

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 mikrobilgisayar ZX80'i piyasaya sunduktan sonra, bunu giderek gücü ve değeri artan modellerle, ZX81, ZX Spectrum ve QL bilgisayarlarıyla sürdürdük. Bunların tasarımında 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 kolay 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

BU KİTABIN KULLANIMI

ZX Spectrum+'unuzun kılavuzu, aynı ayrı 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 yüklemedeki 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 yarablışı ■ Hareketli resimler ■
Müzik ve çeşitli sesler ■ Programlanızı 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 istediği 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 geçirip, gereksinim duyduğunuz bütün gereçlerin bulunup bulunmadığını kontrol edin, sonra da karşı

Listeniz: Kullanacağınız gereçler tamam mı?

Ambalajı açınca, bulacaksınız:

1. ZX Spectrum+
2. ZX Adaptor/ Transformör – Spectrum için gerekli 9 voltluk doğru akım sağlar.
3. Anten bağlantısı – Spectrum'unuzu televizyona bağlar.
4. Teyp bağlantısı – Spectrum'unuzu teyp bağlar.
5. Garanti belgesi – bunu hemen doldurup, postalamanız gerekir.
6. Kullanım Kılavuzuna Ek Kaset.
7. Bu elkitabı.

Elinizde olması gereken diğer araçlar:

1. Bir televizyon.
2. Bir kasetsel teyp.
3. Bir fiş.

ZX Adaptor



Hazırlıklara ilişkin sorular ve yanıtlar

Televizyonun renkli olması gerekir mi?
Hayır. Ancak siyah beyaz televizyonda, Spectrum'un sağlayacağı renkleri 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 durumda karşılaşılabirsiniz. Bu konuda bir kuşku varsa, televizyon satıcınıza danışın.

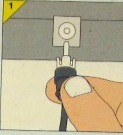
sayfadaki açıklamaları izleyerek, bağlantıları yapın, bilgisayarınızı da fişe takın. Bu aşamada teyp gereksiniminiz yok.

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

Bilgisayarla işiniz bitince, fişi prizden çekin.

ZX Spectrum + 'unuzu cereyana bağlanması

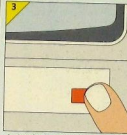
Önce adaptörden çıkan, ucu çiplik kabloyu bir fişe bağlayın. Spectrum'unuzun topraklanmasına gerek yoktur. Bundan sonra, yandaki resimleri izleyerek, Spectrum'u elektrige ve TV alıcısına bağlayın. Bütün bağlantıları tamamladıktan sonra, nasıl yapı yapacağınızı öğrenmek için sayfayı çevirin.



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



Adaptörden çıkan diğer kablounun ucundaki normal fişi elektrik prizine takın.



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



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



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

Spectrum fişleri ve bağlantıları

Cereyan girişi
ZX Adaptor/ Transformöründen gelen 9 voltluk akımın kablosu buraya bağlanır.

Yan bağlantı
Microdrive, yazıcı ve modem gibi çok çeşitli donanımlar buraya bağlanır.



MIC girişi
Programları kasete kaydetmek için kasetsel teyplerin mikrofoni giriş buraya bağlanır.

EAR girişi
Kasete kaydedilmiş programların Spectrum'a yüklenmesi için kasetsel teypin kulaklık girişi buraya bağlanır.



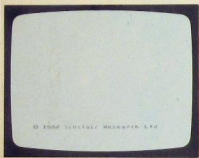
TV girişi
Spectrum'un görüntülerinin ekrana yansıtılabilmesi için TV alıcısının anten giriş buraya bağlanır.



TELEVIZYONUNUZUN 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'unuzun 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 kelimesi 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 tuş buna ayarın.
Böylece Spectrum'u
çalıştırmak
istediğinizde her
seferinde ayar yapma
zorunda kalmazsınız.



Elektronik ayar

Bu sistemde, alıcı
kendini istediğiniz
kanala ayarlar. Elle
ayarlanamayan,
senteseze kanal seçimli
TV alıcısı bu
bilgisayarın kullanımı
için elverişli
olmayabilir.

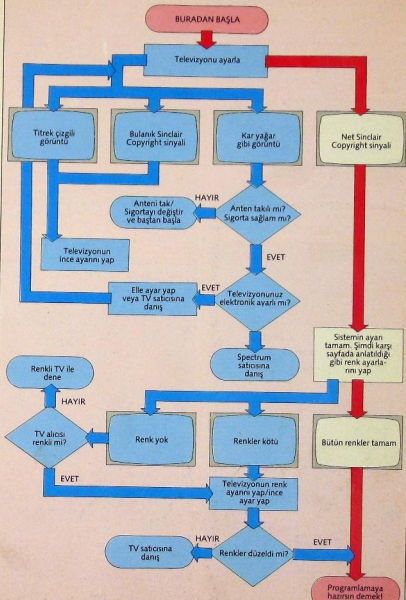


BORDER 3



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

Hazırlamada karşılaşılan sorunlar



ZX SPECTRUM + NELER YAPABİLİR?

Önce, dene

Artık Spectrum'unuzun bağlantıları tamam, TV alıcınız ayarlanmış olduğuna göre, birkaç tuşa basabilirsiniz. Ekranda bazı kelimelerin, harflerin, belki de bazı rakamların belirdiğini göreceksiniz.

Ancak, Spectrum'u programlamak bilmediğinizden, büyük bir olasılıkla, bilgisayar size bir karşılık vermeyecektir. Ama telaşlanmayın — hangi tuza basarsanız basın, bir arzuya yol açmazsınız.

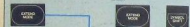
Şimdi, bilgisayarın sol tarafındaki RESET düğmesine basın. Artık Spectrum'unuzda çalışmaya hazırsınız. Şu ilk dört sayfada, size Spectrum'un TV ekranında neler yapabileceğinden bazı örnekler göstereceğiz.

Tuşlar

Bir sözcük, harf ya da rakam yazacağınız zaman, tuşun durumuna dikkat edin. Sonra burada gösterdiğimiz talimatlara uyarak, tuşun durumunu seçin.

Üst komut önce EXTEND MODE'a sonra tuşa bas

Alt komut ya da işaret önce EXTEND MODE'a bas, sonra SYMBOL SHIFT'te aynı anda tuşa bas



BIN
BRIGHT
BORDER
B*

Üst komut (çikintili bölüm) Tuşa bas

Alt komut ya da işaret (çikintili bölüm) Tuşa bas. Büyük harf için CAPS SHIFT'te aynı anda bas.

Harf ya da rakam (çikintili bölüm) Tuşa bas. Büyük harf için CAPS SHIFT'te aynı anda bas.

Alt komut ya da işaret (çikintili bölüm) Tuşa bas. Büyük harf için CAPS SHIFT'te aynı anda bas.

Tuşlar nasıl kullanacağınızın ayrıntılıları 20-21. sayfalarda bulabilirsiniz.

Sıra, Spectrum'un programlanmasında

Spectrum'unuz bir çok şey yapabilir. Ama bunları yapması için, ona bilgisayar programı deneni bir dizi talimat vermeniz gerekir.

Bu sayfalarda gördükleriniz, Spectrum'un yeteneklerini ortaya koyan bir dizi kısa programlardır. Yapmanız gereken, bu program talimatlarını eksiksiz vermeniz. Ekran resimlerini, karşınıza ne çıkacağını gösteriyor. Ama karşı sayfadaki "Program nasıl değiştirilir?" başlıklı kutuyu okursanız, bu programlarla kendiniz de deneyler yapabilirsiniz.

Program yazma ve işleme koyma

Her bir dizi talimat, sıralama deneni bir liste halinde gösterilir. Gördüğünüz gibi program sıralamaları, her bir 10, 20 gibi bir rakama başlayan bir dizi bölümden oluşur.

Programda, her bölüm, (bunlar ekranda iki satır (tuşa bile) satır diye adlandırılır ve bunlar bilgisayar için bir ya da daha çok talimat içerir.

ADLAR

```
10 BORDER 1: INK RND#7
20 PAPER RND#9
40 GO TO 10
```



ZX Spectrum+ adı bütün ekranda çok çeşitli renklerde görünür. Sonra bilgisayar durur ve ekranın altında bir mesa görünür. SCROLL? Görüntünün yukarıya kaymasını istiyorsanız, N, SPACE, BREAK ya da STOP dışında herhangi bir tuşa basmak yeter. Eğer kaymayı durdurur, sırasıyla BREAK, R(RUN) ve ENTER'e basarsanız, bu defa adlarını renklerin değiştirildiğini görürsünüz.

Dene

Satır 30'daki, "ZX Spectrum+" yerine btrak (*) içinde kendi adınızı yazın — örneğin

```
30 PRINT "Ahmet";
```

Noktalı virgüli (;) unutmayın. Bütün ekranın adınızla kaplandığını göreceksiniz.

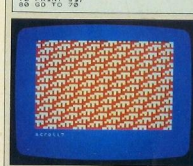
Programın her satırında, gördüğünüz gibi iki ya da daha fazla harftan oluşan kelimeler ya da kısaltmalar vardır — PRINT, LET, RND, PI, PAPER, GOTO gibi. Bunlara komut diyorum ve bu komutları harf harf tuşa basarak veremezsiniz. Bunun yerine, klavyede bu komutların bulunduğu tuşları bulmanız (örneğin PRINT komutu P tuşundadır), sonra da "Tuşlar" kutusundaki talimatlara uymanız gerekir.

Yazdığınız satır, ekranın en altında görünür. Program satırının sonuna geldiğinizde, ENTER tuşuna basmalısınız. Bu defa o satır ekranın üst tarafında görünür. Bundan sonra da her satır aynı yöntemle yazıp, bilgisayara verirsiniz. Yanlış bir tuşa basacak olursanız, gelecek sayfadaki "Hata nasıl düzeltilir?" kutusuna bakın.

Bütün satırlar tamamlandıca, R tuşuna basın. Ekranda RUN sözcüğü görünecek. Şimdi ENTER'e basın ve Spectrum'unuzun programınızın doğrultusunda harekete geçişini izleyin.

DESELER

```
10 LET B=0
20 FOR X=1 TO 7
30 LET B=B+CHR$(RND*16+128)
40 NEXT X
50 INK RND#7
60 BORDER BND#7
70 PRINT B;
80 GO TO 20
```



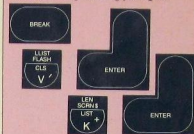
Bu programı işleme koyunca renkli bir geometrik desen bütün ekranı kaplar. Ekran dönlünce, gene SCROLL? sorusu ile görüntü sabitleşir. Ekranın devamını görebilmek için (N, SPACE, BREAK ya da STOP dışında) herhangi bir tuşa basın. Değişik renklerden oluşan bir desen görmek isterseniz, SCROLL? sorusu gelince N'ye, ardından sırasıyla BREAK, R(RUN), ENTER tuşlarına basın.

Dene

Farklı bir desen elde etmek için satır 20'de 7'nin yerine başka bir rakam — örneğin 8'i koyun.

Program Nasıl Değiştirilir?

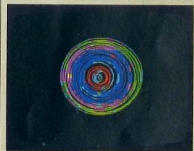
Ya program sona erene kadar bekleyin ya da BREAK'e basıp durdurun. Ardından sırasıyla V(CLS), ENTER, K(LIST), ENTER tuşlarına basın. Ekranda program sıralaması (satırlar) belircektir. Hangi satırı değiştireceğinizi



satırın. Yepyeni bir satır yazın, elbette başında satır numarası da olsun. ENTER'e basın. Yeni satır ekranın yeni alacak. R(RUN) ve ENTER'e basın, yeni program işleme girer.

YANIP SÖNEN HALKALAR

```
10 BORDER 0: PAPER 0: CLS
20 CIRCLE 200 RND#60, FLASH RND
:120 RND#50 RND#45, RND#60
30 BEEP 0.1: RND#60
40 IF RND#2 < 9 THEN GO TO 50
50 GO TO 80
60 FOR Y=2 TO 4
70 FOR X=0 TO 6
80 BORDER RND#5
90 BEEP .05: X=Y
100 NEXT X
110 NEXT Y
120 RUN
```



Ekranda bir dizi içiçe geçmiş, değişik renkli, yanıp sönen halkalar belirir. Anide çevreye aydınlanır, bilgisayar tiz bir ses çıkarır, yepyeni bir dizi daire belirir.

Dene

(K tuşunu kullanarak) program sıralamasından önce PAPER 7 yazın ve ENTER'e basın. Satır 20'yi yeniden yazın ama FLASH RND komutlarını koymayın. Halkaların yanıp sönmeleri durur.

ÇILGIN MOZAK

```

5 BORDER 0 CLS
10 LET N=10
20 FOR I=1 TO 10
30 FOR J=1 TO 10
40 INT RND*2
50 FOR K=1 TO 20
60 PRINT AT V,N:CHR$ 143
70 LET N=N+4
80 LET V=V+4
90 IF N>101 THEN LET N=0
100 IF V>10 THEN LET V=0
110 IF V=21 THEN LET V=0
120 NEXT J
130 GO TO 20

```



Renkli bir kare ekranda ileri geri hareket ederek, renkli desenler oluşturur. Program her başlatışınızda başka deseni biçimlendirir.

Dene

Satır 50'de 143'ün 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 sonra, ekranda son satır olarak OK yazısı belirir. Bu, bütün programın bittiği anlamına gelir. Yeniden başlamak için R(RUN) ve ENTER'e basmak yeterlidir.

Diğer programlar ise, ya ÇILGIN MOZAK gibi süpür güder ya da DOĞAN GÜNEŞ gibi birer 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 R(RUN) ve ENTER tuşlarına basmak yeter.

Hatalar nasıl düzeltilir?

Yanlış bir tuşa basarsanız ya da SHIFT veya EXTEND MODE tuşlarına düzgün basamamışsanız, engellenmeyin. DELETE tuşuna basduğunuzda, son komut işaret, harf ya da rakam silinir. Daha öncelikle de silmek isterseniz DELETE'e basmaya devam edin.



Eğer bir satırda hata yapmış ve ENTER tuşuna da basmışsanız, ekranda yanlış söylen bir soru işareti belirir. Bu, hatayı 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 tamamını yeniden yazıp, sırasıyla ENTER, R(RUN) ve ENTER'e basın. Artık programın çalışması gerekir.

ÇOKGEN

```

5 BORDER 1: PAPER 6: CLS
10 INPUT A
20 FOR I=20 TO 20 STEP 2
30 LET X=100: LET Y=0
40 LET A1=X*A
50 PLOT A1,Y
60 FOR J=30 TO 361 STEP 360/3
70 LET X=X+Y*0.05 (4+PI/180)
80 LET Y=Y+X*0.05 (4-PI/180)
90 LET X=X-Y*Y
100 LET Y=Y+X*Y
110 NEXT J
120 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 oluşur.

YILDIZLAR VE ÇUBUKLAR

```

10 INK 2
20 PAPER 2
30 CLS
40 FOR X=20 TO 140 STEP 20
50 FOR Y=10 TO 110 STEP 10
60 NEXT Y
70 NEXT X
80 NEXT 2
90 PLOT 10,20 DRAW @,131
100 PLOT 32,20 DRAW @,131
110 PAPER 1
120 INK 1
130 FOR X=2 TO 8 STEP 2
140 PRINT AT X,2: "*****"
150 PRINT AT X+1,2: "*****"
160 NEXT X
170 PRINT AT X,2: "*****"

```



Ekranda Amerika Birleşik Devletleri bayrağı belirir.

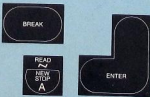
Dene

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

Yeni bir program nasıl başlatılır?

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

Bir programı bilgisayardan belleğinden silmek için iki yol vardır. Biri 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 PND+6
20 INK PND+7
30 PAPER PND+6
40 CLS
50 FOR I=10 TO 174 STEP 2
60 FOR J=10 TO 174 STEP 2
70 PLOT I,J
80 DRAW @,127
90 NEXT J
100 NEXT I
110 INK *
120 FOR X=127 TO 127 STEP 2
130 PLOT 120,0
135 DRAW @,128
140 BEEP .01/60
150 NEXT X
160 FOR X=174 TO 0 STEP -2
170 PLOT 120,0
180 DRAW @,127
190 BEEP .01/60
200 NEXT X
210 PAPER 200
220 GO TO 10

```



Her birkaç saniyede bir, değişik renklerde doğan güneşi andırın bir görüntü belirir. Eğer ekrandaki görüntü kabulursa bekleyin. Ufuktan yeni bir güneş doğacaktır.

Dene

Satır 210'daki 200'un yerine bir başka rakam yazarsanız, her gündüğünün ekranında kalış süresini de 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 yapacağını öğrenmek için, 38. sayfa için "Programlarınızı nasıl saklayabilirsiniz?" bölümünü okuyun.

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

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

HAZIR YAZILIM NASIL KULLANILIR?

Spectrum'a bir program verirseniz, tuşlara basarak bir dizi elektronik olarak kodlanmış sinyal üretirsiniz. Bu kodlu sinyaller Spectrum'un belleğine girer. Bellek bunları program işleme konucuna bilgisayar kullanıldığında işler. Bu kodlu sinyaller, siz (örneğin NEW ya da RESET tuşuna basarak) bunları silmediğiniz ya da Spectrum'u kapatmadıkça, bellekte kalması devam eder.

Ancak Spectrum'unuz her kullanıma 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 kullanıma 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 bunları yükleyip bir kısmı Spectrum'da kullanabilirsiniz. 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 sesli 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.

Yazılıma ilişkin sorular ve yanıtları

"Yazılım" ne demektir?
Yazılım (İngilizcesi Software), bilgisayarları çalıştırmak için verilen programların genel adı. Donanım ise, (İngilizcesi Hardware) kullanılan araçlar - bilgisayar ve bu işlemler yararlanan diğer gereçlerdir.

Yazılım niye kasetli teyple yapılıyor?
Kasetli teypelerin kullanımı kolaydır, başkaca bir özel aygıt gerekmez. Bu tür yazılımın yüklenmesi için gereken tek şey, ucuz bir kasetli teyptir.

Kayda alınmış bir program nasıl bir şey?
Birini, teypi Spectrum'a bağladıktan sonra. Tiz bir duduk sesi duyacaksınız. Bu bilgisayar gidecek yerde operatöre ulaşan kodlu sinyalleri okuyordur sestir. 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 kartuştaki 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 dakikalara 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 ceyran bağlantılı teypler yeğlenmelidir. Teypin bir ses ayar düğmesi olmazdır, ama ton düğmesi zorunlu değildir. Ayrıca özel bilgisayar kasetli teypleri de vardır. Bunlar, programların daha güvenli yüklenmesini ve saklanması sağlamak üzere özel olarak yapılmıştır.

Bir müzik setinin parçası oluşturulan kasetli teypin bilgisayara bağlanması pek kolay olmayacaktır. Ayrıca bu tür araçların ses çıkışı genellikle Spectrum için yeterince güçlü ses sinyali verememektedir.

Teype 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 teypi açar ya da kaparken içinde kaset bırakmamak iyi olur. Bu, kasete kayıtlı programların korunması açısından yararlıdır.

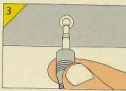
Bağlantıları doğru yapın



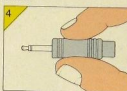
Dört fişten birini Spectrum'un arkasındaki EAR girişine sokun



Obür uçtaki aynı renkli fişi teypin, eğer varsa EAR girişine sokun.



Eğer teypin EAR girişi yoksa, fişi kulaklık girişine, o da yoksa dış operatör çıkışına sokun.



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

Yazılım için ödütlər

■ Spectrum'un teyp kablosunda, bilgisayara teyp arasındaki bağlantıları karıştırmış diye, farklı renkte fişler kullanılmıştır. Teypinizi Spectrum'la kullandığınızda, duzeni deđiřtirmeyin, bir renge EAR, bir renge MIC girişleri için kullanın.

■ Bazı kasetli teyp çevrelerindeki elektrikli araçlardan etkilenirler. Bundan bilgisayar ile teyp arasında geliş giden sinyaller de etkilenir ve bu yüzden programlar düzgun yüklenmeyebilirler. Bu teypin düzgun çalışmaması o zaman teypten televizyon ya da bilgisayarın yanından uzaklaştırın.

EAR ve MIC girişleri

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



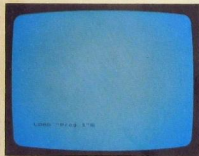
PROGRAM NASIL YÜKLENİR?

Kasetli teypi Spectrum'unuza bağladığınızın 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, eğer ton düğmesi varsa, tize getirilmelidir.

