayın, bu bölümü okuyun. İçini açarsanız, garantiniz geçersiz hale gelir, ciddi zarara neden olabilirsiniz.

Kasanın içinde klavyeyi Spectrum'ün diğer parçalarına bağlayan iki kurdela bağlantı vardır. Bu parçaların hepsi bir tek basılı devre tablosunun üzerine yerleştirilmiştir. Tabloda, direnç, kapasitör gibi standart elektrik parçaları da vardır, ama gen kalkanların çoğu, tek başına ya da bloklar halinde yerleştirilmiş siyah dikdörtgen mikroyongalardır.

### Yonganın içinde

Bir mikroyonganın çalışan bölümü aslında içinde bulunduğu plastik ambalajdan çok daha küçüktür. Bu kaplama, esas olarak yonga için gerekli bağlantıların yapılabilmesini ve bunun devre tablasındaki soketlere takılmasını mümkün kılacak biçimde düzenlenmiştir. Yonganın kendisi ise, üzerinde binlerce elektrik devresi olan çok ince bir silikon dümdür. Her devre, kendisine ulaşan elektrik sinyallerini durduran, aktaran veya depolayan bir elektrik anahtarı görevi yapar. Bu oldukça basit bir işlem olmakla birlikte, aynı anda birlikte hareket eden o kadar çok devre vardır ki, bunlar akıllamaz bir hız ve kesinlikle bilgi depolayan ya da işleyen sinyaller üretebilirler. ZX Spectrum + 'da her biri bilgisayarın işleyişinde belli rol üstlenmek üzere tasarlanmış çok farklı yongalar vardır.

### Yongalar nasıl bağlanır?

Bir bakıma, sonuçta, Spectrum'unuz son derece karmaşık bir elektrik devresidir. Yongaların ve parçaların içindeki ve arasındaki yollardan sürekli gidip gelen kalp atışı gibi elektrik akımlarından oluşan kodlu sinyaller bilgisayarın çalışmasını sağlar.

Peki, nasıl oluyor da, doğru sinyal tam zamanında, tam yerine ulaşabiliyor? Bilgisayarın yongalarından birinin içine gizlenmiş bir saat vardır. Bu, çevreye kalp atışı gibi elektrik akımları saçarak çalışır - saniyede 3,5 milyon atış. Bunlar, düzenli olarak bu devrelerde dolaşarak, her parçanın hareketini denetleyen, her şeyin düzgün akmasını sağlayan kodlu sinyalleri üretir.

**ZX Spectrum + 'unuzun içi**  
Spectrum'un devre tablosunun bu çiziminde, klavye ile bağlantı sağlayan kurdela bağlantılar sökülüştür.

**Spectrum kullanırken, bir tuşa basılması klavyenin altındaki iki kablolu temasını sağlar. Bu da CPU'ya kodlu bir sinyal gönderir.**

**Bağılantısız Mantık Düzeni (ULA)**  
Bu yonga, RAM'da bulunan bilgilerden görüntüler üretir, ayrıca sistem denetçisi olarak görev yapar.

**Klavye bağlantı noktası**  
Klavyeden gelen kurdela bağlantılardan biri buraya bağlanır.

**TV kodlayıcısı**  
Bu, bilgisayarın devrelerinin ürettiği sinyalleri, renkli televizyon sinyallerine çevirir.

**Rastgele Kullanım Belleği (RAM)**  
Bu yongalar, bilgisayarın değerleri gibi, programların gereksinim duyduğu belirli bilgileri bilgisayara besleyen programı içerir. RAM'ın 48 K'lık içeriği klavyeden değiştirilebilir veya reset'e getirilerek ve bilgisayar kapatılarak tımden silinebilir.

**TV çıkışı**  
Bu, televizyon alıcısına giden sinyalleri üretir.

**Teyp soketleri**  
Bunlar, bellekten bilgi ve programların bir teypte gönderilmesi ve teypten yeniden belleğe verilmesi için kullanılır.

**Mantık Yonaları**  
Bu yongalar, CPU ile RAM arasında bilgi akışını denetleyen görevi yapar.

**Merkez İşlem Birimi (CPU)**  
Bilgisayarın "beyni". CPU, bir Z80 mikro işlemcidir. Butün hesapları yapar ve Spectrum'un bütün işlemlerini denetler.

**9 VDC girişi**  
Elektrik bağlantısını sağlar.

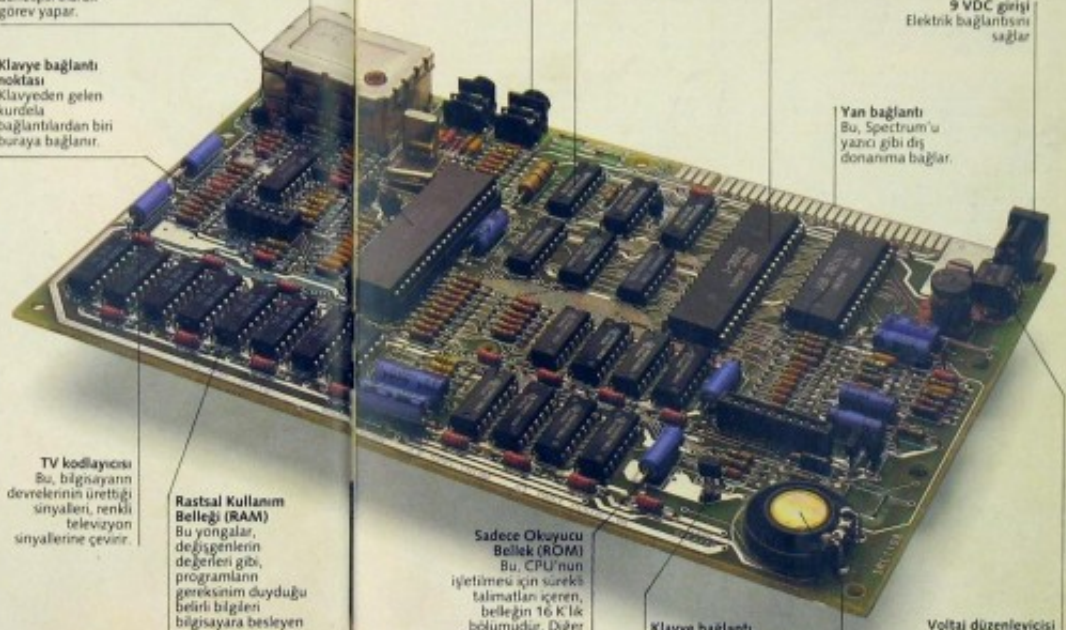
**Yan bağlantı**  
Bu, Spectrum'u yazıcı gibi dış donanıma bağlar.

**Sadece Okuyucu Bellek (ROM)**  
Bu, CPU'nun işletimisi için sürekli talimatları içeren, belleğin 16 K'lık bölümüdür. Diğer işlevlerin yanı sıra bu talimatlar, BASIC komutlarını CPU'nun anlayacağı biçime sokar. Bu bellek yongalarının içeriği, klavyeden değiştirilemez.

**Klavye bağlantı noktası**  
Klavyeden gelen kurdela bağlantılardan biri buraya bağlanır.

**Volajaj düzenleyicisi**  
Bu parça voltajdaki değişimin bilgisayarın etkilemesini önler.

**Oparlör**  
Gerektiğinde ses üretir.



## ZX SPECTRUM+ 'UNUZ NASIL ÇALIŞIR?

Diğer mikrobilgisayarlar gibi, ZX Spectrum + 'un işleyişi dört ana parçaya oluşur. Bunlar, bilgisayar bilgi veya bir program veren klavye gibi *girdi birimleri*; bilgi, program ve işletme talimatları depolayan geçici veya daimi  *bellekler*; bilgiler üzerine program talimatlarını uygulayan *Merkezi İşlem Birimi (CPU)* ve sonucu veren *çıkış birimleri* dir.

### Program verip işleme koyma

Çok basit bir program verip işleme koyunca, Spectrum'un içinde ne olur? İşte tek sabırlık bir örnek.

```
10 PRINT 6+2
```

Önce klavyeyi kullanırsınız. Tuşların altı, kafes gibi tellerle örülmüştür. Her tuşa basışınızda, bir çift tel temas eder ve CPU'ya kodlu bir sinyal gönderir. Bunun üzerine CPU bu kodu depolanacağı RAM'a yollar.

Programı işleme koyunca, CPU, depolanmış kodları, programdaki sırasıyla teker teker RAM'dan alır. Önce, ona ROM'dan belli bir işlem kodu almasını söyleyen PRINT kodunu alır. Bu işlem kodu CPU'ya gider ve CUP, ekranda bir değer

## İkil kodlar

Spectrum'unuzu çalıştıran bütün kodlar ikil biçimdedir. Bunlar iki tür sinyalden oluştuğu için bunlara ikil denir. Bunlar ikil sayılar, yani iki rakamdan 0 ve 1'den oluşan sayılarla ifade edilebilir. Örneğin 6'nın ikil sayısı 00000110'dır.

'Spectrum' unuzun içinde, bu kodlar hızlı elektrik alışlarından oluşan bölümleri haline gelir. Atrış belli bir sürede gelmezse, bu 6'dır. Bu nedenle bilgisayar kodunda 6, *Gelmedi-gelmedi-gelmedi-gelmedi-gelmedi-geldi-geldi-geldi-geldi* dir. Spectrum seki kodu aynı anda işler.

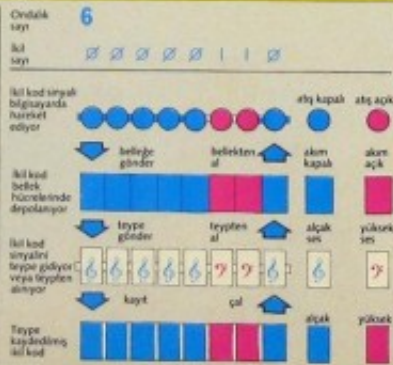
Yandaki şekilde, farklı kil kodların bilgisayar tarafından bilgiyi bir yerden bir başkasına aktarmak için nasıl kullanıldığını görebilirsiniz.

yanıtsızca hareketleri yapmaya hazırlanır. CPU, RAM'dan 6 değeri alır. Bu da, bir tür kod biçimindedir ve CPU bunu "küçük" denen küçük ikili belleğe geçirir. Bunu toplama kodu izler ve CPU gene gerekli işlem kodunu ROM'dan alır. Son olarak da CPU, 2. için RAM'dan bir kod alır. Bu kodu, "Küçük" teki değere ekleyip sonucu (8) elide eder. CPU, sonra bu sonucu başka bir elide kodu çevirip, bunu görüntü kesimine gönderir. Bu, RAM in ekranda gördüğünüz her şeyi için kodları bulunduran kesimdir ve ekranda 8 rakamı görünür.

Bir program depolaması

Spectrum'dan bir programı kasette saklamasını isterseniz, CPU gene kodları RAM'dan alır. Ama buna uygun hareket edecek yerde, kodları ses sinyallerine dönüştürecek bir değişim birimine gönderir. Bu sinyaller sonra teypte yollandı, kasete kaydedilir.

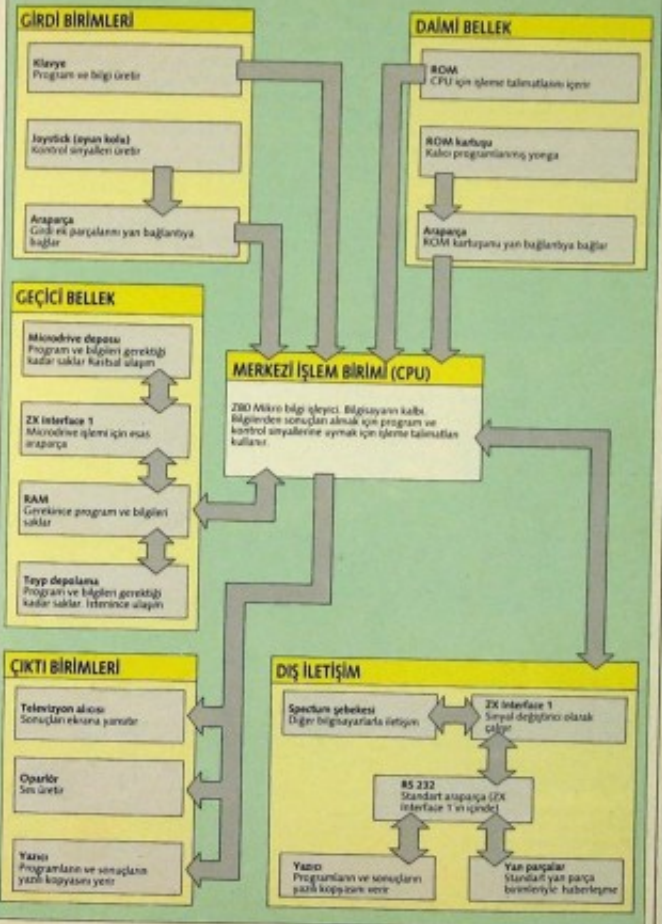
Daha sonra programı yüklediğinizde, teypten gelen ses sinyalleri, değiştirim biriminde bilgisayar kodlarına çevrilir. CPU, bunları gerek olana kadar saklanacakları RAM'a yollar.



Spectrum'un girdi-çıkış yolları

Bu çizelge, kodlanmış bilgilerin, klavye gibi birimlerinden, Spectrum'un işleme sisteminden geçerek, televizyon ekranı gibi çıktı birimlerine

nasıl ulaştığını gösteriyor.  
Tek uçlu oklar, tek yönlü çalışan yolları, çift uçlu  
oklar ise iki yönlü çalışan yolları gösteriyor.





## DONANIM BAĞLANIŞI

Sinclair'in ya da Spectrum'la kullanılabilecek diğer yapımcıların yan donanımlarından yararlanarak ZX Spectrum + 'unuzu tam ve güçlü bir bilgisayar sistemi haline getirebilirsiniz. Bu sistemin merkezini ZX Interface 1 oluşturmaktadır. Programların ve verilerin daha hızlı ve daha kolay işlenmesini sağlayan Microdrive'ları ve diğer Spectrum'lar da dahil olmak üzere çok çeşitli diğer donanımları bilgisayara bağlamanızı bu mümkün kılar. Bu araparçadan yararlanarak, Spectrum'unuzu standart yazıcılara bağlayabilir, telefon hatları aracılığıyla program ve veri alıp vermenizi sağlayan modemle bağlantı kurabilirsiniz. Bilgisayara anında program yüklemek için kullanılan ROM kartuşları ile bağlantı sağlayan başka araparçalar da vardır. Oyun oynamanızı kolaylaştıran oyun

kolu(joystick) da gene bunlar aracılığıyla bağlanır.

### Spectrum'a uyan yazıcılar

Bazı yazıcılar, doğrudan Spectrum'un yan bağlantısına bağlanabilirler. Örneğin elinizde Sinclair ZX yazıcınız varsa, bunu bir araparça kullanmadan bilgisayarınıza bağlayabilirsiniz. Bu tür bir yazıcı aynı zamanda ZX Interface 1'in de arkasına takılabilir. Ancak RS232 çıkışlı bir yazıcı için, ZX Interface 1'deki D girişini kullanmanız gerekir.

### ZX Interface 1

ZX Interface 1 birimi Spectrum'un altına ve arkasına bağlanır. Bu, Spectrum'unuzu sekiz kadar Microdrive'a, 63 kadar diğer Spectrum bilgisayarına ve RS232 standart araparça birimi ile pek çok sayıda diğer standart donanıma bağlar.

