

sinclair

ZX Spectrum +
Guide de
l'utilisateur



LE LOGICIEL DU SPECTRUM

La gamme complète de logiciel destinée aux ordinateurs Spectrum (y compris tous les titres existants) est entièrement compatible avec votre nouveau ZX Spectrum +.

PRESENTATION DU ZX SPECTRUM +

Sinclair Research a longtemps été en avance en ce qui concerne la technologie des microchips et c'est ce qui a mis l'informatique à la portée de tous. Après l'arrivée du micro-ordinateur le moins cher du monde, le ZX 80, nous avons conjugué un pouvoir encore plus important à une valeur encore plus grande pour ces successeurs, le ZX 81, le ZX Spectrum et les ordinateurs QL. La facilité à l'emploi a été aussi notre mot d'ordre à la fois pour la conception de ces produits et pour leurs façons d'opérer.

Le ZX Spectrum + a amené Sinclair Research à faire un pas supplémentaire dans cette voie. En lui, vous avez une machine avec toutes les meilleures caractéristiques du Spectrum dans une version encore améliorée qui fait que ce micro-ordinateur, déjà le plus puissant et le plus connu du monde, est encore plus facile à utiliser. Nous espérons que vous profiterez pleinement des nombreuses possibilités que votre nouvel ordinateur vous apportera.

Chris Smith

SOMMAIRE

VOYEZ PAR VOUS-MÊME 3

ESSAYEZ DE PROGRAMMER 17

**FAMILIARISEZ-VOUS AVEC VOTRE
ZX SPECTRUM + 41**

**FAMILIARISEZ-VOUS AVEC LE
SINCLAIR BASIC 49**

Écrit par Neil Ardley
Publié par Dorling Kindersley Ltd
en association avec
Sinclair Research Ltd.

COMMENT UTILISER VOTRE MANUEL

Le guide de votre ZX Spectrum + contient quatre chapitres séparés par un onglet de couleur différente. Pour l'ouvrir à un chapitre, il vous suffit de choisir l'onglet de couleur correspondante.

1 VOYEZ PAR VOUS-MEME

Comment monter votre ZX Spectrum + ■ Réglage de votre TV ■ Montage du dépanneur ■ Ce que votre ZX Spectrum + peut faire ■ Comment utiliser le logiciel prêt à être exécuté ■ Comment charger un programme ■ Chargement du logiciel dépannage

2 COMMENCEZ A PROGRAMMER

Le clavier — le panneau de commande de votre ordinateur ■ Comment utiliser les touches ■ La calculatrice de la télévision ■ La couleur et sa méthode d'utilisation ■ Graphiques simples à réaliser vous-même ■ Le carnet de croquis sur écran ■ Dessinez vos propres motifs et images ■ Comment créer vos propres caractères ■ Animation ■ Comment produire des effets musicaux et sonores ■ Sauvegarder vos propres programmes ■ Sauvegarde du logiciel

3 FAMILIARISEZ-VOUS AVEC VOTRE ZX SPECTRUM +

Que contient-il? ■ Comment votre Spectrum fonctionne-t-il? ■ Comment raccorder les périphériques ■ Plan de la mémoire du ZX Spectrum +

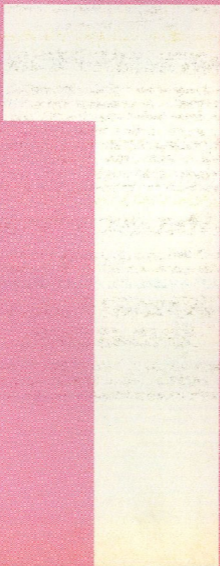
4 FAMILIARISEZ-VOUS AVEC LE SINCLAIR BASIC

Guide des mots-clés du Sinclair BASIC à l'intention du programmeur ■ Etats de l'écran Spectrum ■ Au-delà du BASIC ■ Le jargon informatique — sa signification

VOYEZ PAR VOUS-MEME

Ce chapitre vous explique quelles sont les possibilités de votre ZX Spectrum +. Suivez les instructions qu'il contient pour installer votre ordinateur qui sera alors prêt à fonctionner. A ce moment-là, vous avez le choix.

Par l'intermédiaire du clavier, vous pouvez introduire plusieurs programmes qui mettent le Spectrum à l'épreuve et vous permettent de découvrir ses possibilités — couleurs, graphiques et son — ou bien vous pouvez apprendre à vous servir de programmes prêts à l'emploi comme certains jeux. Quel que soit votre choix, vous ne tarderez pas à vous amuser avec votre nouvel ordinateur.



MONTER VOTRE ZX SPECTRUM +

Avant de monter votre Spectrum, lisez la liste de contrôle ci-dessous pour vous assurer que vous avez tout ce dont vous avez besoin. Pour connecter et brancher

l'ordinateur, suivez les instructions de la page d'en face.

Enfoncez solidement les prises. Si le Spectrum est débranché par accident, vous perdez votre programme, ainsi que toutes les informations et les résultats qu'il contient.

Lorsque vous avez fini d'utiliser l'ordinateur, coupez le courant à la prise murale si cette dernière est munie d'un interrupteur et débranchez la prise secteur.

Liste de contrôle: Avez-vous tout ce dont vous avez besoin?

En déballant votre ZX Spectrum + vous avez trouvé:

- 1 Le ZX Spectrum +
- 2 Une alimentation électrique fournissant au Spectrum l'alimentation 9 V DC dont il a besoin.
- 3 Un câble d'antenne pour connecter le Spectrum à un téléviseur.
- 4 Un câble pour connecter le Spectrum à un magnétophone à cassette.
- 5 Un bon de garantie à remplir et à nous renvoyer.
- 6 Le guide de l'utilisateur sur cassette.
- 7 Ce manuel.

Vous aurez également besoin de:

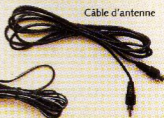
- 1 Un téléviseur
- 2 Un magnétophone à cassette
- 3 Une prise secteur.



Alimentation du ZX



Cordon du magnétophone



Câble d'antenne

Questions et réponses

AI-JE BESOIN D'UN TÉLÉVISEUR EN COULEUR?

Non. Toutefois, sur un téléviseur noir et blanc, vous ne pourrez pas voir les couleurs produites par le Spectrum.

N'IMPORTE QUEL TÉLÉVISEUR FERA-T-IL L'AFFAIRE?

Votre Spectrum devrait produire une image sur n'importe quel téléviseur. Si vous n'avez pas d'image, votre ordinateur et votre téléviseur sont sans doute munis de deux systèmes différents. Il se peut que votre téléviseur soit un modèle trop ancien ou que votre Spectrum et votre téléviseur aient été fabriqués dans des pays différents. En cas de doute, renseignez-vous auprès de votre distributeur de téléviseurs.

PUIS-JE UTILISER UN ÉCRAN DE CONTRÔLE AU LIEU D'UN TÉLÉVISEUR?

Oui. Votre distributeur pourra éventuellement vous fournir un écran de contrôle donnant une image de qualité supérieure.

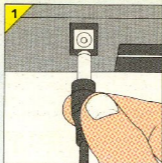
PUIS-JE UTILISER LE ZX 16K RAM?

Non. Le module de mémoire RAM ne peut être utilisé qu'avec l'ordinateur Sinclair ZX81.

Branchez votre ZX Spectrum +

En premier lieu, fixez une prise secteur aux deux extrémités dénudées du cordon d'alimentation. Ensuite, vous munirez la prise d'un fusible de 3A. Notez qu'il n'est pas nécessaire de raccorder votre Spectrum à la terre même s'il est muni d'une prise secteur à trois broches.

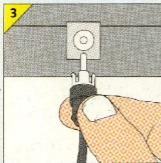
Vous trouverez ci-dessous une série d'illustrations vous montrant comme connecter votre Spectrum au secteur et à votre téléviseur. Une fois que le système aura été branché, tournez la page pour voir comment le régler.



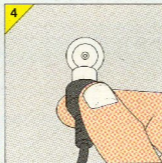
1
Branchez la petite prise du cordon d'alimentation dans l'embase 9 V DC de votre Spectrum.



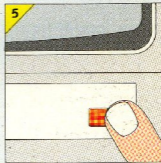
2
Branchez la prise secteur et mettez sous tension s'il y a un interrupteur. Votre Spectrum n'a pas son propre interrupteur marche/arrêt.



3
Introduisez le câble d'antenne dans l'embase TV de votre Spectrum. Seule une des prises du câble d'antenne s'adaptera à l'embase.



4
Détachez le câble d'antenne du téléviseur. Introduisez l'autre prise du câble d'antenne du Spectrum dans l'embase d'antenne du téléviseur.



5
Allumez le téléviseur et mettez son volume à zéro. Vous pouvez maintenant régler le téléviseur de façon à ce qu'il puisse recevoir le signal du Spectrum.

Les prises et les connecteurs du Spectrum

Prise de courant
Le courant 9 V DC produit par le transformateur ZX est connecté par l'intermédiaire de cette prise.



Connecteur latéral
Diverses machines, y compris des micro-disque, des imprimantes et des modems peuvent y être connectés.



EMBASE EAR
La prise des écouteurs d'un magnétophone à cassette est raccordée à cette prise.

EMBASE MIC
La prise du microphone d'un magnétophone à cassette est raccordée à cette prise.

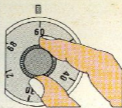
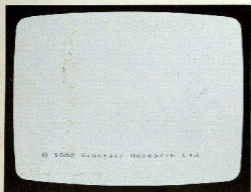
EMBASE TV
La prise d'antenne d'un téléviseur est raccordée à cette prise.



REGLAGE TELEVISEUR

Votre Spectrum émet un signal vidéo de télévision en couleur à la fréquence du canal 36 dans la bande UHF: votre téléviseur doit donc être réglé sur ce canal pour que l'image de l'image de l'ordinateur apparaisse sur son écran.

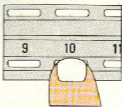
Une fois votre Spectrum branché et connecté au téléviseur, agissez sur la commande de réglage du téléviseur jusqu'à ce que vous obteniez le message de copyright Sinclair comme sur le premier écran ci-dessous. Si vous ne parvenez pas à obtenir le message de copyright ou si les couleurs sont bizarres, lisez les instructions ci-contre.



Commandes de réglage

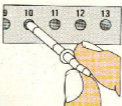
Réglage variable

Une commande de réglage permettant de électionner n'importe quel canal. Faites tourner le bouton jusqu'à ce que vous obteniez le message de copyright.



Poussoir de réglage

Sélectionnez un bouton de réglage utilisé pour les calculs et manœuvrez-le jusqu'à ce que le message de copyright apparaisse. Utilisez si possible un bouton de réserve. Vous n'aurez ainsi pas besoin de régler votre téléviseur chaque fois que vous utilisez votre Spectrum.



Réglage électronique

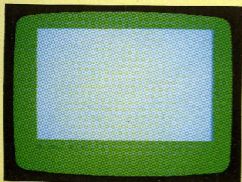
Avec ce système téléviseur se règle automatiquement dans le canal voulu. Le téléviseur avec sélection de canaux synthétisée mais sans option de réglage manuel, peut ne pas pouvoir être utilisé avec cet ordinateur.

Comment essayer les couleurs du Spectrum

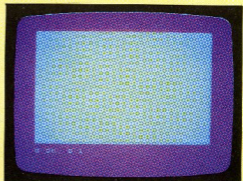
Pour essayer les couleurs du Spectrum, il vous suffit d'appuyer sur la touche B, puis sur l'une des touches numérotées de 1 à 6. Le message de copyright disparaît; le mot BORDER apparaît en premier lieu, suivi du numéro. Appuyez alors sur la touche ENTER.