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

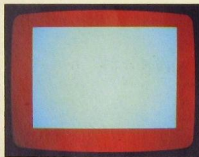


Kasetli teypi çalıştırın. Spectrum'un elektrikle bağlı olup olmadığının bakın. Kasetli teypi takın. Eğer bilgisayarda bir program varsa, bitmesini bekleyin ya da BREAK tuşuna basıp durdurun. Programı Spectrum'un belleğinden silmek için NEW ya da RESET tuşuna da basabilirsiniz, ama bu önemli değil, çünkü zaten yeni bir program yüklenmesi belleği temizler. Önemli olan, yeni bir program yüklenmesinin eskisini bellekten siliceğini hatırlamaktır.

Şimdi sırasıyla aşağıdaki talimatları izleyin. Eğer bir hata olursa, 16. sayfadaki "Yazılım yüklemedeki sorunlar" tablosuna bakın.

4 ENTER'e basın. Ekran silinecektir.

5 Teypi çalıştırın. Ekranın çevresi kırmızı veya mavi olacak ya da kırmızı ve mavi yanıp 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 belirtisidir.

Yazılım yüklemeye öğretir

İşte size yüklemeye zaman 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.

Adı	Adı	Adı
1. Program Adı	2. Program Adı	3. Program Adı
4. Program Adı	5. Program Adı	6. Program Adı
7. Program Adı	8. Program Adı	9. Program Adı
10. Program Adı	11. Program Adı	12. Program Adı

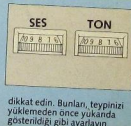
2 Kasetli teypin sayacı varsa, bunu üzerinde birden fazla program bulunan kasetlerde aradığınızı hızla

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

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

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

4 Spectrum'unuzun yüklenmesine yardımcı olacak ses ve ton düzeyine



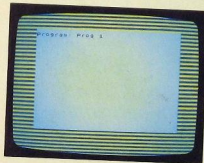
dikkat edin. Bunları, teypinizi yüklemeye önce yukarıda gösterildiği gibi ayarlayın.

7 Ekranda Program sözcüğü, ardından programın adı veya Bytes yazısı, bundan sonra da bir harf ya da rakam görünür. Bu bilgisayarın istenen programı başarıyla 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 uzansa, yüklemeye birkaç dakika sürebilir.



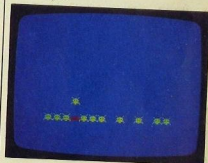
10 Eğer program bölümlere ayrılmışsa, 7, 8 ve 9. adımlar bir veya birkaç kez daha tekrarlanabilir.

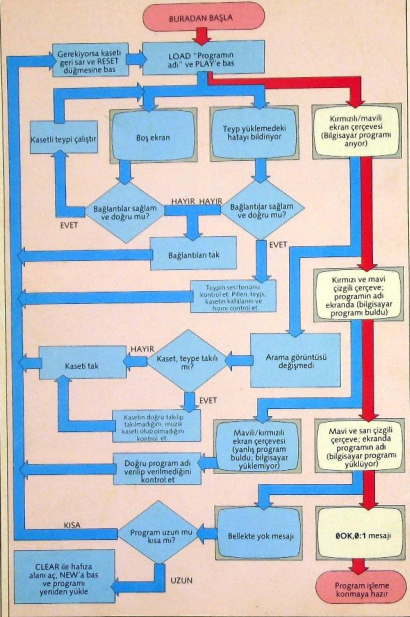
11 Program, yüklenir yüklenmez otomatik olarak başlayabilir. Teypi durdurmayı unutmayın.

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



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





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 - BİLGİSAYA- RINIZIN KOMUTA KONTROL TABLOSU

ZX Spectrum+'un kendine özgü bir dili vardır - BASIC diye bilinen bilgisayar dili. Talimatlarınıza uyması için, Spectrum'ü BASIC dili konuşarak programlamamız gerekir. Bunu, Spectrum'un klavyesini kullanarak yapacaksınız. Ayrıca bu klavye, bilgisayar sizin programınızı işlerken, bilgisayarın kontrol etmenizi 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 bilecektir.

NEW

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

DELETE

Yanlış bir tuşa basmış, bir komut, harf, rakam veya işareti silmek istiyorsanız, bu tuşa basın - bak sayfa 10.

EDIT

Bu tuş, bütün bir programı yeniden yazmak için bunun bir satırını düzeltmesinizi sağlar - bak sayfa 21.

EXTEND MODE

Bu tuşun yardımıyla, tuşların kıçını kısmının üstündeki komutu verebilirsiniz. Bunun ardından SYMBOL SHIFT'i kullanırsanız, kıçını kısmın hemen üstündeki işaret ya da komutu elde edersiniz - bak sayfa 20-21.

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

CAPS LOCK
Eğer sürekli büyük harf yazmaya istiyorsanız, harf tuşuna basmaya başlamadan önce CAPS LOCK'a basarsanız, yazmaya başlarsınız. Büyük harflerle işiniz bitince, tekrar bu tuşa basıp küçük harfe dönelirsiniz.

BEEP
Bu tuş, Spectrum'un ses sensörünü denetleyen komutlarını elde edimesinde kullanılır.

TRUE VIDEO ve INV VIDEO

Bu tuşlar, program sabrınına, normal ya da ters (diği) renklerle elde etmek üzere kontrol kodlarının yerleştirilmesi için kullanılır.

görünmesini sağlayabilirsiniz.

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

Renk tuşları

Bu alt tuş, ekrandaki renkleri kontrol eden komutları verir.

Komutların ve işaretlerin seçimi

Spectrum'un klavyesinde çok kullanılabilecek 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 anlatmıştık. Şimdi klavyeyi biraz daha tanıdıktan sonra, önumuzdeki iki sayfada, bilgisayarın klavyesinde gördüğünüz herhangi bir şeyi nasıl seçebileceğinizi anlatacağız. Bir defa bunu öğrendiniz mi, programlarınızı yazmaya başlayabilirsiniz.

Rakam tuşları

Bunlar, rakamları yazmanın yanısıra, gösterilen renklerin programlara yerleştirilmesi için kontrol kodlarını da verir. (Bak s.31). 8 dışında 4'ten 0'a kadarki tuşları hemen üstünde yer alan bu komutlar ancak ZX Microdrive'larla kullanılabılır.

BREAK

Bu tuş, işleme konan bir programı durdurur. Program bilgisayarın belleğinden silmez.

ENTER

Bu tuşa basınca, bir program satırını Spectrum'un belleğinden çıkarır. Bu tuş, ayrıca sık sık, bir program sırasında bilgisayara bilgi beslemeye de kullanılır.

SYMBOL SHIFT

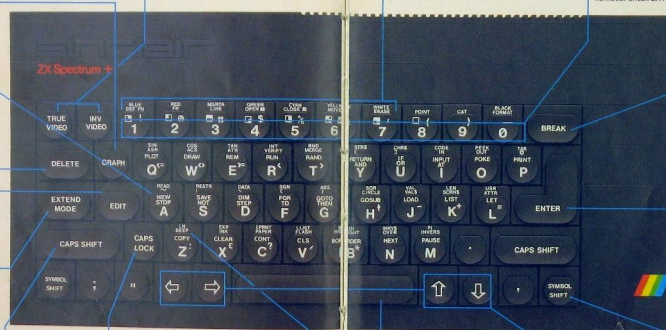
Bu tuşun üstündeki kıçını bölümdeki işareti ya da alt komutu kullanmak isterseniz, bu tuşa SYMBOL SHIFT ile birlikte basmanız gerekir. Eger SYMBOL SHIFT'e, EXTEND MODE'dan sonra basarsanız, o zaman bilgisayar, kıçını bölümün hemen üstündeki komutu uyar - bak sayfa 20-21.

Göstergeç kontrolü

Bu tuşlara basarak göstergecin (İngilizcesi cursor) oklarını yönünde hareket ettirmesi sağlayabilirsiniz. Bu tuşlar, programlarda genellikle şekillerin ekrandaki hareketlerinin denetimini sağlar. Ayrıca program üzerindeki düzenlemelerde de yararlı olur.

Boşluk tuşu

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



TUŞLAR NASIL ÇALIŞIR?

ZX Spectrum'un tuşlarından herhangi biriyi alı kadar değişik komut, harf, rakam ya da işaret yazabilirsiniz. Spectrum'un özelliklerinden birini iyice öğrendiğiniz takdirde klavyedeki herhangi bir harfi ya da komutu seçmek pek o kadar kaşışık 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 yararı şudur: Siz klavyeyi kullanırken, aslında Spectrum, talimat ve bilgileri sırasıyla verebilmeyi için size klavyedeki modları seçmenize yardımcı olur. Bu iki sayfada bu modların ne iş 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

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

neyi, nasıl bulabileceğinizi belirleyin. Daima önce ekrandaki göstergelere bakıp bilgisayarın hangi moda olduğunu saptayın.

Harf tuşu



Komut (K) modu

Sadece tuş BORDER

SYMBO ve tuş *

Genişletilmiş (E) mod

EXTEND sonra sadece tuş B

EXTEND sonra SYMBO ve tuş BRIGHT

Harf (L) modu

Sadece tuş b

CAPS ve tuş B

SYMBO ve tuş *

Büyük harf (C) modu

CAPS sonra sadece tuş B

CAPS sonra SYMBO ve tuş *

Grafik (G) modu

GRAPH sonra A ile U arasındaki tuşların kullanımını belirleyeceği grafik

Komut modu

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

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

Sayı tuşu

MGN TA LINE

Komut (K) modu

Sadece tuş 3

SYMBO ve tuş #

Genişletilmiş (E) mod

EXTEND sonra sadece tuş mor

EXTEND sonra SYMBO ve tuş LINE

Harf (L) modu

Sadece tuş 3

SYMBO ve tuş #

Büyük harf (C) modu

CAPS sonra sadece tuş 3

CAPS sonra SYMBO ve tuş #

Grafik (G) modu

GRAPH sonra sadece tuş

GRAPH sonra CAPS ve tuş

Harf ve büyük harf modları

Bilgisayar komut modundayken, bir komut ya da işaret yazdıktan sonra otomatik olarak göstergesi L'ye çevirir. Artık harf moduna geçmiştir. Herhangi bir harf tuşuna basın, bir küçük harf belircektir. Rakam tuşuna basın, rakam görünüvü. Büyük harf yazmak isterseniz, CAPS SHIFT'e basıp bunu tutarken harfe basın.

Eğer hep büyük harf yazacaksanız, önce CAPS LOCK'a basın. Göstergesi C'ye dönüşür. Spectrum'unuz büyük harf moduna geçti, artık hangi harfe basarsanız büyük harf yazsınız. Rakamlar büyük harf modunda da yazılabilir. Lock'la moduna dönmek için CAPS SHIFT'a bir kez daha basın.

Genişletilmiş mod

Bu modun adı genişletilmiş mod ve bunu EXTEND MODE'a basarak elde edersiniz. Göstergesi E'ye dönüşür. Herhangi bir harf tuşuna bastığınızda, çikintılı bölümün *uzununda* bulunan iki komuttan üsttekini

elde edersiniz. Örneğin B'ye basarsanız BIN çıkar. Altta ki komutu ya da işareti elde etmek için ise SYMBO SHIFT tuşlarından birine basıp tuşun ve harf tuşuna dokununuz. B tuşuna örneğin BRIGHT yazısı çıkar. Yani genişletilmiş moda 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 geçilir. Göstergesi G'ye dönüşür. 1'den 8'e kadar rakamlara basıp bunların üzerindeki grafik karakterlere bakın. Şimdi ise CAPS SHIFT'e basıktan 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 programlarınızı yazmaya başladığınızda, komutlarda ya da program satırlarında düzeltmeler yapmak veya bunları değiştirmek isteyebilirsiniz. Bu oldukça kolaydır.

Hatanın düzeltilmesi

BASIC dilinde hatalı bir satır ya da komut vermeye kalkarsanız, Spectrum bu hatanın önünde bir ? işaretini yazıp söndürmeye başlar.

Hatayı düzeltmek için, göstergesi hatanın sağına götürebilmek üzere, göstergesi 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 SYMBO SHIFT'e basmayı unuttunuz. Bunun yerine

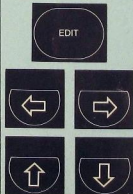
PRINT 7*8

çkar. Spectrum bu komuta uyamaz, bu nedenle siz ENTER'e basarak basmaz hata yaptığınız b'nin önünde bir soru işareti yazıp sönmeye başlar. Bütün

yaşacağınız, göstergesi hatanın sağına götürüp, ardından DELETE'e basıp b'y'i yocketmek. Bundan sonra SYMBO SHIFT ve B'ye basarak * elde edin ve ENTER'e basarak bilgisayarın doğru talimatı yazmasını sağlayın. Göstergesi yeniden satırın sonuna kadar götürmeniz gerekmez. Spectrum talimatı uygular ve sonucu verir.

Bir program satırını nasıl düzeltilir?

Bir program yazarken, sıralama



denilen, birbirini izleyen numaralarla bir dizi satır ardarda koyarsınız. Eğer, programı yazdıktan sonra, bunu K (LIST) ve ENTER tuşlarına basarak "sıralarsanız", programın satırlarından birinin karşısına > işareti göreceksiniz. Eğer görüyorsanız, göstergesi aşağı yukarı kontrol eden tuşlara basın. EDIT tuşuna basarsanız, bu işareti karşınızdaki satırın eşi ekranın altında bir daha çikar. Buna daha önce anlattığımız gibi göstergesi kontrol ve DELETE tuşlarını kullanarak "eğitirebilirsiniz. ENTER'e basınca da o satır eski yerine döner. Başka bir satır daha düzeltmek istiyorsanız, > işareti istediğiniz satırın karşısına gelecek şekilde yukarı aşağı oynatın ve EDIT'e basın. Eğer bu uzun süreceksiz, sırasıyla LIST, istenen satırın numarası ve EDIT'e basın. Her iki halde de istediğiniz satır ekranın altına belircektir. Bunda düzeltme yapabilirsiniz. Programın bütün bu satırını silmek istiyorsanız, sadece satırın numarasını yazıp ENTER'e basın. Eğer bir hata içeren bir programı işleme koyduysanız, ekranda bunun haberini 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 yapilmasını istediğinizi gösteren +, -, = gibi bazı işaretlerdir.

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

PRINT 6+2

Bu bir komuttur. ENTER'e basınca komut kaybolur, ekranda cevap, 6 görünür.

Spectrum'unuz, hesaplamada, *aritmetik işlem* 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 sırada bir hesap makinasının yapmadığı birçok şey yapabilir. Herşeyden önce, hem hesabı hem sonucunu aynı anda gösterir. Şu komutu verir:

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 işaret arasındaiki 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	iki rakamı toplar	8+2=10
-	J	Bir rakam diğerinden çıkarır	8-2=6
*	B	iki rakamı çarpır	8*2=16
/	V	Bir rakamı diğerine böler	8/2=4
↑	H	Birinci rakam ikincisinin üstüne çıkarır	8 ↑ 2=64

İlk programınız

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

10 PRINT 6+2

Bu defa talimatı 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 talimatla artık komut değil *cümle* denir ve bir programın numaralanmış bir satırını oluşturur.

Programdaki cümleler daima satır sırasına göre işleme konur ve bunlar gerektiğinde aralansa satır eklenebilis diye daima onlu sayılarla başlar.

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

SAYI TABLOSU

```
10 LET N=1
20 PRINT N;" "
30 LET N=N+1
40 GO TO 20
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
```

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

Bu program bir *değişgen* kullanıyor. Bu olayda değişgen, n. Değişgen olarak

herhangi bir harf ya da sözcük de kullanılabilir. Burada n sayısını geliyor. Değişgen program boyunca değişen bir değer verir. 10. satırda LET komutu kullanılarak bu değer 1 olarak verildi. 20. satırda bir boşluk bırakılmış. 30. satırda gene LET kullanılmış, ama bu kez değeri 1 artırılmış. 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. satırını 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 bilgisayarına bir program sırasında bilgi alma talimatı verir.

Bir çarpım cetveli programlama

Eski programi temizlemek için RESET düğmesine basın ve yeni programı verir. Bu program Spectrum'a çarpım yaptırır. Herhangi bir rakama bastığınızda, ekranda bu rakam için çarpım cetveli çıkar. 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 cetvelinin programı

ÇARPIM CETVELİ

```
30 LET X=3
40 INPUT N
45 PRINT N;" "
50 GO TO 30
```

```
3
6
9
12
15
18
21
24
27
30
33
36
39
42
45
48
51
54
57
60
63
66
69
72
75
78
81
84
87
90
93
96
99
102
105
108
111
114
117
120
123
126
129
132
135
138
141
144
147
150
153
156
159
162
165
168
171
174
177
180
183
186
189
192
195
198
201
204
207
210
213
216
219
222
225
228
231
234
237
240
243
246
249
252
255
258
261
264
267
270
273
276
279
282
285
288
291
294
297
300
303
306
309
312
315
318
321
324
327
330
333
336
339
342
345
348
351
354
357
360
363
366
369
372
375
378
381
384
387
390
393
396
399
402
405
408
411
414
417
420
423
426
429
432
435
438
441
444
447
450
453
456
459
462
465
468
471
474
477
480
483
486
489
492
495
498
501
504
507
510
513
516
519
522
525
528
531
534
537
540
543
546
549
552
555
558
561
564
567
570
573
576
579
582
585
588
591
594
597
600
603
606
609
612
615
618
621
624
627
630
633
636
639
642
645
648
651
654
657
660
663
666
669
672
675
678
681
684
687
690
693
696
699
702
705
708
711
714
717
720
723
726
729
732
735
738
741
744
747
750
753
756
759
762
765
768
771
774
777
780
783
786
789
792
795
798
801
804
807
810
813
816
819
822
825
828
831
834
837
840
843
846
849
852
855
858
861
864
867
870
873
876
879
882
885
888
891
894
897
900
903
906
909
912
915
918
921
924
927
930
933
936
939
942
945
948
951
954
957
960
963
966
969
972
975
978
981
984
987
990
993
996
999
```



Parantezlerin kullanımı

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

PRINT 6+2/4
PRINT (6+2)/4

İlkinin sonucu 6.5, ikincisinin ise 2. Bu iki farklı sonucu nedeni, bilgisayarın hesaplarda kullandığı bir öncelikli sistemine sahip olması. Önce ↑, sonra veya/, em son olarak da + veya - uygular. Ama daima parantez içindeki işleme öncelik verir. Bu nedenle, yukandaki ilk komutta, önce 2'yi 4'e böler, sonra sonuçla (0.5) 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 sağlamasını etkileyecek biçimde bilgisayara talimat vermeye de yarar.

- **Noktalı virgül** PRINT ile birlikte kullanılınc, bilgisayara iki yanındaki öğelerin ekranda yan yana görünmesini talimatı verir.
- **İki nokta** Program satırındaki cümlelerden birinin sona erdiğini, bir diğerinin başlayacağını haber verir.

¶ **Tırnak işareti** Tırnak işaretleri arasında yazılan rakam ya da değişgen olarak değil, bir ifade olarak değerlendirilir. Dizi tırnak işaretleri ile başlayıp biter.

¶ **Virgül** PRINT ile birlikte kullanılınc, bilgisayara izleyen öğeyi ya bir satırın ortasında, ya da yeni satırın başında ekrana yansıtmasını bildirir. Bunu binler ya da milyonlar hanesini ayırmak için kullanmayın.

- **Nokta** Ya bir ondalık işareti ya da noktadır.

RENKLER VE KULLANIMI

ZX Spectrum+ ile sekiz değişik renk elde edebilirsiniz ve her rengin değişik bir renk kodu vardır. Renkleri çerçeve rengi, mürekkep rengi ya da kağıt rengi olarak değişik biçimlerde kullanabilirsiniz.

ZX Spectrum+ renk kodları

Aşağıdaki çizelge, Spectrum'un kullandığı renkleri ve kodları gösteriyor. Bu kodları ezberlemeniz gerekmez; rengi sağağaç tuşunun üzerindeki rengin adı yazıdır. (Bu adlar komut değildir).

Rakam	Renk
0	Siyah
1	Mavi
2	Kırmızı
3	Acık
4	Yeşil
5	Açık mavi
6	Sarı
7	Bezay

Ekranı göreceğiniz renk tonunu ise televizyon alıcınızın renk, kontrast ve ışık ayarları belirler. Unutmayın bunun için renkli televizyon gerekir.