Microdrive'lar ve Microdrive kartuşları, program ve verilerin depolanmasında kasetli teypin ve kasetlerin yerini alır. Microdrive kartuşu takarak, programlarınızı birkaç

sanideye depolayabilir, doğrultabilir ve yükleyebilirsiniz. Her kartuş 85K kadar veri depolayabilir ve en fazla sekiz Microdrive kullanarak, Spectrum'unuz 680K'lık bir depolama kapasitesine ulaşır!

Araparça birimi ile verilen şebeke bağlantısını kullanarak bilgisayarınızı bir başka ZX Spectrum ya da ZX Spectrum + 'a bağlayabilirsiniz. Bu şebeke, en çok 63 başka Spectrum'u kapsayacak kadar genişletilebilir. Bunların arasında saniyede 10,000 hızıyla bilgi alışverişi yapılabilir.

ZX Interface 1 birimi ayrıca 9 adet D girişli bir RS232 araparçası da içerir. Ve bu girişlerle modemler ve diğer bilgisayarlar Spectrum'unuza bağlanabilir. Ayrıca bir de standart araparça kablosu da vardır.



Standart yazıcılar ZX Interface 1 aracılığıyla bağlanır.

### ROM kartuşları ve oyun kolları

ZX Interface 2 gibi araparçalar, ROM kartuşlarının ve oyun kollarının bağlanmasını sağlar. ROM kartuşları, kasetten yüklenmesi hayli uzun zaman alacak programların derhal yüklenmesini mümkün kılar.



**Microdrive'in yüklenmesi**  
Microdrive kartuşları, sürücünün önündeki yarığa sokulur.



**ROM kartuşunun takılması**  
Kartuş, araparça soketine sokulur. Bilgisayar ceryana bağlanınca, program bilgisayarın içindeki ROM'u atlayarak otomatik olarak yüklenir.

### Donanımların bağlanması

İşlem sırasında, ZX Interface 1 bilgisayarın altına ve arkasına gelecek şekilde yan bağlantıya girer. Bu resimde, sistemin bilgisayara bağlanmadan önceki durumu görülmektedir.



**Microdrive birimleri**  
Bir Spectrum'a bu depolama birimlerinden sekiz tane bağlanabilir.

**Kurdelâ kablo**  
Bu, Microdrive'i bilgisayara ZX Interface 1 aracılığıyla bağlar.

### Dipnot

ZX Spectrum + 'un klavyesine eğim verecek iki ayağı vardır. ZX Interface 1 takıldığında zaman ayakların kullanılması gerekmez.



ROM kartuşu/oyun kolu araparçası

Yan bağlantı  
Yan donanımlar bilgisayara buradan bağlanır.

**Uyarı**  
Yan donanımlar, daima ceryanı bağlanmadan yapılmamalıdır.

## ZX SPECTRUM+'UN BELLEK HARİTASI

42 ve 43. sayılardaki Spectrum'un içeriğini gösteren resme baktığınızda, bir ROM ve 16 daha küçük RAM yongası olduğunu göreceksiniz. Bu yongalar Spectrum'un belleğini oluşturur. Bellekte, her biri bir byte (0 ile 255 arasında bir sayı) içeren 65536 bilgi birimi bulunmaktadır. Her birim, adres denilen bir numara ile tanımlanır.

ROM, Sadece Okuyucu Bellek demektir. Belleğin bu bölümü Merkezi İşlem Birimi için işletme talimatları içerir. Bu 16 K'lık bir ROM'dur, yani  $16 \times 1024$  (16384) byte ya da adres taşımaktadır. Bu bellek, byte'ları sadece okur, yani bunlar değiştirilemez. (Değiştirilirse bilgisayar çalışmaz olur.) Herhangi bir adresteki byte'i PEEK'i kullanarak elde edebilirsiniz.

RAM, Rastsal Ulaşım Belleği anlamına gelir ve bilgisayara verilen programları ve bilgileri içerir. Spectrum'un 48 K'lık bir RAM'i vardır, yani  $48 \times 1024$  (49152) byte ya da adres taşıyabilir. Rastsal ulaşım, buradaki herhangi bir adresteki bir byte'in değiştirilebileceği anlamına gelir ve bu sadece POKE kullanılarak yapılabilir.

Bellekteki adresler 0'dan 65535'e kadar uzanır. Bunların ilk çeyreği ROM, geri kalanı RAM'dir.

### Sistem değişkenleri

Yandaki sütun, Spectrum'un belleğinin nasıl düzenlendiğini göstermektedir. Bunda, bilgisayar denetleyen belli kesimlerin nerede yer aldığını görebilirsiniz. Bunlardan bazılarının yerleri değiştirilebilir, bunun sınırlarını ise sistem değişkenleri belirler.

Spectrum'un sistem değişkenleri, BASIC'te kullanılan değişkenlere benzemez. Aslında bunlar, bellekteki belli adreslerde ya da yerlerde bulunan bazı yararlı değerler için verilmiş isimlerdir. Bu ismin amacı, burada depolanmış belli bir değerin anlamını hatırlatmaya yardımcı olmaktır. Örneğin RAMTOP sistem değişkeni, ROM'un en başındaki adrestir. Belleğin bu bölümünde bir BASIC programı ile değişkenlerinin değerleri bulunmaktadır. RAMTOP'un adresi 23730'dur.

### Bellek haritası

Kullanılan tanımlanmış grafikler	
GOSUB yığını	RAMTOP
Yedek	
Hesaplama yığını	STKEND
Geçici çalışma alanı	STKBOT
INPUT verileri	
Düzeltilen emir ya da satır	WORKSP
Değişkenler	E-LINE
BASIC programı	VARS
Bilgi kanalı	PROG
Microdrive haritaları	CHANS
Sistem değişkenleri	23734
Yazıcı tamponu	23552
Göndermeler	23296
Görüntü dosyası	22528
	16384
16K ROM	

48K RAM

## SINCLAIR BASIC'I ÖĞRENELİM

Bu bölüm, Sinclair BASIC'i tam olarak tanımlamaktadır. Burada, her komutun kullanış biçiminin bir özetini bulacak, Sinclair BASIC'in nasıl çalıştığını daha ayrıntılı olarak öğreneceksiniz. Burada verilen bilgi çok temel olanlardan, en ileri BASIC programlama için gerekli olanlara kadar değişmektedir. Bu, baştan sona okunacak bir bölüm değildir. Aslında bu, sizin Spectrum'un yeteneklerini sonuna kadar kullanmanızı mümkün kılacak bir programcı sözlüğüdür.





# SINCLAIR BASIC'IN KOMUTLARI İÇİN PROGRAMCININ BAŞVURU KILAVUZU

## Komut türleri

Komutlar, aşağıdaki dört gruptan birine girerler:

### Emir

Bir hareketin yapılmasını sağlayan ve doğrudan emir biçiminde komutlardır. Enter'e basarak basmaz işleme konur. Örnek - RUN, LOAD

### Cümle

Bir hareketin yapılmasını sağlayan ve bir program satırında kullanılabilecek komutlardır. Ancak program işleme girince uygulanır. Örnek - DRAW, INPUT

### İşlev

Bir tür değer üreten komutlardır. Bir emrin veya cümlenin bir parçasını oluşturur. Örnek: RND, INT.

### Mantık işlemi

Bir cümle ya da emirde mantık ifade üzere kullanılan komuttur. Belli koşulların geçerliğini veya doğruluğunu belirler ya da değiştirir. Spectrum'da üç mantık işlemi vardır - AND, OR ve NOT.

## Komut biçimi

Komut biçimi her komutun iç dizisini ifade eder - yani, komutla değerler ve değişkenler gibi diğer öğelerin doğru biçimini gösterir.

Bu kılavuz, ZX Spectrum + 'da bulunan bütün BASIC komutlarının tüm tanımını içermektedir. Her birinin altında

- Komutun yeri
- Komutun türü
- Komutun amacı
- Komutun kullanımı
- Programlama biçimi

verilmektedir. Yer, amaç ve kullanım konusunda verilen ayrıntılar, kolayca anlaşılabilir. Tür ve biçim daha karmaşıktır. Bu bakımdan, bu kılavuzdan tam yararlanmak istiyorsanız, önce bu sayfadaki bilgileri dikkatle okumanız gerekir.

## Sayılar ve değişkenler

### Sayılar

9 ya da 10 haneli bir sayıyı tastamam saklayabilir. İşleyebildiği sayılar ise  $10^{28}$  ile  $4 \times 10^{-39}$  arasında yer almaktadır.

### Kabul edebildiği değişkenler

Sayı bir harfle başlamak üzere istenen uzunlukta. Boşlukları dikkate almaz ve bütün harfleri küçük harfe dönüştürür. Büyük küçük harf farkı gözlenmez.

Dizi '\$' in ardından gelen herhangi bir tek harf. Büyük küçük harf aynı yapılmaz.

Düzen Düzen değişken ve altöge (subscript) leni için, DIM maddesine bakın.

Biçimi gösterirken, aşağıdaki kısaltmalar kullanılmıştır.

Kısaltmalar	Açıklama	Örnek
num-const	Sayısal değişmez (rakam)	24.5
num-var	Sayısal değişken (rakam da içerebilen değişken)	sum
num-expr	Sayısal ifade (bir rakam veren komutlar, sayısal değişmezler ve değişkenlerden oluşan geçerli herhangi bir bileşim)	sum * 24.5 RND.7
int-num-const	En yakın tam sayıya yuvarlanmış bir sayısal değişmez, değişken ya da ifade	
int-num-var		
int-num-expr		
string-const	Bir dizi değişmez ya da dizi (büyük karetiler içinde karakterlerin herhangi birleşimi)	"ZX Spectrum +"
string-var	Dizi değişken (dizi içeren değişken)	a\$
string-exp	Dizi ifade (dizi veren komutlar, değişmez ve değişkenlerden oluşan geçerli herhangi bir bileşim)	a\$ + "ZX Spectrum +" a\$(6 TO 8) Y * X
letter	Herhangi bir büyük veya küçük harf	B\$A\$
letter\$	Herhangi bir büyük veya küçük harfi izleyen \$	B\$A\$
cond	Bir koşul veya koşul içinde bir altkoşul	X = 10 AND 1 < 10
statement	Başka bir cümle ile kullanıldığında geçerli olan herhangi bir BASIC cümlesi	IF T > 10 THEN STOP PRINTINK 2: X

[ ] Tekrarlanabilecek seçmeli bir öğe

NOT Metinde sayısal değer ve dizi değeri sayısal veya dizi öğeleri için kullanılmıştır.

## Sinclair BASIC'in işaretleri

İşaret	Yeri	İşlevi/Kullanımı	İşaret	Yeri	İşlevi/Kullanımı
\$	4	Dizi değişkeni	=	L	Eşittir
%	7	Yeni satır başlatır	Z	8	Bir program satırında cümleleri ayırır
(	8	Parantez açar	/	V	Bölme
)	9	Parantez kapatır	*	8	Çarpma
<=	Q	Daha az veya eşit	-	Kendi tuşu	Öndelik işaret
<>	W	Eşit değil	-	Kendi tuşu	Yan satırda gösterir Program cümlesindeki cümleleri ayırır
>=	E	Daha büyük veya eşit	*	Kendi tuşu	Dizi başlatıp bitirir 0 veya 16. satırda gösterir Komutlardan sonra değeriyle ayırır
<	R	Daha az	-	Kendi tuşu	
>	T	Daha büyük			
↑	H	Üssüne yükseltir			
→	J	Çarpma/düzensiz			
+	K	Toplama/düzensiz/Dizi kumedi			

## ZX Spectrum + character set

	0	1	2	3	4	5	6	7	8	9
0					TRUE VIDEO	INV VIDEO	PRINT sayı	EDIT	gösterge ışık	gösterge ışık
10	gösterge ışık	gösterge ışık	DELETE	ENTER	sayı	GRAPHICS MODE	INK kontrol	PAPER kontrol	FLASH kontrol	BRIGHT kontrol
20	INVERSE kontrol	OVER kontrol	AT kontrol	TAB kontrol						
30			boşluk	!	"	#	\$	%	&	'
40	(	)	*	+	-	.	/	0	1	
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[	\	]	↑	-	£	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{		}	-	©	□	■
130	■	■	■	■	■	■	■	■	■	■
140	■	■	■	■	GRAPHICS A	GRAPHICS B	GRAPHICS C	GRAPHICS D	GRAPHICS E	GRAPHICS F
150	GRAPHICS G	GRAPHICS H	GRAPHICS I	GRAPHICS J	GRAPHICS K	GRAPHICS L	GRAPHICS M	GRAPHICS N	GRAPHICS O	GRAPHICS P
160	GRAPHICS Q	GRAPHICS R	GRAPHICS S	GRAPHICS T	GRAPHICS U	BMD	INKETS	PI	PI	POINT
170	SCREENS	ATTR	AT	TAB	VALS	CODE	VAL	LEN	SIN	COS
180	TAN	ASN	ACS	ATN	LN	EXP	INT	SGR	SGN	ABS
190	PRIN	IV	USR	STRS	CHRS	NOT	SIN	OR	AND	<=
200	>=	<>	LINE	THEN	TO	STEP	DRFFN	CAT	FORMAT	MOVE
210	ERASE	OPEN +	CLOSE +	MERGE	VERIFY	BEEP	CIRCLE	INK	PAPER	FLASH
220	BRIGHT	INVERSE	OVER	OUT	LPRINT	LIST	STOP	READ	DATA	RESTORE
230	NEW	BORDER	CON-TINUE	DIM	REM	FOR	GO TO	GO SUB	INPUT	LOAD
240	LIST	LET	PAUSE	NEXT	POKE	PRINT	PLOT	RUN	SAVE	RANDOM-IZE
250	IF	CLS	DRAW	CLEAR	RETURN	COPY				

**ABS** Absolute value

Klavyede yeni  
EXTEND MODE  
G

İşlev

ABS bir sayısal değerin mutlak değerini, yani artı ya da eksi işaretini olmaksızın değerini verir.

**ABS'nin kullanımı**

ABS'yi bir sayısal değer izler. Ifade parantez içine alınmalıdır. Örneğin:

50 LET x=ABS (y-z)

ABS sayısal değerin mutlak değerini verir.

Örnek

Emir

PRINT ABS -34.2

34.2 görüntü verir.

Bigimi

ABS num-const

ABS num-var

ABS (num-expr)

**ACS** ArcCose

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT W

İşlev

ACS, kosinüsünden bir açının

değerini hesaplar.

**ACS'nin kullanımı**

ACS'yi bir sayısal değer izler. Ifade parantez içine alınmalıdır. Örneğin:

60 LET x=ACS (y-z)

ACS'yi izleyen değer (yukarıda y-z) istenen açının kosinüsüdür ve -1 ile 1 arasında olabilir. ACS açının değerini radian olarak verir. Radiana dereceye çevirmek için ACS'nin verdiği değeri 180/PI ile çarpın.

Örnek

Emir

PRINT 180/PI \* ACS 0.5

60 verir, bu kosinüsü 0.5 olan açının değeridir.

Bigimi

ACS num-const

ACS num-var

ACS (num-expr)

**AND**

Klavyede yeni

SYMBOL SHIFT Y

Mantık işlemi/İşlev

AND, bir koşullar bileşiminin gerçeğini inşamada mantık işlemi olarak görev yapar. Ancak bütün koşullar gerçeğe, bileşimin tümü gerçeğe. AND ayrıca iki sayısal ya da dizi değeri üzerinde ikili işlemler yapmak için bir işlev olarak da kullanılır.

**AND'in kullanımı**

Mantık işlemi olarak AND, tümünün gerçeğinin inşamada bir cumlede iki koşulu bağlar. Örneğin

90 IF x=y & AND time < 10 THEN

PRINT "Correct"

Ancak her iki koşul da (x=y & az ve time < 10) gerçeğe, ekranda Correct yazısı görülecektir. Eğer biri ya da her ikisi de yanlışsa, o zaman bileşimin tümü yanlış ve program bir sonraki satıra geçer.