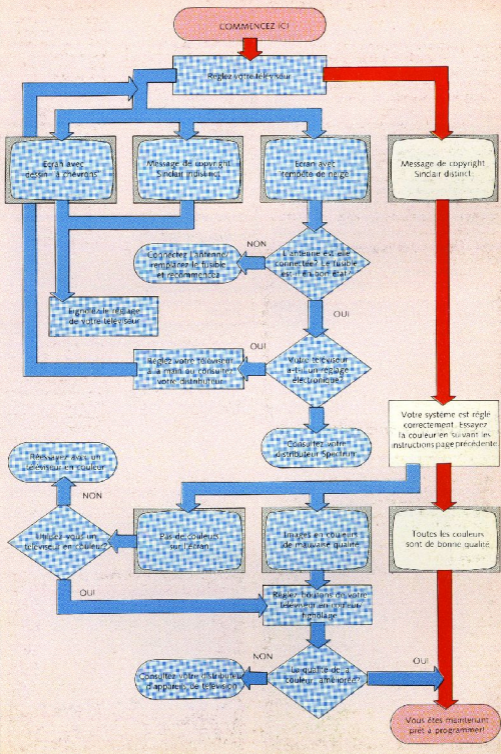
La "bordure" de l'écran devrait prendre la couleur indiquée sur la touche numérotée. Les écrans ci-dessous illustrent ce qui se passe lorsque vous appuyez sur BORDER 4 et ENTER, puis sur BORDER 3 et ENTER. Si vous appuyez sur BORDER 7, la bordure redevient blanche.

BORDER 4



BORDER 3





CE QUE VOTRE ZX SPECTRUM + PEUT FAIRE

Faites des essais

Maintenant que votre Spectrum est branché et que votre téléviseur est réglé, essayez d'appuyer sur quelques touches. Des mots et des lettres apparaîtront sur l'écran et peut-être aussi quelques chiffres.

Toutefois, à moins que vous ne sachiez comment programmer le Spectrum, l'ordinateur ne réagira probablement pas. Mais ne vous en faites pas, vous pouvez enfoncer les touches de l'ordinateur sans crainte de l'abîmer.

Appuyez maintenant sur le bouton 'reset' (remise à l'état initial) se trouvant sur le côté gauche de l'ordinateur et vous êtes prêt à vous servir de votre Spectrum.

Comment utiliser le clavier

Pour introduire par l'intermédiaire du clavier un mot, une lettre ou un chiffre, commencez par repérer leur position sur les touches. Appuyez ensuite sur les touches de sélection dans l'ordre indiqué ici.

Mot-clé supérieur
Appuyez sur EXTEND MODE puis sur la touche.



Mot-clé ou signe inférieurs
Appuyez sur EXTEND MODE et maintenez SYMBOL SHIFT enfoncée; ensuite appuyez sur la touche.



Mot-clé supérieur (section soulevée). Appuyez sur la touche.

Lettre ou chiffre (section soulevée). Appuyez sur la touche. Pour les majuscules, maintenez CAPS SHIFT enfoncée.



Mot-clé ou signe inférieurs (section soulevée). Maintenez SYMBOL SHIFT enfoncée et appuyez sur la touche.

La façon d'utiliser les touches est expliquée en détails aux pages 20-21.

Ensuite, programmez votre Spectrum

Votre Spectrum peut faire des tas de choses. Mais pour qu'il fonctionne, vous devez lui donner une série d'instructions appelée programme machine.

Voici une série de programmes courts qui mettront à l'épreuve votre Spectrum et vous permettront de vous rendre compte de ses possibilités — couleurs, son et graphiques. Il vous suffit d'introduire par l'intermédiaire du clavier les instructions du programme *telles* qu'elles apparaissent ici. Les images de l'écran vous indiquent ce à quoi vous devez vous attendre, mais si vous lisez le paragraphe intitulé 'Comment modifier un programme', vous pourrez faire vos propres expériences.

Comment entrer et exécuter un programme

A chaque série d'instructions correspond une liste. Vous verrez que les listes du programme contiennent plusieurs sections numérotées 10, 200, etc. Dans le programme, chacune de ces sections est appelée une ligne.

NOMS

```
10 BORDER 1, INK RND*7
20 PAPER RND*7
30 PRINT "ZX Spectrum +";
40 GO TO 10
```



Le nom ZX Spectrum + apparaît en plusieurs couleurs à travers l'écran. L'ordinateur s'arrête alors et un message, scroll?, apparaît en bas de l'écran. Pour faire remonter l'image, appuyez sur n'importe quelle touche sauf N, SPACE et STOP. Si vous arrêtez le déplacement et appuyez sur BREAK, puis sur R (RUN) et sur ENTER, les noms apparaîtront dans des couleurs différentes.

Faites l'essai suivant:

A la ligne 30, remplacez 'ZX Spectrum +' par votre nom entre guillemets (") — par exemple

```
30 PRINT "John";
```

N'oubliez pas d'ajouter le point-virgule(,).

Dans chacune des lignes d'un programme, vous trouverez des mots complets ou des abréviations contenant deux ou plusieurs lettres, par exemple PRINT, LET, RND, PI, PAPER ou GOTO. Ce sont des *mots-clés* et il n'est pas nécessaire de les épeler pour les entrer. Il suffit de trouver la touche où figure le mot-clé (par exemple, PRINT se trouve sur la touche P) et de suivre les instructions du paragraphe 'Comment utiliser de clavier'.

Lorsque vous introduisez une ligne par l'intermédiaire du clavier, cette dernière apparaît en bas de l'écran. Lorsque vous arrivez à la fin de la ligne d'un programme, appuyez sur la touche ENTER et la ligne apparaît alors en haut de l'écran. Introduisez et entrez chaque ligne de la même façon. Si vous vous trompez de touche, lisez le paragraphe de la page suivante, intitulé 'Comment corriger les erreurs'.

Une fois toutes les lignes entrées, appuyez sur R. Le mot-clé RUN apparaît. Appuyez ensuite sur ENTER et votre Spectrum se met en marche alors que le programme se déroule.

FORMES

```
10 LET S$=""
20 FOR X=1 TO 7
30 LET S$=S$+CHR$(RND*14+129)
40 NEXT X
50 INK RND*7
60 BORDER RND*7
70 PRINT S$;
80 GO TO 20
```



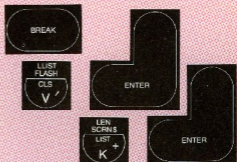
Lorsque vous exécutez le programme, des formes géométriques en couleur apparaissent sur l'écran. Une fois l'écran rempli, le message scroll? est affiché. Pour revoir l'image, appuyez sur n'importe quelle touche (sauf N, SPACE, BREAK et STOP) et la forme remonte. Pour voir une nouvelle forme dans une combinaison de couleurs différente, appuyez sur N lorsque le message scroll? apparaît. Ensuite, appuyez sur BREAK, puis sur R (RUN) et finalement sur ENTER.

Faites l'essai suivant

A la ligne 2, remplacez 7 par un autre chiffre pour obtenir une autre forme. Essayez 8 par exemple.

Comment modifier un programme

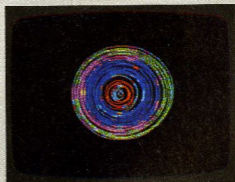
Attendez que le programme soit terminé ou bien arrêtez-le en appuyant sur BREAK. Ensuite, appuyez sur V (CLS), puis sur ENTER, puis sur K (LIST), puis sur ENTER. La liste du programme (liste des lignes) apparaît sur l'écran.



Trouvez la ligne que vous voulez changer et introduisez la nouvelle ligne au complet, y compris son numéro par l'intermédiaire du clavier, appuyez sur ENTER. La nouvelle ligne remplacera l'ancienne. Appuyez sur R (RUN) et sur ENTER et le nouveau programme se mettra en route.

CERCLES CLIGNOTANTS

```
10 BORDER 0; PAPER 0; CLS
20 CIRCLE INK RND*6; FLASH RND
;120+RND*5;RND*5;RND*5;RND*5
30 BEEP 0;1;RND*50
40 IF RND<.9 THEN GO TO 50
50 GO TO 20
60 FOR Y=2 TO 4
70 FOR X=0 TO 6
80 BORDER X
90 BEEP .05;X+Y
100 NEXT X
110 NEXT Y
120 RUN
```



Une série de cercles clignotants presque concentriques se forme sur l'écran. Puis tout à coup, la bordure se met à clignoter, l'ordinateur émet un son perlé et une nouvelle série de cercles apparaît.

Faites l'essai suivant

Avant de lister le programme, introduisez par l'intermédiaire du clavier PAPER 7 et appuyez sur ENTER. Ensuite, à l'aide de la touche K, réintroduisez la ligne 20 manquant sur les deux mots-clés FLASH RND et les cercles s'arrêteront de clignoter.

MOSAÏQUE FARFELUE

```

5 BORDER 0: CLS
10 LET H=15: LET V=11
15 LET I=INT (RND*3-1): LET V=
20 INT (RND*3-1)
30 INK RND*7
40 FOR Z=1 TO 20
50 PRINT AT V,I:CHR$ 143
60 LET H=H+4
70 LET V=V+4
80 IF H=0 THEN LET H=31
90 IF V=0 THEN LET V=30
100 IF H=31 THEN LET V=0
110 IF V=30 THEN LET H=0
120 NEXT Z
130 GO TO 20
130

```



Un carré de couleur zigzague sur l'écran, traçant une forme colorée. Chaque fois que vous remettez en marche le programme, une forme différente est produite.

Faites l'essai suivant

A la ligne 50, remplacez 143 par 42 et vous verrez des étoiles! Essayez d'autres numéros, de 33 à 142. Consultez le tableau des jeux de caractères de la page 51 pour savoir ce qui va se passer.

Comment remettre en marche un programme

Certains des programmes comme STARS AND STRIPES (BANNIÈRE ÉTOILÉE) se terminent et produisent l'état OK ainsi que le numéro de la dernière ligne du programme. Ceci signifie que le programme est complètement terminé. Pour le remettre en marche, il suffit d'appuyer sur R (RUN) et sur ENTER.

D'autres programmes continuent à se dérouler — MANIC MOSAÏC (MOSAÏQUE FARFELUE) par exemple — ou se remettent en marche automatiquement — SHIMMERING SUNRISE (REFLETS DE SOLEIL) par exemple. Pour les arrêter,



appuyez sur BREAK.

Maintenez cette touche enfoncée jusqu'à ce que le programme s'arrête et que l'état BREAK apparaisse. Pour le remettre en marche, il suffit d'appuyer sur R (RUN) et sur ENTER.

Comment corriger les erreurs

Si vous vous êtes trompé de touche ou si vous n'avez pas appuyé convenablement sur les touches shift ou EXTEND MODE, ne vous en faites pas. Appuyez sur la touche DELETE et le dernier mot-clé, le dernier signe, la dernière lettre ou le dernier numéro disparaîtront. Pour continuer à effacer, maintenez la



touche DELETE enfoncée.

Si vous avez fait une erreur après avoir appuyé sur ENTER, il se peut qu'un point d'interrogation clignotant apparaisse dans la ligne. C'est là que se trouve l'erreur. Appuyez sur DELETE pour effacer la ligne jusqu'au point d'interrogation, puis complétez la ligne correctement et appuyez sur ENTER.

Si vous réussissez à entrer une ligne incorrecte, il se peut que le programme s'arrête et produise un état qui donne le numéro de la ligne erronée en bas de l'écran. Introduisez la ligne toute entière par l'intermédiaire du clavier, puis appuyez sur ENTER, R (RUN) et ENTER. Tout doit maintenant être rentré dans l'ordre.