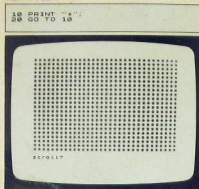
Spectrum'da renk kullanmanın üç yöntemi

Üç tür renk vardır: Çerçeve rengi, orta alanın çevresindeki renktir. Mürekkep rengi, karakterin (harflar, rakamlar, işaretler ve grafik şekillerin) ve çizgilerle noktaların rengini belirler. Kağıt rengi, ya bütün alanın ya da o harfin etrafındaki küçük karenin tonunun rengidir.

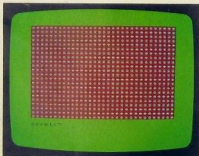
Spectrum'u ilk çalıştırdığınızda, önceden belirlenmiş renkleri kullanır. Mürekkep rengi siyah, çerçeve ve kağıt rengi beyazdır. Klavyedeki tuşlarla vereceğiniz komutla bu renkleri derhal değiştirirsiniz. Bunu zaten daha önce 6 ve 7. sayfalarda görmüştünüz. BORDER komutunu kullanarak televizyonunuzun ve Spectrum'unuzun renkleri için doğru bağlantı bağlanmadığını denetlemiştiniz. Şimdi RESET'e basıp,

aşağıdaki basit programı verip, işleme koyun.

RENK TESTİ



Siyah ve beyaz yıldızlardan oluşan bir desen belirlecektir. BREAK'e basıp, programı yeniden işleme koyun. Aşağıda iki ekranda üsttekinde BORDER 4, PAPER 2 ve INK 7, alttakinde ise BORDER 3, PAPER 5 ve INK 1 komutu verilmiştir.



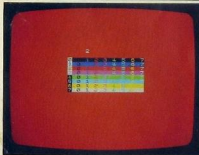
Renk kullanarak program yazma

Bir programda BORDER, PAPER ve INK komutlarını kullanarak metinler, tablolar ve desenler yapabilirsiniz ve ekranda rengarenk resimler belirlir. Bir program satırında BORDER kullanılırsa, Spectrum bu satıra gelince çerçevenin rengi değişir. Bir satırdaki İNK komutu ardından gelen harf ya da satırlar o rengi alır. PAPER komutu kendi başına verilirse, yalnızca karakterlerin (buna nota ve çizgiler de dahildir) çevrelerindeki rengi değiştirir. Bütün fonu belli bir renk yapmak istiyorsanız, PAPER'den sonra CLS'ye basmak gerekir.

PRINT'ten sonra da INK ve PAPER kullanabilirsiniz. Bu durumda, sadece PRINT'ten sonra kullanılmayan karakterler INK ve PAPER'in renklerini alır. Şimdi vereceğimiz program, çerçeve, mürekkep ve kağıt renklerinin tüm kullanımlarını, ayrıca INK ve PAPER'in PRINT'ten sonra nasıl kullanılacağını gösteriyor.

RENK BİLEŞİMLERİ

```
10 FOR I=0 TO 9
20 BORDER 0: PAPER 0: CLS
30 PRINT AT 0:0: INK 0:0
40 FOR P=0 TO 7: INK P:0:0
50 PRINT AT P+0:0: INK P: PAPER
60 BEEP 0:0:144-20*P
70 FOR I=0 TO 9
80 PRINT INK I: I: PAPER P: P: I: I
90 BEEP 0:0:1:145
100 NEXT P
110 NEXT I
120 NEXT E
```



Bu programı işleme koyduğunuzda, çerçeve, mürekkep ve kağıt renkleriyle yapılabilecek bütün bileşimleri görebilirsiniz. Programda üç değişken var: b çerçeve, i mürekkep, p kağıt rakamının değişkeni. BEEP sesi sağlıyor. FOR ve NEXT'le başlayan satırlar, sırasıyla 0 ile 7 arasındaki bütün renk rakamlarının kullanılacağı döngünün başlangıcı ve bitişini belirliyor. Döngüde FOR ve NEXT'in kullanımı hakkında 27. sayfada daha ayrıntılı bilgi bulabilirsiniz. Gördüğünüz gibi, INK ve PAPER'in değeri 9 olabilir. Bu, mürekkep ya

da kağıt renginin, herhangi bir fon ya da karakterin üzerinde görülebileceği, siyah ya da beyaz olmasını sağlar.

Renkli çubuklu grafiklerin programlanması

Bu program, Spectrum'un renklerinin kullanılmasını sağlayan renkli çubuklu grafik yapılmasını sağlar. 28 günlük isiyi, gösteren rakam yazılı çubuklar olarak gösterir. 60. satırda tırnak işaretlerinin arasında iki boşluk bırakın.

ÇUBUKLU ÇİZELGE

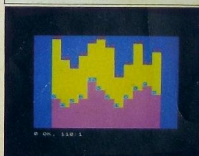
```
10 BORDER 0: PAPER 1: CLS
20 LET C=4
30 FOR X=1 TO 12
40 READ C
50 FOR I=0 TO 23: STEP -1
60 PRINT PAPER C:AT I,C: " "
70 PRINT INK 2:AT 20-I,C: " "
80 LET C=C+2
90 NEXT X
100 NEXT C
110 DATA 20,15,13,10,10,20,10,1
1,1,9,10,14,1,9
```



Şimdi aşağıdaki satırları ekleyin ve 110. satırı gösterdiği gibi yeniden yazın. Artık çizelge iki renkli olmuştur. READ ve DATA hakkında daha fazla bilgi için 33. sayfaya bakın.

ÇİFT ÇUBUKLU ÇİZELGE

```
85 READ I
86 FOR I=0 TO 23: STEP -1
87 PRINT PAPER 3:AT I,C: " "
88 NEXT I
89 PRINT INK 1: PAPER 5:AT 20-
115 DATA 20,6,15,1,13,6,10,6,10
10,20,6,10,6,11,1,1,1,6,1,6,1,9,1,7,7
```



BASİT BAZI GRAFİKLER

ZX Spectrum+'unumuzla alçak ya da yüksek tanımlı bazı çizimler yapabiliriz. 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 sabir rakamı verilir. Bu, aşağıda kaç satır kaldığını gösterir. En üstteki satırın rakamı 0, en alttakini ise 21'dir. Bunu yazayız sütun rakamı izler. Sol baştaki sütunun rakamı 0, en sağdakinin ise 31'dir. (80. sayfada alçak tanımlı çizim çizelgesinin tümünü görebilirsiniz). Şimdi göreceğiniz program, bu pozisyonların renklerle nasıl doldurulacağını gösteriyor. R tuşundaki RND komutu rastsal bir mürekkep rengi seçer.

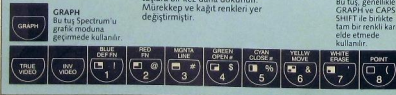
Grafik karakterlerin seçimi

ZX Spectrum+'unuzun klavyesinde bazı grafiklerin programlanmasını kolaylaştıran bir dizi grafik karakter vardır. Bunları 1-8 arasındaki tuşlarla 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 belircektir. Bu karakterlerin tuşların üzerinde beyaz olan bölümü ekrana mürekkep renginde, siyah bölümü de kağıt renginde yansır. Bu defa CAPS SHIFT'e basarak, bu tuşlara bir kez daha dokunur. Mürekkep ve kağıt renkleri yer değiştirmiştir.

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

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



RASTSAL KARELER



Bütün ekranı 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 noktalı virgül tuşlarına basın. Örneğin, şu komutta

PRINT AT 11,16; "*" "

11. satır, 16. sütunda bir yıldız çıkacaktır.

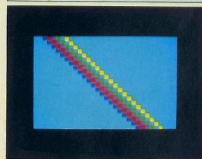
Gökkuşuğu deseni nasıl çizilir?

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

Her defasında sadece bir döngü kullanılmakla sınırlı değilizdir. Bir döngüyü, bir diğeri için yerleştirebiliriz ve bu çoğu kez yararlı sonuçlar verir. Aşağıda (bir diğeri için "yuvalanmış") iki FOR NEXT döngüsünde INK ve AT'ın sağlayacağı pozisyonların ve renklerin değişebileceğini göreceksiniz. Bu döngülerin nasıl hazırlanacağını bu sayfanın sonundaki karede bulabilirsiniz.

GÖKKUŞAĞI

```
5 BORDER 0: PAPER 0: CLR
10 LET X=1
20 FOR L=0 TO 21
30 FOR C=0 TO 21
40 PRINT INK C:AT L,C: *
50 NEXT C
60 LET X=X+1
70 NEXT L
```



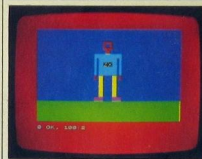
Resimlerin programlanması

Alçak tanımlı çizimlerde, grafik karakterleri pozisyonları ve renkleri ile oynayarak resimleri "boyayabiliriz". 80. sayfadaki basit çizim çizelgesinden yararlanarak resimlerini programlayabiliriz. Bundan sonra, karşı sayfada gösterildiği gibi grafik karakterleri seçerek, resmi oluşturacak program satırlarını birer birer verebiliriz.

Yandaki program nasıl bir sonuç elde edeceğimizi gösterecek. Buradaki bütün şekiller rakam tuşlarındadır. Programı işleme koymak için, tümünü verene kadar bekleyebileceğiniz gibi, her satır verdiğiniz işleme koyacak olursanız, robotun çeşitli konumlarının nasıl oluştuğunu görebilirsiniz. (Bölmeyin eğer yanlış grafik karakteri verecek olursanız, bunları da tıpkı yanlış rakam veya harfleri düzelttiğiniz gibi düzeltebilirsiniz.)

ZX ROBOT

```
5 BORDER 0: PAPER 1: CLR
10 PRINT INK 0: AT 11,13: *
20 PRINT INK 0: AT 11,13: *
30 PRINT INK 0: AT 11,13: *
40 FOR L=7 TO 10: PRINT INK 0: AT L,14: *
50 PRINT INK 0: PAPER 0: AT 0,1
60 PRINT INK 2:AT 11,13: *
70 FOR L=11 TO 15: PRINT INK 0: AT L,14: *
80 PRINT INK 3:AT 15,13: *
90 LET X=17 TO 21: FOR C=0 TO 21: PRINT INK 4:AT L,C: *
100 NEXT C: NEXT L
```



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

FOR NEXT döngüsünün kullanımı

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

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

Burada değişgen C, bununda başlangıç değeri, bilgisayarın bu programı tekrarlamasını sağlayacak satırları içerecektir. Bunlarda da c değişgeni kullanılabilir. FOR NEXT döngüsü, daima NEXT komutu ve değişgenle sona erer. Örnek

```
50 NEXT C
```

Program işleme konunca, FOR'dan NEXT'e kadarlı bütün kendisini defalarca tekrarlar. Değişgen TO'dan önceki değeriyle başlar ve TO'dan sonraki limite kadar her seferinde 1 artar. Bu durumda kendisi 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 yapılmaktadır.

RESİM DEFTERİ OLAN EKRAN

ZX Spectrum + ile yapılabilecek çizimler, iri parçalarından oluşan alçak tanımlı çizimlerle sınırlı değildir. Karmaşık yetenekleri neredenyele 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 noktalardan 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 cells*' in, resim hücrelerinin kısaltılmışıdır. Tıpkı 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 piksel 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

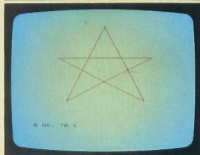
Yüksek tanımlı çizimler için üç komut yeterlidir: PLOT, DRAW ve CIRCLE. PLOT' u virgülle ayrılmış yatay ve dikey koordinatlar izler ve komut buraya bir piksel kondurur. DRAW' u da virgülle ayrılmış iki rakam izler ama bunlar koordinat *değildir*. Bunlar piksel olarak, iki noktanın arasındaki yatay, dikey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

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

Kullanılmıyorsa, hangisi en son kullanılmıyorsa, PLOT' daki son nokta ya da DRAW ile ulaşılan son noktadır. Bundan sonraki DRAW komutu yeni noktaya bir çizgi çeker. Eger çizgi sola ya da aşağı gidecekse, yatay veya dikey mesafeler eksi (-) değerler verir. Şu programı deneyin:

YILDIZ

```
10 INK=2
20 PLOT 50,174
30 DRAW 50,140
40 DRAW 150,50
50 DRAW 140,0
60 DRAW 100,50
70 DRAW 70,140
```

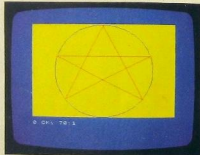


PLOT başlangıç noktasını ekranın tepesine götürür, ardından DRAW komutunun beş kırmızı çizgi çizerdir.

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 komutunu 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 artırarak FOR NEXT döngüsü ile elde edebilirsiniz.

DOLU ÜÇGEN

```
10 BORDER 1: PAPER 0: INK 0: C
20 FOR X=100 TO 100
30 PLOT 128,100
40 DRAW X,-100
50 NEXT X
```



Eğer çizgileri biraz aralıksız 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 çizildiğinde x'i 1 ilerletecek yerde 4 sıratmasıdır.



Ekrandaki resim defteriniz

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

Program size mürekkep rengi soracak olan INPUT komutu ile başlıyır. Ardından (bu defa bir dikey belirleme üzere 5 işaretini kullanarak) INPUT komutu ile dört tuşa - u, d, l ve r - bastığınız bilgisayarın kısa bir çizgi çizmesini sağlar.

RESİM DEFTERİ VE ÖRNEK

```
10 INPUT "INK=";I
20 BORDER 3: PAPER 0: INK I: C
30 PLOT 55,25
40 LET X=0
50 INPUT "X=";X
60 IF X#0 THEN DRAW 0,X
70 IF X#-1 THEN DRAW 0,-X
80 IF X#1 THEN DRAW 0,X
90 IF X#-1 THEN DRAW -X,0
100 GO TO 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 bastığınıza bakar. Eğer (eğer) bu tuşlara basmışsanız, THEN (o zaman) bilgisayara bir çizgi çizmesi söylenmiş demektir. Bastığınız büyük harfe çizgi çizmez.

IF'i her zaman, Spectrum'un yapılmış olduğunu sinayacağı bir olgu izler - belli bir tuşa basılıp basılmadığı gibi. Gerçekse ya da oyuysa, 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.

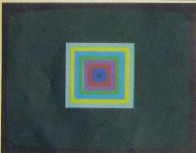
DESEN VE RESİM ÇİZİMLERİ

Alçak veya yüksek tanımlı çizimleri ya da her ikisini de kullanarak ZX Spectrum+ ile her tür desen ya da resim çizebilirsiniz. Bu sorunun üstesinden gelmenin en iyi yolu, deseni 80. sayfadaki çizimlerin bir kopyası üzerine çizmektir. Bundan sonra istenen yerde çizgi ve biçimler oluşturularak programı çıkarmak gerekir.

Desen ya da resimleri çizerken, programın belli bir bölümünü birkaç kez tekrarlayacak FOR NEXT döngülerini kullanabilirsiniz. Her defasında pozisyonları, karakterlerini ya da çizgi renklerini, genellikle düzenli bir biçimde değiştirebilirsiniz. İşte, bu tekniği kullanan bir program:

KARELER

```
10 BORDER 0: PAPER 0: CLS
20 FOR X=7 TO 8 STEP -1
30 INK 0
40 FOR Y=133 TO 137
50 PRINT "X TO 16*X"
60 NEXT Y
70 NEXT X
```



Bu programda üç FOR NEXT döngüsü içeriyor. x döngüsü, yapılan büyük karenin rengini ve boyutunu değiştirirken, l döngüsü ile c döngüsü her defasında küçük karenin satır ve sütun olarak pozisyonunu değiştirir. 60 satırdaki kareyi bir yıldız çevin ya da klavyedeki başka herhangi bir karakterle değiştirin.

Rastal etkiler ve alt programlar

Döngülerin kullanılmasında her defasında benzer şekiller elde edilemez anlamına gelir. Döngülerde RND (İngilizce RANDOM, rastalsal kısıltımsız) komutunu kullanarak, her defasında renkleri,

pozisyonları ve diğer görüntüleri değiştirebilirsiniz. 10. sayfadaki mozaikçe bakın. Bu, mürekkep rengi RND 7 olduğu için bu sonuca veriyor. Çünkü bu 0 ile 7 arasında herhangi bir ondalık sayı demek İNK bunu en yakın tam sayıya çeviriyor. Bu nedenle her defasında bir kare görünürce, rengi İNK 0 ile İNK 7 arasında bir renk oluyor.

Bir sonraki program, ekranda grafik karakterlerle simetrik desenler çiziyor. Bu karakterleri ve renklerini değiştirmek için RND kullanıyor. i ve p değişimleri mürekkep ve kağıt renklerini veriyor, a ise kaç desen çizileceğini (bu örnekte 4 kez) gösteriyor. n değişimi her desende kaç karakter olduğunu belirtiyor. x ise 129 ile 142 arasında rastal bir rakam.

50. satırdaki GOSUB 1000 bilgisayarı alt programa gönderir.

SİMETRİK DESELER

```
10 BORDER 4: PAPER 0: CLS
20 LET X=4: I=1: P=0
30 FOR X=1 TO 4
40 LET X=RND(4)
50 FOR N=1 TO 40: GO SUB 1000:
NEXT X
60 LET I=1+I: LET P=P+1
70 STOP
80 NEXT X
1000 LET I=INT (RND(4)+1)
1010 LET C=INT (RND(4)+1)
1020 INK C: PAPER 0
1030 PRINT AT (I,C) CHR$(X)
1040 PRINT AT (I-1,C) CHR$(X)
1050 PRINT AT (I+1,C) CHR$(X)
1060 PRINT AT (I,C-1) CHR$(X)
1070 PRINT AT (I,C+1) CHR$(X)
1080 RETURN
```



Alt program, program içinde program gibi hareket eden bir grup satırdır. Bu programda, alt program 1000. satırdan başlıyor. Ekranın dört çeyreğine de, hepsi merkezden (11,16 pozisyonu) aynı uzaklıkta bir dizi grafik karakter yazılıyor. İNT rastal sayıyı, I'nin 0 ile 10, c'nin de 0 ile 15 arasında olabilmesi için bir tam sayıya çeviriyor. 1030'dan 1060. satıra kadarkiler kodu x olan grafik karakterleri belirtiyor. (51. sayfadaki karakter listesine bakın).

BEEP tonu bulunduğu noktaya göre değişen bir ses çıkartıyor, ardından 1060. satırdaki RETURN, programı 50. satırdaki GOSUB'dan sonra gelen son cümleye gönderiyor.

60. satır mürekkep ve kağıt rengini değiştiriyor, 70. satırdaki PAUSE 100, programı döngü yeniden başlamadan 2 saniye durduruyor. 90. satırdaki STOP ise döndürücü döngüden sonra programın derhal alt programa geçmesini önlemek için gerekli.

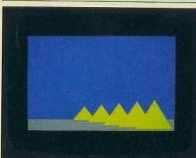
30. satırdaki 4 ve 50. satırdaki 40'ı başka rakamlara çevirerek, bu programı değiştirebilirsiniz. 40. satırdaki x'in alanını daha genişleterseniz, başka karakterlerin ekrana yansımaları önlersiniz. i ve p'nin de 7'den büyük olmaması gerekir.

Çizimlerde FOR NEXT döngüsünün kullanımı

FOR NEXT döngüleri, yüksek tanımlı çizimlerde düzgün şekli ve çizgilerden oluşan resimleri yapmanın çok iyi kullanılabilir. Aşağıdaki programı yazıp, işleme koyun. Sadece PLOT ve DRAW komutlarıyla, iki FOR NEXT döngüsü, önce yerdiki çizgiyi, sonra da içi dolu beş üçgen ya da piramidi çizer.

PIRAMİTLER

```
10 BORDER 0: PAPER 1: INK 6
20 CLS
30 FOR Y=8 TO 20 STEP 2
40 PLOT 0
50 DRAW 255,0
60 MEAT Y
70 FOR X=100 TO 250 STEP 30
80 FOR A=13.0 TO 10.0+7.0
90 PLOT 0,35+7.0
100 DRAW X
110 NEXT X: NEXT Y
```



Şimdi de yan sütundaki satırları ekleyin. İşleme koyduğunuzda, gece karlılığında yerden yansıtılan bir lazer ışınının göğün yıldızlarıyla kaplanmasına yol açtığını göreceksiniz. Çizgi, ekranın köşesinden x,y pozisyonuna çiziliyor. x,y,değişimleri rastal sayılır. Bunlar daha sonra yıldızlar için alçak tanımlı çizimdeki pozisyon rakamlarına dönüşür.