**İşlev olarak AND**

İşlev olarak AND iki sayısal değeri çarpabilir. Örneğin

50 LET x=y AND z

AND ancak ikinci değeri 0'a eşit değilse ilk değeri 1'i verir, ikinci değeri 0 ise 0 zaman 0 verir.

AND, kendinden önce gelmesi koşullarla bir dizi değeri de işleyebilir. AND mutlak bir sayısal değerin işlemesi gerekir. Örneğin

50 LET a5=b5 AND z

AND, ikinci değeri 0'ı ıfırlı değilse iki (b5) verir, ikinci değeri 0 ise boş dizi (" ") verir.

Dikkat edin, ZX Spectrum + doğru bir kopya 1 değeri, yanlış olana da 0 verir. Sıfır olmayan değeri gerçek, 0'ı yanlış olarak kabul eder. Sayısal değerlerin bileşimini standart gerçek tablosu uyarınca değeri vermez.

**Örnekler**

60 LET correct=(x=y & z) AND time < 10

70 LET score=score + 10 \* (1 AND correct)

80 LET a5=(1 "Out Of Time Or Not" AND NOT correct) + "Correct"

Eğer 60. satırdaki iki koşul gerçekleşir, sayısal değeri correct'e 1 değeri verir. O zaman score 10 artar ve a5 "Correct" olur. Eğer koşullardan biri yanlışsa, o zaman correct'in değeri 0 olur, score değeri 10 artmaz ve a5 "Out Of Time Or Not Correct" (zamanınız ya da doğru değil) olur.

**Bigimi**

cond AND cond  
num-expr AND num-expr  
string-expr AND num-expr

**ASN** ArcSine

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT Q

İşlev

ASN, sinüsünden bir açının değerini hesaplar.

**ASN'nin kullanımı**

ASN'yi bir sayısal değer izler. Ifade parantez içine alınmalıdır. Örneğin:

60 LET x=ASN (y-z)

ASN'yi izleyen değer (yukarıda y-z) istenen açının sinüsüdür ve -1 ile 1 arasında olabilir. Bu durumda ASN açının

değerini radian olarak verir. Radiana dereceye çevirmek için ASN'nin verdiği değeri 180/PI ile çarpın.

Örnek

Emir

PRINT 180/PI \* ASN 0.5

ekranda 30 görünür. Bu sinüsü 0.5 olan açının değeridir.

**Bigimi**

ASN num-const

ASN num-var

ASN (num-expr)

**AT**

Klavyede yeni

SYMBOL SHIFT I

Box INPUT/LPRINT/PRINT

**ATN** ArcTangent

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT E

İşlev

ATN bir açının değerini tanjantından hesaplar.

**ATN'nin kullanımı**

ATN'yi bir sayısal değer izler. Ifade parantez içine alınmalıdır. Örneğin:

60 LET x=ATN(y-z)

ATN'yi izleyen değer (yukarıda y-z) istenen açının tanjantıdır. ATN değeri radian olarak verir. Radiana dereceye çevirmek için ATN'nin verdiği değeri 180/PI ile çarpın.

Örnek

Emir

PRINT 180/PI \* ATN 1

ekranda 45 görünür. Bu, tanjantı 1 olan açının değeridir.

**Bigimi**

ATN num-const

ATN num-var

ATN (num-expr)

**ATTR** ATTRibute

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT

İşlev

ATTR ekrandaki belirli bir karakter pozisyonunun gördürmelerini verir. Bu gördürmeler, bu pozisyondaki karakterin murekkep ve kağıt rengi, parlaklığı ve yazıya dönme durumudur.

**ATTR'nin kullanımı**

ATTR'yi bir birinden vüğüle aynıymı ve tümü parantez içine iki sayısal değeri izler. Örneğin

150 IF ATTR(y-h)=115 THEN GOSUB 2000

ATTR'yi izleyen ilk değeri (yukarıda y) 0 ile 23 arasında olabilir ve ekrandaki bir pozisyonun satır numarasıdır. İkinci değeri (yukarıda

h) 0 ile 31 arasında olabilir ve ekrandaki pozisyonun sütun numarasıdır. Bunun üzerine ATTR 0 ile 255 arasında bir rakam verir. Bu rakam, belli bir pozisyondaki gördürmelerin toplamıdır ve şöyle hesaplanır:

Murekkep rengi: Renk kodu (0-7)  
Kağıt rengi: Renk kodunun 8 katı  
Parlak: 64  
Yazıya dönme: 128

**Örnek**

Eğer 11.16 pozisyonundaki karakterin murekkep rengi 3 (mor), kağıt rengi 6 (sarı) ve karakterin parlak ama yazıya dönme varsa, o zaman

PRINT ATTR (11,16)

emir, 115 verir: (3 + 8\*6 + 0)

**İkili biçimde ATTR**

ATTR bir byte a verildiğinde, 7. bit (en önemli) yazıya dönme için 1, normal için 0, 6. bit parlak için 1, normal için 0, 5 ile 3 arasındaki ilk rakam yazıya dönme kağıt rengi, 2 ile 0 arasındaki murekkep rengidir.

**Bigimi**

ATTR(num-expr,num-expr)

**BEEP**

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT Z

Cumle/Emir

BEEP verilen uzunlukta ve tonda bir sesin oparörden çıkmasını sağlar.

**BEEP'in kullanımı**

BEEP bir program cümlesi ya da doğrudan bir emir olarak kullanılabilir. BEEP'i vüğüle aynıymı iki sayısal değeri izler. Örneğin

80 BEEP x,y

İkili değeri (x) 0 ile 10 arasında olabilir ve notanın saniye olarak süresi belirler. İkinci değeri (y) -60 ile 69 arasında olabilir ve yazıya dönme halinde notanın bittirini belirler. Ortadaki C'nin (do) alındıysa eksi, üstündeyse artı değeri verir.

**Örnek**

Emir

BEEP 0.5,1

Ortak C'nin (do) bir üstündeki C diyez sesinin yarı saniye süreyle çıkmasını sağlar.

**Bigimi**

BEEP num-expr, num-expr

**BIN** Binary number

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT B

İşlev

BIN bir iki sayısal değeri ondan sayıya çevirir.

**BIN'nin kullanımı**

BIN'ı, 1'lerden ve 0'lardan oluşan en fazla onaltı haneli bir iki sayı izler. Örneğin

**50 POKE USR "a", BIN 1010 1010**

BIN bu iki sayısal değeri ondan sayıya çevirir. Bu, genellikle yukarıda da olduğu gibi POKE ve USR ile birlikte, kullanımı tanımladığı grafiklerin yapımında kullanılır. Burada 1 murekkep, 0 ise kağıt renginde bir pixel'e işaret eder.

**Örnek**

Emir

PRINT BIN 111111110

Ekranda, bu iki sayısal değeri ondan sayıya çevirir. Bu, genellikle yukarıda da olduğu gibi POKE ve USR ile birlikte, kullanımı tanımladığı grafiklerin yapımında kullanılır. Burada 1 murekkep, 0 ise kağıt renginde bir pixel'e işaret eder.

**Bigimi**

BIN (1) (0)

**BORDER**

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT

Cumle/Emir

BORDER, ekranın çevresindeki alanın rengini belirler.

**BORDER'in kullanımı**

BORDER, doğrudan bir emir ya da bir program cümlesi olarak kullanılabilir. Bunu bir sayısal değeri izler. Örneğin

30 BORDER RND \* 7

BORDER'i izleyen değeri en yakın tam sayıya yuvarlanacak ve bu da aşağıda gösterildiği gibi ekran çerçevesinin rengini verecektir:

0 Siyah

1 Mavi

2 Kırmızı

3 Mor

4 Yeşil

5 Açık mavi

6 Sarı

7 Beyaz

Ayrıca BORDER ekranın alt bölümünün kağıt rengini de belirler. INK ve PAPER dan farklı olarak BORDER cümlesi, bir PRINT cümlesinin içine yerleştirilemez.

**Bigimi**

BORDER int-num-expr

**BRIGHT**

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT B

Cumle/Emir

BRIGHT, karakterlerin ekranda normalden daha parlak görünmesini sağlar.

**BRIGHT'in kullanımı**

BRIGHT doğrudan bir emir olarak kullanılabilir, ama genel kullanımı bir program cümlesi biçimindedir. Bunu bir değeri izler. Örneğin

80 BRIGHT 1

Gerekirse BRIGHT'ı izleyen değeri en yakın tam sayıya yuvarlanabilir. Bu sayı 0, 1 veya 8 olabilir. Değeri 1 olmasa, bunu izleyen PRINT veya

INPUT cümleleri de ekrana yazdıracak bütün karakterlerin murekkep ve kağıt renginin daha parlak görünmesini sağlar. Değeri 8 olmasa, yeni karakterler yazıldığında, ekranda parlak karakterlerin bulunduğu pozisyonların, parlak, normalden

0 ise hem BRIGHT 1'i hem de BRIGHT 8'i iptal eder, bundan sonra gelen bütün karakterler normal görünür.

BRIGHT ayrıca PRINT, INPUT, PLOT, DRAW ve CIRCLE gibi ekranın görüntü yaratma komutlarına oluşturulan cümlelere de yerleştirilebilir. BRIGHT komutları sonra, ama veri ya da görüntü öğelerinden önce gelir. BRIGHT'i kendi değeri ve bir noktalı vüğüle izler. Örneğin

50 PRINT BRIGHT 1: "WARNING"

Bu durumda BRIGHT'in etkisi biter ve ancak kendisini izleyen karakterler, ya da bu cümle ile komutun noktası, çizilen çizgiyi kaplar. Unutmayın BRIGHT 8 ile murekkep renginde bir pixel, bir karakter pozisyonuna yerleştirilecek olursa, 8\*8 pixel boyutundaki bu karakter pozisyonunun tümünün kağıt rengi de parlak olur.

**Bigimi**

BRIGHT int-num-expr [:]

**CAT** CAtalogue

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT

Cumle/Emir

Microdrive dosya işleme emri. Microdrive ve Interface 1'in el kitabına bakın.

**CHR\$** Character string

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT

Cumle/Emir

Spectrum'un karakter tablosu, klavyede bulunan karakter ve komutların yanı sıra kullanımlarını tanımladığı grafik karakterlerinden oluşur. CHR\$ ve bir kod numarası kullanarak, bunlardan herhangi birini bir dizi olarak elde edilebilir. Karakter tablosu ayrıca, karakterlerin ekrana yazdırılmasını sağlayan bazı kontrol kodları da içerir. CHR\$ den önce PRINT kullanılarak bu kodlar harekete geçirilip karakterler ekrana yazdırılabilir. Karakter tablosunun tamamı ve kod numaraları 51 sayfa da bulabilirsiniz.

**CHR\$'nin kullanımı**

CHR\$'yi bir sayısal değeri izler. Örneğin

80 PRINT CHR\$(x)

Ifade parantez içine alınmalıdır. CHR\$'yi izleyen değeri (yukarıda x) en yakın tam sayıya yuvarlanır. 32 ile 255 arasında ise, CHR\$ bir dizi











Örnek  
Emir



Görüntü şöyle olur:

45 -8

Bijimi  
INT num-const  
INT num-var  
INT (num-exp)

**INVERSE**

Klavyede yeni  
EXTEND MODE  
SYMBOL SHIFT M

Cümle/Emir

INVERSE, karakter pozisyonlarında kağıt rengini murekkep, murekkep rengini kağıt rengine döndürmesini sağlar.

INVERSE'in kullanımı  
INVERSE normal olarak bir program cümlesi olarak kullanılır. Bunu bir sayısal değer izler. Örneğin,

70 INVERSE 1

INVERSE 1 izleyen değer en yakın tam sayıya yuvarlanır ve bu ya 0 ya da 1 olur. INVERSE 1, bundan böyle PRINT ve INVERSE ile gerçekleştirilecek bütün görüntülerin renklerin yer değişmesini sağlar. INVERSE 0 ise murekkep ve kağıt renklerini normale döndürür.

INVERSE, aynı INK gibi, görüntü sağlayan cümlelerin içine yerleştirilebilir. Ancak CIRCLE, PLOT veya DRAW ile kullanılsa INVERSE 1 bir çizgi veya noktanın kağıt rengini almasına, dolayısıyla da görülmemesine neden olur.

Bijimi

INVERSE int-num-exp

**LEN**

Length of string

Klavyede yeni

EXTEND MODE

K

izley

LEN, bir dizinin uzunluğunu verir.

LEN'in kullanımı

LEN'i bir dizinin değeri izler. Örneğin,

50 LET X=LEN A\$

İfade paranteze alınmalıdır. LEN dizindeki karakterlerin sayısını verir.

Örnek

Aşağıdaki satır

120 INPUT A\$, IF LEN A\$ &gt; 9

THEN GOTO 120

sadece en fazla dokuz karakterli dizilerin geçmesine izin verir.

Bijimi

LEN string-const

LEN string-var

LEN (string-exp)

**LET**

Klavyede yeni

L

Cümle/Emir

LET, bir değişim değeri belirlemede kullanılır. Sinclair BASIC'te belirleme cümlesi LET olmadan olmaz.

LET'in kullanımı

LET normal olarak bir program cümlesi oluşturur ama ayrıca doğrudan emir olarak da kullanılabilir. Bunu bir sayısal ya da dizinin değeri, bir eylem, sonra da bir değeri izler. Bu değeri LET'i izleyen değeriye bağı olarak bir sayısal ya da dizinin değeri olabilir. Örneğin,

60 LET X=X+1

80 LET A\$= "Correct"

Böylece değeriye bir değeri vermiş olur.

Unutmayın, basit değişimler, LET, READ veya INPUT tarafından bir değeri verilmemiş belirlenir.

Ancak dizinin değeri, daha başından ya 0 ya da boş dizi (bak DİZİ) olarak belirlenir.

Bijimi

LET num-var=num-exp

LET string-var=string-exp

**LINE**

Klavyede yeni

K

Emir/Cümle

LIST, o anda bellekte bulunan programın sıralamasını verir.

LIST'in kullanımı

LIST normal olarak doğrudan emir olarak kullanılır, ama bir program cümlesi yapmak için de kullanılmak mümkündür. Bir programın tümünün sıralaması için kendi başına kullanılır.

LIST

diye verilen bir doğrudan emir ardından, önce sıralamanın ilk sayfası ekranda belirlir. N. ara tuşu, STOP veya BREAK dışında herhangi bir tuşa basınca da diğer sayfalar sırayla görünür.

LIST'i ayrıca sayısal değeri biçiminde bir satır numarası izleyebilir. Örneğin,

LIST 100

Bunun üzerine eğer gerekiyorsa, LIST'i izleyen değeri en yakın tam sayıya yuvarlanır ve sıralama bu sayıdan başlar. Bu numarası taşıyan bir satır yoksa sıralama bir sonraki satırdan başlar.

Bijimi

LIST (int-num-exp)

LIST

Line printer LIST

Klavyede yeni

EXTEND MODE

V

Emir/Cümle

LIST, Sinclair tipi yazıcıların, o anda bellekte bulunan programın sıralamasını yazıya aktarmasını sağlar.

LIST'in kullanımı

LIST aynı LIST gibi kullanılır (aynısı için LIST'e bak). Dikkat edin, sıralama yazdırırken ekrandaki görüntü değişmez.

Bijimi

LIST (int-num-exp)

**LN**

Logarithm (Natural)

Klavyede yeni

EXTEND MODE

Z

izley

LN bir değerin doğal logaritmasını (e tabanı üzerine logaritmasını) verir. EXP'nin tersi gibi işler.