POLYHEDRA (POLYHEDRES)

```

1 BORDER 1: PAPER 0: CLS
10 INPUT N
20 FOR I=20 TO 80 STEP 2
30 LET X=120: LET Y=87
40 LET H1=V1: LET V1=4
50 PLOT H1,V1
60 FOR A=0 TO 361 STEP 360/N
70 LET H2=X+Y+COS (A*PI/180)
80 LET V2=Y+C*PI/180
90 LET H1=H2: LET V1=V2
100 BEEP 0:02:R-20
110 NEXT A
120 NEXT I

```



Au départ, l'écran est blanc. Appuyez sur la touche 6, puis sur ENTER. Une forme à 6 côtés apparaît. Lorsque le programme est terminé, remettez-le en marche et appuyez sur une touche portant un autre numéro pour obtenir une forme ayant un nombre de côtés différent.

Faites l'essai suivant

A la ligne 20, remplacez 2 par un autre chiffre. Plus le chiffre est élevé, plus la forme apparaît rapidement et plus la distance entre les polyèdres (figures à côtés multiples) est grande.

BANNIERE ETOILEE

```

10 INK 2
20 PAPER 7
30 CLS
40 FOR X=20 TO 140 STEP 20
50 FOR Z=0 TO 11
55 PLOT 16,Z+X: DRAW 216,0
60 NEXT Z
70 NEXT X
80 PLOT 16,28: DRAW 0,131
90 PLOT 232,28: DRAW 0,131
110 PAPER 1
120 INK 7
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,"* * * * *"
```



Le drapeau des Etats-Unis apparaît sur l'écran.

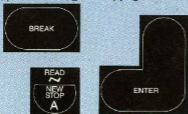
Faites l'essai suivant

Changez le numéro des couleurs du drapeau. La couleur des bandes est à la ligne 10, celle des étoiles à la ligne 120 et celle qui sert de fond aux étoiles à la ligne 110.

Comment commencer un nouveau programme

Lorsque vous n'avez plus besoin d'un programme et que vous voulez entrer un programme complètement différent, attendez que le premier se termine ou arrêtez-le en appuyant sur BREAK.

Il y a deux façons d'effacer l'ancien programme de la mémoire de l'ordinateur. Soit en appuyant sur deux touches, A (NEW) et ENTER. L'écran sera noir pendant quelques instants, puis le message de copyright



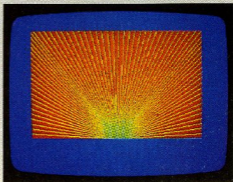
apparaîtra.

Soit en appuyant sur le bouton "reset" (remise à l'état initial), ce qui est la solution la plus facile. L'effet est identique à celui obtenu lorsque vous débranchez le Spectrum.

REFLETS DE SOLEIL

```

10 BORDER RND*6
20 INK RND*7
30 PAPER RND*6
40 CLS
50 LET Z=RND*10+2
60 FOR X=0 TO 174 STEP Z
70 PLOT 120,0
80 DRAW -128,X
90 BEEP .01,X/3
100 NEXT X
110 FOR X=-127 TO 127 STEP Z
120 PLOT 120,0
130 DRAW X,175
140 BEEP .01,50
150 NEXT X
160 FOR X=174 TO 0 STEP -Z
170 PLOT 120,0
180 DRAW 127,X
190 BEEP .01,X/3
200 NEXT X
210 PAUSE 200
220 GO TO 10
```



Toutes les quelques secondes, une image de couleur différente ressemblant aux reflets du soleil apparaît sur l'écran. Si l'écran se vide, patientez, le soleil ne tardera pas à réapparaître.

Faites l'essai suivant

A la ligne 210, remplacer 200 par un autre chiffre pour modifier la période pendant laquelle le soleil reste sur l'écran. 200 correspond à 4 secondes.

Et ensuite?

Vous avez maintenant le choix. Si vous souhaitez conserver l'un des programmes pour le repasser plus tard, vous pouvez l'enregistrer sur bande magnétique. La façon de procéder est expliquée à la page 38.

Si vous souhaitez continuer à faire des essais, lisez le chapitre 2 intitulé "Commencez à programmer". Jusqu'à présent, vous n'avez fait qu'essayer les programmes sans nécessairement comprendre comment ils fonctionnent. Le chapitre 2 vous expliquera certaines des caractéristiques de la programmation du Spectrum.

Si vous voulez essayer des bandes portant les programmes, lisez le paragraphe intitulé "Comment utiliser le logiciel prêt à être exécuté".

COMMENT UTILISER LE LOGICIEL PRET A ETRE EXECUTE

Lorsque vous entrez un programme dans le Spectrum, vous produisez, en appuyant sur les touches, une série de signaux codés électroniquement. Les codes sont stockés dans la mémoire du Spectrum de façon à ce que l'ordinateur puisse les utiliser lorsque le programme est exécuté. Les codes restent dans la mémoire jusqu'à ce que vous les enleviez (en entrant NEW ou en appuyant sur le bouton "reset" par exemple) ou que vous débranchiez le Spectrum.

Toutefois, lorsque vous souhaitez utiliser votre Spectrum, il n'est pas toujours nécessaire d'introduire un programme par l'intermédiaire du clavier. Vous pouvez acheter du logiciel prêt à être exécuté, contenant des programmes qui peuvent être introduits directement et automatiquement dans l'ordinateur. Non seulement le logiciel prêt à l'emploi vous évite de devoir introduire un programme par l'intermédiaire du clavier chaque fois que vous voulez utiliser votre Spectrum, mais encore il vous permet de disposer d'une bibliothèque de programmes prêts à l'emploi qu'il vous faudrait plusieurs jours ou même plusieurs semaines pour écrire vous-même. Les fabricants de logiciel mettent au point toutes sortes de programmes écrits par des programmeurs extrêmement compétents et il existe un large éventail de programmes convenant au Spectrum. Jetez un coup d'oeil au catalogue Sinclair Spectrum pour vous faire une idée du genre de programmes que vous aimeriez passer.

Comment charger les programmes dans le Spectrum

Les signaux codés d'une bande portant les programmes sont des bip-bip aigus et graves enregistrés au rythme d'environ 1.500 par seconde. Lorsque vous repassez une bande portant les programmes dans un magnétophone à cassette, ce dernier émet la série de bip-bip qui constitue le programme. Il vous suffit de connecter le magnétophone à cassette à votre Spectrum pour les codes soient directement stockés dans la mémoire de ce dernier. C'est ce que l'on appelle *charger* un programme.

Ces deux pages vous expliquent comment connecter votre magnétophone à cassette.

Le logiciel: Questions et réponses

Que veut dire "logiciel"?

Logiciel est le nom général donné aux programmes introduits dans des ordinateurs et leur permettant de fonctionner. Le "hardware" désigne l'ensemble du matériel — les ordinateurs eux-mêmes et n'importe quel autre dispositif.

Pourquoi le logiciel est-il produit sur des bandes magnétiques?

Les bandes magnétiques sont faciles à utiliser et n'exigent pas d'équipement spécial. Un magnétophone à cassette bon marché est tout ce dont vous avez besoin pour charger ce genre de logiciel.

Quel genre de sons les programmes enregistrés sur bande produisent-ils?

Passer une bande sur votre magnétophone sans le connecter au Spectrum. Vous entendrez un son aigu qui s'explique par le fait que les signaux codés passent dans le haut-parleur du magnétophone au lieu de l'ordinateur. Les signaux passent tellement rapidement de la cassette au Spectrum qu'il est impossible de distinguer les sons.

Existe-t-il d'autres sortes de logiciel?

Oui. Vous pouvez vous procurer des programmes sur cartouche ROM au lieu de bandes magnétiques. Les cartouches s'introduisent dans une interface qui se place à l'arrière de votre Spectrum. Un programme sur cartouche ROM est chargé instantanément, sans le moindre délai.

Il existe également du logiciel sur cartouches à micro-disque. Ces cartouches contiennent des programmes enregistrés sous forme magnétique, tout comme une bande magnétique. Une cartouche peut contenir plusieurs programmes et à la différence des bandes magnétiques, un programme peut être chargé en quelques secondes au lieu de quelques minutes. Les cartouches à micro-disque sont utilisées avec un dispositif à micro-entraînement (voir page 46).

Quel est le meilleur magnétophone à cassette?

Le Spectrum se contente d'un magnétophone à cassette portatif bon marché, fonctionnant de préférence sur le secteur plutôt que sur piles. Il doit être muni d'un bouton de réglage du volume, mais un bouton de réglage de la tonalité n'est pas essentiel. Il existe également des magnétophones à cassettes spéciaux pour ordinateurs. Ils offrent plus de garanties que les magnétophones ordinaires en ce qui concerne le stockage et le chargement des programmes.

Les programmes enregistrés sur bande ont-ils besoin d'un entretien spécial?

Comme n'importe quelle forme d'enregistrement magnétique, les programmes sur cassette peuvent subir l'influence de puissants champs magnétiques. Evitez donc de ranger vos cassettes à proximité d'un appareil utilisant un fort courant électrique. Vos cassettes doivent également être protégées contre la poussière.

N'importe quelle sorte de logiciel fera-t-il l'affaire?

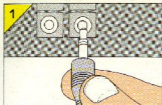
Non. Vous ne pouvez charger que le logiciel mis au point pour le ZX Spectrum et le ZX Spectrum +.

Comment connecter votre magnétophone à cassette

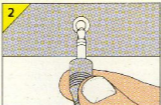
Le cordon livré avec votre Spectrum est prévu pour être connecté à votre magnétophone à cassette. C'est le cordon muni de deux petits jacks à chaque extrémité. Placez le magnétophone à côté du Spectrum et branchez le cordon comme illustré.

Le magnétophone et le Spectrum peuvent être en marche ou non pendant l'opération mais il vaut mieux enlever la cassette du magnétophone avant de le mettre en marche ou de l'arrêter. Ceci sauvegardera les programmes qui y sont stockés.

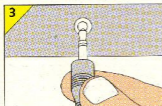
Faire les branchements correctes



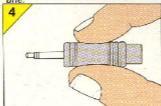
1 Introduire l'un des quatre jacks dans l'embase EAR à l'arrière de votre Spectrum.



2 Introduire l'autre jack de la même couleur dans l'embase EAR du magnétophone si ce dernier en a une.



3 Si le magnétophone n'a pas d'embase EAR, connectez le jack à une embase d'écouteur s'il y en a une. Autrement, essayez de le connecter à une embase de haut-parleur extérieure.



4 Si le jack du cordon ne s'adapte pas dans l'embase du magnétophone à cassette, vous aurez besoin d'un adaptateur ou d'un cordon spécial munis des jacks corrects en vente dans les magasins d'appareils électro-ménagers. L'embase EAR du Spectrum exige un jack de 3,5mm et un signal d'entrée d'environ 1 volt.

EMBASES EAR ET MIC

Lors du chargement des programmes, les embases EAR et MIC peuvent toutes deux être connectées, comme illustré ici. Mais lorsque vous sauvegardez des programmes (voir page 38), le cordon EAR doit être déconnecté.

Quelques conseils pour le logiciel

■ Le cordon du Spectrum a des jacks codés par des couleurs pour éviter des connections entre croisées entre les embases de l'ordinateur et le magnétophone. Quand vous utilisez le magnétophone avec votre Spectrum, essayez de charger toujours le même système, avec une couleur pour les embases EAR et l'autre couleur pour les embases MIC.

■ Quelques magnétophones peuvent être affectés par d'autres appareils électriques placés à côté. Cela peut quelquefois modifier les signaux envoyés entre l'ordinateur et le magnétophone, et empêcher ainsi les programmes d'être chargés correctement. Si votre magnétophone semble ne pas fonctionner occasionnellement, essayez de le bouger de façon à ce qu'il ne se trouve pas le long du téléviseur ou de l'ordinateur.



COMMENT CHARGER UN PROGRAMME

Maintenant que vous avez connecté un magnétophone à cassette à votre Spectrum, vous êtes prêts à charger et à passer un programme. Vous pouvez utiliser une bande portant des programmes prêts à être exécutés ou votre propre bande contenant vos programmes. La procédure est identique dans les deux cas.