```
120 LET X=RND(255)
130 LET I=INT (135-Y)/8
140 LET C=INT (X/4)
150 PLOT 0,0: DRAW OVER I,X,Y
160 PLOT 0,0: DRAW OVER I,X,Y
170 PLOT 0,0: DRAW OVER I,X,Y
180 PRINT AT I,C:
200 GO TO 120
```



160 ve 180. satırlardaki OVER 1, ilk satırın lazer ışını çizdikten sonra, ikinci satırının olmasını, ancak resmin geri kalanını bozulmamasını sağlıyor. İleriye gerekeceği için bu programı saklayın.

FLASH, BRIGHT ve INVERSE

Bu üç komut Spectrum'un renklerini gerçekten harekete getirebilir. Her komutu 0 veya 1 izler. 0 veya 1'dan sonra noktalı virgöl kullanmak koşuluyla, bunları PRINT cümleleri olarak verilebilir. FLASH 1 karakterlerin bulunduğu yerlerin, mürekkep ve kağıt rengi arasında yansımasını, BRIGHT 1 renklerin parlaklaşmasını sağlar. INVERSE 1 mürekkebi kağıt rengi, kağıdınkini mürekkep rengi yapar. Bu komutlardan sonra 0 kullanarak ekranı normal hale getirir.

Bu komutların ne sonuç verdiğini görmek için, bu iki sayfadaki programlarda şu değişiklikleri yapın. Kareler programında 60. satırdaki kareyi yıldız çevirip, şunu ekleyin:

15 INVERSE 1

Değişim mürekkep renkli bantlar üzerinde siyah (kağıt rengi) eklezdir belirlir. Devam etmeden INVERSE 0 ekleyin.

Simetrik Desenler programında BRIGHT ve FLASH'ın çalışmasını görmek için şu satırları ekleyin:

15 BRIGHT 1

15 FLASH 1

FLASH deseni sanki ileri geri gidiyormuş gibi gösterir. Ekrandaki yansıma somneyi durdurmak için FLASH 0:CLS komutu verir.

Bu değişiklikler her programdaki bütün görüntüye etkilir. FLASH, BRIGHT ve INVERSE 1 bir PRINT cümlesinde kullanırsanız, bu üç komutun etkisi o satıra yazılanlarla sınırlı kalır.

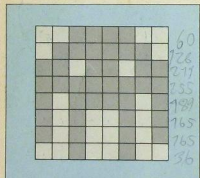
BİLGİSAYAR KARAKTERLERİNİN YARATILIŞI

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

Her karakter mükrekpe renginde 64 küçük nokta ya da pixel'den oluşur. Bunlar her bir sırada 8 tane olmak üzere sekiz sıra halinde ve her karakter alıca 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, mükrekpe rengindeki pixel'lerdir. Sonra ya yazarak ya da kafanızdan, dolu kareleri 1, boşları da 0 olarak alın. Şu bir örnek çizimi:



Kullanılan belirlediğiniz 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. Örnek karakteri s olsun.

```
10 POKE USR "S" : BIN 00111100
11 POKE USR "2" : BIN 01111110
12 POKE USR "6" : BIN 11011101
13 POKE USR "7" : BIN 10111111
14 POKE USR "8" : BIN 10111101
15 POKE USR "9" : BIN 01000101
16 POKE USR "0" : BIN 10100101
17 POKE USR " " : BIN 00000000
```

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

ÖRÜMCEKLER

```
20 BORDER 1 PAPER 0 CLD
30 PRINT "S" RND*7; " "
40 GO TO 30
```



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

Renklerin benekli karelerle "karıştırılması"

Spectrum'unuzda renkleri kolayca karıştırabilirsiniz. Bunu yapmak için bir kareyi yüzde 50'si mükrekpe, yüzde 50'si kağıt renginde dolduracak bir karakter oluşturmak gerekir.

```
10 FOR X=0 TO 5 STEP 2
20 POKE USR "S" : BIN 01010100
30 POKE USR "4" : BIN 11010101
40 NEXT X
```

Bütün yapmanız gereken, iki pixel satırı belleğiniz 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 mükrekpe ve kağıt renklerinin karışımını yansıtacaktır.

READ ve DATA ile çizimleri basitleştirme

Bilgisayar grafiklerinizi oluşturmanın daha kolay bir yolu vardır. O da READ ve DATA ile ondaklık sayılar kullanmaktır. Önce 0 ile 1'lerden oluşan sekiz ikil sayıyı ondaklı sayıya çevirin. Bunu PRINT BIN arından sayıyı verecek yapın. Örneğin

PRINT BIN 00111100

Spectrum'da 60 görünür, bu 00111100'ün ondaklık halidir. Örümceğin satırının ondaklık sayıları 60, 126, 219, 255, 189, 165, 165 ve 36'dır.

Artık READ ve DATA'yı kullanabilirsiniz. Bu iki komut, bir programa değişen olarak rakam gibi birçok değeri kolayca verebilmeyi sağlar. READ'ı bir değişen izler. Bu, eğer rakamsa, bir veya birkaç harf, bir dizidir işlemlen tek bir harftir. Burada ondaklık sayıları okumak (READ) istediğinize göre, rakamsal bir değişen gerekir. Buna y diyelim.

Spectrum'unuz READ ile karşılığınca, programdaki DATA (veri) cümlesine bakar. Bu cümle, virgüllerle ayrılmış bir dizi değerden oluşmaktadır. Bilgisayar buradaki ilk değeri alır. READ'e geldiğinde ise ikinci değer verir ve bu sırasıyla böyle gider. İste size örümcek yapacak yeni bir program.

```
10 FOR X=0 TO 7
20 READ V:BIN " " : X : U
30 NEXT X
40 DATA 60,126,219,255,189,165,165,36
```

Aynıda bu program bir karakter oluşturmak üzere belleğe herhangi sekiz ondaklık sayıyı yerleştirir. 30. satırdaki sekiz ondaklık sayı harfe çevirin. 50. satırdaki DATA'dan sonra herbin virgül ile ayrılmış sekiz sayı yazın. Programı işleme koyduktan sonra GRAPH ile seçtiğiniz harfe basın.

Renk kontrol kodlarının kullanımı

INK (mükrekpe) ve PAPER (kağıt) gibi komutları kullancağınza, PRINT cümlelerinde ilk brnaktan sonra kontrol kollarını koyabilirsiniz.

Ekrana yazıyacak 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 hangi nasıl seçeceğinizi gösteriyor:

BAS		1	2	3	4	5	6	7	8	9	0
MAVİ MÜR	KIRMIZI MÜR	MOR MÜR	YEŞİL MÜR	A.MAVİ MÜR	SARI MÜR	BEYAZ MÜR	FLASH AÇ	FLASH AÇ	SIYAH MÜR		
MAVİ KAGIT	KIRMIZI KAGIT	MOR KAGIT	YEŞİL KAGIT	A.MAVİ KAGIT	SARI KAGIT	BEYAZ KAGIT	BRIGHT KAPA	BRIGHT AÇ	SIYAH KAGIT		

Satranç tahtası çizimi

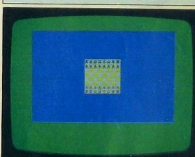
Bu, ekrana satranç tahtası çizip, taşlarını oynattırma hazır yerleştiren bir programdır. Aşağıdaki çerçevede gibi, renk kontrol kodlarını kullanarak, renklerin bir sıralamada yer almasını sağlayarak bunu yapabilirsiniz.

SATRANÇ

```
10 FOR X=1 TO 8
20 READ S#
30 GO SUB 800
40 NEXT S#
50 BORDER 4 PAPER 1 CLS
60 FOR L=7 TO 14 STEP 2:FOR C
=15
70 NEXT C
80 PRINT AT L,C; " " : AT L+1,C;
90 NEXT C
100 PRINT AT 7,15; " "
110 PRINT AT 13,15; " "
120 PRINT AT 13,7; " "
130 GO TO 13 TO 7
140 READ V
150 POKE USR "S" : X : U
160 NEXT V
```

500. satırda başlayan alt program oyun taşlarını belirler.

```
150 DATA "0",0,0,16,56,56,16,12
160 DATA "1",0,64,124,56,56,124
170 DATA "2",0,16,56,120,64,56,124
180 DATA "3",0,16,40,60,100,56,124
190 DATA "4",0,16,56,16,56,56,5,6
200 DATA "5",0,64,40,16,100,124,124,0
```



HAREKETLİ GÖRÜNTÜLER

Bilgisayar çizimleri, eğer karakterler ve çizgiler ekranda hareket edebilirse çok daha hoş görünür. Spectrum'unuzda bu hareketliliği yaratmak çok zor değildir. Bütün yapmanız gereken, bir karakterin ya da çizginin 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'unuzdu kapatmadınız 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 5: CLS
20 FOR X=0 TO 7
30 BEEP V
40 POKE USR "S" * X, V
43 NEXT X
50 DATA 60, 120, 219, 255, 109, 105
60 35
55 FOR X=0 TO 7
70 BEEP USR "S" * X, V
80 NEXT X
90 DATA 10, 10, 16, 16, 16, 10, 10
0
110 FOR Y=0 TO 20
120 BEEP AT 11.3, INK 2: "S"
130 PRINT AT 11.3, INK 2: "S"
140 NEXT Y

```

Bu programı her işleme koyuşunuzda örümceğin iplikten indiğini göreceksiniz.

Programda 60'tan 100. satıra kadar olanlar bir başka grafik karakter ("S") olan ipliği yapıyor. Hareketi, satır sayısı 1'i 0 ile 20 arasında değiştiren FOR NEXT döngüsünü oluşturan 110 ile 140 arasındaki satırlar sağlıyor. Döngünün kendini her tekrarlayışında, belli bir pozisyonda bir miktar iplik oluşuyor ve örümcek bir sonraki pozisyonda beliyor. Bir sonraki turda bir miktarda iplik daha oluşuyor ve örümcek yeniden işleme başlıyor. Böylece örümcek 21. satırda yani görüntünün en altına kadar hızla ipliğin ucunda iniyor.

Spectrum'unuz yeni pozisyonları hızla hesaplayabilir ve bu nedenle örümcek hızla iner. Bunu yavaşlatmak için şu satırı verin:

135 PAUSE 10

Bu bilgisayarın, örümceği bir sonraki pozisyonda yazınsızdan önce saniyenin beşte biri kadar beklemesine neden olur. 10'u bu başka bir rakama çevirin ve etkisini izleyin.

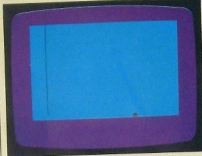
Şimdi de programa şu satırları ekleyin ve yeniden işleme koyun. Bu, daha hareketli farklı bir yönde göreceksiniz.

KAÇAN ÖRÜMCEK

```

150 FOR C=3 TO 30
160 PRINT AT 21.3, C: INK 2: "S"
170 PRINT AT 21.3, C+1: INK 2: "S"
180 NEXT C

```



Bu daha örümcek yere değer değmez bir tarafa çıkarıyor. Ek satırlar sütun pozisyonu 'c'yi denetleyen bir başka FOR NEXT döngüsü oluşturuyor. Bakın önce bir boşluk beliyor, ardından örümcek bir sonraki sütun pozisyonunda ortaya çıkıyor. Böylece örümcek sağa giderken, bir pozisyonda kaybolup, bir diğerinde beliyor. Bir karakteri eski bulunduğu pozisyonda yok etmeden, yeni bir yere yazımsıztaksana, eskisini yok ettikten sonra yeniden yazımsıztık saha iyidir. Bu, hareketli çizimlerde titremeyi yok eder ya da azaltır.

Hedef uygulaması

Birçok bilgisayar oyunlarında, olay, ya hareket halindeki iki şekil çarpışına ya da bir şekil ışına isabet alırsa meydana gelir. Bilgisayar çarpışma ya patlamanın olacağını nasil biliyor?

Çarpışma olacağını kestirmek güç değil. Eğer iki karakterden biri (satır), c(sütun) pozisyonunda, diğeri de v(dikey), h(yatay) pozisyonundaysa ve |v-v|, c-h ise, ikisi de aynı pozisyonlarda demektir. Bunu şöyle bir cümle olarak yazabilirsiniz:

```
160 IF |v-v| AND c=h THEN PRINT "CRASH!"
```

Çarpışmanın kontrol etmenin bir başka yolu da renkleri kullanmaktır. NEW komutuyla örümcek programını silin. 31. sayfadaki Pırımtiller programının tümünü verin, ya da bu yeni teypte sakladığınızda bilgisayar yazabilir ve (eğer bu arada kapatmadysanız ya da reset'e getirmedyseniz bilgisayarın belleğindeki)

örümceğimize birleştirerek yeni bir program yaratılabiliriz.

Önce grafik bir patlama, "e", yaratacak aşağıdaki satırları ekleyin.

```

5 FOR X=0 TO 7
6 BEEP V
7 POKE USR "E" * X, V
8 NEXT X
9 DATA 145, 82, 44, 121, 159, 52, 7
4, 137

```

Şimdi de 190. satırı silip, aşağıdakileri de ekleyin ya da değiştirin.

ÖRÜMCEKLER VE PIRAMİTLER

```

114 LET H=END*3
115 FOR S=0 TO 20
117 PRINT AT V, H: "S" * S: "T" * V: H, S:
118 4, 137
200 NEXT V
205 PRINT AT 21.3, FLASH 1: "INK
210 PAUSE 5: "S"
210 GO TO 114

```



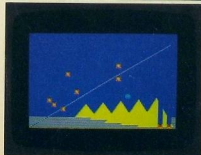
Artık yıldız patlamaları olmaz. Bunun yerine gökten düşecek örümcekler, piramitler ve yeni yiyeceklerdir. Yaptığınız, v ve h'nin örümceğin pozisyonunu belirlediği, bir FOR NEXT döngüsü eklemek oldu. h değisgeni ratsaldır. Bu nedenle örümcekler ekranın değişik yerlerinde dikey düşmeye başlarlar. Şimdi de şu satırları ekleyin.

PATLAYAN ÖRÜMCEKLER

```

190 IF ATTR (V+1, H) = 14 THEN GO
TO 500
200 PRINT AT V+1, H: FLASH 1: PA
PER 5: "E"
510 PAUSE 100
720 GO TO 114

```



Lazer ışını örümceklere isabet sağladıkça, kısa bir süre için bunlar san olacak. DRAW'ın oluşturduğu çizgi örümcek karakterinin pozisyonuna uzağına, murekkep rengi, çizginin rengi olan sanya döner. 190. satırda, ATTR örümceğin sanya dönuştüğünü fark edince, bilgisayar 500. satırdaki patlama alt programına gönderir.

Zıplayan top

Bir çok grafik programda cisimler ekranın kenarından yansır. Bu program, bunun nasıl yapılabileceğini gösteriyor. v ve h değisgenleri patlayan örümcekler programındaki gibi işlerler, ama toplan aşağı yukarı sağa sola gitmesi için bunlara +1 veya -1 eklenir. SCREEN, c, h pozisyonunda bir X olup olmadığını kontrol eder.

ZIPLAYAN TOP

```

10 BORDER 1
20 FOR Z=0 TO 10
30 PRINT AT Z, Z: "S" * Z: "T" * Z: Z:
40 PRINT INK 2: PAPER 5: FLASH
50 NEXT Z
60 PRINT AT V, H: "S" * S: "T" * V:
70 NEXT V
80 IF H=0 OR H=251 THEN LET X=X-
1: BEEP 2: 10
90 IF H=0 OR H=251 THEN LET X=X+
1: BEEP 2: 10
110 IF SCREENS (V, H) = "X" THEN P
RINT INK 1: PAPER 5: AT V, H: "X"
120 STOP
130 PAUSE 5
140 GO TO "0"

```



Gönderme kullanımı

ATTR komutu, ekrandaki belli bir pozisyona "gönderme" leni satırlar. "Gönderme" ler, murekkep ve kağıt renginden, belli bir pozisyonun yanıp sönmelerinden ya da patlak olup olmasından kaynaklanır. Patlayan Örümcekler programında, ATTR, örümceğin san renk alıcısı yok olmasını sağlar. Burada gönderme murekkep rengi (6 rakamıdır). Kağıt rengi mavimsi (1 rakamıdır) ve örümcek parlak ya da yansır söner de gildir. Bu durumda göndermelerin toplamı 14'tür. Programların Başvuru Kılavuzu bu rakamların nasıl bulunduğunu gösterir.

MÜZİK VE ÇEŞİTLİ SESLERİN YAPIMI

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

Seslerin programlanması

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

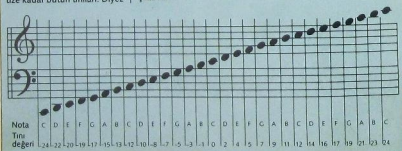
Spectrum' unuzun çıkaracağı bütün sesleri duyabilmek için şu programı verin:

```
10 FOR P=0 TO 60 STEP -1
20 BEEP P*.05,P: BEEP P*.05,P
30 NEXT P
```

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

Müzik yapmada tınıların değeri

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



tıkırtı gibi geldiğini farkedeceksiniz. Çünkü bu sesler insan kulağının duyabileceği sınırların ötesindeki seslerdir. Bu sayfanın altındaki çizgele, notaların tını değerini göstereyor. 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 (RND*40) -30
20 BEEP P*.05,P: BEEP P*.05,P+7:
30 GO TO 10
```

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

MAKİNA

```
10 FOR P=+10 TO 30
20 BEEP .05, P-34-X
30 NEXT P
40 GO TO 10
```

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

KALKIŞ

```
10 FOR P=1 TO 40 STEP 0.2
20 BEEP P*.01,P: BEEP .01,P-.6
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 - yarımtonunun beşte biri - kadar değişir. Bu da sesin tınısının daha yavaş yükselmesine yol açar. STEP değerini değiştirerek, diğer tını değişikliklerini de deneyin.

KLAVVE DEĞİŞİKLİĞİ

```
10 LET P=CODE INKEY$
20 IF P=0 THEN GO TO 10
30 BEEP P*.05, P-30: P
40 GO TO 10
```

Bu program sizin herhangi bir tuşa basmanız bekler. Basınca her biri değişik bir ses çıkarır. Bir tuşa basarsanız, CAPS SHIFT e basarsanız, sesin baslaştığını göreceksiniz. Bu program, her tuşa basılışında 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 görülebilir.

Spectrum' un sesini nasıl yükseltebilirsiniz?

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

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

sarma) düğmelerine basın. Kasetli tepyin ses düğmesini ayarlayınca, bilgisayarsının sesinin tepyin oparlarından geldiğini duyacaksınız. Ya da kulaklığınızı da kasetli tepye takabilirsiniz.

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

gerekir. Spectrum, kasetli tepyeleri ya da diğer tepyeleri gibi düz sinyaller ürettiğinden, amplifikatördeki REPLAY ya da LINE IN girişleri kullanılabilir. Eğer bir sorununuz 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 eklemişti. Programı böyle yavaşlatarak yerde, ses çıkaracak bir duraklama da programlayabilirsiniz. 135. satırı şöyle değiştirin:

135 GOSUB 500

Şimdi de programa şu satırları ekleyin:

```
200 STOP
500 FOR P=40-1 TO 30-1 STEP -1
510 BEEP P*.02,P
520 NEXT P
530 RETURN
```

Program işlemeye koyunca, örümcek aşağı inerken bir vızıltı çıkaracaktır. Alt program, örümceğin her bir alttaki pozisyona geçişinde çok hızlı 0.02' yi notalar. 500. satırı ve 510. satırdaki 0.02' yi değiştirip notaları hızlandırıp yavaşlatarak yeni notalar ekleyin.

PROGRAMLARINIZI NASIL SAKLAYABİLİRSİNİZ?

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

Kendi programlarınızı kaydı

- 1 Önce Spectrum'unuzu 14. sayfada anlatıldığı gibi uygun bir kasetli teyp, kaset bağlantısı ile bağlayın. Ancak Spectrum'un sadece MIC girişini 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"

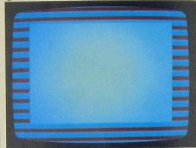


Ad için, on harf ya da rakamdan oluşan herhangi bir bileşim kullanılabilir. ENTER'e basın. SAVE satın kaybolacak ve ekranda Spectrum'dan gelen kasetli teyp kullanma talimatlarını göreceksiniz.



bilgisayardaki programı burada saklayın. Spectrum, programı teyp, kasete kaydedilebilecek biçimde yollar. Bundan sonra programı her faydalarda anlattığı gibi bilgisayara yükleyerek kullanabilirsiniz. Bu iki sayfada, programların nasıl saklanabileceğini ve programın doğru olarak kaydedildiği 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 aktarır. Önce ekranın çevresinde aşağı yukarı oynayan mavi-kırmızı çizgiler göreceksiniz.