LN'nin kullanımı

LN'yi bir sayısal değeri izler. Örneğin,

60 LET X=LN Y

İfade paranteze alınmalıdır. LN'yi izleyen değeri 0'dan büyük olması gerekir. Bunun üzerine LN bu sayının doğal logaritmasını verir.

Bijimi

LN num-const

LN num-var

LN (num-exp)

**LOAD**

Klavyede yeni

J

Emir/Cümle

LOAD, bir programın tamamını teyipten belleğe yükler.

LOAD'in kullanımı

LOAD normal olarak doğrudan emir olarak kullanılır, ama yeni bir program yüklenmesi için bir programın bir cümlesi de oluşturulabilir. LOAD'i bir program adı izler. Bu en fazla on karakterden oluşan bir dizi değeridir. Örneğin,

LOAD "dosyaadı"

Uygulamada, o anda bellekte olan program ile bunun değişimlerinin bütün değerleri iptal edilir. Spectrum daha sonra adı verilen programı arar, bulunca da yükler. Dikkat edin, bilgisayar program adlarında büyük harf ile küçük harf ayrıtılır.

Su emirde olduğu gibi LOAD'ı bir boş dizi izlene.

LOAD = "

Spectrum bulduğu ilk tam programı yükler.

Dikkat edin, Microdrive bağlanmış zaman LOAD farklı kullanılır. Bunun için Microdrive ve Interface 1 el kitabına bakın.

Bijimi

LOAD string-exp

**LOAD CODE**

Klavyede yeni

J

EXTEND MODE

Emir/Cümle

LOAD CODE, belleğin belli bir kesimine teyipte depolanmış bilgileri yüklemek için kullanılır. Bilgi bir dizi byte halinde ve bunlar bellekteki belli adreslere gönderilir. LOAD CODE bir görüntüyü ya da örneğin kullanımların tanımladığı karakterler için bilgileri yüklemek için de kullanılır.

LOAD CODE'un kullanımı  
LOAD CODE doğrudan bir emir olarak kullanılabileceği gibi, programın bir cümlesi de oluşturulabilir. LOAD'ı bir dizi değeri olan dosya adı ve CODE izler. Örneğin,

LOAD "data" CODE

LOAD'ı izleyen dosya adı, yüklenmek bilgilerin adı ve aynı program adının sıralamasına sahiptir. (Bak LOAD). Bunun üzerine LOAD CODE bu adı bilgileri arar, bulunca ekrana byte'ları, ardından da adını yazar. Spectrum daha sonra bu byte'ları, bellekte aldıkları adreslere yükler. Buralarda bulunan bilgileri ise sunar.

CODE'u ayrıca bir veya iki sayısal değeri izleyebilir. Bunlar vurgulu ayrılar. Örneğin,

LOAD "picture" CODE 16384 6912

CODE'u izleyen değerler en yakın tam sayıya yuvarlanır ve ardından adı verilen bilgilerin yüklenmesi başlangıç adresini (yukarıda 16384) ve bu adrese başlayan yere gönderilecek byte'ların sayısını (6912) verir. Eğer nokam yanlışsa, teyit yüklemesi hatası raporu gelir.

CODE'u eğer sadece bir değeri izlerse, bu bütün byte'ları yerleştirileceği başlangıç adresini tanımlıyordur.

Yukarıdaki örnek ayrıca LOAD SCREENS komutlarıyla da uygulanabilir.

Byte'ların saklanması konusunda ayrıntılı bilgi için SAVE CODE'a bakın.

Bijimi  
LOAD string-exp CODE (int-num-exp) (int-num-exp)

LOAD string-exp CODE (int-num-exp) (int-num-exp)

**LOAD DATA**

Klavyede yeni

J

EXTEND MODE

D

Cümle/Emir

LOAD DATA teyipten dizileri yüklemek için kullanılır. Dizileri SAVE DATA kullanılarak kaydedilir.

LOAD DATA'nın kullanımı  
LOAD DATA, ya bir program cümlesi oluşturmak üzere ya da doğrudan emir olarak kullanılabilir. LOAD'in önce bir dizi değeri olan dosya adı, sonra da sırasıyla DATA, bir harf veya bir harfle birlikte \$ ve son olarak bir boş parantez izler. Örneğin,

270 LOAD "numbers" DATA n()

300 LOAD "names" DATA n()

LOAD'ı izleyen dosya adı, dizine teyipte verilen adres ve LOAD'la birlikte kullanılan program adlarına uygulanmış kodlamalara tabidir. DATA'yi izleyen harf ya da harfle \$ yüklem kullanılabileceği zaman dizine programda verilecek addır. Uygulamaya kondığında,

Spectrum teyipte dizinin adını arar, bulunca ekrana Number array (sayı dizini), Character array (karakter dizini) mesajlarıyla dizinin adı belirlenir ve dizin yüklenir. O sırada bellekte olup da aynı harfle adlandırılmış (yukarıda n veya n\$) başka bir dizin varsa, bu iptal edilir.

Bu harf adını ve teyipte depolanmış değerleri taşıyan yeni bir dizin yaratılır. Dikkat edin, karakter dizinlerinde, o sırada bellekte olan ve aynı harfle adın taşıyan herhangi bir dizin değeri de iptal edilir.

Bijimi  
LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

LOAD string-exp DATA letter(\$)

aynıları konusunda SAVE SCREENS'e bakın.

Bijimi  
LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS

LOAD string-exp SCREENS



## MOVE

Microdrive dosya işleme emri. Microdrive ve Interface 1 el kitabına bakın.

## NEW

Klavyede yeni A

Emir/Cümle

NEW, BASIC bellek alanını (RAMTOP'a kadar olan alan) temizlerken, belleğin bu komuda bulunan herhangi bir program yok eder.

## NEW'nin kullanımı

NEW normal olarak bir doğrudan emir olarak kullanılır ama bir program cümlesi de oluşturabilir. Kendi başına kullanılır. Uygulamaya konulan program ve değişkenler yok olur. Bellek RAMTOP'a kadar temizlenir, böylece RAMTOP'un altında depolanmış olan kullanımların tanımladığı grafik karakterler bundan etkilenmez.

Bilgisi

NEW

## NEXT

Klavyede yeni N

Cümle/Emir

NEXT, bir FOR NEXT döngüsü yaratılmak için her zaman FOR ile bağlantılı kullanılır.

## NEXT'in kullanımı

NEXT, bir FOR NEXT döngüsünü tamamlamak için bir program cümlesi oluşturmak üzere kullanılır. NEXT'i, döngüde kontrol değişkeni olan bir harf izler. Örneğin,

90 NEXT a

Sindair BASIC'te kontrol değişiminin mutlaka konulması gerekir.

FOR NEXT döngüsünün ayrıntıları için FOR'a bakın.

Bilgisi

NEXT letter

## NOT

Klavyede yeni SYMBOL SHIFT S

Mantık işlemi/İşlev

NOT, bir koşulun gerçekleşmesini ters yüz etmek, yanlış bir koşulu doğru, doğruyu yanlış hale getirmek için kullanılır.

## NOT'un kullanımı

NOT'u bir koşul veya bir sayısal değer izler. Örneğin,

90 IF NOT x=y+2 THEN PRINT "Wrong"

90 LET correct=x+y+2; IF NOT correct THEN PRINT "Wrong"

NOT'u bir koşul (yukarıda x=y+2) iflediği zaman, Spectrum bu koşulu doğru ise bina 1, yanlışsa 0 değeri verir. NOT bunun üzerine işlev gibi hareket eder, önce ortaya çıkan değeri tersine çevirir. Koşulun teminin sonmasını sağlar. Dikkat edin, koşul AND veya OR işleviyle, o zaman bunun paranteze alınması gerekir.

Eğer NOT'u bir sayısal değer izliyorsanız, bu değer sıfır değeri NOT 0. 0 ise 1 sonucunu verir. Bu nedenle yukarıdaki örnekte x<y+2 ise Spectrum "Wrong" (yanlış), diğer değeri 0 ise correct (doğru) sonucu verir.

Bilgisi

NOT cond

NOT num-expr

## OPEN

Microdrive dosya işleme emri. Microdrive ve Interface 1 el kitabına bakın.

## OR

Klavyede yeni SYMBOL SHIFT U

Mantık işlemi/İşlev

OR, bir koşullar bileşkesinin doğruluğunu sınamak için bir mantık işlemi olarak hareket eder. Eğer koşullardan bir veya birkaçı doğru ise o zaman bütün bileşke doğru demektir. OR ayrıca iki sayısal değer üzerinde iki işlemler yapmak için de fonksiyon olarak görev yapar.

## OR'un kullanımı

Bir mantık işlemi olarak OR, tümünün gerçekleşmesi için cumledeki iki koşulu bağlar. Örneğin,

70 IF INKEY\$="N" OR INKEY\$="n" THEN STOP

Eğer bu koşullardan biri veya her ikisi de doğru ise tüm bileşke doğru demektir. Yukarıdaki satırda, CAPS SHIFT veya CAPS LOCK tuşlarına basılısa da, basılımsa da N tuşuna basılır basılmaz koşullardan biri INKEY\$="N" ve INKEY\$="n" doğru hale gelecektir. Bu durumda bütün bileşke doğrudur ve program durur.

## İşlev olarak OR

ZX Spectrum'a, doğru koşula 1 sayısal değeri, yanlış koşula ise 0 sayısal değeri verir. Sıfır dışındaki her değeri doğru, 0'ı yanlış kabul eder. Bu yüzden OR'dan önce veya sonra bir sayısal değer gelir. Örneğin,

40 LET x=y OR z

Bu durumda, z sıfır değeri ya da doğru bir koşula x değerine 1 değeri, diğer z 0 ise veya yanlış bir koşul ise y'nin değeri verilir. Bu anlamda yararlıdır. Aşağıdaki örnekte, yokunun yanı

14'ten küçükse bilet ücreti yan

yarıya düşüyor

60 PRINT fare(0.5 OR age > 13)

Eğer yaş (age) 14'ten küçükse, age > 13 koşulu yanlış, ücret (fare) 0.5 ile çarpılır. Eğer age > 13 doğru ise ücret 1 ile çarpılır.

Dikkat edin, Spectrum sayısal değerlerin bileşkesini, standart gerçek tablosu gibi değerlendirmez.

Bilgisi

cond OR cond

num-expr OR num-expr

## OUT

Klavyede yeni EXTEND MODE SYMBOL SHIFT T

Cümle/Emir

OUT, bir çıktı donanımını harekete geçirmek için belli bir gördü/çıkı

yükleme adresine bir byte gönderir.

## OUT'un kullanımı

OUT bir program cümlesi ya da doğrudan emir olarak kullanılabilir. Bunu birbiriinden vurgula aynıymış iki sayısal değer izler. Örneğin,

40 OUT 254,3

Her iki değeri de en yakın tam sayıya yuvarlanır. İlk değeri (yukarıda 254) 0 ile 65535 arasında olabilir ve yüklem adresini gösterir. İkinci değeri (3) 0 ile 255 arasında olabilir ve bu adrese gönderilecek byte'tir.

254 yüklem adresindeki byte'ın 0 ile 2 arasındaki bit'leri çerçeve rengini belirler. Bu nedenle yukarıdaki örnek, çerçeve kenarını mora dönüştürür. Bu adresteki 3. bit MIC girişini, 4. bit optik girişini hareket ettirir. 254, 247 ve 239 numaralı yüklem adresleri diğer yan donanımları kullanılır.

Bilgisi

OUT int-num-expr,int-num-expr

## OVER

Klavyede yeni EXTEND MODE SYMBOL SHIFT N

Cümle/Emir

OVER, bir karakterin bir diğerinin üzerine yazdıkları için kullanılır. Ayrıca murekkep rengi yerine kağıt renginde noktalar kondurmak, doğru ya da eğri çizmek için de kullanılabilir.

## OVER'in kullanımı

OVER normal olarak bir program cümlesi olarak kullanılır. OVER'i bir sayısal değer izler. Örneğin,

80 OVER 1

OVER'i izleyen değeri en yakın tam sayıya yuvarlanır ve bu değeri 0 veya 1 olabilir. OVER 0 zaten makinenin normal olarak ayarlanmış olduğu durumdur ve yeni karakter bir

karakter pozisyonundaki eski karakteri silip yenisini alır. OVER 1 ise aynı karakter pozisyonunda iki karakterin üstüste birleşmesini sağlar.

OVER'in, aynı INK gibi, bir PRINT veya INPUT cümlesinin içine yerleştirilerek, sadece bu cümlede yer alan karakterleri etkilemesi sağlanabilir. Örneğin şu cümle, bir karakterin altını çizler:

60 PRINT AT 11,15: "YES"; OVER 1; AT 11,15

Ancak dikkat edin, burada murekkep renklerinin (çalıştığı) yere kağıt rengi verimlidir.

## Yüksek tanımlı çizimde OVER

OVER, PLOT, DRAW ve CIRCLE ile de kullanılabilir. OVER olmadan da çizgiler, eğriler çizebilir, ancak murekkep renkleri aynı değeri, kesiklikleri karakter pozisyonunun tümünün rengi değeri. Eğer OVER 1 kullanırsanız, çizgiler ve eğrilerin kesiklikleri ya da diğer karakterlerle çalıştıkları yerler kağıt rengini alır. OVER 1 kullanarak, aynı pozisyona noktalar koyup, çizgiler, eğriler çizime ekler silinir.

Bilgisi

OVER int-num-expr

## PAPER

Klavyede yeni EXTEND MODE SYMBOL SHIFT C

Cümle/Emir

PAPER, ekrandaki görüntü için kağıt ya da fon rengi seçmede kullanılır. Bu renk ya bütün görüntü alanını, ya da sadece belirli karakterlerin, nokta ya da çizgilerin bulunduğu tek bir karakter pozisyonunu kapsar.

## PAPER'in kullanımı

PAPER, bir program cümlesi oluşturmak üzere ya da doğrudan bir emir olarak kullanılır. PAPER'i bir sayısal değer izler. Örneğin,

80 PAPER x

PAPER'i izleyen değeri en yakın tam sayıya yuvarlanır ve 0 ile 9 arasında bulunur. PAPER'in renkleri aynı INK gibidir. Kağıt rengi genel olarak beyazdır, fakat murekkep renkleri gibi, bütün görüntü yaratacak cümlelerin içine yerleştirilerek belli bir yerle sınırlamak mümkündür. Ayrıntılar için INK'e bakın.

PAPER çizimi ardından bazı karakterlerin yazdırılması istenirse, genel de olsa doğru ya da yanlış, bu karakterlerin yazıldığı karakter pozisyonlarının rengi, seçilen renge döner. Bu ayrıca bir genel emir ardından değil de, sadece bir PAPER cümlesini izleyen nokta kondurma, çizgi ya da daire çizme işlemleri için de geçerlidir.

Bütün bir ekranın fonunu belli bir renkle kaplamak için PAPER cümlesi ardından CLS ile kullanılması gerekir. Bütün görüntü

alanını bir renk kaplar ve fon rengi olarak kalır.

Bilgisi

PAPER int-num-expr

## PAUSE

Klavyede yeni A

Cümle/Emir

PAUSE bir programı belirli ya da belirsiz bir süre için alıkoymayı sağlar.

## PAUSE'un kullanımı

PAUSE normal olarak bir program cümlesi olarak kullanılır. Bunu sayısal bir değeri izler. Örneğin,

130 PAUSE 100

PAUSE'u izleyen değeri, en yakın tam sayıya yuvarlanır ve 0 ile 65535 arasında olabilir. Televizyon ekrandaki görüntüde kaç karelik bir duraklama olacağını belirler. Bu nedenle kare frekansı 50 Hz olan İngiltere ve Avrupa'da, 50 değeri 1 saniyelik bir duraklama neden olur.