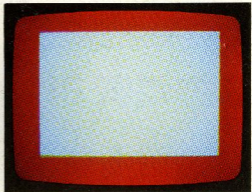
- 1 Introduire la cassette et la rembobiner jusqu'à son début.
- 2 Réglez le volume et la tonalité du magnétophone à cassette au niveau voulu. Le volume peut être réglé à deux tiers du maximum et s'il y a un bouton de réglage de la tonalité, mettez-le sur la tonalité aiguë maximum.
- 3 Appuyez sur J et LOAD doit apparaître sur l'écran. Ensuite, introduisez par l'intermédiaire du clavier le nom du programme entre guillemets, par exemple
LOAD "Prog 1".



Mettez le magnétophone à cassette en marche. Assurez-vous que le Spectrum est branché, puis introduisez la cassette dans le magnétophone. S'il y a déjà un programme dans l'ordinateur, attendez qu'il se termine ou arrêtez-le. Vous pouvez alors entrer NEW ou appuyer sur le bouton qui enlève le programme de la mémoire du Spectrum, mais ceci n'est pas essentiel car en chargeant un nouveau programme, vous remettez la mémoire à zéro.

Suivez maintenant les instructions numérotées. En cas de problème, consultez le paragraphe de la page 16 intitulé 'Chargement du logiciel dépannage'.

- 4 Appuyez sur ENTER. L'écran se videra.
- 5 Mettez la bande en route. La bordure de l'écran devrait devenir bleue ou rouge ou bien bleue et rouge en alternance. Ceci indique que le Spectrum est à la recherche d'un programme.



- 6 Quelques secondes plus tard, des bandes bleues et rouges montent et descendent dans la bordure. Ceci indique que le Spectrum a commencé à recevoir un signal.

Chargement du logiciel: quelques petits trucs

Les quelques petits trucs qui suivent vous feront gagner du temps.

- 1 Munissez vos bandes d'une étiquette de façon à pouvoir trouver les programmes facilement. Si une bande contient plus d'un programme, inscrivez le nom de chaque programme par ordre sur l'étiquette. N'oubliez pas que le nom inscrit sur l'étiquette doit être orthographié de la même façon que le nom utilisé par l'ordinateur.

1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6

- 2 Si votre magnétophone à cassette a un compteur, servez-vous en pour localiser rapidement un programme sur une bande contenant plus d'un

programme par côté. Mettez le compteur à zéro au début de la bande, puis entrez LOAD, suivi des noms de programme (entre guillemets) qui ne se trouvent pas sur la bande. Mettez la bande en marche et le Spectrum nommera les programmes qu'il rencontre sans les charger. Inscrivez le chiffre indiqué sur le compteur à côté du nom des programmes figurant sur l'étiquette. Par la suite, vous pourrez facilement et rapidement trouver le programme que vous cherchez.

7 Le mot Program; suivi du nom du programme ou Bytes; suivis d'un nom ou d'une lettre, apparaissent sur l'écran lorsque vous appuyez sur la touche BREAK. Ceci indique que l'ordinateur a réussi à localiser le programme.

10 Les opérations 7, 8 et 9 peuvent être recommencées une ou plusieurs fois si le programme est divisé en plusieurs sections:

11 Une fois chargé, le programme peut se mettre en route automatiquement. N'oubliez pas d'arrêter la bande.

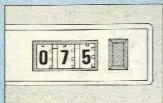
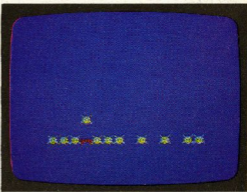
12 Si une fois chargé, le programme ne se met pas en route automatiquement, l'écran se vide et l'état **OK, 0:1** apparaît. Arrêtez la bande.



8 Les bandes bleues et rouges réapparaissent pendant que l'ordinateur s'apprête à charger le programme.

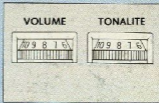
13 Appuyez sur R (RUN) et ENTER et le programme peut alors commencer.

9 Des lignes jaunes et bleues apparaissent dans la bordure. Ceci indique que le Spectrum est en train de charger le programme. Si ce dernier est très long, le chargement peut prendre plusieurs minutes.



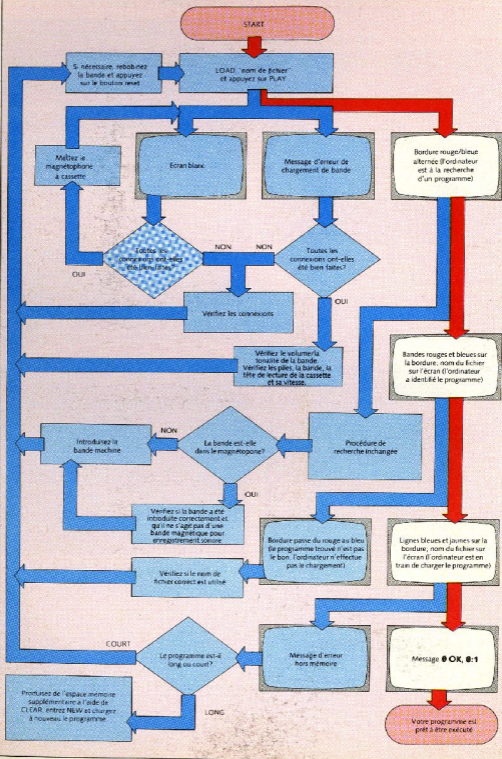
3 Si la bande est arrêtée au programme voulu ou si vous ne connaissez pas le nom du programme, entrer LOAD " " au lieu

de LOAD suivi du nom entre guillemets. Il ne doit pas y avoir d'espace entre les guillemets. Votre Spectrum chargera alors le premier programme qu'il rencontre sur la bande. Si le nom du programme qui apparaît n'est pas celui que vous cherchez, appuyez sur BREAK, rembobinez recommencez.



4 Prenez note des niveaux de volume et de tonalité qui permettent à votre Spectrum de

charger. Réglez les boutons de votre magnétophone à cassette sur ces niveaux avant le chargement.



COMMENCEZ A PROGRAMMER

Ce chapitre sert d'introduction à la rédaction de programmes sur le ZX Spectrum +. Il vous explique comment vous servir de votre Spectrum en vous apprenant à utiliser son clavier. Vous verrez ainsi comment vous pouvez faire fonctionner votre Spectrum. Les programmes courts que vous essayerez ici sont centrés sur les caractéristiques spéciales du Spectrum de telle sorte que lorsque vous commencerez à écrire vos programmes, vous serez à même d'exploiter toutes les possibilités de votre ordinateur.



LE PANNEAU DE COMMANDE DE VOTRE ORDINATEUR

Le ZX Spectrum + a son propre langage, le langage machine appelé BASIC. Pour qu'il obéisse à vos instructions, vous devez programmer le Spectrum en lui parlant en BASIC. Pour vous adresser à votre ordinateur, vous devez manoeuvrer son clavier. Ce dernier

GRAPH

Cette touche est utilisée pour sélectionner les formes ou les caractères graphiques des touches 1 à 8. Si vous appuyez sur cette touche, puis sur une touche numérique, avec ou sans la touche CAPS SHIFT, un caractère graphique apparaîtra sur l'écran. Vous devez réappuyer sur GRAPH si vous voulez que l'ordinateur fonctionne à nouveau normalement.

NEW

Cette touche remet à zéro la mémoire BASIC de l'ordinateur, effaçant n'importe quel programme qui y était stocké.

DELETE

Cette touche est utilisée lorsque vous vous êtes trompé de touche ou que vous voulez enlever un mot-clé, une lettre, un nombre ou un signe

EDIT

Cette touche est utilisée pour changer une ligne d'un programme sans le réécrire complètement

EXTEND MODE

Cette touche sélectionne le mot-clé qui se trouve dans la partie supérieure de n'importe quelle touche. Suivie de SYMBOL SHIFT et d'une touche, elle sélectionne le signe ou, le mot-clé se trouvant immédiatement au-dessus de la section soulevée de la touche.

CAPS SHIFT

Pour obtenir une majuscule, appuyez sur cette touche et sur une touche alphabétique. Si vous souhaitez plusieurs majuscules, utilisez CAPS LOCKS.

vous permet en outre de donner des commandes à l'ordinateur pendant qu'il exécute vos programmes.

Le dialecte (la version) du BASIC que le Spectrum comprend est une forme simple, mais riche du langage en question. En outre, le Spectrum a une fonction qui facilite énormément la programmation. Il s'agit du système d'entrée de mot-clé par touche unique.

Keys and keywords

Les mots-clés sont des mots spéciaux du BASIC qui donnent l'ordre à l'ordinateur de faire quelque chose — des mots comme PRINT ou INPUT par exemple. Sur la plupart des

TRUE VIDEO et INV VIDEO

Ces touches introduisent des codes de commande dans les lignes d'un programme pour produire des couleurs normales ou inversées.



CAPS LOCK

Utilisez cette touche si vous voulez que les touches alphabétiques ne produisent que des majuscules. Débloquez-la pour obtenir des minuscules.

BEEP
Cette touche produit le mot-clé qui commande le synthétiseur de son du Spectrum.

ordinateurs, vous devez taper chacune des lettres d'un mot-clé comme si vous tapiez à la machine et chaque mot doit être épilé sans la moindre erreur. Mais sur le Spectrum, il vous suffit d'appuyer sur une touche pour obtenir un mot-clé complet.

Sinclair BASIC compte plus de 80 mots-clés auxquels vous pouvez accéder par 36 touches au total (26 touches alphabétiques et 10 touches numériques). Bon nombre de ces touches ne produisent pas seulement un mais plusieurs mots-clés qui seront identifiés par l'ordinateur. En fait, la plupart des touches vous donnent des mots-clés, plus une lettre, un nombre, un signe ou même une forme (caractère graphique) qui peuvent tous être utilisés dans des programmes.

Touches d'affichage des couleurs

Ces six touches produisent des mots-clés qui commandent la façon dont le Spectrum affichera des couleurs sur l'écran.

Sélection de mots-clés et de signes

Sur le clavier du Spectrum se trouvent deux touches que vous utiliserez souvent. Il s'agit d'EXTEND MODE et de SYMBOL SHIFT qui sont les touches qui vous permettent de choisir les mots-clés et les signes des autres touches que vous souhaitez voir apparaître sur votre écran. Maintenant que vous vous êtes familiarisé avec le clavier, les deux pages suivantes vous expliquent comment sélectionner ce qui apparaît sur le clavier de l'ordinateur. Une fois que vous êtes capable faire cette sélection, vous pouvez commencer à écrire vos propres programmes.

Touches numériques

Ces touches produisent des nombres et peuvent en outre introduire dans les programmes des codes de commande correspondant aux couleurs illustrées — voir page 33. Les mots-clés se trouvant immédiatement au-dessus des touches 4 à 0, à l'exception de la touche 8, ne sont utilisés qu'avec les micro-entraînements ZX.

BREAK

Cette touche arrête un programme, mais ne l'efface pas de la mémoire de l'ordinateur.

ENTER

Appuyez sur cette touche pour entrer une ligne de programme dans la mémoire du Spectrum. Cette touche est souvent utilisée pour introduire des informations dans l'ordinateur pendant un programme.

SYMBOL SHIFT

Maintenez cette touche enfoncée et appuyez sur une touche numérique ou alphabétique pour sélectionner le mot-clé ou le signe inférieur se trouvant dans la partie supérieure de la touche. Lorsque vous l'utilisez après EXTEND MODE, elle sélectionne le symbole ou le mot-clé qui se trouve immédiatement au-dessus de la section soulevée de la touche

Barre d'espace

Elle produit un espace, comme sur une machine à écrire.