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

- 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 program aktarmı epey zaman olabilir.



- 7 Program teyp'e aktarılınca, OK, 0:1 mesajı görünür. Teypi durdurun. Program artık saklanmıştır. Eğer isterseniz bunu kontrol edebilirsiniz ya da "doğrulatılabilirsin."

Programınızı nasıl doğrulatabilirsiniz?

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

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



İkinci mavili-sarı bölümün sonunda, şu mesaj belirir.

OK, 0:1

Bu, Spectrum'unuzun belleğinde olan programı, teypteki program ile karşılaştırıp, tastaman aynı bulunduğunu gösterir.



Aynı programınızın, kasete geçtiğinden kesinlikle emin olabilirsiniz.

Yazılım saklamada öğütler

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

5 REM SPIDER program Version 3

Program işleme konusunda, bilgisayar bütün REM cümlelerini gözardı eder. Programın istediğini yerlerine REM'i kullanarak, görüş ve düşüncelerinizi ekleyebilirsiniz.

Eğer bu mesajı alamazsanız, bir aksilik var demektir. Önce 16. sayfadaki **Yazılım yüklemeye**deki sorunlar tablosuna bakın. Çünkü program kasete aktarılmış ama bir hatadan ötürü, doğrulama için bilgisayara yüklenemiyor olabilir. Eğer burada bir hata varsa, hatayı düzeltin, kaseti başa sarın ve programı yeniden doğrulattın. Bilgisayar program hâlâ doğrulanıyorsa, 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:

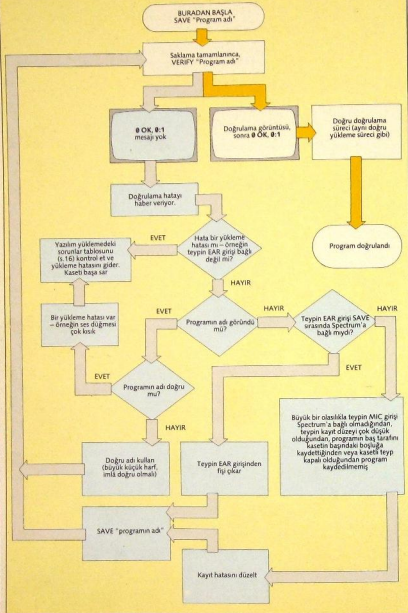
SAVE "SPIDER" LINE 1

Saklama süreci ikkinden farklı değildir. Ancak bu defa doğrulatırken, VERIFY ve programın ardından sonra LINE 1'i dahil etmeyin. LINE 1 ile herhangi programlar, yüklenmişinde otomatik olarak başlar. RUN'i kullanmaya gerek yoktur, (ama program başlanınca 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 bölümünü saklamak için kullanılabilir. Bunlar için Programcının Başvuru Kılavuzuna bakın.



ZX SPECTRUM + 'U TANIYALIM

Bu bölümde, ZX Spectrum + 'u' un içine girecek, klavyenin altındaki çeşitli öğelerin neye yaradığını ve bilgisayarın çalışabilmesi için bunların birbirine nasıl bağlandığını öğreneceksiniz. Ayrıca yan donanımları, yani Spectrum'unuzu tam bir bilgisayar sistemi haline getirecek, eklenebilir parçaları nasıl kullanacağınızı öğreneceksiniz. Nihayet, burada bilgisayarınızın teknik yönleri - örneğin belleğin düzenlenişi, Spectrum' un teknik özellikleri - hakkında daha çok bilgi bulacaksınız.



İÇİNDE NELER VAR?

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

Kasanın içinde klavyeyi Spectrum' un 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 dilimidir. 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 giylenmiş 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.

Bağlantısız 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.

Rastsal Kullanım Belleği (RAM)
Bu yongalar, değişimlerin değerleri gibi, programların gereksinim duyduğu belirli bilgileri bilgisayara besleyen programı içerir. RAM' in 48 K' lık içeriği klavyeden değiştirilebilir veya reset' e getirerek ve bilgisayar kapatarak tunden silinebilir.

TV çıkışı
Bu, televizyon alıcısına giden sinyalleri üretir.

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

Teyp soketleri
Bunlar, bellekten bilgi ve programları bir teypte gönderilmesi ve teypten yeniden belgeye verilmesi için kullanılır

Mantık Yongaları
Bu yongalar, CPU ile RAM arasında bilgi alışverişinde aracı görevi yapar.

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

9 VDC giriş
Elektrik bağlantısını sağlar

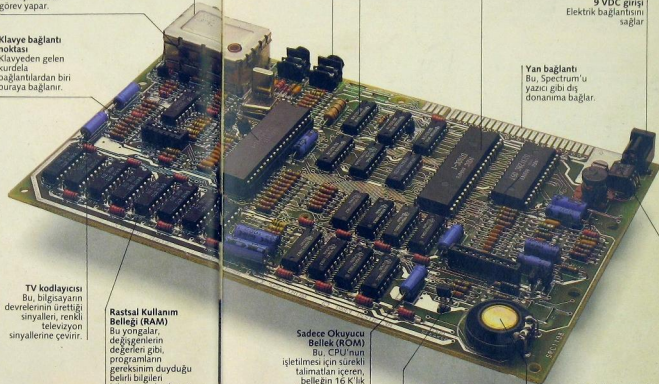
Yan bağlantı
Bu, Spectrum' u yazıcı gibi dış donanıma bağlar.

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

Klavye bağlantı noktası
Klavyeden gelen kurdela bağlantılardan biri buraya bağlanır

Voltaj düzenleyicisi
Bu parça voltajdaki değişikliklerin bilgisayarın etkilenmesini önler.

Operatör
Cerektiginde ses üretir.



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 hızla ve daha kolay işlenmesini sağlayan Microdrive'lar ve diğer Spectrum'lar da dahil olmak üzere çok çeşitli diğer donanımların bilgisayara bağlanmasını bu mümkün kılar. Bu arapaçardan yararlanarak, Spectrum'unuzu standart yazıcılara bağlayabilir, telefon hatları aracılığıyla program ve veri alıp vermesini salayan modemle balantı kurabilirsiniz. Bilgisayara anında program yüklemek için kullanılan ROM kartuşları ile bağlantı sağlayan başka arapaçalar 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 arapaçla kullanmadan bilgisayarınıza bağlayabilirsiniz. Bu tür bir yazıcı aynı zamanda ZX Interface 1'in de arkasına takılabilir. Ancak RS232 çikşli 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'unuz sekiz kadar Microdrive' a, 63 kadar diğer Spectrum bilgisayarına ve RS232 standart arapaç 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

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

Arapaça 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 alışverişini yapabilir.

ZX Interface 1 birimi ayrıca 9 adet D girişli bir RS232 arapaçası da içerir. Ve bu girişleri yazıcılar, diğer standart donanımlar, modemler ve diğer bilgisayarlar Spectrum'unuza bağlanabilir. Ayrıca bir de standart arapaça kablosu da vardır.



Standart yazıcılar ZX Interface 1 aracılığıyla bağlanır.

ROM kartuşları ve oyun kolları

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

Yan bağlantı
Yan donanımlar bilgisayara buradan bağlanır.

ROM kartuşu/oyun kolu arapaçası



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



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

Dipnot

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



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

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



Uyan
Yan donanımlar, daima ceryan bağlantısı yapılmadan bağlanmalıdır.

ZX SPECTRUM+'UN BELLEK HARİTASI

42 ve 43. sayfalardaki Spectrum'un içini gösteren resme baktığımızda, bir ROM ve 16 daha küçük RAM yongası olduğunu göreceğiz. 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 16×1024 (16384) byte ya da adres taşımaktadır. Bu bellek, byte'ları sadece okur, yani bunlar değiştirilemez. (Değiştirilirse bilgisayar çalışamaz 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 48×1024 (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, bilgisayarın 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 tanımladığı grafikler	
GOSUB yığı	RAMTOP
Yedek	
Hesaplama yığı	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 kullanılış biçiminin bir özeti 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'İN 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 basar 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 koşulları geçmesini 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 bileşimini gösterir.

Kısaltmalar	Açıklama	Örnek
num-const	Sayısal değişmez (rakam)	24.5
num-var	Sayısal değişgen (rakam da içerebilen değişgen)	sum
num-expr	Sayısal ifade (bir rakam veren komutlar, sayısal değişmezler ve değişgenlerden oluşan geçerli herhangi bir bileşim)	sum * 24.5 RND * 7
int-num-const	En yakın tam sayıya yuvarlanmış bir sayısal değişmez, değişgen ya da ifade	
int-num-var		
int-num-expr		
string-const	Bir dizi değişmez ya da dizi (tüm karakterleri içinde karakterlerin herhangi birleşimi)	"ZX Spectrum +"
string-var	Dizi değişgen (dizi içeren değişgen)	a\$
string-exp	Dizi ifade (dizi veren komutlar, değişmez ve değişgenlerden oluşan geçerli herhangi bir bileşim)	a\$ + "ZX Spectrum +" a\$(5 TO 8) Y * X
letter	Herhangi bir büyük veya küçük harf	B\$a\$
letter\$	Herhangi bir büyük veya küçük harfi izleyen \$	X = 10 AND T < 10
cond	Bir koşul veya koşul içinde bir altkoşul	IF T > 10 THEN STOP
statement	Başka bir cümle ile kullanıldığında geçerli olan herhangi bir BASIC cümlesi	PRINT INK 2 : X

[] Tekrarlanabilecek seçmeli bir öğe

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

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 değişgenler

Sayılar

9 ya da 10 haneli bir sayıyı tamamen saklayabilir. İşleyebildiği sayılar ise 10^{38} ile $4 \cdot 10^{-39}$ arasında yer almaktadır.

Kabul edilebilir değişgenler

Sayı Bir harfle başlamak üzere istenen uzunlukta. Boşlukları dikkate almaz ve bütün harfleri küçük harfe dönüştürür. Büyük küçük harfler farkı gözetmez.

Dizi '\$' in ardından gelen herhangi bir tek harf. Büyük küçük harf ayrımlı yapmaz.

Düzen Düzen değişgen ve altöge (substring) leri için, DIM maddesine bakın.

Biçimi gösterirken, aşağıdaki kısaltmalar kullanılmıştır.

Kısaltmalar	Açıklama	Örnek
num-const	Sayısal değişmez (rakam)	24.5
num-var	Sayısal değişgen (rakam da içerebilen değişgen)	sum
num-expr	Sayısal ifade (bir rakam veren komutlar, sayısal değişmezler ve değişgenlerden oluşan geçerli herhangi bir bileşim)	sum * 24.5 RND * 7
int-num-const	En yakın tam sayıya yuvarlanmış bir sayısal değişmez, değişgen ya da ifade	
int-num-var		
int-num-expr		
string-const	Bir dizi değişmez ya da dizi (tüm karakterleri içinde karakterlerin herhangi birleşimi)	"ZX Spectrum +"
string-var	Dizi değişgen (dizi içeren değişgen)	a\$
string-exp	Dizi ifade (dizi veren komutlar, değişmez ve değişgenlerden oluşan geçerli herhangi bir bileşim)	a\$ + "ZX Spectrum +" a\$(5 TO 8) Y * X
letter	Herhangi bir büyük veya küçük harf	B\$a\$
letter\$	Herhangi bir büyük veya küçük harfi izleyen \$	X = 10 AND T < 10
cond	Bir koşul veya koşul içinde bir altkoşul	IF T > 10 THEN STOP
statement	Başka bir cümle ile kullanıldığında geçerli olan herhangi bir BASIC cümlesi	PRINT INK 2 : X

[]

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 değeri	=	L	Eyitir
:	7	Yeni satır başlatır	:	Z	Bir program satırında cümleleri ayırır
{	8	Parantez açar	/	V	Bölme
}	9	Parantez kapatır	*	B	Çarpma
<	Q	Daha az veya eşit	<	Kendi tuğu	Ondaklı işareti
>	W	Eşit değil	:	Kendi tuğu	Yan sütunda gösterilen cümleleri ayırır
>	E	Daha büyük veya eşit	:	Kendi tuğu	Program cümlesindeki cümleleri ayırır
>	R	Daha az	*	Kendi tuğu	Dizi başlatıp bitirir
>	T	Daha büyük	*	Kendi tuğu	0 veya 16. sabrda gösterilen komutlardan sonra değeri ayırır
]	H	Üssüne yükseltir			
]	J	Çıkartma/olumsuz			
+	K	Toplama/olumlu/Dizi kümesi			

ZX Spectrum + character set

	0	1	2	3	4	5	6	7	8	9
0										
10	göstergeç sağı	göstergeç yukarı	DELETE	ENTER	TRUE VIDEO sayı	INV VIDEO GRAPHICS MODE	PRINT virgül kontrol	INK kontrol	PAPER kontrol	göstergeç sola
20	INVERSE kontrol	OVER kontrol	AT kontrol	TAB kontrol					FLASH kontrol	BRIGHT kontrol
30										
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	RND	INKEY\$	PI	FN	POINT
170	SCREENS	ATTR	AT	TAB	VALS	CODE	VAL	LEN	SIN	CGS
180	TAN	ASN	ACS	ATN	LN	EXP	INT	SQR	SIN	ABS
190	PEEK	IN	USR	STRS	CHR\$	NOT	BIN	OR	AND	<=>
200	>=<	<>	LINE	THEN	TO	STEP	DEF FN	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 C

İşlev

ABS bir sayıya değerini mutlak değerini, yani aksi etkiyi işaretli olmaksızın değer verir.

ABS'nin kullanımı

ABS'yi bir sayıya 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.

Bilgisi

ABS num-const

ABS num-var

ABS (num-expr)

ACS Arc Cosine

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT W

İşlev

ACS, kosinüsünden bir açının

değerini hesaplar.

ACS'nin kullanımı

ACS'yi bir sayıya değer izler. Ifade parantez içine alınmalıdır. Örneğin,

60 LET x=ACS (y-z)

ACS'yi izleyen değer (yukarıda y-z) istenen açının kosinüsünü verir -1 ile 1 arasında olabilir. ACS'nin değeri radiant olarak verir. Radiant derecede çevirmek için ACS'nin verdiği değer 180/PI ile çarpın.

Örnek

Emir

PRINT 180/PI * ACS 0.5

60 verir, bu kosinüsü 0.5 olan açının derecesidir.

Bilgisi

ACS num-const

ACS num-var

ACS (num-expr)

AND

Klavyede yeni

SYMBOL SHIFT Y

Mantık işlemi/İşlev

AND, bir koşullar bileşiminin gerekliliğini sinamada mantık işlemi olarak görev yapar. Ancak bütün koşullar gerçekleşirse, bileşimin tamamı gerçekleşir. AND ayrıca iki sayısal ya da dizi değeri üzerinde iki işlem yapar için bir işlev olarak da kullanılır.

AND'nin kullanımı

Mantık işlemi olarak AND, tümünün gerekliliğini sinamada bir cümlede 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) gerçekleşirse, ekranda "Correct" yazısı görünür. Eğer biri ya da her ikisi de yanlışsa, o zaman bileşimin tümü yanlış olur ve program bir sonraki satıra geçer.

İşlev olarak AND işlev olarak AND, iki sayısal değeri işleyebilir. Örneğin

50 LET x=y AND z

AND ancak ikinci değeri (z) 0'a eşit değilse iki değeri (y) verir, ikinci değeri (z) 0 ise o zaman 0 verir. AND, kendinden önce gelmesi koşuluyla bir dizi değeri de işleyebilir. AND'1 mutlak bir sayısal değerin izlemesi gerekir. Örneğin

50 LET a=b AND z

AND, ikinci değeri (z) sıfırı değilse iki (b) verir, ikinci değeri (z) 0 ise boş ("") verir.

Dikkat edin, ZX Spectrum + doğru bir koşula 1 değeri, yanlış olana da 0 verir. Sifir olmayan değeri gerçek 0'ı yanlış olarak kabul eder. Sayısal değerlerin bileşimini standart gerçek tablosu yanınca da değerlendirmez.

Örnekler

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

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

80 LET a\$="1" Out Of Time Or Not" AND NOT correct="1" Correct"

Eğer 60. satırdaki iki koşul gerçekleşirse, sayısal değeri (score) 10 artırır ve a\$ "Correct" olur. Eğer koşullardan biri yanlışsa, o zaman correct 0'dır ve diğer 0'dur. score değeri ve a\$ "Out Of Time Or Not Correct" (zamanınız ya da doğru değil) olur.

Bilgisi

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

ASN Arc Sin

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT Q

İşlev

ASN, sinüsünden bir açının değerini hesaplar

AND, bir koşullar bileşiminin gerekliliğini sinamada mantık işlemi olarak görev yapar. Ancak bütün koşullar gerçekleşirse, bileşimin tamamı gerçekleşir. AND ayrıca iki sayısal ya da dizi değeri üzerinde iki işlem yapar için bir işlev olarak da kullanılır.

ASN'nin kullanımı

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

60 LET x=ASN (y-z)

ASN'yi izleyen değeri (yukarıda y-z) istenen açının sinüsünü verir -1 ile 1 arasında'dır. Bu durumda ASN açının

değerini radiant olarak verir. Radiant derecede 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 derecesidir.

Bilgisi

ASN num-const

ASN num-var

ASN (num-expr)

AT

Klavyede yeni

SYMBOL SHIFT I

Bak INPUT:PRINT

İşlev

ATN bir açının değerini tanjantından hesaplar.

ATN'nin kullanımı

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

60 LET x=ATN(y+z)

ATN'yi izleyen değeri (yukarıda y+z) istenen açının tanjantıdır. ATN değeri radiant olarak verir. Radiant derecede 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 derecesidir.

Bilgisi

ATN num-const

ATN num-var

ATN (num-expr)

ATTR ATTRIBUTES

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT

İşlev

ATTR ekrandaki belirli bir karakter pozisyonunun göndermelerini verir. Bu göndermeler, bu pozisyondaki karakterlerin murekkep ve kağıt rengi, parlaklığı ve yanıp sönmeye durumudur.

ATTR'nin kullanımı

ATTR'yi birbirinden virgülle ayrılmış ve tümü paranteze alınmış iki sayısal değeri izler. Örneğin

150 IF ATTR(y,h)=115 THEN GOTO 2000

ATTR'yi izleyen iki 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 göndermelerin toplamları ve şöyle hesaplanır:

Murekkep rengi. Renk kodu (0-7) Kağıt rengi. Renk kodunu 8 katı Parlaklık Yanıp sönmeye 128

Örnek

Eğer 11,16 pozisyonundaki karakterinin murekkep rengi 3 (mor), kağıt rengi 6 (sarı) ve karakter parlak ama yanıp sönmüyorsa, o zaman

PRINT ATTR (11,16)

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

İkili biçimde ATTR

ATTR bir byte'a verildiğinde, 7 bit (en önemlisi) yanıp sönmeye için 1, normal için 0, 6. bit parlak için 1 normal için 0, 5. ile 3. arasında diğer ikili rakamları yazılmış kağıt rengi, 2. ile 0. arasında murekkep rengidir.

Bilgisi

ATTR(num-expr,num-expr)

BEEP

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT Z

Cümle/Emir

BEEP, verilen uzunlukta ve tonda bir sesin optörden çıkmasını sağlar.

BEEP'in kullanımı

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

30 BEEP y

iki değeri (x) 0 ile 10 arasında olabilir ve notlanan sayıya olarak çünüsü belirler. İkinci değeri (y) -60 ile 69 arasında olabilir ve yanıp sönmeye halinde notlanan süresi belirler. Ortadaki C'nin (do) altındaysa aksi, üstündeyse azar deregedir.

Örnek

Emir

BEEP 0.5, 1

Ortak C'nin (do) bir üstündeki C diye sesinin yanıp sayıya süreyle çikmasını sağlar.

Bilgisi

BEEP num-expr,num-expr

BIN

Binary number

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT B

BIN bir ikil sayıyı ondalık sayıya çevirir.

BIN'in kullanımı

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

50 POKE USR "a", BIN 101010

BIN bu ikil sayının ondalık karşılığını verir. Bu, genellikle yukarıda olduğu gibi POKE ve USR ile birlikte kullanılır. Tanımadığı grafiklerin yanında kullanılır. Burada 1 murekkep, 0 ise kağıt renginde bir pixel'e işaret eder.