Ancak, duraklama devam ederken herhangi bir tuşa dokunup bu süreyi kısaltmak mümkündür. PAUSE 0 ise bir tuşa basılana kadar sürecektir süresi duraklamayı sağlar.

Bilgisi

PAUSE int-num-expr

## PEEK

Klavyede yeni EXTEND MODE O

İşlev

PEEK, belleğin belli bir adresinde depolanmış byte'ın değerini verir.

## PEEK'in kullanımı

PEEK'i bir sayısal değeri izler. Örneğin,

80 LET x=PEEK(256+y)

Dikkat edin ifade paranteze alınmalıdır. PEEK'i izleyen değeri, genellikle en yakın tam sayıya yuvarlanır ve bellekteki adresi vermesi için 0 ile 65535 arasında olur. PEEK bunun üzerine bu adresteki byte'ın değerini (0 ile 255) arasında bir rakam olarak verir.

## Örnek

Spectrum'un son çalıştırmasından bu yana meydana gelen televizyon görüntü karesinin numarası 23672 ile 23674 arasındaki adreslerde. Bu kareleri belli bir hızla oluşturup için, bu yerlerin PEEK edilmesi bir zaman ölçüm yöntemi verir. Aşağıdaki satır, Spectrum'un certyana başlatılmasından bu yana geçen zamanı saniye olarak verir. (Buna sayı üretmekle ve kasıtlı teyp, yazıcı gibi yan donanımları çalıştırmakla geçen zaman dahil değildir.)

10 PRINT (PEEK 23672+256\*PEEK 23673+65536\*PEEK 23674)/30

Not: Eğer kullandığınız elektrik frekansı İngiltere'deki gibi 50 Hz değil de 60 Hz ise, o zaman 50'yi 60'a çevirin.

Bilgisi

PEEK int-num-const

PEEK int-num-var

PEEK (int-num-expr)

## PI

Klavyede yeni EXTEND MODE M

İşlev

PI, hesaplamalarda kullanılan pi sayısının (pi) değerini verir. PI, bir dairenin çemberinin çapına oranıdır.

## PI'nin kullanımı

PI bir cümlede ve bir emir olarak kullanıldığında değerlere ya da değişkenlere gereksinim duymaz. Örneğin,

DRAW 244,0,-PI

Yukarıdaki emrin ekranda büyük bir yarı daire çizilmesi için PI 3.1415927 değerini verir.

Bilgisi

PI

## PLOT

Klavyede yeni Q

Cümle/Emir

PLOT, yüksek tanımlı çizimlerde ekrandaki belli bir pozisyona bir pixel ya da renkli bir nokta yerleştirmek için kullanılır.

## PLOT'un kullanımı

PLOT, bir program cümlesi veya bir emir olarak kullanılır. Normal olarak bunu, birbiriinden vurgula aynıymış iki sayısal değeri izler. Örneğin,

50 PLOT 128,87

Eğer gerekiyorsa, PLOT'u izleyen her iki değeri de en yakın tam sayıya yuvarlanır. İlk değeri 0 ile 255 arasında değeri ve dikey koordinatı verir. Bunun üzerine belirli bir pozisyonda (yukarıdaki örnekte ekranın ortasında) mevcut murekkep rengi ile bir nokta yerleştirir.

Renk cümleleri ya da emirleri PLOT üzerinde aşağıdaki etkileri yapar. OVER 1'den sonra aynı pozisyondaki mevcut noktaları rengi, kağıt rengine döner. INVERSE 1'den sonra mevcut kağıt renginde nokta yerleştirir. BRIGHT 1 veya FLASH 1'den sonra alçak tanımlı ekranda, pixel in yerleştirildiği karakter pozisyonunun tümü parlak veya yanar söner hale gelir. Bu dört komut ve INK, PRINT ile olduğu gibi bir PLOT cümlesine de yerleştirilebilir. Örneğin,



## 160 PLOT INK 2: x, y

İki ayrı ama, bu cümle ile yerleştirilen pixel ve çevresi ile renklendirir. Bir PLOT cümlesine PAPER yerleştirilirse, pixel çevresindeki karakter pozisyonunun tümünün kağıt rengi, verilen renge döner.

Dikkat edin, PLOT aynı zamanda bir sonraki DRAW cümlesinin başlangıç pozisyonunu da tanımlar.

**Bijimi**  
PLOT (statement)  
int-num-exp1; int-num-exp2

## POINT

**Klavyede yeni**  
EXTEND MODE  
SYMBOL, SHAFT B

İşlev

POINT, yüksek tanımlı bir ekranda belirli bir pozisyonun rengini mürekkep rengi mi, kağıt rengi bir olduğunu anlamak için kullanılır. POINT, rengin hangisi olduğunu araştırır.

## POINT'ın kullanımı

POINT, birbirinden virgüle ayrılmış ve hepsi paranteze alınıp iki sayısal değer alır. Örneğin,

240 IF POINT (x, y) = 1 THEN  
GOSUB 600

POINT'ı izleyen iki değer, genellikle tam sayıya yuvarlanır. 0 ile 255 arasında değişen iki değer ekrandaki pixel'in yatay koordinatını verir. 0 ile 175 arasında değişen ikinci değer ise dikey koordinatı verir. Bunun üzerine POINT, eğer belirlenen pozisyondaki pixel mürekkep rengindeyse 1, kağıt rengindeyse 0 verir.

**Bijimi**  
POINT (int-num-exp1; int-num-exp2)

## POKE

**Klavyede yeni**  
O

Cümle/Emir

POKE, bellekte belli bir adresdeki byte'in değerini değiştirmek için kullanılır. Bellekteki belli yerlere POKE ile değerler, normal olarak, BASIC komutlarının sağlandığı bir hanelerle yapılmak için verilir.

## POKE'in kullanımı

POKE bir program cümlesi ya da emir olarak kullanılır. Buna bir virgüle ayrılmış iki sayısal değer alır. Örneğin,

POKE 21609, 255

POKE'ı izleyen iki değer gerekirse en yakın tam sayıya yuvarlanır. İlk değer, 16384 ile 65535 arasıdır ve RAM'daki bir adrestir. İkincisinin değeri ise 0 ile 255 arasında değişir ve belirlenen adrese yazılacak byte dir.

Yukarıdaki örnekte, 255, 21609 sayılı adrese POKE ile verilecektir. Bu adres, tuşa basıldığında okunan ses kontrol etmektedir. 255 değeri, normal kık sesinin yerine uzun bir bip sesinin çıkmasını sağlar. Diğer değerler daha kısa bir ses çıkarılır.

**Bijimi**  
POKE int-num-exp1,  
int-num-exp2

## PRINT

**Klavyede yeni**  
P

Cümle/Emir

PRINT, verileri ekrana yansıtır. Veri bir tek veya bir dizi karakter olabilir. PRINT cümlesi, verinin pozisyonunu ve rengini belirleyen diğer komutları da içerir.

**PRINT'in kullanımı**  
PRINT kendi başına kullanılabilir gibi, bunu veriler de izleyebilir. Bu veriler, herhangi bir sayısal ya da dizi ifade veya bunların bir bileşimi de olabilir.

PRINT'ı veri ile birlikte kullanırken, iki ve daha fazla öge birbirinden virgüller, nokta-virgüller ve kesme işaretleriyle ayrılır.

PRINT ile verilen arasında diğer bazı komutlar da yerleştirilebilir. Ancak bir komuta olacağı her cümle bir nokta-virgüle bitmelidir. Böyle kullanılacak komutlar: CHR\$, TAB, AT, INK, PAPER, FLASH, BRIGHT, INVERSE ve OVER.

## PRINT'le diziler

PRINT kendi başına ya da bir boş dizi (" ") ile kullanıldığında, bir boş satır oluşturur ve göstergeç bir sonraki satırın başına geçer.

PRINT'ı bir değişmez dizi (bina) içeren içinde herhangi bir karakteri izlerse, ekrana bu tek karakteri içeren karakterler yansır. Örneğin şu emir

PRINT "3/542/76/21"

ekrana şu görüntüyü yansıtır:

3/542/76/21

PRINT'ı izleyen bir dizi değişken veya ifade bunları temsil ettiği dizi veya dizi ekranı yansıtır.

## PRINT'le sayılar

PRINT'ı herhangi bir sayısal ifade izlerse, ekrana bu ifadenin değeri yansır. Sayılar ondalık sayılar halinde 8 haneli olarak verilir, ondalık işaretinden sonra gelen sıfır yansımaz.

Çok büyük ya da çok küçük rakamlar, birbirinden 8 haneli ile ayrılmış iki sayı halinde, daha kısa bir bilimsel ifade ile yansır. Bu, ilk bölümün (kök) ikinci bölümdeki (uz) kuvveti yüksekliği anlamına gelir. Şu emir, örneğin,

PRINT 1/542/76/21

ekrana şu görüntüyü verir:

## 1.4680798E-6

## PRINT'in noktalarla işaretleriyle kullanımı

PRINT'ı izleyen veri ögesi birbiriinden nokta-virgüle ayrılmış, ekranda bunlar aralarında hiç boşluk olmadan yan yana görünürler.

PRINT 1, 2, 3

emir, şu görüntüyü verir:

123

PRINT'ı izleyen veri ögesi birbiriinden virgüle ayrılmış, ekranda her öge, ilk ögenin pozisyonuna göre, ya bir satırın başında ya da ortasında görünür.

PRINT 1, 2, 3

emir, şu görüntüyü verir:

1 2 3

PRINT'ı izleyen veri ögesi birbiriinden kesme işaretleriyle ayrılmış, ekranda her bir veri bir satırın başında görünürler.

PRINT 1, 2, 3

emir, şu görüntüyü verir:

1  
2  
3

Bir PRINT cümlesi veya emir nokta-virgül, virgül veya kesme işaretleriyle sona ererse, bir sonraki PRINT cümlesinde yer alan öge de bu durumdan ayrı biçimde etkilenebilir.

## PRINT ve diğer komutlar

PRINT'ı sırasıyla TAB, bir sayısal değer, bir nokta-virgül ve bir de veri ögesi izleyebilir. Örneğin,

60 PRINT TAB x; a\$

TAB'ı izleyen değer (yukarıda x) gerekiyorsa, en yakın tam sayıya yuvarlanır, ardından 32'ye bölünür ve kalan 0 ile 31 arasında bir değer verecek biçimde ekrana yansır. Bundan sonra, aynı satırda veya bunu izleyen satırda bir veri ögesi hep bu satır pozisyonunda yansır.

PRINT'ı AT, virgüle ayrılmış iki sayısal değer, bir nokta-virgül ve bir veri ögesi izleyebilir. Örneğin,

50 PRINT AT 11, 16, "Data"

İk değer (yukarıda 1) 0 ile 21 arasında ve verinin yansıtılacağı satır ya da sırayı gösterir. İkinci değer (0) 0 ile 31 arasında değişir ve veriden 0 karakter veya rakamın yansıtılacağı sütunun numarasını kabul edilir ve en yakın tam sayıya yuvarlanır. PRINT AT 11, 16, "Data" emir, ekranın ortasına bir yıldı yansıtır.

PRINT'ı ayrıca bir veya daha fazla CHR\$ fonksiyonu da izleyebilir. Bunun ayrıntıları için CHR\$'e bakın.

**PRINT ve renk komutları**  
PRINT'in sağlayacağı görüntü, INK, PAPER, FLASH, BRIGHT, INVERSE

ve OVER'in verdiği ve uygulanmakta olan renk emirleri ve cümlelerinden etkilenebilir. PRINT'ı, bu alt cümleden bir veya birkaç nokta-virgüle ayrılmış olarak izleyebilir. Veri, bunlardan sonra gelir. Örneğin,

50 PRINT AT 11, 16, INK 2, FLASH 1, "

Bunun üzerine veri ögesi, renk komutlarının belirlendiği göndermeler ekrana yansır. Bu göndermeler sıradır ve sadece o ögeye uygulanır. PRINT cümlesinin uygulamaya konmasıyla, en baştaki veya bundan bir önceki genel renklere döner. PRINT ayrıca veriyi yerleştirilmiş yeni renk kontrol kodlarına da yarar. Ötek s. 33).

## Bijimi

PRINT (TAB int-num-exp1;)

[AT int-num-exp1;

int-num-exp2;]

[CHR\$ (int-num-exp1;)

int-num-exp2;] (int-num-exp3;

string-exp1;)[...]

## RANDOMIZE

**Klavyede yeni**  
T

Cümle/Emir

Klavyede RAND olarak gösterilen RANDOMIZE, rastsal veya belirlenmiş aralıklarla bir dizi sayı elde etmek için RND ile bağlantılı olarak kullanılır.

## RANDOMIZE'in kullanımı

RANDOMIZE ya bir program cümlesi ya da ya da emir olarak kullanılır. İzlenirse bir sayısal değer de izleyebilir. Örneğin,

RANDOMIZE 1

10 RANDOMIZE

RANDOMIZE'ı izleyen değer, gerekirse en yakın tam sayıya yuvarlanır ve 0 ile 65535 arasında bir değer olabilir. Bu değer 0 dan büyüğe buna sistem değişkeni SEED tahsis edilir. SEED'ı izleyen RND daha ayrı sayı sırası verir (sistem değişkeni hakkında bilgi için s. 48'e bakın). Sayı sırasını belirleyen RANDOMIZE'in değeridir.

RANDOMIZE'ı bir 0'lıya veya bir 1'e değer izleniyorsa, o zaman SEED e FRAMES diye adlandırılan bir başka sistem değişkeni tahsis edilir. Buna FRAMES (kareler) denir. Bunun, bunun Spectrum'un geliştirilmesinden bu yana geçen televizyon görüntü karelerini saymasıdır. SEED saniyede 50 ya da 60 kez değiştirildiğinden, RANDOMIZE veya RANDOMIZE'ı izleyen RND'in oluşturacağı sayı sıraları hızla rastsaldır.

Eğer RANDOMIZE kullanılmazsa, RND, cümlesinin başlanmasından, reset düğmesine ya da NEW tuşuna basılmasından bu yana ürettiği

sayıları sırasıyla aynı yerde sayı üretir.

**Bijimi**  
RANDOMIZE (int-num-exp)

## READ

**Klavyede yeni**  
EXTEND MODE  
A

Cümle/Emir

READ, DATA ile bağlantılı olarak, bir DATA cümlesindeki değerleri kullanan değişkenlere değer vermek üzere kullanılır.

## READ'in kullanımı

READ normal olarak bir program cümlesi olarak kullanılır. Bunun bir veya birkaç sayısal değişken veya dizi değişken izler ve bunları birbirinden virgüllerle ayırır. Örneğin,

20 READ a\$, x

READ ilk işleme konduğunda, ilk DATA listesinin başından itibaren kaç değişken varsa aynı sayıda değer alır. Bu değerler de değişkenlere aynı sırada verir. READ işleme konmasa, bu READ cümlesindeki değişkenlere bir sonraki DATA değerleri verilir ve bu böyle gider.

Ayrıntılı bilgi için DATA ya bak.

## Bijimi

READ num-var (num-var);

(string-var)

READ string-var (num-var)

(string-var)

## REM

REM

**Klavyede yeni**  
E

Cümle

REM, programa giriş ve habiretler yerleştirmek için kullanılır. Bunlar programın adı, yazan ile bir değişkenin amacı, açıklama için program satırlarına eklenmiş açıklamalar olabilir. Bu giriş, programın işleyişinde hiç bir etki yapmaz, ancak sıralamada görünürler.

## REM'in kullanımı

REM, program içinde kendi başına bir satır olabilir, gibi bir satırın son cümlesi de olabilir. Bunu istediği gibi tuşlara basılarak yazılmış bir giriş olabilir.

50 INPUT r\$; REM r\$ = name

Bilgisayar REM'e rastladığında, o satırda onu izleyemeyi dikkate almaz.