C Commandes du curseur

Si l'on appuie sur ces touches, le curseur se déplace dans la direction des flèches. Ces touches sont souvent utilisées par les programmes pour contrôler le mouvement des formes sur l'écran. Elles sont également utilisées lors de l'édition de programmes.



COMMENT UTILISER LES TOUCHES

La plupart des touches de votre ZX Spectrum + peuvent vous donner six mots-clés, lettres, numéros ou signes différents. Lorsque vous appuyez sur une touche le résultat qui apparaît sur l'écran dépend du mode dans lequel l'ordinateur se trouve à ce moment-là. Les divers modes vous permettent d'introduire par le clavier différents types d'information, comme des mots-clés, des lettres ou des caractères graphiques. L'avantage est que pendant que vous manœuvrez le clavier, le Spectrum vous aide à choisir les modes voulus de telle sorte que vous entrez les instructions et les informations dans l'ordre correct.

Mode mot-clé

Mettez votre Spectrum en route ou remettez-le à l'état initial pour faire apparaître

le message de copyright. Ensuite, appuyez sur ENTER. Un K clignotant apparaît dans le coin inférieur gauche. Le carré clignotant s'appelle le curseur. Il vous indique l'endroit où quelque chose va apparaître sur l'écran et le K indique que l'ordinateur est dans le mode *mot-clé*. Appuyez sur n'importe quelle touche alphabétique et le mot-clé supérieur de la section soulevée de la touche apparaît sur l'écran. Essayez Q par exemple et le mot-clé PLOT apparaît. Appuyez sur la touche DELETE pour effacer le mot-clé et essayez d'autres touches. Les touches numériques vous donnent des nombres, mais dès que vous appuyez sur une touche alphabétique, le mot-clé supérieur de la section soulevée apparaît.

Utilisez à nouveau DELETE pour faire réapparaître le curseur K. Ensuite, appuyez sur une touche SYMBOL SHIFT. Cette fois, le mot-clé ou le signe se trouvant juste au-dessus de la lettre de la section soulevée qui apparaît. S'il s'agit d'une touche numérique, le signe se trouvant à droite de la section soulevée apparaît.

Comment sélectionner un mot-clé, un symbole et un caractère

Vous voyez ici comment sélectionner un mot-clé, un symbole ou un caractère sur une touche alphabétique ou sur une touche

numérique. Lorsque vous sélectionnez une fonction-clé, notez sa position sur la touche et ensuite, en vous basant sur les deux

exemples donnés ici, choisissez les touches dont vous avez besoin pour obtenir le mode correct.

Lettre-clé



Mode mot-clé (K)

Touche unique **BORDER**

et touche *

Mode étendu (E)

Ensuite, touche unique **BIN**

Ensuite, et touche **BRIGHT**

Mode lettre (L)

Touche unique **b**

et touche **B**

et touche *

Mode majuscules (C)

Ensuite, touche unique **B**

Ensuite, et touche *

Mode graphique (G)

Ensuite touches A à U uniquement (UDC)

Nombre-clé



Mode mot-clé (K)

Touche unique **3**

Et touche #

Mode étendu (E)

Ensuite touche unique magenta (Code commande couleurs)

Ensuite, et touche **LINE**

Mode lettre (L)

Touche unique **3**

Et touche #

Mode majuscules (C)

Ensuite, touche unique **3**

Ensuite, et touche #

Mode graphique (G)

Ensuite touche unique

Ensuite, et touche

Modes lettre et majuscules

Après avoir produit un mot-clé ou un signe en mode mot-clé, l'ordinateur remplace automatiquement le curseur par L. Il se trouve maintenant dans le mode *lettre*. Si vous voulez obtenir une majuscule, appuyez sur CAPS SHIFT, puis sur la touche alphabétique.

Si vous ne voulez que des majuscules, appuyez d'abord sur CAPS LOCK. Le curseur est remplacé par C. Votre Spectrum est maintenant dans le mode *majuscules* et chaque fois que vous appuyez sur une touche alphabétique, vous obtenez une majuscule. Pour revenir au mode lettre (L), appuyez à nouveau sur CAPS LOCK.

Mode étendu

Le mode suivant est le mode *étendu* et il s'obtient en appuyant sur la touche EXTEND MODE. Le curseur est maintenant remplacé par E. Appuyez sur n'importe quelle touche alphabétique et le mot-clé supérieur de la paire de mots-clés se trouvant au-dessus de la section soulevée apparaît. Par exemple, appuyez sur B et vous obtiendrez BIN. Pour

obtenir le mot-clé ou le signe inférieurs se trouvant au-dessus de la section soulevée, appuyez d'abord sur une touche SYMBOL SHIFT et maintenez-la enfoncée, puis appuyez sur la touche alphabétique. Avec la touche B, par exemple, vous obtenez maintenant BRIGHT. Après avoir appuyé sur une touche (ou sur EXTEND MODE) dans le mode étendu, l'ordinateur revient automatiquement au mode lettre ou majuscules.

Mode graphique

Le cinquième mode est le mode *graphique* obtenu en appuyant sur la touche GRAPH. Le curseur est remplacé par G. Appuyez sur les touches 1 à 8 et voyez si les caractères graphiques qui figurent sur ces touches apparaissent. Ensuite, appuyez sur CAPS SHIFT et sur n'importe quel numéro de 1 à 8. Les caractères graphiques réapparaissent, sans cette fois, le blanc et le noir sont inversés. Pour abandonner le mode graphique, vous devez toujours réappuyer sur GRAPH. Puisque l'ordinateur ne l'abandonne pas automatiquement.

Éditer sur le Spectrum

Lorsque vous donnez des commandes ou que vous écrivez des programmes pour votre Spectrum, vous devrez corriger les erreurs qui se sont glissées dans les commandes ou les lignes des programmes ou bien vous devrez les modifier. Ceci est facile à réaliser en les éditant.

Comment corriger une erreur

Si vous essayez d'entrer une ligne ou une commande erronées en BASIC, un clignotant apparaîtra sur l'écran avant l'erreur. Pour corriger l'erreur, maintenez enfoncée la touche de commande droite ou gauche du curseur pour amener ce dernier à la droite de l'erreur. Ensuite, effacez l'erreur en appuyant sur DELETE ou bien ajoutez ce qui est requis. Puis, appuyez sur ENTER.

Comment éditer une ligne de programme

Lorsque vous écrivez un programme, vous établissez une

série de lignes d'instructions numérotées appelée une liste. Si après avoir écrit un programme, vous le "listez" en appuyant sur K (LIST) et sur ENTER, vous devriez voir le signe > à côté d'une des lignes du programme. Si le signe n'apparaît pas, enfoncez ou relâchez la touche de commande du curseur. Si vous appuyez sur EDIT, cette ligne est reproduite en bas de l'écran et peut être modifiée comme dans le cas précédent à l'aide de la touche du curseur et de DELETE. Appuyez sur ENTER pour introduire la nouvelle ligne dans le programme. Si vous souhaitez éditer une autre ligne, amenez le signe jusqu'à la ligne que vous souhaitez modifier à l'aide de la touche de commande du curseur, puis appuyez sur EDIT. Si cela prend trop longtemps, introduisez LIST, suivi du numéro de la ligne, puis, appuyez sur EDIT.

Pour effacer une ligne entière d'un programme, il vous suffit d'introduire par le clavier le numéro de la ligne et d'appuyer sur ENTER. Si vous exécutez un programme contenant une erreur, vous verrez un état d'erreur. Ces états sont expliqués à la page 74.



LA CALCULATRICE DE LA TELEVISION

Le ZX Spectrum + peut faire des calculs très rapidement et avec beaucoup de précision. Il n'a besoin que de quelques nombres et des signes, + ou - par exemple, lui disant ce qu'il doit faire avec les nombres.

Introduisez l'instruction suivante par le clavier (vous trouverez le signe + sur la touche K):

PRINT 6+2

Il s'agit d'une *commande*. Lorsque vous appuyez sur ENTER, la commande disparaît et la réponse 8 est imprimée sur l'écran.

Votre Spectrum utilise cinq signes appelés des *opérateurs arithmétiques*. Leur rôle est expliqué dans le tableau qui se trouve en bas de la page. Vous pouvez les utiliser tous de la même façon à l'aide de PRINT.

Le Spectrum peut également afficher les calculs et leurs résultats. Entrez la commande suivante:

PRINT "6+2=";6+2

L'ordinateur affiche:

6+2=8

Appuyer sur PRINT fait apparaître sur l'écran tout ce qui se trouve entre les guillemets, c'est pourquoi 6+2 apparaît. Le point-virgule donne l'instruction au Spectrum d'afficher le résultat immédiatement après le signe égale.

Les signes mathématiques du Spectrum

Les signes ci-dessous ou 'opérateurs arithmétiques' sont utilisés par le Spectrum pour effectuer des opérations mathématiques. Veuillez noter que l'ordinateur n'utilise pas les signes x ou ÷.

Symbole	Touche	Fonction	Exemple
+	K	Addition deux nombres	8+2=10
-	J	Soustrait deux nombres	8-2=6
*	B	Multiple deux nombres	8*2=16
/	V	Divise deux nombres	8/2=4
^	H	Élève le premier nombre à la puissance du second	8 1 2 = 64

Votre premier programme

Une fois qu'il a exécuté une commande, votre Spectrum l'oublie. Si vous voulez que l'ordinateur recommence le calcul, ce dernier peut être écrit en tant que programme. Introduisez l'instruction par le clavier, puis appuyez sur ENTER.

10 PRINT 6+2

Cette fois-ci, l'instruction n'est pas exécutée sur-le-champ, mais elle est affichée sur l'écran par l'ordinateur. Appuyez ensuite sur R (RUN) et sur ENTER et le résultat 8 apparaîtra.

L'instruction complète constitue maintenant un programme machine. Placer un nombre au début signifie que votre Spectrum stockera l'instruction dans sa mémoire, mais ne l'exécutera pas avant que vous ne lui en ayez donné l'ordre. Chaque fois que vous passez un programme en appuyant sur R (RUN), puis sur ENTER, l'instruction est exécutée. L'instruction — qui n'est désormais plus une commande — constitue l'une des lignes numérotées d'un programme. Les instructions d'un programme sont toujours exécutées par ordre de numérotation des lignes qui sont généralement groupées par dix.

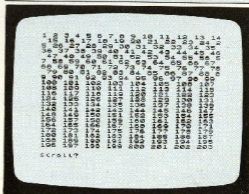
Vous pouvez maintenant vous mettre sérieusement au travail. Entrez le programme. N'oubliez pas d'appuyer sur ENTER après avoir introduit chaque ligne par le clavier; une fois que vous avez terminé, appuyez sur R (RUN) et sur ENTER. Lorsque vous avez passé le programme, vous devriez voir ce qui suit:

TABLEAU NUMERIQUE

```

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

```



Tous les nombres de 1 à 203 sont affichés. Appuyez maintenant sur n'importe quelle touche, sauf N, la barre d'espacement, STOP et BREAK, et une autre série de nombres apparaît.

Ce programme utilise une *variable* qui est ici appelée n. N'importe quelle lettre ou mot ferait l'affaire — n remplace tout simplement nombre.

Une valeur est attribuée à la variable et cette valeur se modifie au cours de l'exécution du programme. A la ligne 10, on utilise le mot-clé LET pour fixer la valeur à 1. La ligne 20 affiche la valeur suivie d'un espace. A la ligne 30, on utilise à nouveau LET, cette fois pour augmenter la valeur de 1, ce qui veut dire que n devient 2. La ligne 40 utilise le mot-clé (unique) GOTO pour renvoyer le programme à la ligne 20 qui affiche maintenant 2. La même opération se répète jusqu'à ce que l'écran soit entièrement couvert de chiffres.