Örnek

Emir

PRINT BIN 111111110

Ekranda, bu ikil sayının ondalık karşılığı olan 254 görünür.

Bilgisi

BIN (1) (0)

BORDER

Klavyede yeni

Cümle/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 INDF -D

BORDER' izlenen 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

Arka-BORDER ekranın alt bölümünün kağıt rengi de belirler. INK ve PAPER dan farklı olarak BORDER değeri, bir PRINT cümlesinin içine yerleştirilemez.

Bilgisi

BORDER int-num-expr

BRIGHT

BRIGHT, karakterlerin ekranda hareketlenirken 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

Yerine BRIGHT'1 izleyen değeri en yakın tam sayıya yuvarlanabilir. Bu sayı 0, 1 veya 4 olabilir. Değeri 1 olması, bunu izleyen PRINT veya

INPUT cümleleri ile ekrana yansıtacak bütün karakterlerin murekkep ve kağıt renginin daha parlak görünmesini sağlar. Değeri 8 olması, yeni karakterler yazıldığında, eskiden parlak karakterlerin bulunduğu pozisyonları, parlak; normalden bulunduğu pozisyonların da normal kalmasını sağlar. BRIGHT'1 izleyen ise hem BRIGHT'1 hem de BRIGHT'0'ı izler. bundan sonra gelen bütün karakterler normal görünür.

BRIGHT ayrıca PRINT, INPUT, PLOT, DRAW ve CIRCLE gibi ekranda görüntü yaratan komutlarla oluşturulan cümlelere de yerleştirilebilir. BRIGHT komutları sonra, ama yeni ya da görüntü öğeldenden önce gelir. BRIGHT'1 kendi değeri ve bir noktali virgül izler. Örneğin

50 PRINT BRIGHT:1 "WARNING!"

Bu durumda BRIGHT'1 etkisi sınırlıdır ve ancak kendisini izleyen karakterleri, ya da bu cümle ile konulan nokteli, çizilen çizgiyi kapsar. Uzunmayan BRIGHT'1 ile murekkep renginde bir pixel, bir karakter pozisyonuna yerleştirilecek bu karakter pozisyonunun tümünün kağıt rengi de parlak olur.

Bilgisi

BRIGHT int-num-expr [:]

CAT Catalogue

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

CHR\$ Character string

Klavyede yeni

EXTEND MODE

İşlev

Spectrum'un karakter tablosu, klavyede bulunan karakter ve komutlarının yanı sıra kullanılabilecek grafik karakterlerini oluşturur. CHR\$ ve kod numarası kullanılarak, bunlardan herhangi bir bir dizi olarak elde edilebilir. Karakter tablosu ayrıca, karakterlerin ekrana yansıtılmasını etkileyen baz kontrol kodları da içerir. CHR\$ den önce PRINT kullanılarak bu kodlar hareketlenirken karakterler ekrana yansıtılabilir. Karakter tablosunun tamamı ve kod numaralarını 51. sayfada bulabilirsiniz.

CHR\$'nin kullanımı

CHR\$'yi bir sayısal değeri izler. Örneğin

80 PRINT CHR\$(5)

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

DEF FN kullananla kavlayımla bulunulması bir işlevi tanımlama olanağı verir. Bu işlevi hareketle geçirip, sonuçta sayısal ya da dizi değeri veren bir FN cümlesine bu işlevin çığılması belirleyici olabilir.

DEF FN'nin kullanımı

DEF FN sadece bir program cümlesi olarak kullanılabilir. Bir sayısal işlev tanımlanacaksa, DEF FN'yi önce bir harf, ardından da parantez içine almamız ve virgüleyle ayırmış bir veya birkaç sayısal değeri ardı. DEF FN (x,y) gibi, bunun izlenir bir çift parantez ile değiştirilmesi çeren bir sayısal ifade gibi. Örneğin,

```
1000 DEF FN (x,y)=
  SQR (x * 2 + y + 1)
```

DEF FN (x,y) zeytin haf (yukanda r), işlevi anlayışta bir addr. Değişigener aynca tek bir harf de olabilir. Her iki halinde de Spectrum büyük harf - küçük harf ayırmı yapmaz.

Eğil gerebilen işlev ifade, bu işlevi tanımlamak için değiştirilen (yukanda x y) kulllanır.

DEF FN cümlesi, programın herhangi bir yerine yerleştirilebilir, bunun tanımladığı işlevi yapılmak için bir FN cümlesi kullanılır. Bunu işlevin adı olan harf ile hepsi parantez içine almış, birbiriinden virgüle ayırmış bir dizi sayısal değer gelir. Örneğin,

```
50 PRINT FN (1,3,4)
```

Parantez içindeki değerler, DEF FN cümlesindeki değeri yerine aynı sırada fonksiyona aktarılır. Yani, bu örnekte x=3, y ve d=4 değeri verilecektir. FN bu işlevi çağırarak değerlerdir ve sonucu verir.

Bazen de DEF FN'yi sadece bir hafife boş bir parantez ile kullanırız. Örneğin,

```
1000 DEF FN ( )=INT (x + 0.5)
```

Değişigene (yukanda x) verilen değeri; FN tarafından istenince fonksiyona aktarılar. Bu örnekte, FN (r), x'e verilmis değeri en yakın tam sayıya yuvarlatmış gibidir bulur.

DEF FN ve dizi

DEF FN ve FN aynı biçimde bir dizi fonksiyonu tanımlayabilir ya da işleme koyabilir. Bu durumda fonksiyonu adı bir harf ile onu işleyen b'şimdedir. Tanımlaymaya da buna uygun bir dizi ifade yapar. Örneğin,

```
1000 DEF FN a$(b$,x,y)=
  b$(x TO y)
```

Bu örnekte eğil gerebilen işlevin dizi ifade bir dizi bölücüdür. x ve y b'ş'nin bir bölünümü ile ve son karakterlerdir. FN'yi milaka işlevin adı ve parantez içinde bir dizi değer ile fonksiyona aktararak diğer belirleyiciler izlemelidir. Bu durumda

```
PRINT FN a$
(*FUNDAMENTAL* ,1,3)
```

emr. FUN yazısız

```
PRINT FN a$
(*FUNDAMENTAL* ,5,8)
```

emr. AMEN yazısız ekrana yansır.

Bicim

DEF FN letter{letter} [letter] ;
 [open >] [close <] [open >] DEF FN letter\$(letter\$) [letter\$] [letter\$] ;
 [letter\$] ;
 FN letter [int-num-exp] [int-num-exp] FN letter\$ [string-exp] [int-num-exp] [string-exp]

DİM Dimesion

KLAVYE YERİ

D

Cümle

DİM, belirli sayıya sayısal ya da dizi değeri veren bir düzen için değiştirilen FN cümlesidir. Düzen, aynı adı taşıyan değerlerin bir listesidir. Bu değeri veren birbirlerinden ayrılmış. (Altoge bir düzen içinde her değeri veya öğeyi belirleyene değeri)

DİM'in kullanımı

DİM bir program cümlesi oluşturmak için kullanılır. Bunu, düzen adlandırarak bir harf ve parantez içine almamız ve virgüleyle ayırmış bir veya fazla sayısal değer izler. Örneğin,

```
20 DIM x(10)
```

```
80 DIM x(20,5)
```

İlk durumda, 1'den 10'a kadar altoge en öge içeren, tek yönlü sayısal düzen oluşturulmuştur. Düzenin adı x'tir ve altoge değeri (x) (1) ile x(10) arasında'dır. Aynı adı taşıyan mevcut herhangi bir düzen iptal edilmis ve değeri yeniden her birine 0 değeri verilir. Unutmayın bir düzen kullanırken Spectrum bu adın büyük ya da küçük harf olması ayırt etmez -x(2) değeri X(2) değeri ile ayırdır. Ancak, x harfini taşıyan başka sayısal değeri ile aynı adı taşıyan (x veya X) düzen birlikte kullanılabilir ve gerekiyorsa aynı FN kullanılır.

Parantez içindeki değerlerin sayısı, bir sayısal düzen yaratmış olduğuna sayısına eşittir. İkinci örnekte ise, birime boyutu 20, ikinci 5 öge, 100 ögeden oluşan 4 boyutlu bir düzen kurmaktadır. Bu öge (x(1,1)) den (x(20,5)) e kadar numara almaktadır.

İstediği kadar çok sayıda boyutlu düzen kurulabilir. Bir sayısal düzen ögele, o düzenin aynı işlevi parantez içindeki bir değeri adlandırabilir. Örneğin,

```
70 PRINT x(a)
```

```
160 PRINT z(7,b)
```

DİM ve dizi düzenler

DİM burada da aynı sayısal düzen olduğu gibi kullanılabilir, fakat düzenin adı bir harf ve onu izleyen 0'dur. Ayrıca, parantez içindeki boyut değerlerine, her bir dizinin uzunluğunu belirleyecek bir değer daha eklenmelidir. Örneğin

```
30 DIM a$(20,5)
```

```
90 DIM b$(20,5,10)
```

İlk cümle, her bir 5 karakterli bir dizi öge ve 20 öge bir düzen oluşturmuştur. Altoge değeri değiştirilen a\$(1) den a\$(20) ye kadar adlar alır ve bunlara başlangıçta bir boş ()'erir. Aynı adı taşıyan mevcut dizi iptal edilir. Sayısal düzenlerin aksine, aynı adı taşıyan basit bir dizi değeri bulunma aynı adla varolamaz.

İkinci örnekte ise ilk boyutu 20, ikinci boyutu 5 ögele 100 ögeden oluşan iki boyutlu bir dizi düzen oluşturulmuştur. Bunun ögeleinin uzunluğu 10 karakterdir. Daha sonra bir dizi düzenine dizi değeri verildiğinde, gerekirse ya düzenin sonuna boşluk bırakılır ya da düzeni uyu kesilir.

Bir dizi düzenin ögele, düzenin adını izleyen, altogeinin sayısını ya da sayıları veren parantez içindeki bir ya da daha çok sayısal değeriyle adlandırılır. Örneğin, a\$(2) ögesi "SMITH" ve b\$(7,2) ögesi "DERBYSHIRE" olabilir. Ancak dizideki belli bir karakter tanımlamak için bir değeri daha ögele olabilir. Verilen örnekte b\$(2,2) "M" ("SMITH" in ikinci karakter) ve b\$(12,4,5) "Y" olacaktır.

Sıfır boyutlu düzen

Parantez içinde tek bir değer kullanılarak sıfır boyutlu bir dizi düzen yaratmak da mümkündür. Örneğin,

```
10 DIM c$(15)
```

Bu düzenin tek bir ögesi var: c\$. Uzunluğuda belirli bir değerde (15 karakter) sabitlenmiştir.

Bicim

DİM letter {num-exp [num-exp] int-num-exp}

DRAW

KLAVYE YERİ

W

Cümle/Emir

DRAW, ekranda düz ya da eğri çizgi çekmek için kullanılır.

DRAW'ın kullanımı

DRAW normal olarak bir program cümlesi biçiminde kullanılır. Düz bir çizgi gerekiyorsa, DRAW'ı virgüle ayırmış iki sayısal değer izler. Örneğin,

```
40 DRAW x,y
```

Bunun dışında, yükektin çizim çizimi çizilmesini daha sonra ise, ya bir önceki PLOT cümlesi ile tanımlanan bir pozisyonun veya bir önceki DRAW cümlesi ile ulaşılan noktadan bir dizi çizgi çizer. DRAW'ı izleyen her iki değeri gerekiyorsa en yakın tam sayıya yuvarlanır. İlk değeri (yukanda x) bu pozisyonun yatay çizgiyi, ikinci

değeri (y) ise dikey çizgiyi belirler. Eğer çizgi sola ya da sağa giderekse bu değeri ek olur. Ulaşılabacak noktadan da ekran çerçevesi içinde olması gerekir.

Eğer daha önce bir PLOT veya DRAW cümlesi yoksa, DRAW özyeme 80 pozisyonunun (ekranın sol alt köşesinden) başlar. Benek cümleleri ya da emirler DRAW'ı etkiler. PLOT ve CIRCLE gibi, belli sonuçlar verecek başka cümleler de çerebilir.

DRAW ve eğri çizgiler

Bir darenin parçası olarak bir eğri de çekmek için DRAW cümlesinden üçüncü bir değeri de yer alabilir. Üçüncü değeri (yukanda z), çizgi çizilince çıkacak açıyı (radyan olarak) tanımlar. Bu değeri artsı se -z'ye sola, eksi se sağa döner. Pi ya da π -Pi değerleriyle dairesi daire çizer.

Örnek

Aşağıdaki program bir üçgen çizer:

```
10 PLOT 127,150
```

```
20 DRAW 70,-10
```

```
30 DRAW -140,0
```

```
40 DRAW 70,100
```

DRAW cümlesine 1 ya da -1 eklemeye üçgenin kenarlarını çizmek veya dibkünye olmasını sağlar.

Bicim

DRAW [statement:] int-num-exp, int-num-exp [,int-num-exp]

ERASE

Microdrive dosya işleme emri. Microdrive ve Interface 1'e kitabına bakın.

EXP EXPONENT

KLAVYE YERİ

EXTEND MODE

X

İşlev

EXP, bir sayı (e) belli bir kuvvette yükseltken bir matematik işlevidir.

EXP'nin kullanımı

EXP'i bir sayısal değer izler. Örneğin,

```
60 LET Y=EXP X
```

ifade parantez almaktadır. Bunun üzerine EXP ifade ettği e sayısını verilen kuvvette (yukanda x) yükseltir.

Emir

Bunun dışında, fonksiyonu tanımlayan DEF FN ile birlikte kullanılır.

PRINT EXP 1

ekranda 2.7182828'ı yansır. Bu, e'nin değeri.

Bicim

EXP num-var

EXP num-exp

FLASH

KLAVYE YERİ
 EXTEND MODE
 SYMBOL SHIFT V

Cümle/Emir

FLASH, mürekkep ve kağıt renklerini kontrol eder. Birimlere değiştirilerek, bir karakter pozisyonunun yanıp sönmelerini sağlar.

FLASH'ın kullanımı

FLASH doğrudan bir emir olarak kullanılır, ama normal kullanımı, bir program cümlesi biçimindedir. Bunu bir sayısal değer izler. Örneğin

```
50 FLASH 1
```

FLASH'ı izleyen sayı gerekiyorsa bir tam sayı yuvarlanır ve bu değeri 0, 1 veya 8 olur. Değeri 1 olması, bunu izleyen PRINT veya INPUT cümleli ile ekrana sayısal kaç tütün karakterlerin yanıp sönmelerini sağlar.

Değeri 8 olması, yanıp karakter yazıldığı, eksi yanıp son karakterlerin bulunduğu pozisyonların yanıp söm, eskiden normal olanların normal 8-kalmasını sağlar.

"FLASH" izleyen 0 de hem FLASH 1, hem de FLASH 8 iptal eder, bundan sonra gelen bütün karakterler normal görünür. FLASH aynıca PRINT, INPUT, PLOT, DRAW ve CIRCLE gibi ekranda görüntü yaratan komutlarla oluşturulan cümlelere de yerleştirilebilir. FLASH komuttan sonra, ama veri ve diğer görünümlerinden önce gelir. FLASH'ı kendi değeri ve bir noktali virgül izler. Örneğin,

```
120 PRINT FLASH:1;INK 2;
```

```
PAPER.6;"WARNING";
```

Bu durumda FLASH'in etkisi sınırlıdır ve ancak kendisini izleyen karakterleri, ya da bu cümle ile konum noktayı, çizilen çizgiyi kapar. Unutmayın, FLASH'ı ile mürekkep renginde bir piksel, bir karakter pozisyonuna yerleştirilecektir. 8x8 piksel boyutundaki bu karakter pozisyonunun tümü yanıp sönmeye başlar.

Bicim

FLASH int-num-exp [:]

FN FxNctioN

KLAVYE YERİ

EXTEND MODE

SYMBOL SHIFT Z

İşlev

FN, kullananın tanımladığı fonksiyonun hareketi geçirir. Daimi bir fonksiyonu tanımlayan DEF FN ile birlikte kullanılır.

FN'nin kullanımı

Eğer bir sayısal fonksiyon kullanılacaksa, FN'yi bir harf ve bir parantez izler. Eğer fonksiyona bazı belirleyiciler verilecekse, o zaman parantez birbiriinden virgüle ayrılmış bir dizi işlev alınır. Örneğin,

```
170 LET X=FN (1,3,4)
```

Belirleyiciler (yukanda 3 ve 4) r'dye adlandırılan fonksiyona aktarılır. Bunun üzerine FN sonucu verir. Eğer herhangi bir belirleyici verilmemişse, parantez gene de konumlandırılır. Örneğin,

```
70 PRINT FN (1)
```

Bu durumda fonksiyon, değeri verecek verilen son değeri kulllanır.

FN dizi fonksiyonu da aynı şekilde hareketle geçirir. Ancak fonksiyonu belirleyen harf 'n'in de eklenmesi gerekir. Ayrıntılar için DEF FN'ye bakın.

Bicim

FN {int-num-exp}

[int-num-exp]

FN letter\$ [string-exp]

[num-exp] [int-num-exp] [string-exp]

FOR

KLAVYE YERİ

Cümle/Emir

FOR her zaman TO ve NEXT komutlarıyla birlikte bir FOR NEXT döngüsü yaratmak için kullanılır. Bu sayılar, programın bir bölünümü belirler tek tekrarlama sayısını sağlar.

FOR'un kullanımı

FOR daima TO ile bir cümle oluşturur. FOR' u bir harf, bir çift işaret ile birbiriinden TO ile aynımlı iki sayısal değeri izler. Örneğin,

```
60 FOR I=1 TO 9
```

Harf (yukanda a) bir kontrol değeri oluşturur. Bunu tekrarlama cümleleri izler ve normal olarak bunların en az birinde bu kontrol değeri değişen kullandırılır. Döngü bir NEXT cümlesi ile biter. Bunda NEXT'i cümlesi ile değiştirilebilir.

90 NEXT a

Uygulamaya konunca FOR, kontrol değeri ile aynı adı taşıyan diğer değeri izleyen iptal eder ve başlangıçta buna TO dan önceki sayı kadar bir değeri verir (yukanda 1). Bundan sonra kontrol değeri bu değeri taşıyan cümleler uygulamaya konur. NEXT'e varınca kontrol değeri değişimin değeri 1 artar. Eğer bu değeri TO dan sonraki değere az ya da ona eşitse (yukanda örnekte limit 9) program FOR cümlesine döner ve FOR NEXT döngüsü tekrarlama. Kontrol değeri değişim değeri limitin dışındadır, fazla ise, o zaman döngü sona erer ve program NEXT ten sonraki cümle ile devam eder.

Yukandaki örnekte, kontrol değeri 9 den 9'a yükselirken, döngü 1 kez tekrarlama. Döngü sona ererken değeri 9'den 10'a çıkarılır. Eğer kontrol değeri değeri limitin dışındadır, fazla ise, o zaman döngü sona erer ve program NEXT ten sonraki cümle ile devam eder.

Spectrum kontrol değeri adlandırılan büyük harf küçük harf ayırmı yapmaz.

FOR NEXT döngüsünde STEP kullanımı
Eğer kontrol değeriinin değeri n 1 den farkli bir değerde artması ya da azalmasi itenişyorsa FOR cümlesine STEP komutu yerleştirilbilir. STEP limiti değeri izler ve ardından da bir sayisal değer gelir. Örneğin,

```
60 FOR a=1 TO 10 STEP 2
```

Kontrol değeriini limite varinceya kadar step değeri (yukarıda 2) kadar artacaktır. Kontrol değeriini arıyışla 1,3,5,7,9 değerlerini alır ve değeri 11'e çıkınca döngüyü bırakır. Eki step değeri, kontrol değeriinin değeriini azaltmasına neden olur. Bu durumda başlangıç değeri bitiş değeriinden yüksek olacaktır ve döngü, kontrol değeriinin değeriini limiti altına düşünce sona erer. Örneğin,

```
60 FOR a=9 TO 1 STEP -1
```

Değer 9'dan 1'e döğru iner ve 0'a varınca program döğünden ayrılır.

Döngülerin yuvulanması
Birden çok FOR NEXT döngüsü içine yerleştirilirse bu buna 'yuvulanmak' denir. NEXT cümlesindeki kontrol değeriinlenin sırası FOR cümlesindeki kontrol değeriinlenin sırasına tesisi yönünde olmalıdır. FOR NEXT döngüleri istenen yere konabilir ve istenildiği kadar çok döngü birbirinin içine yuvulanabilir.