## Bijimi

REM istenen karakterler

## RESTORE

**Klavyede yeni**  
EXTEND MODE  
S

Cümle/Emir

RESTORE, READ ve DATA ile birlikte, READ'in programdaki ilk veya bir sonraki DATA cümlesinden değeri belli bir DATA cümlesinden değeri alınmasını sağlamak için kullanılır.

## RESTORE'in kullanımı

RESTORE normal olarak bir program satırı oluşturur. İzlenirse bunu bir sayısal değer alır. Örneğin,

160 RESTORE 100

RESTORE'ı izleyen değer, gerekiyorsa, en yakın tam sayıya yuvarlanır. Bu değeri, programın DATA cümlesi içeren bir satırın numarası olması gerekir. RESTORE'un ardından, bir sonraki READ cümlesi, bu DATA cümlesindeki değerleri verecektir. Eğer bu numarası takip bir satır yoksa ve bu satırda bir DATA cümlesi bulunmuyorsa, o zaman READ, bundan sonraki ilk DATA cümlesine gider.

Eğer RESTORE'ı bir 0'lıya veya bir 1'e değer izleniyorsa, o zaman READ cümlesi programdaki ilk DATA cümlesine gider.

## Bijimi

RESTORE (int-num-exp)

## RETURN

**Klavyede yeni**  
Y

Cümle/Emir

RETURN, bir altprograma son vermek ve bilgisayarı ana programa ya da bir önceki altprograma göndermek için kullanılır.

## RETURN'in kullanımı

RND, bir cümlede veya bir emir olarak kendi başına kullanılır. Örneğin,

1000 RETURN

İşleme konduğunda, program, uygulanan son GOSUB cümlesini izleyen cümleye uyar. Ayrıntılar için GOSUB'a bakın.

## Bijimi

RETURN

## RND

RND Random number

**Klavyede yeni**  
EXTEND MODE  
T

İşlev

RND, bir rastsal sayı elde etmek için kullanılır.

**RND'in kullanımı**  
RND, bir cümlede veya bir emir olarak kendi başına kullanılır. Örneğin,

60 LET X=RND

Bunun üzerine RND'iden küçük, 0 dan büyük veya 0'a eşit bir rastsal sayı verir.



Spectrum çağrıldığında, reset değerine veya NEW tuşuna basıldığında, RND'nin vereceği sayılar aynı sıradadır. Şöyle, 75 in kuvvetini alarak (75, 75+75, 75+75+75, vb) ve çıkan sayıyı 65537'ye bölüp sadece kalanı kullanıp, kalandan 1 çıkıp, bu sonucu 65536'ya bölerek elde edilir.

Daha rastsal veya daha sabit bir sıra gerekiyorsa, RND den önce RANDOMIZE'yi kullanın.

**Rastsal tam sayılar**

Spectrum'un INK ve CHG gibi binyok cümlesi ya da işlevi sayılan en yakın tam sayıya yuvarlanır. RND bunlarla birlikte doğrudan kullanılabilir. Örneğin, INK RND\*7 rastsal bir mükemmel rengi belirler. Diğer bazı cümle ve işlevler ise tam sayı isteyebilirler. INT (RND\*x)+1, 1'den x'e kadar hep tam sayı verir. 0'dan x'e kadar rastsal bir tam sayı elde etmek için INT (RND\*x+0.5) kullanın.

**Biçimi**  
RND

**RUN**

**Klavyede yeri**  
R

**Emir/Cümle**

RUN, programı normal olarak ilk satırdan başlatmak üzere işleme koyar.

**RUN'un kullanımı**

RUN, doğrudan bir emir olabileceği gibi bir program satır da oluşturabilir. İhtiyaçla bunu sayısal bir değer de isteyebilir. Örneğin,

**RUN 50**

Eğer RUN'u bir değer izlemiyorsa, program ilk satırdan işleme konur.

Eğer bir değer eklenmişse, bu gerekiyorsa en yakın tam sayıya yuvarlanır ve program bu satırdan başlar. Eğer böyle bir satır yoksa, program bundan sonra gelen satırdan başlar. Dikkat edin, RUN, programı işleme koymadan önce CLEAR'de uygular, bu yüzden mevcut değişken değerleri silinir.

Bunu önlemek için, ardından bir satır numarası gelen GOTO kullanın.

Eğer bir program LINE kullanılarak saklanıyorsa, yüklenir yüklenmez otomatik olarak işleme konur, RUN'a gerek olmaz.

**Biçimi**  
RUN [int-num-exp]

**SAVE**

**Klavyede yeri**  
S

**Emir/Cümle**

SAVE, bir programı kasette saklamak için kaseti teypte gönderir.

**SAVE'nin kullanımı**

SAVE normal olarak doğrudan bir emir olarak kullanılır ama bir program cümlesi de oluşturabilir. Buna, bir dizi değer olan dosya adı izler. Örneğin,

**SAVE "dosyaad"**

Dosya adı en fazla on karakter olabilir. Uygulamaya konduğunda ekranda

**Start tape, then press any key**

"Teypti çalıştırın, sonra herhangi bir tuşa basın" mesajı görünür.

Herhangi bir tuşa basılınca program kaseti teypte gönderilir ve sonuçta

**BOOK, 1** mesajı ekranda belirir.

Dikkat edin, Microdrive bağlandığında SAVE farklı kullanılır.

Bunun ayrıntısı için Microdrive ve Interface 1 el kitabına bakın.

**Otomatik işleme koyma**

Eğer depolanmış program, yüklenince otomatik olarak işleme konacaksa, SAVE LINE ile bağlantılı kullanılmalıdır. Program adını LINE ve bir sayısal değer izler. Örneğin,

**SAVE "dosyaad" LINE 1**

LINE'i izleyen değer, gerekiyorsa en yakın tam sayıya yuvarlanır. Bu durumda 1 veya programdaki bir satırın numarası olmalıdır. Program daha sonra teypte aynı SAVE'de yapıldığı gibi gönderilir. Programı yüklenince, bu verimsiz olan satırdan veya böyle bir satır yoksa programda bunu izleyen satırdan başlayarak otomatik olarak işleme konur.

Pratikte LINE 1'in kullanılması bütün programın otomatik olarak başlatılmasına neden olur.

**Biçimi**  
SAVE string-exp [LINE int-num-exp]

**SAVE CODE**

**Klavyede yeri**  
S

**Emir/Cümle**

SAVE CODE, belirlenmiş bulunan bir bölümün bilgisi kasette saklanmak üzere kaseti teypte gönderir. Daha sonra bu bilgi LOAD CODE kullanılarak tekrar belleğe yerleştirilebilir.

**SAVE CODE'nin kullanımı**

SAVE CODE doğrudan bir emir veya programın bir cümlesi olarak kullanılabilir. SAVE'i, bir dizi değer olan dosya adı ile CODE izler. Örneğin

**SAVE "picture" CODE 16384,6912**

SAVE'i izleyen dosya adı en fazla on karakterden oluşur. CODE'u izleyen iki değerdendir bir bir gerekiyorsa en yakın tam sayıya yuvarlanır. İki (yukarıda 16384) bellekteki bölgenin başlangıç adresini, ikinci (6912)

ise, depolanacak byte'ların sayısını verir. Bunun üzerine bilgi, tipik SAVE ile programa yapıldığı gibi, kaseti teypte gönderir.

Yukarıdaki emirle saklanan bilgi ekrandaki görüntüdür.

**Biçimi**  
SAVE string-exp CODE int-num-exp [int-num-exp]

**SAVE DATA**

**Klavyede yeri**  
S

**Emir/Cümle**

SAVE DATA, bir düzengi teypte depolar. Bundan sonra da DİZEN LOAD DATA ile yüklenir.

**SAVE DATA'nın kullanımı**

SAVE DATA, bir program cümlesi olarak kullanılabilir. Doğrudan emir de olabilir. SAVE'i bir dosya adı, DATA, bir harf veya bir harf \$ ve nihayet bir boş parantez izler. Örneğin,

**450 SAVE "numbers" DATA R)**

750 SAVE "names" DATA R\$)

DİZEN'in dosya adı en fazla on karakterden oluşur. DATA'yi izleyen harf veya harf\$ teypte saklanacak olan programdaki DİZEN'in addır.

Bunun üzerine DİZEN, tipik SAVE ile programa yapıldığı gibi, kaseti teypte gönderir.

**Biçimi**  
SAVE string-exp DATA letter [S] O

**SAVE SCREENS**

**Klavyede yeri**  
S

**Emir/Cümle**

SAVE SCREENS ekrandaki görüntüyü teypte depolar. İleride bu LOAD SCREENS kullanılarak yeniden bilgisayara yüklenebilir.

**SAVE SCREENS'in kullanımı**

SAVE SCREENS doğrudan bir emir veya bir program cümlesi biçiminde kullanılabilir. SAVE'i bir dizi değer olan bir dosya adı ile SCREENS izler. Örneğin,

**SAVE "picture" SCREENS**

Dosya adı en fazla on karakterden oluşur. Bunun üzerine görüntü, tipik SAVE ile programa yapıldığı gibi, kaseti teypte gönderir.

**Biçimi**  
SAVE string-exp SCREENS

**SCREENS**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SCREENS, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

SIN (num-exp)

**SQR**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SQR, bir rakamın kare kökünü verir.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SGN y**

İfade paranteze alınmalıdır. SGN, önemsenin (yukarıda y) değeri artı ne 1, eksi ne -1, sıfır ise 0 görüntüsü verir.

**Biçimi**  
SGN num-const

**SGN (num-exp)**

**SIN**

**Klavyede yeri**  
EXTEND MODE

**Emir/Cümle**

SIN, bir rakamın artı, eksi veya sıfır olduğunu işaret eder.

**SGN'nin kullanımı**

SGN'yi bir sayısal değer izler. Örneğin,

**50 LET x=SIN y**

İfade paranteze alınmalıdır. SIN'i izleyen değer radian olarak bir açının değeridir. SIN açının sinüsünü verir.

**Biçimi**  
SIN num-const

**SIN (num-exp)**

**SQR**

**Klavyede yeri**  
EXTEND MODE

Klavyede yerli  
SYMBOL SHIFT F

İşlev

TO'nun Sincir BASIC'te işlevi değişik kullanımı vardır. FOR'a bağlanış olarak FOR NEXT döngüsü oluşturmak için (ayrım) için bak FOR'da ve dizi bölücü (dizinin daha küçük alt dizilerine bölmek için) olarak kullanılır.

TO'nun dizi bölücü olarak kullanımı

TO bir ana dizindeki bir alt dizinin ilk ve son karakterlerinin belirlenmek için kullanılır. TO'dan önce bir dizi değeri gelir. Parantez açılır, istenirse bir sayısal değer verilir. TO'dan sonra da istenirse bir başka sayısal değer verilir parantez kapanır. Örneğin:

80 PRINT a\$(4 TO 7)

Dizi ifade paranteze alınmalıdır. Dizi değeri (yukarıda a\$) bölünecek olan dizedir. İki sayısal değer (4 ve 7) bu dizi içindeki alt dizinin ilk ve son karakterleridir. Bunun üzerine TO alt diziyi (a\$'nin 4'ten 7'ye kadar olan karakterlerini) verir. Her bir şey verimse iki sayısal değer 1, son sayısal değer de dizideki son karakterin pozisyonunun değeridir. Eğer alt dizi dizinin ilk karakteri ile başlayacaksa ilk değer verimseylebilir. Eğer alt dizi dizinin son karakteri ile sona eriyorsa, son değer verimse de olur.

Biçimi  
string:const [(num-exp) TO  
(num-exp)]  
string-var [(num-exp) TO  
(num-exp)]  
(string-exp) [(num-exp) TO  
(num-exp)]

USR User SubRoutine

Klavyede yerli  
EXTEND MODE

İşlev

USR, bellekte belli bir adrese yerleştirilmiş olan bir makina kodu altprogramını işletmek için kullanılır. USR ayrıca, kullanıcının tanımladığı grafikler için verilen bellegin üst kısmında aynımsi yere yerleştirilmesi için de kullanılır.

USR ve makina kodu

Makina kodunu kullanabilmek için USR'yi bir sayısal değer ister. Örneğin:

80 PRINT USR 65000  
100 RANDOMIZE USR 65000

İfade paranteze alınmalıdır. USR'yi isteyen değer bir tam sayıya yuvarlanır ve makina kodunun bellekteki başlangıç adresini oluşturur. USR'çeren herhangi bir cümle bu adresteki altprogram ister ve USR "bc" kutuk çıkışın işlevinin değerini verir. Örneğin,

RANDOMIZE USR veya RESTORE USR sadece altprogram işleme koyarken, PRINT USR buna ek olarak "bc" kutuk değerini de ekrana yansıtır.

USR ve kullanıcının tanımladığı

grafikler  
Kullanıcının tanımladığı grafikler için USR, POKE ile birlikte kullanılır. Bunun bir dizi değeri ya da değişken izler ve bunlar POKE cümlesi için bir adres verir. Örneğin:

50 POKE USR "a", 255

USR'yi isteyen dizi değeri, A ile U veya a ile u arasında bir tek harftir. Büyük harflerle küçük harfler arasında bir ayırım göçmez. Bunun üzerine USR, bellegin kullanıcının tanımladığı grafiklere ayrılmış 21 bölümden birinin başlangıç adresini verir. Her bölüm, bir grafik karakter yaratacak şekilde sekiz byte'in POKE edilebileceği sekiz adres içerir. Byte'ler ondalık veya iki biçimde (bak BIN) verilebilir.

Biçimi

USR int-num-const

USR int-num-var

USR (int-num-exp)

USR string-const

USR string-var

VAL Value

Klavyede yerli  
EXTEND MODE

İşlev

VAL, sayısal değere sahip bir diziyi sayıya çevirir.

VAL'in kullanımı

VAL i bir dizi değeri veya

değişken izler. Örneğin,

70 LET x=VAL a\$

Dizi değeri veya değişkenin değeri tmak işaretlerinden çıkarılır ve bu durumda sayısal bir değer olması gerekir. VAL bunu değerlendirir ve bu bir sayısal değeri olarak verir.

Örnekler

a\$'nin değeri "435" ise, o zaman yukarıdaki cümle x'e 435 değeri verir. Ancak VAL, ifadelerin değerlendirilebilir.

10 INPUT a\$,x

20 PRINT VAL a\$

a\$'yi verilen değer x'i kullanan bir ifade olmadıkça, "x=x" gibi. Bunun üzerine x'e örneğin 5 gibi bir sayısal değer verilir. VAL, bu ifadeden tmak işaretlerini çıkıp x=x elde eder ve x'e verilen değeri değerlendirilerek ekrana sonuç olarak 25'i yansıtır.

Biçimi

VAL string-const

VAL string-var

VAL\$ Value (string)

Klavyede yerli

EXTEND MODE

SYMBOL SHIFT J

İşlev

VAL\$ bir diziyi bir dizi ifade olarak değerlendirir.

VAL\$'in kullanımı

VAL\$ i bir dizi değeri izler.

Örneğin,

130 PRINT VAL\$a\$

Dizi değeri değeri, tmak işaretlerinden ayrılır. Bu durumda bunun bir dizi ifade olması gerekir. VAL\$ ifadesi değerlendirilir ve değeri bir dizi değeri olarak verir.

Örnekler

Bu programı deneyin:

10 INPUT a\$,x\$

20 PRINT VAL\$a\$

a\$'e verilen dizi değeri x\$'yi kullanan bir ifade, örneğin "x\$+x\$" olmalıdır. Bu dizi değeri, bunun üzerine, x\$ "DO" ya verilir örneğin, VAL\$a\$, x\$'nin değerinden tmak işaretlerini ayıklayıp x\$+x\$'i alır ve x\$'ye verilen değeri değerlendirir, örneğin sonuç olan DODO'ya ekrana yansıtır.