Que faire pour qu'un programme demande un nombre

Arrêtez le programme en appuyant sur BREAK. Introduisez une nouvelle ligne par le clavier

10 INPUT n

Cette ligne remplace l'ancienne ligne 10 du programme. Lorsque vous passez le programme, l'ordinateur attend que vous entriez un nombre. Introduisez n'importe quel nombre par le clavier, puis appuyez sur ENTER. Les nombres commenceront maintenant au nombre que vous venez d'entrer et ceci parce que INPUT n donne à n une valeur égale au nombre qui a été entré.

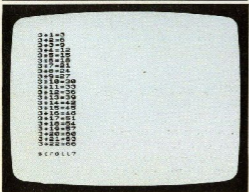
Programmer une table de multiplication

Appuyez sur le bouton 'reset' (remise à l'état initial) pour extraire l'ancien programme et entrer le nouveau. Grâce à ce programme, le Spectrum effectuera des multiplications.

Introduire n'importe quel nombre par le clavier et une table de multiplication de ce nombre clignotera sur l'écran. Appuyez sur n'importe quelle touche, sauf N, BREAK ou la barre d'espacement, et la table s'allonge. Appuyez sur BREAK, puis repassez le programme pour créer une nouvelle table. Voici le programme et ce que vous devriez voir si vous introduisez 3, puis T46.

TABLE DE MULTIPLICATION

```
10 LET K = 1
20 INPUT N
30 PRINT "1 2 3 4 5 6 7 8 9 10 *";
40 LET K = K + 1
50 GO TO 30
```



Pourquoi vous devez utiliser des parenthèses

Vous devrez parfois utiliser des parenthèses dans un calcul. Entrez les deux commandes ci-dessous et comparez les résultats:

```
PRINT 6+2/4
```

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

La première donne 6,5 et la seconde 2. Ces différents résultats s'expliquent par le fait que l'ordinateur a un système de priorités incorporé qu'il utilise pour effectuer des calculs. Il exécute d'abord, puis * ou /, et enfin + ou -, mais il commence *toujours* par effectuer les calculs entre parenthèses. Dans la première commande ci-dessus, il commence par diviser 2 par 4 et ajoute le résultat (0,5) à 6. Dans la deuxième commande, il additionne 6 et 2 et divise le résultat par 4.

Comment ponctuer avec votre Spectrum

Le Spectrum utilise une série de signes de ponctuation. Ils sont très importants car beaucoup d'entre eux servent également d'instructions machine et déterminent la façon dont l'ordinateur comprend une ligne de programme ou le contenu de l'écran.

- **Point-virgule.** Utilisé avec PRINT, le point-virgule demande à l'ordinateur d'afficher l'un à côté de l'autre les deux éléments d'information se trouvant à sa gauche et à sa droite.
- **Deux points.** Indique la fin d'une instruction dans une ligne de programme et le début de la suivante.
- // **Guillemets.** Tout caractère entre guillemets ne sera pas traité en tant que nombre ou variable, mais en tant que texte. Les guillemets commencent et terminent une chaîne.
- **Virgule.** Utilisée avec PRINT, la virgule demande à l'ordinateur d'afficher l'élément d'information qui suit, soit au centre de la ligne, soit au début de la ligne suivante. Ne pas l'utiliser pour indiquer des milliers ou des millions.
- **Point.** Soit une décimale ou un point.

LA COULEUR ET SON UTILISATION

Votre ZX Spectrum + peut produire huit couleurs différentes et chaque couleur possède son numéro de code. Chaque couleur peut être utilisée de trois façons différentes — en tant que couleur de la bordure, en tant que couleur de l'encre et en tant que couleur de papier.

ZX SPECTRUM + COULEURS CODEES

La liste vous donne les couleurs et les codes utilisés par le Spectrum. Il n'est pas nécessaire de mémoriser ces codes. Le nom des couleurs figurent sur les touches numériques qui les produisent. (Ces noms ne sont pas des mots-clé.)

Nombre	Couleur
0	noir
1	bleu
2	rouge
3	magenta (pourpre)
4	vert
5	cyan (bleu de prusse)
6	jaune
7	blanc

Les nuances que vous obtiendrez sur votre téléviseur dépendent du réglage de la couleur, du contraste et de la brillance. N'oubliez pas que vous avez besoin de la télévision.

Les trois façons dont le Spectrum utilise la couleur

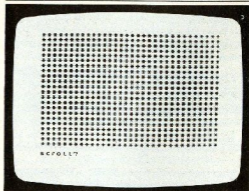
Vous pouvez utiliser la couleur de trois façons différentes. La couleur de la *bordure* est celle qui entoure la zone d'affichage centrale. La couleur de l'*encre* est celle dans laquelle les caractères (lettres, nombres, signes et formes graphiques) et les points ou lignes apparaissent. La couleur du *papier* est la couleur de l'arrière-plan, c'est-à-dire celle de la zone d'affichage tout entière ou celle du carré entourant chaque caractère.

Lorsque vous mettez votre Spectrum en route, il utilise les couleurs prédéfinies. La couleur de l'encre est noire, celle du papier et de la bordure est blanche. Ces couleurs peuvent être instantanément modifiées en entrant des commandes par le clavier. Vous avez déjà vu comment procéder aux pages 6 et 7, où la commande BORDER a été utilisée pour vérifier que les couleurs de votre téléviseur et de votre Spectrum sont correctement réglées. Appuyez maintenant sur le bouton "reset", introduisez le

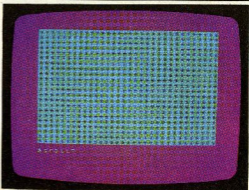
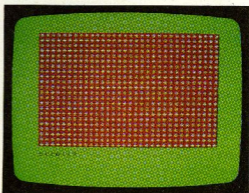
programme simple ci-dessous et exécutez-le.

TEST DES COULEURS

```
10 PRINT "0"
20 GO TO 10
```



Des étoiles noires et blanches apparaissent. Appuyez sur BREAK et introduisez quelques commandes de couleur. Introduisez par le clavier les mots-clés BORDER, INK et PAPER, suivis chacun d'un nombre de 0 à 7 et appuyez sur ENTER après chaque nombre. Ensuite, repassez le programme. Deux images seront affichées: la première avec BORDER 4, PAPER 2 et INK 7 et la seconde avec BORDER 3, PAPER 5 et INK 1.



Comment écrire des programmes en couleurs

Vous pouvez utiliser les mots-clés BORDER, PAPER et INK pour que le texte, les tableaux, les formes et les images d'un programme apparaissent en plusieurs couleurs différentes. Si vous utilisez BORDER dans une ligne du programme, la couleur de la bordure change aussitôt que le Spectrum atteint cette ligne.

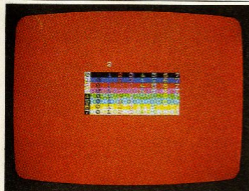
Utiliser INK dans une ligne donne une nouvelle couleur d'encre lorsqu'un caractère ou une ligne apparaît ensuite sur l'écran. Utiliser PAPER dans une ligne change la couleur du papier, mais uniquement autour des caractères (et aussi des points et des lignes). Si vous souhaitez que la totalité de l'arrière-plan de la zone d'affichage adopte une couleur déterminée, vous devez appuyer sur PAPER, puis sur CLS.

Vous pouvez aussi utiliser INK et PAPER après PRINT. Dans ce cas, seuls les caractères produits par PRINT adoptent les couleurs de l'encre et du papier. Le programme suivant illustrent les couleurs que peuvent prendre la bordure, l'encre et le papier. Il vous montre également comment utiliser INK et PAPER après PRINT.

COMBINAISONS DE COULEURS

```

10 FOR b=0 TO 7
20 BORDER b: PAPER b: CLS
30 PRINT AT 6,12: INK 9: b
40 FOR p=0 TO 7
50 PRINT AT p+8,8: INK p: PAPER p
R 9:p
60 BEEP 0.5,b*p-20*p
70 FOR i=0 TO 7
80 PRINT INK i: PAPER p: " ";i;
90 BEEP 0.01,i*5
100 NEXT i
110 NEXT p
120 NEXT b
  
```



Lorsque vous passez ce programme, vous verrez quelles sont les combinaisons possibles de bordure, papier et encre. Le programme a trois variables, b pour le numéro de la bordure, i pour le numéro de l'encre et p pour le numéro du papier. BEEP produit le son et les lignes commençant par FOR et NEXT indique le début et la fin de la boucle du programme qui modifie tous les numéros des couleurs par ordre de 0 à 7. Veuillez noter que INK et PAPER peuvent tous deux avoir la valeur 9. Le papier et l'encre sont alors blancs ou noirs de façon à contraster avec l'arrière-plan ou un caractère.

Programmer des diagrammes de Gantt en couleurs

Le programme suivant utilise les couleurs du Spectrum pour produire un diagramme de Gantt. Il donne douze températures diurnes, sous forme de colonnes jaunes numérotées. A la ligne 60, entrez deux espaces entre les guillemets.

DIAGRAMME DE GANTT

```

10 BORDER 0: PAPER 1: CLS
20 LET c=4
30 FOR x=1 TO 12
40 READ t
50 FOR l=21 TO 21-t STEP -1
60 PRINT PAPER 6:AT l,c: " "
70 NEXT l
80 PRINT INK 2:AT 20-t,c:t
90 LET c=c+2
100 NEXT x
110 DATA 20,15,13,16,19,20,18,1
112,19,14,17
  
```



Ajoutez maintenant les lignes suivantes et introduisez par le clavier la nouvelle ligne 110, comme indiqué. Le diagramme apparaît maintenant en deux couleurs. Pour en savoir plus sur READ et DATA, consultez la page 33.

DIAGRAMME DOUBLE DE GANTT

```

85 READ t
86 FOR l=21 TO 21-t STEP -1
87 PRINT PAPER 3:AT l,c: " "
88 NEXT l
89 PRINT INK 1: PAPER 5:AT 20-t,c:t
110 DATA 20,6,15,4,13,5,16,6,19,10,20,8,18,6,11,4,14,6,10,6,14,6,17,7
  
```



GRAPHIQUES SIMPLES A FAIRE VOUS-MEME

Votre ZX Spectrum + peut vous donner des graphiques à haute et basse résolution qui peuvent apparaître ensemble sur l'écran. Les graphiques à basse résolution sont constitués de blocs de couleur. Ces deux pages vous expliquent comment produire ces blocs à l'aide du clavier et comment les positionner sur l'écran.

L'écran à basse résolution

L'écran à basse résolution compte 32 positions auxquelles les caractères peuvent être placés horizontalement et 22 positions auxquelles ils peuvent être placés verticalement. Chaque position est identifiée par deux nombres. D'abord, le numéro de ligne qui est le nombre de lignes à descendre sur l'écran pour atteindre la position. La ligne supérieure est la ligne 0 et la ligne inférieure, la ligne 21. Ensuite vient le numéro de colonne qui est le nombre de colonnes à traverser sur l'écran pour atteindre la position. La colonne de gauche est la colonne 0 et la colonne de droite, la colonne 31. (Jetez un coup d'oeil à la grille à basse résolution de la page 80.) Le programme suivant introduit des couleurs dans ces positions de caractère. Le mot-clé RND (sur la touche R) choisit une couleur d'encre aléatoire.

CARRÉS

```
10 BORDER 1: INK RND+7
20 PRINT " ";
30 GOTO 10
```



Des carrés apparaissent partout sur l'écran. Pour qu'un caractère apparaisse à une position déterminée, vous devez utiliser le mot-clé AT en même temps que PRINT. AT vient après PRINT et est suivi du numéro de ligne, d'une virgule, du numéro de colonne et d'un point-virgule. La commande

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

par exemple, produit une astérisque à la ligne 11, colonne 16, c'est-à-dire au centre de l'écran.