Bicim:
FOR letter= num-exp1 TO num-exp2 [STEP num-exp3]
NEXT letter

FORMAT

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

GOSUB

Klavyede yeri
H
Cümle/Emir
GOSUB programın altprograma sarmasını sağlar. Altprogram, programın aynısı bir bölümdür. Bu, özellikle altprograma birkaç kez gerek olacağı çok yararlıdır.

GOSUB'ın kullanımı
GOSUB bir cümle ya da döğrudan emir olarak kullanılır ve ardından bir sayisal değer gelir. Örneğin,

```
GOSUB 1000
```

Uygulamaya konunca, GOSUB'ü izleyen değeri (yukarıda 1000) en yakın tam sayıya yuvulanır ve program, bu değeriye sahip satıra geçer. Bir değeriye veya ifadelerin kullanımını programın hesaplanabilir bulunmuş bir satır sayısına sarmasını sağlar. Eğer oyle bir satır numarası yoksa, program gere de sapa ve kargına çıkan iki cümleye devam eder.

Altprogram RETURN ile sona erer ve program gene sapa ve GOSUB'un ardından gelen satıra devam eder. Altprogramlar birbirini izleyecek biçimde yuvulanılmıyorsa, bu durumda RETURN, programı uygulamaya konan son GOSUB cümlesini izleyen cümleye döğnderir.

GOSUB yubini
GOSUB uygulamaya konduğı zaman satır numarası belirlekti GOSUB yubiniya yerleşir. RETURN den önce iki veya daha fazla yazıya fazla GOSUB uygulanması. Bu durumda THEN'in numarası en işe gelecek şekilde sıralanır. RETURN daima son satır numarasını bu yubından alır, programu devam emri olarak işe bu satıra gider.

Dikkat eden, yetenice RETURN cümlesi yoksa 4. hata (Belletek yok) ortaya qakabilir.

Bicim:
GOSUB int-num-exp1

GOTO

Klavyede yeri
G
Cümle/Emir
GOTO bir programın belli bir satıra sarmasını sağlar.

GOTO'nu kullanımı
GOTO, önce ekranı temizlededen programın belli bir satır numarasından işleme koylemek üzere döğrudan emir olarak kullanılır. Aynı zamanda bir program cümlesi içinde THEN statement [statement] bir sayisal değeri gelir. Örneğin,

```
60 GOTO 350
```

Uygulamada, GOTO'yu izleyen değeri en yakın tam sayıya yuvulanır ve program bu numaryayı taşıyan satıra sapa. Bir değeriye ya da ifade kullanılmıyorsa, programın hesaplanabilir bulunmuş bir satıra sarmasını sağlar. Dikkat eden eğer böyle bir satır yoksa, program gene de sapa ve karışıklığı iki cümleden devam eder.

Bicim:
GOTO int-num-exp1

IF

Klavyede yeri
I
Cümle/Emir
If'da, daha sonraki bir eylemi etkileyecek bir karar alacak olan THEN komutu ile birlikte kullanılır. Bunun için, bilgisayar bir şeyin doğru olup olmadıđını arar. Eğer doğrusu bunu belli bir yubinde bir hareket izler. Eğer değilise, bir başka yubine hareket gelir.

IF ve THEN'ın kullanımı
If normal olarak THEN ile bir cümle döğöğner. If'ın önce sayisal değeri veya bir koşul, sonra da THEN ve

onun ardından gezeeri bir veya birkaç BASIC cümlesi izler. Örneğin

```
80 IF > THEN GOTO 250  
240 IF <= NO THEN PRINT  
"THE END" : STOP
```

Bir değeriye, değeriye veya ifade (yukarıdaki x gibi) sıfırdan başka bir değeriye sahip döğru kabul edilir. Bu durumda THEN'in izleyen cümle ve ayrı satırdaki diğer cümleler de uygulanır. Program daha sonra bir sonraki satıra geçer. Eğer değeri 0 ise o zaman değeriye, değeriye veya ifade yanlı sayılır. İzleyen cümleler uygulamaya konmaz ve program bir sonraki satıra atılır. Örneğin, eğer x 0 ise o program 250. satıra gilmeyecektir.

If izleyen bir koşul (<=>="NO") doğrusu, THEN'in izleyen cümleler uygulamaya konmaz satıra geçer. Bu ornekte, eğer <=>="nin değeri" NO ise ekranda "THE END" göğürür ve program döğner. Eğer <=>="nin değeri" başka bir şeyse program bir sonraki satırdan devam eder.

Spectrum döğru koşulu 1, yanlı koşulu 0 değeri verir. Şi değirdaki her değeri döğru, 0'ı yanlı kabul eder. Bir cümle ile bir koşulu değeri olarak bir değeriye verilebilir. Örneğin,

```
70 LET <=>="NO"
```

Dikkat eden, değeri bazı BASIC'lerden farkli olarak THEN'in GOTO'dan önce de konması gerekir.

Bicim:
IF num-exp1 THEN statement [statement]
IF cond THEN statement [statement]

IN

Klavyede yeri
EXTEND MODE SYMBOL SHIFT

İşlev
IN, klavyenin ve diğer giridi çıkı donanımının durumunu denetler. IN belli bir giridi adresinden gelen, bu giride başka donanım durumu döğöğner bir iş olur.

IN'ın kullanımı
IN'i bir sayisal değeri gelir. Örneğin,

```
150 LET x=IN y  
If izleyen değeri ile 65535 arasında değeriye ve okunacak giridi adresini belirler. IN bunun üzerine bir adressten okuduğıu byte'i ekrana yandırır.

```

Klavye adresleri
Klavye 8 adreses vardır. Bu adreslerden her biri be değışik bit da birimini girer. Hangi bite döğöğner, INK komutu izler. Bu adresler 65278, 65022, 64510, 63486, 61438, 57342, 49150 ve 32766'dır. Bu adreslerdeki byte döğöğnerleri ise 175, 183, 187, 189 veya 190 olabilir.

Bicimi
IN num-const
IN num-var
IN (num-exp1)

INK

Klavyede yeri
EXTEND MODE SYMBOL SHIFT X

Cümle/Emir
INK, ekranda karakterlerin belleteği, notkalanı konacağı, düz veya eđri çizgiğlerin çizileceđi rengi belirler.

INK'ın kullanımı
INK döğöğner emir olarak da kullanılır, ama normal kullanım bicimi bir program cümlesindedir. Bunun bir sayisal değeri gelir. Örneğin,

```
70 INK 5
```

INK'ı izleyen değeri en yakın tam sayıya yuvulanır ve 0-9 arasında herhangi bir sayı olabilir. Sayılar şu renkleri verir:
0 Siyah
1 Mavi
2 Kırmızı
3 Mor
4 Yeşil
5 Açık mavi
6 Sarı
7 Beyaz
8 Sıydam
9 Fon koyu ise beyaz, akıksa siyah

INK 8, ekranda INK 8'in kullanımıldığı döğöğner mevcut rengin değeriinden kalmasını sağlar. INK 9 ise kağıt (fon) rengine göre mürekkep renginin göğünecek şekilde siyah veya beyaza döğöğmesini sağlar.

Sınırlı veya genel mürekkep rengi
INK yukarıda öđüğü gibi kendi başına bir cümle oluştuyorsa, o zaman renk değeri ve bunu izleyen her göğürünte bu üstünde döğöğner bir rengi alır. INK ayrıca PRINT, INPUT, PLOT, DRAW ve CIRCLE ile yapılan ve göğürünte oluşturan cümlelerin içine de yerleştirilebilir. INK komutu izler, ancak veriler ya da göğürünte öğelerinden önce gelir. INK'i kendi değeriye ve notkallı yubine izler. Örneğin,

```
60 CIRCLE INK 4, 128, 88, 8  
Bu durumda INK'ın etkisi sınırlıdır ve sadece göğünecek karakterler, konacak noktaya da çizilecek çizgi budan etkilenir. Bu ornekte, sadece bir yeşil daire çizilir. Bundan sonra mürekkep rengi genel renge ya da orijinal renge sayıya döğöğür.

```

Bicim:
INK int-num-exp1 [:]

INKKEYS [INPK KEY Shift]
Klavyede yeri
EXTEND MODE N

İşlev
INKKEYS klavyede belli bir tuşa basılıp basılmadıđını saptamak için kullanılır.

INKKEYS'ın kullanımı
INKKEYS e bir öğnerme gerekecek, genellikle bir diz değeriye bir karakteri tahsis etmek ya da belli bir karakteri sınamak için kullanılır. Örneğin,

```
70 LET <=>=INKKEYS  
130 INPUT INK; "N THEN STOP"
```

Bu uygulamaya konduğı zaman, INKKEYS 0 veya basılan tuşun karakterini yandırır. Eğer bir tuşa basılmıyorsa bir boş diz (" ") verir. Dikkat eden, INKKEYS büyük ve küçük harflerle SHIFT tuşuna basılarak elde edilenlere basmadan elde edilenleri ayrıyettir. (Bu aynı okunaklı bir tuşa basılıp basılmadıđını belirlemek için INK kullanılır.)

INK'ın Y'tan farkli olarak INKKEYS eğer belteden bir sonraki cümleye geçer. Bu yuzden normal olarak, istenen tuşa basılana kadar kendini tetkik etmaması mümkün kılacak bir döğöğeriye yerleştirilir.

Önek
Bu satır (CAPS SHIFT ve CAPS LOCK'a basılmadan) Y tuşuna basılana hareketi askoya alır.

```
60 IF INKKEYS <> "Y" THEN GOTO 60
```

Bicim:
INKKEYS

INPUT

Klavyede yeri
I
Cümle/Emir
INPUT, bir program işleme konmuş bir bası verilerini yerilebilmesini sağlar.

INPUT'ın kullanımı
INPUT normal olarak bir program cümlesi oluşturun ve PRINT'e çok beşer yerleştirilebilir. En basit biçimde bir num sayısal ya da diz değeriye gelir. Örneğin,

```
60 INPUT x  
90 INPUT a$
```

Bunun üzerine bilgisayar bir sayı ya da diz verilece kadar bekler. Bu değeri verirken, ekranın altında satır başında göğürür. ENTER'e basınca, bu değeri alınımdış değeriye verilir oyle ve program devam eder.

Bir INPUT cümlesi birden fazla değeriye gerebilir ve bir soru (prompt) oluşturma şeklinde bazı karakterleri ekranı yanlıştır. Bu, PRINT ile de yapıldığı gibi, soruyu oluşturan karakterin bmk işaretleri işleme olarak ve gerektildiğinde öğeleri birbinden ayrı ayrı (iki notkallı virgöl veya birünç ilçiler kullanılarak yapılabılır. Bunun için INK, FLASH

ve PAPER gibi göğürünte etkileyen cümleler de eklenir. Örneğin,

```
80 INPUT INK 2: "What is your name?"; n$ (How old are you, +$+ "?); a$
```

PRINT'te arasımdış yu farklara dikkat eden INPUT bir değeriye gelince bekler, bu yuzden sorulara şahit edilecek butun değeriye ler i ifadeler (yukarıda n\$ de içine asılan öđüğü gibi) parantez içine almaktadır. Göğürünte en alt satırdan başlar ve birden fazla satır kullanımışına yuğru döğru çıkar. AT bir INPUT cümlesinde ayrı PRINT'e öđüğü gibi kullanılır. AT 0'ın öđüğü satırdan bir üstünden göğürünte yu başlar ve göğürünte iki satırdan fazla tuşlara yuğru döğru çıkar.

INPUT'un durdurulması
Eğer INPUT'u bir sayisal değeriye izliyorsun, STOP'a basınca program döğner. Diz değeriye ile işi beliren bmk işareti silinebilir ve STOP'a basınca program durdurulabilir.

INPUT'UN LINE ile kullanımı
INPUT LINE sadece diz değeriye lerle kullanılır. Normal olarak diz değeriye ler kullanılır. Normal olarak diz değeriye lerin değeriinde nitelik olmur. Düzünü rakamlarıni tuşlara basılıklıca bunlar bmk işaretiğinin içine belirtilir. Bu bmk işaretiğinin silmek için, arında diz değeriye ler olan bir INPUT LINE kullanılır. Eğer bir soru gerekiyorsa, bu INPUT ile LINE'in arısına yerleştirilir. Örneğin,

```
70 INPUT "What is your name?"; LINE n$  
Bicim:  
INPUT [prompt] [:] [:] [:]  
INPUT [prompt] [:] [:] [:]  
string-var  
INPUT [prompt] [:] [:] [:] LINE string  
[prompt] = [string-const] (string-exp1) [AT int-num-exp1, int-num-exp2] [statement] [:] [:] [:]
```

INT INTEGER

Klavyede yeri
EXTEND MODE R
İşlev
INT tam sayı olmayan tam sayıya döğöğürür.

INT'ın kullanımı
INT: bir sayisal değeriye izler. Örneğin,
70 LET x=INT y
İfade, parantez almaktadır. INT bunun üzerine, bu değeri en yakın tam sayıya yuvulanır ya da yevir.

Önek
Emir

PRINT INT-45.67. INT-7.66

Görüntü söyle olur.

45

-8

Biçimi

INT num-const

INT num-var

INT (num-expr)

INVERSE

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT M

Cümle/Emir

INVERSE, karakter pozisyonlarında kağıt rengini murekkep, murekkep renginin kağıt rengine dönüşmesini sağlar.

INVERSE'ni kullanımı

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

70 INVERSE 1

INVERSE'ni 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 renklerinin yer değişmesini sağlar. INVERSE 0 ise murekkep ve kağıt renklerini normale döndürür.

INVERSE, aynı İNK gibi, görüntü sağlayan cümlelerin içine yerleştirilebilir. Ancak CIRCLE, PLOT veya DRAW ile kullanılırsa INVERSE 1 bir çizgi veya noktadan kağıt rengini almasın, dolayısıyla da görülmemesine neden olur.

Biçimi

INVERSE int-num-expr

LEN Length of string

Klavyede yeni

EXTEND MODE

K

İşlev

LEN, bir dizinin uzunluğunu verir.

LEN'ni kullanımı

LEN'i bir dizin değeri izler. Örneğin

50 LET X=LEN A\$

ifade parantez almıyorsa, LEN dizideki karakterlerin sayısını verir.

Ömek

Aşağıdaki sabir

120 INPUT A\$,IF LEN A\$ > 9

THEN GOTO 120

sadece en fazla dokuz karakterli dizilerin geçmesini izin verir.

Biçimi

LEN string-const

LEN string-var

LEN (string-expr)

LET

Klavyede yeni

EXTEND MODE

V

Cümle/Emir

LET, bir değeriğine değer belirlemeye kullanılır. Sıncilar BASIC'te belirleme cümlesi LET olmadan olamaz.

LET'ni kullanımı

LET normal olarak bir program cümlesi oluşturur ama ayrıca doğrudan emir olarak da kullanılabilir. Bunu bir sayısal ya da dizin değeri, ben eşit işaret, sonra da bir değer izler. Bu değer LET'i izleyen değeriğine bağlı olarak bir sayısal ya da dizin değeri olabilir. Örneğin,

60 LET X=+1

80 LET A\$=Correct

Boylece değeriğine bir değer veriliyor olur.

Unutmayın, basit değeriğeler, LET, READ veya INPUT tarafından bir değer verilmeyecektir. Ancak dizin değeriğeler, daha başından ya 0 ya da boş dizin (bak DIM) olarak belirtilerler.

Biçimi

LET num-var=num-expr

LET string-var=string-expr

LINE

Klavyede yeni

K

Emir/Cümle

LIST, o anda bellekte bulunan programın sıralamasını verir.

LIST'ni 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.

diye verilen bir doğrudan emir ardından, önce sıralamanın ilk sayfası ekrana belirlir. N ara tuşu, STAY veya BREAK dışında herhangi bir tuşa basılınca da diğer sayfalar sıraya görünür.

LIST'i ayrıca sayısal değer biçiminde bir sabir numarası izleyebilir. Örneğin,

LIST 100

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

Biçimi

LIST [int-num-expr]

LLIST Line printer LIST

Klavyede yeni

EXTEND MODE

V

Emir/Cümle

LIST, Sıncilar tipi yazılmalı, o anda bellekte bulunan programın sıralamasını yazıya aktarmasını sağlar.

LIST'ni kullanımı

LLIST aynı LIST gibi kullanılır (aynı için LIST'e bak). Dikkat edin, sıralama yazılacak ekrandaki görüntü değizmez.

Biçimi

LLIST [int-num-expr]

LN Logarithm (Natural)

Klavyede yeni

EXTEND MODE

Z

İşlev

LN bir değeriğin doğal logaritmasını (e taban uzmanden logaritmasını) verir. EXP'ni tersi gibi izler.

LN'ni kullanımı

LN'yi bir sayısal değer izler. Örneğin,

60 LET X=LN Y

ifade parantez almıyorsa, LN'yi izleyen değeriğin 10'dan büyük olması gerekir. Bunun üzerine LN bu sayının doğal logaritmasını verir.

Biçimi

LN num-const

LN num-var

LN (num-expr)

LOAD

Klavyede yeni

Emir/Cümle

LOAD, bir programın tümünü teyipten belleğe yükler.

LOAD'ni kullanımı

LOAD normal olarak doğrudan emir olarak kullanılır, ama yeni bir program yüklenmesi için bir programın bir cümlesini de oluşturabilir. LOAD'ı bir program adıyla izler. Bu en fazla on karakterden oluşan bir dizin değeriğedir. Örneğin,

LOAD "dosya1"

Uygulamada, o anda bellekte olan program ile bunun değeriğelenin bütün değeriğini iptal eder. Spectrum daha sonra ad verilen program arar, bulunca da yükler. Dikkat edin, bilgisayar program adlarında büyük harf ile küçük harf aynı değildir. Bu emirde olduğu gibi LOAD'ı bir boş dizin izler.

LOAD " "

Spectrum bulduğu ilk tam programı yükler.

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

Biçimi

LOAD string-expr

LOAD CODE

Klavyede yeni

EXTEND MODE

I

Emir/Cümle

LOAD CODE, belleğin belli bir kesimine teypte depolanmış bilgilerin yüklenmesi için kullanılır. Bilgi bir dizin ya da harfle birlikte ve bunlar bellekteki belli adreslere gönderilir. LOAD CODE bir görüntüye ya da örneğin kulanılan tanımladığı karakterler için bilgileri yüklemek için de kullanılır.

LOAD CODE'ni kullanımı

LOAD CODE doğrudan bir emir olarak kullanılabileceği gibi, programın bir cümlesini de oluşturabilir. LOAD'ı bir dizin değeriğedir olan dosya adı ve CODE izler. Örneğin,

LOAD "data" CODE

LOAD'ı izleyen dosya adı, yükleniecek bilgilerin adresi ve aynı program adını sıralamasına sahiptir. (Bak LOAD). Bunun üzerine LOAD CODE bu adlı bilgileri arar, bulunca ekrana byte'lan, ardından da adını yansıtır. Spectrum daha sonra bu byte'lan, bellekte alındıkları adreslere yükler. Buralarda bulunan bilgileri ise silinir.

CODE'u ayrıca bir veya iki sayısal değeriğeyebilir. Bunlar virgüle ayrılır. Örneğin,

LOAD "picture" CODE

16384-6912

CODE'u izleyen değeriğeler en yakın tam sayıya yuvarlanır ve ardından adı verilmiş bilgilerin yüklenieceği başlangıç adresini (yukarıda 16384) ve bu adrese başlayan yere gönderilecek byte'lan sayısını (6912) verir. Eğer rakam yanlışsa, teypte yüklem hatası raporu gelir. CODE'u eğer sadece bir değeriğeyizse, bu bütün byte'lanı yerleştirilieceği başlangıç adresini tanımlıyordur.

Yukarıdaki örnek ayrıca LOAD SCREENS komutlarıyla da uygulanabilir.

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

Biçimi

LOAD string-expr CODE

[int-num-expr],[int-num-expr]

LOAD DATA

Klavyede yeni

EXTEND CODE

J

Cümle/Emir

LOAD DATA teyipten düzenleri yüklemek için kullanılır. Düzenler SAVE DATA komutlarıyla kaydedilir.