Biçimi

VAL\$ string-var

VERIFY

Klavyede yerli

EXTEND MODE

SYMBOL SHIFT R

Emir/Cümle

VERIFY, SAVE'den sonra bir programın doğru bir biçimde teypte aktarılıp aktarılmadığını denetler.

VERIFY'in kullanımı

VERIFY, Normal olarak, tpki LOAD gibi doğrudan bir emir olarak verilir ve bunu program adı izler. Örneğin,

VERIFY "dosyaad"

Teyp çalışırken, bulunan her programın adı ekrana yansır ve bellekteki programla aynı adı taşıyan teypteki program karşılaştırılır. Eğer ikisi de aynı ise

OK,diğer

mesaj verilir.

VERIFY bir Microdrive bağlı ise farklı kullanılır. Aynımsi için Microdrive ve Interface 1 el kitabına bakın.

VERIFY CODE ve VERIFY DATA

VERIFY CODE, tpki LOAD CODE gibi, bellekteki bir bilgini bölünmüş teypte aktarılmasını doğrulamak için kullanılır. VERIFY DATA da LOAD DATA gibi bir dizinin teypte aktarılmasını denetlemek için kullanılır. Ayrıntılar için LOAD CODE ve LOAD DATA'ya bakın.

Biçimi

VERIFY string-exp

VERIFY string-exp CODE

[(int-num-exp) [(int-num-exp)

VERIFY string-exp DATA

letter (SIO)

VERIFY CODE

Klavyede yerli

EXTEND MODE

SYMBOL SHIFT R

EXTEND MODE

İ

Bak VERIFY

VERIFY DATA

EXTEND MODE

SYMBOL SHIFT R

EXTEND MODE

D

Bak VERIFY









## ZX SPECTRUM + EKRAN MESAJLARI

Spectrum, BASIC'i uygularken durursa, ekranın alt kesiminde bir mesaj belirir. Bu, ya bir emrin ya da programın tamamlandığını ya da bir hata olduğunu gösterir. Her mesaj, bir kod numarası veya harf ile kısa bir mesaj ve bilgisayarın durduğu satır ya da cümle numarasından oluşur. Emir, 0. satır ve bir satır başında ise 1. cümle, ilk nokta virgülden sonra ise 2. cümle vb olarak gösterilir. CONTINUE, normal olarak programın ekranda gösterilen cümleden yeniden başlamasını sağlar.

**0 OK**  
İşlemin başını ile tamamlandığını veya programda bulunan satır numaralarının daha büyük bir satıra atlama girişimini gösterir. CONTINUE, bu mesajı dikkate almaz ve bir önceki mesajda belirtilen cümleden yeniden başlar.

**1 NEXT without FOR**  
Kendisini izleyen bir FOR olmayan bir NEXT ile karşılaşılmıştır ve kontrol değişimi ile aynı adı taşıyan bir başka değişim daha vardır.

**2 Variable not found**  
Bir değer verilmemiş veya teypten değer yüklenmeden bir basit değişim kullanılmış; veya önce FOR cümlesi ile tanımlanmadan NEXT ile bir kontrol değişimi kullanılmış; veya DIM ile düzenin boyutu tanımlanmadan ya da bir düzen teypten yüklenmeden önce bir altöge değişimi kullanılmış demektir.

**3 Subscript wrong**  
Altöge, düzenin boyutları dışında.

**4 Out of memory**  
Cümleyi ya da emri tamamlamak için bellekte yeterli yer yok.

**5 Out of screen**  
INPUT, ekranın alt bölümü için 23 satırdan fazla satır vermiş ya da PRINT AT ile birlikte 22 ya da daha yüksek bir satır numarası kullanılmış demektir.

**6 Number too big**  
Bilgisayar 10<sup>9</sup>'den daha büyük bir rakam elde etmeye çalışmıştır.

**7 RETURN without GOSUB**  
RETURN cümlelerinin numarası GOSUB cümlesinin numarasından bir büyüktür.

**8 End of file**  
Microdrive dosya işleme emri.

**9 STOP statement**  
Programı durdurmak için STOP kullanılmıştır. CONTINUE bir sonraki cümleden başlatacaktır.

**A Invalid argument**  
İşleve yanlış değer verilmiştir.

**B Integer out of range**  
Bir değer en yakın tam sayıya yuvarlanmış, ancak boyutları dışında olduğu için kabul edilememiştir.

**C Nonsense in BASIC**  
Cümle, program çeviricisinde BASIC açısından bir anlam ifade etmiyor.

**D BREAK - CONT repeats**  
BREAK'e basılmıştır. CONTINUE programı durdurulduğu cümleyi tekrarlayacaktır.

**E Out of DATA**  
READ, programdaki son DATA cümlesinin ötesinde birşeyler okumaya kalkışmıştır.

**F Invalid file name**  
SAVE, ondan fazla karakterden oluşan bir adla birlikte kullanılmıştır.

**G No room for line**  
Yeni program satırının gireceği yeterli bellek yeri yoktur.

**H STOP in INPUT**  
INPUT'a karşılık olarak veya veri verilirken STOP'a basılmıştır. CONTINUE INPUT cümlesini tekrarlar.

**I FOR without NEXT**  
Limitli veya STEP değeri yanlış olduğu için (örneğin FOR x=5 TO 0 STEP 1 kullanıldığında) ve bunu karşılayacak NEXT bulunmadığından bir FOR NEXT döngüsü uygulanamamaktadır.

**J Invalid I/O device**  
Microdrive dosya işleme emri.

**K Invalid colour**  
INK, PAPER, FLASH, BRIGHT, INVERSE veya OVER ya da bunlara ilişkin kontrol karakteri için belirtilen değer, kullanım alanı dışındadır.

**L BREAK into program**  
BREAK 'e basılmıştır. Bu mesaj uygulanacak son cümleyi gösterir. CONTINUE bir sonraki cümleden başlatacaktır.

**M RAMTOP no good**  
RAMTOP için belirlenen değer, ya çok büyük ya da çok küçüktür.

**N Statement lost**  
Artık mevcut olmayan bir cümleyle atlama girişimi olmuştur.

**O Invalid stream**  
Microdrive dosya işleme emri.

**P FN without DEF**  
Bunu karşılayan bir DEF FN cümlesi olmadan bir FN cümlesi kullanılmıştır.

**Q Parameter error**  
FN cümlesi, fonksiyona iletilecek üzere yanlış sayıda değerler içermektedir veya değerlerden biri yanlış türdedir (örneğin sayı yerine dizi ya da ters).

**R Tape loading error**  
Yükleme, birleştirme veya doğrulama işlemleri başarısızlığa uğramıştır.

## BASIC'İNÖTESİ

BASIC, birçok uygulamada çok iyi iş gören, çok amaçlı bir bilgisayar dildir. Ancak bu, Spectrum'da kullanılabilecek tek bilgisayar dili değildir. FORTH, micro-PROLOG ve LOGO gibi diğer dillerdeki yazılar da bulunmaktadır. Bu diğer BASIC'ten çok farklı işler ve bilgisayarınıza yeni olanaklar açar.

BASIC çok amaçlı bir dil olduğundan, bazı uygulamalarda hantal kalabilir. Göreceli olarak yavaşır. Diğer diller, daha basit bir programlama ve daha hızlı işleme ile birleşik daha büyük esneklik sağlayabilirler. Örneğin, FORTH, size kendi kelimelerinizi tanımlama ve bunları bilgisayarınızın anlayacağı, BASIC'deki benzer emirlerden on kat hızla uygulayacağı talimatlarınızda kullanma olanakları sağlar. Micro-PROLOG ile bilgisayar basit İngilizce cümleleri anlar, bunları kullanan ile basit bir karşılıklı konuşma için belleginde tutar. LOGO eğitimde kullanılmak üzere geliştirilmiş bir bilgisayar dildir. Çok esnek bir biçimde kullanılabilecek çok basit emirler içerir. Ancak ZX Spectrum+ 'unuzu gerçekten hızlı programlar yazmak istiyorsanız, makina kodu ile nasıl programlama yapacağınızı anlamamız gerekir.

**Makina kodu**  
BASIC sizin kolaylıkla anlayabileceğiniz biçimde bilgisayara talimat vermenizi sağlar. Spectrum'un Merkezi İşlem Birimi (CPU), yani göçü Z80A yongası, aslında BASIC'i anlamaz. Belleğin bir kesimi, BASIC çevirmeni denen, sizin BASIC ile verdiğiniz talimatları kodlu sinyallere dönüştüren bir kâğıt program içerir. Aslında Z80A'nın sizin talimatlarınızı uygulamasını sağlayan bu kodlardır.

Bu çevirmenin sizin BASIC ile verdiğiniz talimatları Z80A koduna ya da yaygın adıyla makina koduna çevirmesi biraz zaman alır. Ancak isterseniz BASIC çevirmenini atlayıp makina kodlarınızı doğrudan Z80A'ya gönderebilirsiniz. Bunun üzerine programınız hızla işleme konur. Bunun için ödenmesi gereken bedel, makina kodu ile program yazmak için harcanacak fazladan zamandır. BASIC ten farklı olarak bu hiç de "dostça olmayan" bir dildir. Yani bunu öğrenmek epey bir zaman alır. Makina kodu ile programlama bu kitabın boyutları dışındadır. Ancak Spectrum'un makina kodunu çok ileri düzeyde öğrenen kitaplar vardır. Makina kodunun hızının tadına varabilmemiz için şu kısa göster programını yazıp verin.

### ANINDA ÇİZGİLER

```
10 FOR X=0 TO 15
20 REPEAT N: FOR Y=0 TO 255:PRINT X,Y,N
30 NEXT Y
40 NEXT X
50 DATA 22,255,60,1,1,26,22
60 DATA 35,1,120,1,77,200,114,
70 DATA 22,255,60,1,1,26,22
80 DATA 35,1,120,1,77,200,114,
90 DATA 22,255,60,1,1,26,22
```



50. satırdaki 55'in değerini 1 ile 255 arasındaki başka herhangi bir rakama çevirin ve çubukların değişimini izleyin. Ayrıca başta bir INK emri kullanarak renkli çubuklar da elde edebilirsiniz. Makina kodu kullanınca ne kadar hızla görüntü elde ettiğinize bakın. Bu, adeta anında olmaktadır. BASIC in ise ekranı doldurması iki saniye sürer.

Bu program, 10 ile 30. satırlarla, bellekte 65000 adresinden sonrasına DATA cümlesi ile 16 kod yerleştirildiğinden işlemektedir. 70. satır kodları Z80A'ya göndermekte ve derhal görüntü oluşmaktadır. Dikkat edin sekizinci kod, çubukların genişliğini kontrol etmektedir.

Spectrum'unuz için mevcut oyunların çoğu, mümkün olan en hızlı sonucu elde etmek için makina kodu ile yazılmıştır. Makina kodu yazabilmemiz için "düzenleyici" (assembler) denen bazı programlar mevcuttur. Makina kodu sadece bazı rakamları isterken, bunları size tuflara basarak vereceğiniz talimatları sağlar. Bu talimatlar, BASIC komutları gibi İngilizce kelimeler değil, bilgisayarın mutlakla yerine getirmesi gereken işlemleri ifade eden bazı kısaltmalar ya da özel deyimlerdir. Bu yüzden, "düzenleme" dilini kullanmadan önce bilgisayarın nasıl çalıştığını adım adım öğrenmeniz gerekir.

## BİLGİSAYAR JARGONU — NE ANLAMA GELİYOR

Bilgisayarda kullanılan kelimelerin çoğu günlük hayatta kullanılan kelimelerdir ama çoğunun anlamı farklıdır. Burada, bu kitapta kullanılan kelimelerden bazıları ile bilgisayarla ilgili bazı deyimlerin açıklamasını bulacaksınız. İtaliye yazılmış kelimeler bu sayfalarda ayrıca açıklanmak. Eğer bu kitapta anlamadığınız bir kelime ya da deyim varsa ve burada bulamazsanız, o zaman dizine bakın.

**Address (Adres)** Belleğin bir tek birimi. ZX Spectrum + 'da 65536 adres vardır.

**Argument (Önerme)** Bir sonuç almak için bir işlemin kullandığı değer.

**Array (Düzen)** Belleğin bir kesiminde bir arada tutulan, birbiriyle bağlantılı, birgrup veri.

**Attributes (Gönderme)** Karakterlerin renklerini veren kodlar.

**BASIC** ZX Spectrum + ve diğer ve bilgisayarlarının kullandığı bilgisayar dili.

**Binary code** (İkili kod) Bilgisayarların kullandığı bir tür kod. Akimin açık ya da kapalı olması durumundan oluşan bazı obeklerdir.

**Bit** (İkili kodda açık ya da kapalı olma durumu). İkili rakamın kısaltması.

**Byte** Değeri 0 ile 255 arasında bir rakamı ifade eden sekiz bit'ten oluşan bir dizi. Bellekteki her adreste bir byte vardır.

**Character** (Karakter) Ekrana yansıtılan veya yazdırılan, tek bir harf, (0 ile 9 arasında) bir rakam, bir işaret veya grafik birimi.

**Character set** (Karakter tablosu) Bilgisayarın kullandığı önceden belirlenmiş karakterlerin ve bazı kontrol kodlarının tam listesi.

**Command** (Emir) Bilgisayarın uyguladığı bir tek talimat ve doğrudan emir.

**Concatenation** (Küme) Birbirine ekleyerek dizilen birleştirme.

**Constant** (Değişmez) Bir sayı veya bir ya da birkaç harf ya da başka karakterden oluşan grup.

**CPU (Central Processing Unit)** (Merkezi İşlem Birimi) Bilgisayarın işlemleri yapan ve

diğer birimleri denetleyen ana parçası. ZX Spectrum + bir Z80 mikro bilgi işlem aygıtı kullanır.

**Cursor** (Göstergeç) Ekranda bir sonraki görüntünün belireceği pozisyon. Bunu bilgisayarın hangi modda olduğunu gösteren yanıp sönen bir işaret belirir.

**Data** (Veri) Bir sonuç elde etmek için bilgisayara beslenen veya bilgisayarın bir programdan elde ettiği bilgi.

**Direct command** (Doğrudan emir) Bilgisayara verilir verilmez derhal uygulanan bir ya da birkaç talimat.

**Edit** (Düzeltilme) Bir programdaki ayrıntıları değiştirme.

**Enter** (Verme) Tamamlanmış bir talimat veya bilgiyi bilgisayara verme.

**Expression** (İfade) Değişmez, değişken komutlardan oluşan bir birleşim

**False** (Yanlış) Bilgisayarın gerçek dışı ya da yanlış olduğuna karar verdiği bir durum veya sonuç. False'ın sayısal değeri 0'dır.

**Function** (Fonksiyon/İşlev) Bilgisayarın bir veya birkaç değer (veya önerme) alıp, bunları kullanarak bir başka değer olan sonucu verdiği işlem.

**Graphics** (Grafik işaretler) Bilgisayarın resim, çizim veya şema gibi görüntüler oluşturmaları.

**Hardware** (Donanım) Bilgisayarın kendisi veya bununla ilişkili yan donanım gibi araç ve makineler.

**Information** (Bilgi) Bilgisayarın işlemesi istenen kelimeler, rakamlar ve işaretlerden oluşan herhangi bir bileşim.

**Input** (Girdi) Bilgisayara verilen program ve veriler.

**Interface** (Araparça) Bilgisayar ve/veya yan donanım birbirine bağlayan ve bunların birbiriyle haberleşmesini sağlayan bir birim.