Comment dessiner des arc-en-ciel

Utilisées dans vos programmes graphiques, les boucles FOR NEXT sont idéales pour produire des motifs en couleur. Un programme contenant des boucles FOR NEXT refait ces boucles un certain

Comment sélectionner les caractères graphiques

Le clavier de votre ZX Spectrum + comporte une série de caractères graphiques grâce auxquels il est facile de programmer des graphiques à basse résolution. Ils se trouvent sur les touches 1 à 8.

Pour produire les caractères graphiques sur l'écran, appuyez sur le touche GRAPH, puis sur les

touches 1 à 8, en utilisant à chaque fois la barre d'espacement. Les caractères graphiques apparaissent en bas de l'écran. La partie blanche de chaque caractère est la couleur de l'encre et la partie noire, la couleur du papier. Appuyez à nouveau sur les touches tout en maintenant CAPS SHIFT enfoncée.

Cette fois-ci, la couleur de l'encre et du papier sera inversée.

Voilà comment vous pouvez introduire des caractères graphiques dans les lignes d'un programme. Si vous ne voulez plus produire des caractères graphiques, appuyez seulement sur GRAPH encore une fois.

GRAPH

Cette touche sert à mettre l'ordinateur dans le mode graphique.



Touche 8

Cette touche est souvent utilisée avec GRAPH et CAPS SHIFT pour produire un carré plein en couleur.



nombre de fois. A la ligne où la boucle commence, vous pouvez dire à l'ordinateur combien de fois vous souhaitez que la boucle soit exécutée. Lors de son exécution, elle peut être utilisée pour placer des caractères sur l'écran par exemple.

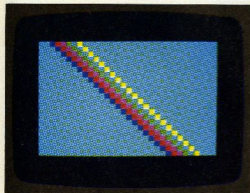
D'autre part, vous pouvez programmer plus d'une boucle à la fois. Vous pouvez placer une boucle à l'intérieur d'une autre, ce qui donne souvent des résultats très intéressants. Le programme ci-dessous vous montre comment utiliser deux boucles FOR NEXT ("emboîtées" l'une dans l'autre) pour changer les couleurs et les positions produites par INK et AT. La programmation de ces boucles est expliquée dans l'encadré du bas de la page.

ARC-EN-CIEL

```

5 BORDER 0: PAPER 5: CLS
10 LET X=1
20 FOR L=0 TO 21
30 FOR C=1 TO 6
40 PRINT INK C:AT L,C:X:
50 NEXT C
60 LET X=X+1
70 NEXT L

```



Programmation d'images

Dans les graphiques à basse résolution, vous pouvez "peindre" des images en déterminant les positions et les couleurs des caractères graphiques. Vous pouvez mettre au point votre propre image en utilisant la grille à basse résolution de la page 80. Ensuite, sélectionnez les caractères graphiques comme expliqué à la page d'en face et entrez une à une les lignes du programme pour construire votre image.

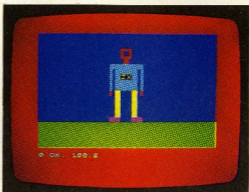
Le programme ci-dessous illustre le genre de résultats que vous pouvez obtenir. Toutes les formes qu'il contient se trouvent sur les touches numérotées. Vous pouvez attendre que toutes les lignes aient été entrées par le clavier avant d'exécuter le programme ou bien vous pouvez l'exécuter après avoir entré chaque ligne et vous verrez alors comment les différentes parties du robot sont assemblées. (Souvenez-vous que si vous avez entré un caractère graphique incorrect, vous pouvez l'éditer, tout comme vous éditeriez un nombre ou une lettre incorrects.)

ZX ROBOT

```

5 BORDER 2: PAPER 1: CLS
10 PRINT INK 2;AT 0,15:
15 PRINT INK 2;AT 4,15:
20 PRINT INK 2;AT 8,15:
30 PRINT INK 2;AT 12,15:
40 FOR L=7 TO 10 PRINT INK 5:
AT L,13:
45 PRINT INK 5: PAPER 0;AT 8,15:
5:
50 PRINT INK 2;AT 11,13:
60 FOR L=11 TO 15: PRINT INK 6:
AT L,14:
70 PRINT INK 3;AT 15,13:
AS 17:
80 FOR L=17 TO 21: FOR C=0 TO 31
90 PRINT INK 4:AT L,C:
100 NEXT C: NEXT L

```



Le mot-clé TAB qui apparaît après PRINT à la ligne 70 est utilisé pour positionner un caractère sur la ligne à laquelle l'ordinateur est alors occupé à travailler. TAB est suivi d'un nombre entre 0 et 31, spécifiant une position de colonne.

Comment utiliser les boucles FOR NEXT

Une boucle FOR NEXT commence toujours par une ligne contenant les mots-clés FOR et TO, en même temps qu'une variable accompagnée de sa valeur initiale et de sa valeur limite, par exemple

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

Dans ce cas-ci, la variable est C. La boucle qui commence ici contiendra la ou les ligne(s) qui font répéter une opération par l'ordinateur. Ces lignes peuvent également utiliser elles-mêmes la variable C. Les boucles FOR NEXT se terminent toujours par le mot-clé NEXT et la variable, par exemple

```
50 NEXT C
```

Lors de l'exécution, le programme refait la totalité de la boucle de FOR à NEXT un nombre déterminé de fois. La variable commence à la première valeur précédant TO et s'accroît de 1 jusqu'à ce qu'elle ait atteint la limite qui se trouve après TO. Dans ce cas-ci, le programme refait la boucle six fois, commençant à 1, puis devenant 2, 3, 4, 5 et finalement 6.

Dans le premier programme de la page 25, trois boucles "emboîtées" sont utilisées. Ceci signifie qu'à chaque cycle de la boucle "extérieure", la boucle "du milieu" passe par tous ses cycles. La boucle "intérieure" passe systématiquement par tous ses cycles, chaque fois que la boucle "du milieu" accomplit un cycle. Vous pouvez emboîter autant de boucles que vous voulez, mais faites attention qu'elles ne se chevauchent pas.

LE CARNET DE CROQUIS SUR ECRAN

Les graphiques que vous pouvez produire sur le ZX Spectrum + ne sont pas simplement des formes ou des images grossières à basse résolution. La haute résolution de votre Spectrum vous permet de créer des images détaillées aux bords nets, ainsi que des lignes et des bords droits ou courbes.

Les graphiques à haute résolution sont constitués par une série de points placés les uns à côté des autres pour former une ligne ou pour colorer une forme. Chaque point équivaut au soixante-quatrième des carrés utilisés dans les graphiques à basse résolution. Entrez la commande

PLOT 128, 87

et vous en verrez un au milieu de l'écran.

Les points utilisés dans les graphiques à haute résolution sont appelés des *pixels*, l'abréviation de "picture cells" (éléments d'image). Tout comme les caractères à basse résolution, chaque pixel exige deux numéros pour spécifier sa position.

La grille à haute résolution

La grille à haute résolution se compose de 256 pixels horizontaux et de 176 pixels verticaux. Toutefois, à la différence des images à basse résolution, le premier numéro est la coordonnée horizontale — sa position à travers l'écran. Ces numéros de position vont de 0 à gauche jusqu'à 255 à droite. Le second numéro est la coordonnée verticale et les numéros vont de 0 en bas de la grille jusqu'à 175 en haut. La position 0, 0 est le coin inférieur gauche et non le coin supérieur gauche comme dans la basse résolution. Jetez un coup d'oeil à la grille à haute résolution de la page 80.

Traçage et dessin

Trois mots-clés suffisent pour produire des graphiques à haute résolution — PLOT, DRAW et CIRCLE. PLOT est suivi des coordonnées horizontale et verticale séparées par une virgule et il place un pixel à cette position. DRAW est lui aussi suivi de deux numéros séparés par une virgule, mais ce sont pas les coordonnées d'une position. Ce sont les distances horizontale et verticale séparant deux positions; ces distances sont exprimées en pixels et DRAW trace alors une ligne entre les deux positions.

Si PLOT et DRAW n'ont pas encore été utilisés dans le programme, la première position est 0, 0.

S'ils ont déjà été utilisés, cette position est la dernière atteinte par PLOT ou la dernière atteinte par DRAW, mais c'est toujours la position la plus récente. L'instruction DRAW tire alors une ligne jusqu'à la nouvelle position. Exécutez le programme ci-dessous.

ETOILE

```
10 INK = 2
20 PLOT 128, 174
30 DRAW 70, -140
40 DRAW 150, 80
50 DRAW -150, 80
60 DRAW 70, 140
```



PLOT amène la position de départ en haut de l'écran. Puis les cinq instructions DRAW tirent les cinq lignes rouges.

Ajoutez maintenant ces lignes au programme.

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

Repassez le programme et l'étoile rouge apparaît entourée d'un cercle bleu.



CIRCLE a besoin de trois valeurs. Les deux premières donnent la position du centre du cercle et la troisième donne son rayon. Vous pouvez également ajouter une troisième valeur aux

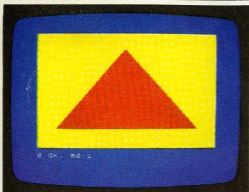
instructions DRAW. Essayez des valeurs allant de 2 à -2 et voyez ce qui se passe.

Comment remplir des formes

Vous pouvez facilement produire des formes pleines en traçant une série de lignes proches les unes des autres. Ceci s'exécute à l'aide d'une boucle FOR NEXT qui augmente de 1 à chaque fois les positions DRAW.

TRIANGLE PLEIN

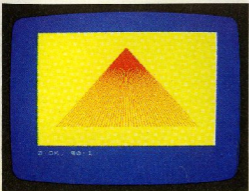
```
10 BORDER 1: PAPER 6: INK 2: C
LS
20 FOR x=-100 TO 100
30 PLOT 128,150
40 DRAW X,-120
50 NEXT X
```



En écartant légèrement les lignes, vous obtenez un effet intéressant. Pour les écarter, vous devez ajouter le mot-clé STEP et un numéro à l'instruction FOR. Entrez une nouvelle ligne 20 et repassez le programme.

```
20 FOR x=-100 TO 100 STEP 4
```

Cette fois la forme en éventail représentée ci-dessous apparaît et ceci parce que STEP fait augmenter x de 4 au lieu de 1 chaque fois qu'une ligne est tirée.



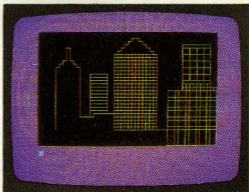
Votre carnet de croquis sur écran

Vous n'avez pas besoin d'écrire un programme chaque fois que vous voulez produire une image ou une forme. Vous pouvez utiliser un programme qui vous permet de construire une image directement sur l'écran.

Le programme commence par le mot-clé INPUT, vous demandant un numéro d'encr. Puis, toujours à l'aide de INPUT (cette fois avec le signe \$ pour affecter une chaîne), il ordonne à l'ordinateur de tracer de courtes lignes chaque fois que vous appuyez sur l'une des quatre touches suivantes — U, D, L ou R.

CARNET DE CROQUIS ET EXEMPLE

```
10 INPUT "INK ":I
20 BORDER 3: PAPER 0: INK I: C
LS
30 PLOT 25,25
40 LET X=0
50 INPUT X,$
60 IF $="U" THEN DRAW 0,X
70 IF $="D" THEN DRAW 0,-X
80 IF $="L" THEN DRAW X,0
90 IF $="R" THEN DRAW -X,0
100 GO TO 50
```



Prendre des décisions avec IF et THEN

Les lignes 60 à 90 du programme du carnet de croquis contiennent les instructions IF et THEN qui permettent à votre Spectrum de prendre une décision. L'ordinateur essaie alors de voir si la lettre entrée est u, d, l ou r. Si vous avez appuyé sur l'une de ces touches, l'ordinateur reçoit alors l'ordre de tracer une ligne en haut, en bas, à gauche ou à droite.