LOAD DATA'ni kullanımı

LOAD DATA, ya bir program cümlesi oluşturmak üzere ya da doğrudan emir olarak kullanılabilir. LOAD'ın önce bir dizin değeriğedir olan dosya adı, sonra da sırayla 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, düzene teypte verilen adresi ve LOAD'la birlikte kulanılan program adlarına uygulanacak kısımlarına tabidir. DATA'ya izleyen harf ya da harfle \$ yüklenip kulanılacağı zaman düzene programı verilecek addır. Uygulamaya konduğunda, Spectrum teypte düzenin adını arar, bulunca ekranda Number array (sayı dizini), Character array (karakter dizini) mesajlarıyla düzenin adı belirlenir ve düzen yüklenir. O sırada bellekte olup da aynı harfle adlandırılmış (yukarıda n veya n\$) başka bir düzen varsa, bu iptal edilir. Bu harf adını ve teypte depolanmış değeriğeler taşıyan yeni bir düzen yaratır. Dikkat edin, düzen değeriğelerinde, o sırada bellekte olan ve aynı harf adını taşıyan herhangi bir dizin değeriğeyi de iptal eder.

Biçimi

LOAD string-expr DATA

letter(\$)

LOAD SCREENS

Klavyede yeni

EXTEND MODE

SYMBOL SHIFT X

Cümle/Emir

TO

Klavyede yeni
SYMBOL SHIFT F

İşlev

TO'nun Sincirler BASIC'te iki değişik kullanımı vardır. FOR'la bağlantılı olarak FOR NEXT döngüleri oluşturmak için (aynınlık için bak FOR) ve dizi bölücü (dizileri daha küçük altdizilere bölmek için) olarak kullanılır.

TO'nun dizi bölücü olarak

kullanımı
TO bir ana dizideki bir alt dizinin ilk ve son karakterinin belirlemesi 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 dizidir. İki sayısal değer (4 ve 7) bu dizi içindeki alt dizinin ilk ve son karakterleridir. Bunun üzerine TO alt dizini (a\$'nin 4'ten 7'ye kadar olan karakterlerini) verir.

Hiç bir şey verilmese ilk sayısal değer 1, son sayısal değer de dizideki son karakterin pozisyonunun değeridir. Eğer alt dizi dizinin ilk karakteri ile başlayacaksa ilk değer verilebilir. Eğer alt dizi dizinin son karakteri ile sona eniyorsa, son değer verilmese de olur.

Bicimi
string-const ([num-expr TO [num-expr])
string-var ([num-expr] TO [num-expr])
(string-expr) ([num-expr] TO [num-expr])

USR User Subroutine

Klavyede yeni
EXTEND MODE L

İşlev

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

USR ve makina kodu
Makina kodunu kullanabilmek için USR'yi bir sayısal değer izler. Örneğin,

80 PRINT USR 65000
100 RANDOMIZE USR 65000

İfade paranteze alınmalıdır. USR'yi izleyen değer bir tam sayıya yuvarlanır ve makina kodunun bellekteki başlangıç adresini oluşturur. USR içeren herhangi bir cümle bu adresdeki altprogramı ister ve USR "bc" kutuk çıktının içiniğini değini verir. Örneğin,

RANDOMIZE USR veya RESTORE

USR sadece altprogramı işleme koyarsın. 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 değimeş ya da değişgen izler ve bunlar POKE cümlesi için bir adres verir. Örneğin,

50 POKE USR "a", 255

USR'yi izleyen dizi değeri, A ile U veya a ile u arasında bir tek harftir. Büyük harflerle küçük harfler arasında bir ayırım guduüzmez.

Bunun üzerine USR, belleğin kullanıcının tanımladığı grafiklere ayrılmış 21 bölümden birinin başlangıç adresini verir. Her bölüm, bir grafik karakter yaratacak şekilde sektiz byte'in POKE edilebileceği sektiz adres içine. Byte'ler ondalık veya ikili biçimde (bak BIN) verilebilir.

Bicimi

USR int-num-const
USR int-num-var
USR (int-num-expr)
USR string-const
USR string-var

VAL VALUE

Klavyede yeni
EXTEND MODE J

İşlev

VAL, sayısal değere sahip bir diziye sayıya çevirir.

VAL'in kullanımı
VAL'ı bir dizi değimeş veya değişgen izler. Örneğin,

70 LET x=VAL a\$

Dizi değimeş veya değişgenin değeri birak işaretilerinden çıkarılır ve bu durumda sayısal bir değer olması gerekir. VAL bunu değerdendirir ve bu bir sayısal değimeş olarak verir.

Örnekler

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

10 INPUT a\$,x
20 PRINT VAL a\$

a\$'ye verilen değeri x'i kullanan bir ifade olmalıdır. "x*x" gibi. Bunun üzerine x'e örneğin 5 gibi bir sayısal değer verilir. VAL, bu ifadeden birak işaretilerini çıkıp x*x elde eder ve x'e verilen değeri değerdendirirerek ekrana sonuç olarak 25'i yansıtır.

Bicimi

string-const
VAL string-var

VAL\$ VALUE (string)

Klavyede yeni
EXTEND MODE SYMBOL SHIFT J

İşlev

VAL\$ bir diziye bir dizi ifade olarak değerdendirir.

VAL\$'in kullanımı
VAL\$'i bir dizi değişgen izler. Örneğin,

130 PRINT VAL\$a\$

Dizi değişgenin değeri, birak işaretilerinden çıkarılır. Bu durumda bunun bir dizi ifade olması gerekir. VAL\$ ifadedi değerdendirir ve değeri bir dizi değimeş 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\$, a\$'nin değerdendirir birak işaretilerini ayıklayıp x\$+x\$'i alır ve x\$'ye verilen değeri değerdendirir, örneğin sonuç olan DODO'yu ekrana yansıtır.

Bicimi

VAL\$ string-var

VERIFY

Klavyede yeni
EXTEND MODE SYMBOL SHIFT R

Emir/Cümle

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

VERIFY'in kullanımı

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

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

0 OK:0:1

mesajı verilir.

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

VERIFY CODE ve VERIFY DATA
VERIFY CODE, tipki LOAD CODE gibi, bellekteki bir bölgenin bölümünün teypte aktarıldığını değerdendirir için kullanılır. VERIFY DATA da LOAD DATA gibi bir düzenden teypte aktarılmış denetlemek için kullanılır. Ayrıntılar için LOAD CODE ve LOAD DATA'ya bakın.

Bicimi

VERIFY string-expr
VERIFY string-expr CODE
[int-num-expr],[int-num-expr]
VERIFY string-expr DATA
letter (\$!)

VERIFY CODE

Klavyede yeni
EXTEND MODE SYMBOL SHIFT R
EXTEND MODE L

Bak VERIFY

VERIFY DATA

EXTEND MODE SYMBOL SHIFT R
EXTEND MODE D

Bak VERIFY

ZX SPECTRUM + EKRAN MESAJLARI

Spectrum, BASIC'i uylarken durursa, ekranın alt kısmında bir mesaj belirir. Bu, ya bir emrin ya da programın tamamlanmışlığı 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, iki noktalı virgülden sonra ise 2, cümle vb olarak gösterilir. CONTINUE, normal olarak programın ekrana gösterilen cümleden yeniden başlamasını sağlar.

0 OK
İşlemin başını ile tamamlanmışlığını veya programa bulunan satır numaralarından daha büyük bir satıra atlama girişimini gösterir. CONTINUE, bu mesajı dikket almaz ve bir önceki mesajda bulunan cümleden yeniden başlar.

1 NEXT without FOR
Kendisini izleyen bir FOR olmayan bir NEXT ile karşılaşılmıştır ve kontrol deşğini ile aynı adı taşıyan bir başka deşğin daha vardır.

2 Variable not found
Bir deęer verilmenden veya teypten deęer yüklemenden bir basit deşğin kullanılmış; veya önce FOR cümlesi ile tanımlanmadan NEXT ile bir kontrol deşğini kullanılmış; veya DIM'le düzenin boyutu tanımlanmadan ya da bir düzen teypten yüklenmeden önce bir altdeęer deşğini kullanılmış demektir.

3 Subscript wrong
Altdeęer, düzenin boyutundan dşında.

4 Out of memory
Cümleli 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 10³⁰'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 deęer verilmiştir.

B Integer out of range
Bir deęer en yakın tam sayıya yuvarlanmış, ancak boyutları dşında olduğu için kabul edilmemiştir.

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

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

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

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

G No room for line
Yeni program satırının gireceęi yeterli bellek yeterli olmadığı için.

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

I FOR without NEXT
Limitleri veya STEP deęeri yanlış olduğu için (örneğin FORx=5 TO 0 STEP 5 kullanıldığında) ve bunu karşılayacak NEXT bulunamadığından bir FOR NEXT döngüsü uygulanmamaktadır.

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 belirlenen deęer, kullanılmı alanı dşındadır.

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 deęer, 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 iletilmek üzere yanlış sayıda deęerler içermektedir veya deęerlerden biri yanlış türdedir (örneğin sayı yerine dizi ya da tersi).

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

BASIC'İNÖTESİ

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

BASIC çok amaçlı bir dil olduğundan, bazı uygulamalarda hantal kalabilir. Göreceli olarak yavaşdır. Diğer diller, daha basit bir programlama ve daha hızlı işleme ile birleştiğinde 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 bekleğinde tutar. LOGO eğitimed kullanılmak üzere geliştirilmiş bir bilgisayar dilidir. Çok esnek bir biçimde kullanılabilecek çok basit emirler içerir. Ancak ZX Spectrum+ 'unuzla 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 gücü Z80A yongası, aslında BASIC'i anlamaz. Belleğin bir kısmını, BASIC çevirmeni denir, sizin BASIC ile verdiğiniz talimatları kodlu sinyallerle dönüştüren bir kılıcı program içerir. Aslında Z80A'nın sizin talimatlarınızı uygulamasını sağlayan bu kodlardır.

Bu çevirmenin sizin BASIC'le verdiğiniz talimatları Z80A koduna ya da yaygın adıyla makina koduna çevirmesi biraz zaman alır. Ancak isterseniz BASIC çevirmeni atlayıp makina kodlarınıza doğrudan Z80A'ya gönderebilirsiniz. Bunun üzerine programınız hızla işleme konur. Bunun için ödemesi gereken bedel, makina kodu ile program yazmak için harcanacak fazladan zamandır. BASIC'ten farklı olarak bu hiç de "doğsta olmayan" bir dildir. Yani bunu öğrenmek epey bir zaman alır. Makina kodu ile programlama bu kitabın boyutları dşındadır. Ancak Spectrum'un makina kodunu çok ileri düzeyde öğreten kitapları vardır. Makina kodunun hazırlanma tadına varabileceğiniz için şu kısa gösteren programını yazıp verin.

ANINDA ÇIĞIRILAR

```
10 FOR X=0 TO 35
20 PER N: POKE 65000+X,N
30 NEXT X
40 DATA 33,255,60,1,1,24,82
50 DATA 35
60 DATA 35,11,120,177,200,114,
70 RNDHNDZ USR 65000
```



50. satırdaki 55'in deęerini 1 ile 255 arasındaki başka herhangi bir rakama çevirin ve çubukların deşğini izleyin. Ayrıca başta bir INK emri kullanarak renkli çubuklar da elde edebilirsiniz. Makina kodu kullanınca ne kadar hızlı görüntü elde edildiğine bakın. Bu, adeta anında olmaktadır.

BASIC 'in ise ekranı doldurması iki saniye sürer. Bu program, 10 ile 30. satırlarla, bellekte 65000 adresinden sonrasında DATA cümlesi ile 16 kod yerleştirildiğinden işlemektedir. 70. satır kodları Z80A'ya göndermek ve derhal görüntü oluşturmaktadır. Dikkat eden 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 yazabilmeniz için "düzenleyici" (assembler) denilen bazı programlar mevcuttur. Makina kodu sadece bazı rakamlar isterken, bunlar size tuşlara basarak vereceğiniz talimatları sağlar. Bu talimatlar, BASIC komutları gibi İngilizce kelimeler değil, bilgisayarın mutlakla yerine getirmesi gereken işlemleri ifade eden bazı kısımlara ya da özel deyimlere, Bu yüzden, "düzenleyici" 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 kelimelerin bazılarını ile bilgisayarla ilgili bazı deyimlerin açıklanmasını bulacaksınız. *İtaliye* yazılmış kelimeler bu sayfalarda ayrıca açıklanmamak. 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 işlevin kullandığı *değer*.

Array (Düzen) Belleğin bir kesiminde bir arada tutulan, birbirine bağlantılı, bir grup veri.

Attributes (Gönderme) Karakterlerin renklerini veren kodlar.

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

Binary code (İkil kod) Bilgisayarların kullandığı bir tür kod. Akımın açık ya da kapalı olması durumundan oluşan bazı öbeklerdir.

Bit (İkil koddan) açık ya da kapalı olma durumu. İkil 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) Ekranı yansıtılan veya yazılan, 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) Bilgisayarı uyguladığı bir tek talimat ve *doğrudan emir*.

Concatenation (Küme) Birbirine ekleyerek dizileri 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 belirler.

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

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

Edit (Düzeltilme) Bir *programdaki* ayrıntıları değiştirme.

Enter (Verme) Tamamlanmış bir talimat veya bilgili 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'in sayısal değeri 0'dir.

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

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

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

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

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

Interface (Araparça) Bilgisayarı ve/veya yan *donanım*ı birbirine bağlayan ve bunları birbirine 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 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 bilgisayar *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 üretilebilecek *komut* veya *karakterlerin* elde edilebilecekleri bes durumundan biri. Program sırasında *modun* hangisi olduğunu *göstergeçte* yanıp sönen harf gösterir.

Nesting (Yuvalama) Programda bir *dönüşüm* 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 işlevi yapılmasını sağlayan talimat.

Output (Çıktı) bilgisayarın sağladığı sonuçlar.

Peripheral (Yan donanım) Bilgisayara bağlı herhangi bir araç.

Pixel Ekranda görünebilecek en küçük renkleri nokta. İngilizce picture cells – resim hücrelerinin kısaltılmışı.

Print (Yaz) Ekranda sonuçları veya *grafik işaretleri* yansıtın ya da yazıcıya yazdırın emir.

Program (Program) Bilgisayarı uygulayacağı bir sıra talimat.

RAM (Random Access Memory) (Rastal Ulaşılan Bellek) Belleğin bir *program*, *veriler* veya diğer değişebilen *değerler* verilebilen kesimi. Ayrıca *değişken* bellek diye de bilinir. Cereyan kesildiği zaman RAM'ın içeriği silinir. ZX Spectrum + 'in 48K 'lık bir RAM'ı 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 verilebileceğini göstergesi.

ROM (Read Only Memory) (Sadecce Okuyucu Bellek) Belleğin kalıcı bazı *programlar* ya da bilgisayar için talimatlar içerir 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 bilgilerin ekranı 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 tırnak işaretleri içine alınmış bir veya birkaç karakterden oluşan grup.

String variable (Bir dizi içerir 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 söylediği bir durum veya sonuç. Doğrunun sayısal değeri 1'dir.

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

Variable (Değişgen) Bilgisayar tarafından kullanılmak üzere belli bir *değişmez* 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.

DİZİN

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

Adlar programı 8
Alçak tanımlı çizimler 26-7
Alf programı 30-1
Anten girişi ve bağlantısı 4-5
Ana çubuğu (tusa) 19
Araparçalar (interface) 45, 46-7
Aritmetik işlemler 22, 22
ATTR 35

Bağlantılar 5
Elektrik 5
Kaselli teyp 13
Televizyon 4

Bağlantısı Mantık Düzeni (ULA) 42
BASIC 18, 49-73
BEEP 36; 18
Bellek 12; 42, 43, 44-8
Bellek 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, 5, 43
CIRCLE 28
Cümleler 22, 50
Çarpım Tablosu programı 23
Çarpışma 34-5
Çerçeve rengi 24-5, 6
Çiğm mozaik programı 10
Çizgele, alçak tanımlı 26, 80
yüksek tanımlı 28, 80

Çizimler ve hareketli görüntüler 34-5
alçak tanımlı 26-27
desenler 30-1
karakter yaratılışı 32-3
rastsal etkiler 30
renk 24-5

sekillerin doldurulması 29; 29
yüksek tanımlı 26, 28-9
Çöğgen 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
DVD girişi 5, 43
Donanım, tanımlı 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
Ekrandaki mesajlar 74
Emirler 22, 50
ENTER 9, 10, 11, 19
EXTEND MODE 8, 21; 18
Genişletilmiş mod 21; 20

FLASH 31
FORTH 26-7, 29, 30, 31, 34
FORTH 75
Girdi-çıkış yolları 45
Ginsler 5
GOTO 23
Göksuay programı 26-7
Göstergeç kontrolleri 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
Hazır Yazılım 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
iki kodlar 44
işlevler 50
işleme koyma, programların 8-9

Kağıt rengi 24-5
Karakter Tablosu 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
Etiketlenmesi 14
Saklanması 12
Sesi 12
Kaselli teyp, amplifikatör olarak 37; 37
Bağlantısı 5-13; 13
Program saklama 38-40
Program yüklenme 14-16
Sayacı 14
Seçimi 12
Ton kontrolü 14, 15, 16
Ton kontrolü 14, 15, 16
Klavye 18-19
Grafik karakterleri 26-7

Modlar 20-1
Komut modu 20; 20
Komutlar 9, 18-19, 50, 52-73, 20-1
Seçimi 19-20
Kullanılan tanımladığı karakterler 80; 32-3
Kuruldu 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
Mikroişlemcisi 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
Noktalamla işaretleri 23, 51
Noktali virgül 23, 51

Oparılır 43
Oyun kolu (joy stick) 45, 47
Parantez 23
Parazit 4
Piramit programı 31
Pixel 28
PLOT 28 28
POKE 48
PRINT 22
Program satırları, numara 21
Düzeltilme 21
Program verme 8-9
Programda değişiklik 9
Programlama 17-40
Programlar
Değiştirme 9
Doğrulama 39
Hataları düzeltme 10
İşleme koyma 8-9, 44
Saklama 13, 38-40
Verme 8-9, 44
Yeniden başlatma 10
yenisine başlama 11
Yüklenme 12-13, 14-15, 14-15

Programların yeniden başlatılması 10
Rakam tuşları 19
Rakamları 50
RAM (Rastсал Kullanım Belleği) 42-48, 42, 45
RAM takımı 4
RAMTOP 48
Rastсал etkiler 9
READ 33
REM 39
Reset düğmesi 11, 12; 5
Resim defteri programı 29
Resimler, çizimi 30-1
Yüksek tanımlı 26-7
RND 26, 30
ROM (Sadace Okuyucu Bellek) 48, 43, 45
ROM kartuşları 12, 47, 46-7
RS232 Araparçası 47; 45

Saklama 13, 38-40
Satırlar 8
İptal etme 21
Uygunluğu 4
Satranç programı 51
SAVE 38-9
Scroll (kayıdırma) 8
Seçme, işaretleri 20
Ses düğmesi, kaselli teyp 12, 14, 15
Ses, efektler 36-7
Sesi yükseltme 37
Sıralama 8-21
Simetrik desenler programı 30
Sinclair BASIC 49-73
Sistem değişimleri 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
Tırmak işaretleri 23, 51
Ton düğmesi, kaselli teyp 12, 14, 15
TRUVEVIDEO 18
Tuşlar 18-19, 18-19
Kullanımı 20-1, 20-1
Tuşları kullanarak yazma 8, 9
TV klonasyonu 42
Voltaj regülatörü 43
Yan bağlantısı 5, 43, 47
Yan donanımlar 45, 46-7
Yanip sonen halkalar programı 9

Yazıcılar 45, 47; 45, 47
Yazılım 12
kullanıma hazır 12-13; 13
türleri 12
uygunluk 12
yüklenme 14-16; 14-16
Yeni programlar 11
Yıldız programı 28
Yıldızlar ve çubuklu programı 11

Yongalar 42-3
yüklenme 13, 14-15; 14-16
yüksek tanımlı çizimler 26; 28-9

Zıplayan top programı 35
280 mikrobilgisayar 43, 75; 45
ZX interface 1, 45, 46-7
ZX Robot programı 27
ZX 16K RAM 4

ilk 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 Modium** ve **ZX Interface Sinclair Research Limited**'in tescilli markalarıdır.

Bütün hakları mahfuzdur. Copyright sahiplerinin yazılı izni önceden alınmadan, 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 fotoğrafı: Vincent Oliver
Yönetim sorumlusu: Alan Buckingham
Diği: The Letter Box Company (Woking) Ltd., Woking, Surrey, England
Reprodüksiyon: A. Mondadori, Verona
Basım ve cilt: A. Mondadori, Verona, İtalya

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

DORLING KINDERSLEY LTD
SINCLAIR RESEARCH LTD'nin
işbirliği ile