**K** Bir bilgisayarın bellek kapasitesinin ölçüsü. 1K bir kilobyte'a, yani 1024 byte'a eşittir. K olarak bellek kapasitesi, her bir byte depolayabilecek bellekteki adreslerin toplamına eşittir. ZX Spectrum + 'in 48K RAM ve 16K ROM, toplam 64K kapasitesi vardır.

**Keyword** (Komut) Bilgisayara BASIC'le verilen talimat. İşleme girebilmesi için bazı değerlere gerek olabilir.

**Line** (Satır) Bir programdaki bir talimat veya bir dizi talimat. Bunun diğer satırlarla birlikte

doğru bir noktada işleme girebilmesi için bir numara verilir.

**Listing** (Sıralama) Sırayla dizilmiş program satırları.

**Load** (Yükleme) Bir kartuş veya kaset gibi bir depolama aygıtından bilgisayara program veya veri verme.

**Logic** (Mantık) Bilgisayarın sonuçların doğru veya yanlış, bir durumun geçerli veya geçersiz olup olmadığına karar verdiği süreç.

**Machine code** (Makina kodu) ZX Spectrum + 'un anladığı dil. Bilgisayar BASIC'le verilmiş programları işlerken bunları makina koduna çevirir.

**Memory** (Bellek) Bilgisayarın gerektiğinde program ve verileri, ayrıca kalıcı işletme talimatlarını saklayan kesimi.

**Mode** (Mod) Spectrum'da klavyedeki her tuşla üretililecek komut veya karakterlerin elde edilebilecekleri beş durumdan biri. Program sırasında modun hangisi olduğunu göstergeçte yanıp sönen harf gösterir.

**Nesting** (Yuvalama) Programda bir döngünün ya da birkaçının bir başkasının içinde uygulanmasını sağlayan düzenleme.

**Numeric variable** (Sayısal değişken) Değişken bir sayı içerir. Sayısal değişkenler, bir veya birkaç harften oluşur.

**Operator** (İşlem) Aritmetik ya da mantık işlemi yapılmasını sağlayan talimat.

**Output** (Çıktı) bilgisayarın sağladığı sonuçlar.

**Peripheral** (Yan donanım) Bilgisayara bağlı herhangi bir araç.

**Pixel** Ekranda görünebilecek en küçük renkli nokta. İngilizce picture cells – resim hücrelerinin kısaltılması.

**Print** (Yaz) Ekranda sonuçları veya grafik işaretleri yansıtan ya da yazıya yazdıran emir.

**Program** (Program) Bilgisayarın uygulayacağı bir sıra talimat.

**RAM** (Random Access Memory) (Rastal Ulaşılan Bellek) Belleğin bir program, veriler veya diğer değişkenler depolayabileceği kesimi. Ayrıca değişken bellek diye de bilinir. Cereyan kesildiği zaman RAM'in içeriği silinir. ZX Spectrum + 'in 48K'lık bir RAM'i vardır.

**Register** (Kütük) Ana bellek dışında bir küçük bellek birimi. CPU içindeki kütükler,

bilgisayar işlemlerinin yapılmasında kullanılır.

**Report** (Mesaj) Bilgisayarın bazı eylemlerini haber veren, ekrana yansıtılan bilgiler.

**Resolution** (Tanım) Bilgisayar çizimlerinde ne ölçüde ayrıntı verilebileceğinin göstergesi.

**ROM** (Read Only Memory) (Sadece Okuyucu Bellek) Belleğin kalıcı bazı programlar ya da bilgisayar için talimatlar içeren kesimi. ZX Spectrum + 'un 16K'lık bir ROM'u vardır.

**Save** (Sakla) Bir program veya verinin kartuş veya kaset gibi bir araçta depolanması.

**Scroll** (Kaydır) Bir ekrandan fazlasını doldurabilecek bilginin ekrana yansımaları sağlayacak hareket.

**Software** (Yazılım) ROM ve kartuşlardaki daimi programlar da dahil herhangi bir program.

**Statement** (Cümle) Bir program satırında bir talimat oluşturmak üzere kullanılan bir komut veya talimatın kendisi.

**String** (Dizi) Rakamlardan veya sayısal değişkenlerden ayrı tutulmak üzere bırak işaretleri içine alınmış bir veya birkaç karakterden oluşan grup.

**String variable** Bir dizi içeren bir değişken. Dizi değişkenler daima bir harf ve \$ işaretinden oluşur.

**Syntax** (Söz dizini) Geçerli bir BASIC talimatı oluşturmak üzere, doğru bir biçimde sıralanmış komutlar, değişmezler, değişkenler ve ifadeler.

**True** (Doğru) Bilgisayarın doğru veya geçerli saydığı bir durum veya sonuç. Doğrunun sayısal değeri 1'dir.

**Value** (Değer) Bir değişmez, değişken veya ifade tarafından verilen veya temsil edilen bir sayı veya dizi.

**Variable** (Değişken) Bilgisayar tarafından kullanılmak üzere belli bir değişmezi saklayan, belleğin bir veya birkaç birimi. Her birine kolayca tanımlanabilmesi için bir ad veya bir harf verilmiştir. ZX Spectrum + , sayısal değişkenlerle dizi değişkenler arasındaki farkı ayırt eder.



## Dizin

İtali olarak verilen sayfa numaraları, resimler veya başlıklara işaret eder.

Adlar programı 8  
Alçak tanımı çizimler 26-7  
Altprogram 30-1  
Anten girişi ve bağlantısı 4-5  
Ara çubuğu (tuşu) 19  
Araparçalar (interface) 45, 46-7  
Animetik işlemi 22, 22  
ATTR 35

Bağlantılar 5  
Elektrik 5  
Kasetli teyp 13  
Televizyon 4  
Bağlantısı Mantık Düzeni (ULA) 42  
BASIC 18, 49-73  
BEEP 36, 18  
Belek 12, 42, 43, 44-8  
Belek haritası 48  
BIN 33  
BREAK 19  
BRIGHT 31  
Büyük harf modu 21, 20

CAPS LOCK 21, 18  
CAPS SHIFT 8, 21, 18  
Cereyan bağlantısı 4, 5, 43  
CIRCLE 28  
Cümleler 22, 50  
Çarpım Tablosu programı 23  
Çarpma 34-5  
Çerçeve rengi 24-5, 6  
Çiğın mozaik programı 10  
Çözge, alçak tanımı 26, 80  
yüksek tanımı 28, 80  
Çizimler ve hareketli görüntüler 34-5  
alçak tanımı 26-27  
desenler 30-1  
karakter yaratıcısı 32-3  
rastsal etkiler 30  
renk 24-5  
şekillerin doldurulması 29, 29  
yüksek tanımı 26, 28-9  
Çokgen programı 10  
Çubuklu grafikler 25, 25

DATA 33  
Değişkenler 22-3, 50  
DELETE 10  
Depolama 44, 45  
Desenler programı 9  
Dizi 22  
Doğan güneş programı 11  
9VDC girişi 5, 43  
Donanım, tanımı 12  
Dongüler 26-7, 30  
DRAW 28-9  
Düzeltilme, hataların 10, 21  
EAR (kulaklık) girişi 37, 5, 13

EDIT (düzeltme) 18, 21  
program satırı 21  
Ekranındaki mesajlar 74  
Emirler 22, 50  
ENTER 9, 10, 11, 19  
EXTEND MOOE 8, 21, 18  
Genişletilmiş mod 21, 20

FLASH 31  
FOR NEXT 26-7, 29, 30, 31, 34  
FORTH 75  
Girdi-çıkışı yolları 45  
Ginşler 5  
GOTO 23  
Gökkuşuğu programı 26-7  
Görüntü kontrolü 19  
Grafik modu 21, 20  
GRAPH 21, 18, 26

Hareketli görüntüler (animasyon) 34-5  
Harf modu 21, 20  
Hatalar, düzeltilme 10, 21  
ekran mesajları 74  
Hızlı Yanım 12, 13, 13  
Hesaplama 22-3, 22, 23  
IF THEN 29  
INPUT 23, 29  
INV VIDEO 18  
INVERSE 31  
İşaretler, hesaplama 22, 50  
Seçimi 19  
İkili kodlar 44  
İşlevler 50  
İşleme koyma, programların 8-9

Kağıt rengi 24-5  
Karakter tabloları 51  
Karakterler, seçimi 20  
yapımı 32-3  
Kareler programı 30  
Kartuşlar, Microdrive 12, 46, 46  
ROM 12, 47, 47  
Kaset 12, 44, 45  
Bakımı 12  
Etiketlemesi 14  
Saklanması 12  
Sesi 12  
Kasetli teyp, amplifikatör olarak 37, 37  
Bağlantısı 5-13, 13  
Program saklama 38-40  
Program yükleme 14-16  
Sayacı 14  
Seçimi 12  
Ses kontrolü 14, 15, 16  
Ton kontrolü 14, 15, 16  
Klavye 18-19  
Grafik karakterleri 26-7  
Modlar 20-1  
Komut modu 20, 20  
Kartuşlar 9, 18-19, 50, 52-73, 20-1  
Seçimi 19-20  
Kullanımın tanımladığı karakterler 80, 32-3  
Kundela kablo 46

LET 23  
LIST 21  
LOAD 14-16  
LOGO 75  
Makina kodu 75  
Mantık yongaları 43  
Merkezi İşlem Birimi (CPU) 43-44, 48, 75, 43, 45  
MIC girişi 37, 5, 13  
Microdrive 46, 5, 46  
Kartuşları 12, 45  
Yüklenmesi 46  
Micro-PROLOG 75  
Modemler 46  
Modlar 20-1  
Mürekkep rengi 24-5  
Müzik 36-7

NEW 11, 12, 18  
Nokta 23, 51  
Noktalama işaretleri 23, 51  
Noktalar virgöl 23, 51  
Oparlar 43  
Oyun kolu (joy stick) 45, 47  
Parantez 23  
Parazit 4  
Piramit programı 31  
Pixel 28  
PLOT 28, 28  
POKE 48  
PRINT 22  
Program sabırları, sıma 21  
Düzeltilme 21  
Program verme 8-9  
Programda değişiklik 9  
Programlama 17-40  
Programlar

Değiştirme 9  
Doğrulama 39  
Hataları düzeltilme 10  
İşleme koyma 8-9, 44  
Saklama 13, 38-40  
Verme 8-9, 44  
Yeniden başlatma 10  
Yenisine başlatma 11  
Yükleme 12-13, 14-15, 14-15  
Programların yeniden başlatılması 10  
Rakam tuşları 19  
Rakamlar 50  
RAM (Rastsal Kullanım Belleği) 42-48, 42, 45  
RAM takımı 4  
RAMTOP 48  
Rastsal etkiler 9  
READ 33  
REM 39  
Reset düğmesi 11, 12, 5  
Resim defteri programı 29  
Resimler, çizimi 30-1  
Yüksek tanımı 26-7  
RND 26, 30  
ROM (Sadece Okuyucu Bellek) 48, 43, 45  
ROM kartuşları 12, 47, 46-7  
RS232 Araparcası 47, 45

Saklama 13, 38-40  
Satırlar 8  
İptal etme 21  
Düzeltilme 21  
Satır programı 51  
SAVE 38-9  
Scroll (kaydırma) 8  
Seçme, işaretler 20  
Ses düğmesi, kasetli teyp 12, 14, 15  
Ses efektleri 36-7  
Sesi yükseltme 37  
Sıralama 8-21  
Simetrik desenler programı 30  
Sinclair BASIC 49-73  
Sistem değişkenleri 48  
STEP 29  
SYMBOL SHIFT 8, 21, 19  
Şekiller, içini doldurma 29, 29

Televizyon alıcısının ayarlanması 6, 6  
Televizyon, bağlantısı 5  
Uygunluğu 4  
Ayarlanması 6, 6  
Tini, müzikte 36  
Tımak işaretleri 23, 51  
Ton düğmesi, kasetli teyp 12, 14, 15  
TRIPLE VIDEO 18  
Tuşlar 18-19, 18, 19  
Kullanımı 20-1, 20-1  
Tuşları kullanarak yazma 8, 9  
TV kodlayıcısı 42  
Voltaj regülatörü 43  
Yan bağlantısı 5, 43, 47  
Yan donanımlar 45, 46-7  
Yanıp sönen halkalar programı 9

Yazılar 45, 47, 45, 47  
Yazılım 12  
kullanıma hazır 12-13, 13  
türleri 12  
uygunluk 12  
yükleme 14-16, 14-16  
Yeni programlar 11  
Yıldız programı 28  
Yıldızlar ve çubuklar programı 11  
Yongalar 42-3  
Yükleme 13, 14-15, 14-16  
Yüksek tanımı çizimler 26, 28-9  
Zıplayan top programı 35  
Z80 mikrobilgisayar 43, 75, 45  
ZX interface 1, 45, 46-7  
ZX Robot programı 27  
ZX 16K RAM 4

İlk yayımı 1984 – Yayınlayan: Dorling Kindersley Ltd, 9 Henrietta Street, London WC2E 8PS – Sinclair Research Ltd, 25 Willis Road, Cambridge ile işbirliği yapılmıştır.

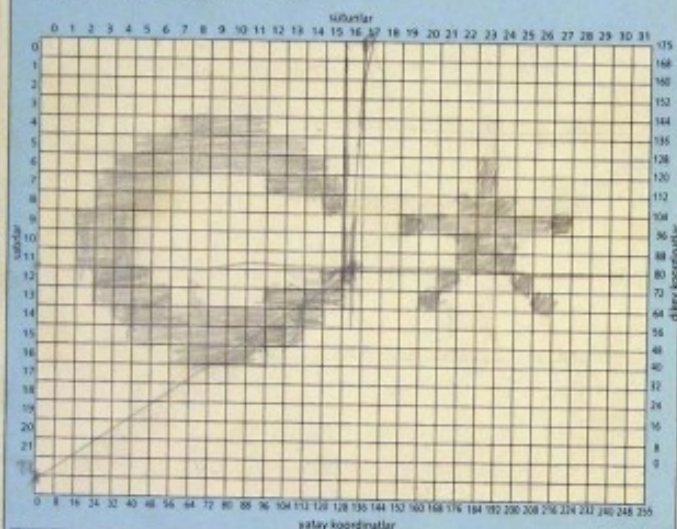
Copyright © 1984 Sinclair Research Ltd ve Dorling Kindersley Ltd, London'a aittir. Resimler için copyright © 1984 Dorling Kindersley Ltd, London'a aittir.  ZX Spectrum +, ZX Spectrum Microdrive ve ZX Interface Sinclair Research Limited'in tescilli markalarıdır.

Bütün hakları mahfuzdur. Copyright sahiplerinin yazılı izni olmadan elektronik, mekanik, fotokopi, kayıt ve başka herhangi bir yöntemle, bu yayının hiç bir kısmı çoğaltılamaz, bilgisayarda saklanamaz veya yayınlanamaz.

Yayımcı: David Burnie  
Sanat yönetmeni: Peter Luff  
Tasarımcı: Debra Lee  
Fotoğrafçı: Trevor Melton  
Ekran fotoğrafları: Vincent Oliver  
Yönetim sorumlusu: Alan Buckingham  
Dizgi: The Letter Box Company (Woking) Ltd., Woking, Surrey, England  
Reproduksiyon: A. Mondadori, Verona  
Basım ve cilt: A. Mondadori, Verona, İtalya

Yüksek tanımlı çizim çizelgesi sadece ana

**Geçmiş**



Bu çizelgenin her sırasının ondalık değerini elde etmek için sayıları toplayın ve karakteri programlamak için POKE USR'yi kullanın.

[illegible]



**SPECTRUM YAZILIMLARI**  
Spectrum bilgisayarları için  
kullanılabilen yazılımların tümü yeni  
ZX Spectrum+ 'unuza da uyar.

DORLING KINDERSLEY LTD  
SINCLAIR RESEARCH LTD'nin  
işbirliği ile