IF est toujours suivi de quelque chose dont le Spectrum met la véracité ou la réalité à l'épreuve — vérifiez si certaines sont enfoncées par exemple. Si la chose est vraie ou réelle, l'action qui suit THEN est effectuée. Dans le cas contraire, le programme passe à la ligne suivante. Tout ce qui dans une ligne vient après THEN est soumis à la décision. Dans cette ligne:

```
110 IF b=5 THEN PRINT "*" GOTO 200
```

L'ordinateur n'avancera à la ligne 200 que si b égale 5.

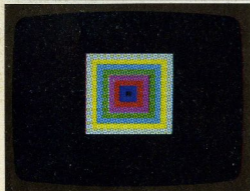
CREER DES FORMES ET DES IMAGES

Votre ZX Spectrum + vous permet de produire toutes sortes de formes et d'images en utilisant des graphiques soit à basse, soit à haute résolution ou bien les deux. Le mieux est de tracer votre dessin sur une copie des grilles de la page 80.

Pour dessiner des formes et des images, vous pouvez souvent utiliser les boucles FOR NEXT qui répètent une partie de programme un nombre déterminé de fois. La position et la couleur des caractères ou des lignes peut changer à chaque fois, habituellement de façon régulière. Voici un programme qui utilise cette méthode.

CARRÉS

```
10 BORDER 0: PAPER 0: CLS
20 FOR x=7 TO 0 STEP -1
30 INK x
40 FOR l=11-x TO 11+x
50 FOR c=-x TO 10+x
60 PRINT AT l,c;"■"
70 NEXT c
80 NEXT l
90 NEXT x
```



Ce programme contient trois boucles FOR NEXT. La boucle x modifie la couleur et la dimension des grands carrés qui sont produits, alors que les boucles l et c changent la ligne et la colonne du petit carré chaque fois qu'il est imprimé. Essayez de remplacer le carré à la ligne 60 par une étoile ou un autre signe du clavier.

Effets aléatoires et sous-programmes

Les boucles ne produisent pas des formes identiques chaque fois que vous exécutez un programme de graphiques. En utilisant le mot-clé RND (abréviation de RaNDom) dans des boucles, vous pouvez changer la position, la couleur et les autres caractéristiques de l'écran à chaque fois.

Jetez un coup d'oeil au programme de mosaïque de la page 10. L'on a pu produire une mosaïque parce que la couleur de l'encre est RND*7, c'est-à-dire n'importe quel nombre muni d'une décimale entre 0 et 7. INK arrondit le nombre au nombre entier supérieur. Chaque fois qu'un carré apparaît sur l'écran, sa couleur est l'une des couleurs produites par INK 0 à INK 7.

Le programme suivant dessine sur l'écran des formes symétriques de caractères graphiques. Il utilise RND pour changer ces caractères et leur position. Les variables i et p donnent la couleur de l'encre et du papier et a indique le nombre de formes dessinées (ici quatre). La variable n donne le nombre de caractères de chaque forme et x est un nombre aléatoire, entre 129 et 142.

L'instruction GOSUB de la ligne 50 envoie l'ordinateur à un sous-programme.

FORMES SYMETRIQUES

```
10 BORDER 4: PAPER 0: CLS
20 LET i=4: LET p=0
30 FOR j=1 TO 4
40 LET x=RND*13+129
50 FOR n=1 TO 40: GO SUB 1000:
NEXT n
60 LET i=i+1: LET p=p+1
70 PAUSE 100
80 NEXT j
90 STOP
1000 LET l=INT (RND*11)
1010 LET c=INT (RND*16)
1020 INK l: PAPER p
1030 PRINT AT l,c;CHR$ x
1040 PRINT AT l,31-c;CHR$ x
1050 PRINT AT 31-l,c;CHR$ x
1060 PRINT AT 31-l,31-c;CHR$ x
1070 BEEP 0.01,l*c/3
1080 RETURN
```



Un sous-programme est un groupe de lignes agissant comme un "programme dans un programme". Dans ce programme-ci, le sous-programme est à la ligne 1000. Il affiche un caractère graphique dans quatre sections de l'écran et chaque caractère est à la même distance du centre (position 11, 16). La distance est donnée par les lignes 1000 et 1010, l donnant la distance en lignes et c, la distance en colonnes. INT remplace le nombre aléatoire par un nombre entier, de telle sorte que l est n'importe quel nombre entier entre 0 et 10 et c, n'importe quel nombre entier entre 0 et 15. Les lignes 1030 à 1060 affichent le caractère

graphique dont le code est x (voir le tableau des caractères de la page 51). BEEP produit un son dont la hauteur est en rapport avec la position et RETURN à la ligne 1080 renvoie le programme à l'instruction qui suit GOSUB de la ligne 50.

La ligne 60 change la couleur de l'encre et du papier, puis PAUSE 100 à la ligne 70 retarde le programme de deux secondes avant qu'il ne se reboucle pour recommencer. STOP est nécessaire à la ligne 90 pour empêcher le programme de passer directement dans le sous-programme après la quatrième boucle.

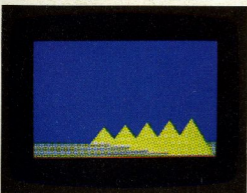
Vous pouvez changer le programme en remplaçant 4 dans la ligne 30 et 40 dans la ligne 50 par d'autres nombres. Si vous allonger la gamme des x de la ligne 40, d'autres caractères apparaîtront sur l'écran (i et p ne doivent pas être supérieurs à 7).

Comment utiliser les boucles FOR NEXT dans les graphiques

Les boucles FOR NEXT peuvent être utilisées avec beaucoup de succès dans les graphiques à haute résolution pour produire des images. Introduisez par le clavier le programme ci-dessous et exécutez-le. En n'utilisant que PLOT et DRAW, les deux boucles FOR NEXT commencent par tracer des lignes sur le sol, puis cinq triangles ou pyramides pleins.

PYRAMIDES

```
10 BORDER 0: PAPER 1: INK 6
20 CLS
30 FOR y=0 TO 20 STEP 2
40 PLOT 0,y
50 DRAW 255,0
60 NEXT y
70 FOR n=100 TO 220 STEP 30
80 FOR x=-10-n/10 TO 10+n/10
90 PLOT n,35+n/10
100 DRAW x,n/4
110 NEXT x: NEXT n
```



Ajoutez maintenant les lignes de la colonne suivante. Lorsque vous repassez le programme, vous verrez qu'un rayon laser traverse sans interruption le ciel sombre, semant des myriades d'étoiles. Il est dessiné à partir du coin de l'écran jusqu'à la position x, y, les variables x et y étant des nombres aléatoires. Ceux-ci sont transformés en coordonnées de l'étoile à basse résolution.

```
120 LET X=RND*255
130 LET Y=RND*104+71
140 LET L=INT(175*(Y)/8)
150 LET C=INT(X/8)
160 PLOT 0,0: DRAW OVER 1,x,y
170 BEEP 0,01*X/4
180 PLOT 0,0: DRAW OVER 1,x,y
190 PRINT AT 1,C:"*"
200 GO TO 120
```



OVER 1 dans les lignes 160 et 180 permet à la première ligne de dessiner un rayon laser et à la seconde ligne d'enlever ce dernier sans changer le reste de l'image. Conservez ce programme, vous en aurez besoin par la suite.

FLASH, BRIGHT et INVERSE

Grâce à ces trois mots-clés, vous pourrez exploiter au maximum les couleurs de votre Spectrum. Chaque mot-clé est suivi de 0 ou de 1 et vous pouvez les placer dans les instructions PRINT à condition de faire suivre 0 ou 1 d'un point-virgule. FLASH 1 fait clignoter la position des caractères entre les couleurs de l'encre et du papier et BRIGHT 1 rend les couleurs plus brillantes. INVERSE 1 remplace la couleur du papier par celle de l'encre et vice-versa. Si vous utilisez 0 après ces mots-clés, l'image redevient normale.

Essayez d'introduire ces changements dans les programmes de ces deux pages pour voir comment les mots-clés agissent. Dans le programme "Carrés", remplacez le carré de la ligne 60 par une étoile, puis ajoutez

15 INVERSE 1

Les étoiles sont maintenant noires (la couleur du papier) et ressortent sur des bandes de couleur (les couleurs changeantes de l'encre). Entrez INVERSE 0 avant de continuer.

Dans le programme de "Formes symétriques", ajoutez les lignes ci-dessous pour voir comment BRIGHT et FLASH agissent.

15 BRIGHT 1

16 FLASH 1

Notez que FLASH semble faire bouger la forme. Pour que l'image s'arrête de clignoter, entrez FLASH 0: CLS. Tous ces changements modifient toute l'image produite par chaque programme. Si vous utilisez FLASH, BRIGHT ou INVERSE sans l'ordre PRINT, les trois mots-clés sont restreints à ce qui est imprimé par cette ligne.

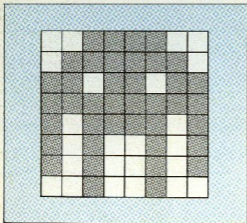
COMMENT CREER DES CARACTERES MACHINE

Votre ZX Spectrum + ne se contente pas de produire les caractères graphiques que vous introduisez par le clavier. Il peut également stocker les caractères que vous avez dessinés vous-même dans une section spéciale de sa mémoire. Ces derniers s'appellent des caractères graphiques utilisateurs et chaque programme peut en compter un maximum de 21.

Chaque caractère est composé de 64 petits points ou pixels de la couleur de l'encre, disposés en huit rangs de huit pixels chacun. Chaque caractère occupe une position de caractère sur la grille à basse résolution.

Dessinez un caractère graphique

Commencez par tracer une grille de 8x8 comme ci-dessous. Ensuite, remplissez certains des carrés pour créer le caractère. Ces carrés représentent les pixels de la couleur de l'encre. Imaginez que chaque carré plein est 1 et chaque carré vide 0. Voici le dessin d'une araignée.



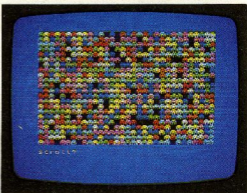
Chaque caractère graphique utilisateur est identifié par une lettre, de a à u (ou A à U — c'est la même chose). Pour programmer le caractère, vous entrez huit instructions POKE USR, se terminant chacune par BIN, suivi d'un nombre binaire composé des huit 0 et 1 des rangs de la grille. Appelons l'araignée s.

```
10 POKE USR "S",BIN 0111100
11 POKE USR "S",BIN 11011110
12 POKE USR "S",BIN 10110111
13 POKE USR "S",BIN 11111111
14 POKE USR "S",BIN 10111101
15 POKE USR "S",BIN 10100101
16 POKE USR "S",BIN 10100101
17 POKE USR "S",BIN 00100100
```

Exécutez ce programme, puis appuyez sur GRAPH et sur S. Au lieu d'un "S", c'est l'araignée qui apparaît! Puis ajoutez les lignes suivantes pour obtenir l'araignée à la ligne 30 et exécutez le programme. L'écran est alors couvert d'araignées.

ARAIGNEES

```
20 BORDER 1: PAPER 0: CLS
30 PRINT INK AND #7: "S";
40 GO TO 30
```



Lorsque vous dessinez vos propres caractères, souvenez-vous que vous ne pourrez pas les voir sur l'écran avant d'avoir exécuté un programme qui les définit. Jusqu'à ce moment-là, ils apparaîtront en tant que lettres dans les listages.

Comment colorier avec des carrés tachetés

Il est facile de mélanger des couleurs sur votre Spectrum. Vous devez créer un caractère qui lorsqu'il est imprimé, produit un carré à 50% de la couleur de l'encre et à 50% de la couleur du papier. Il vous suffit de définir deux lignes de pixels, puis de donner à l'ordinateur

```
10 FOR X=0 TO 255 STEP 2
20 POKE USR "S",BIN 10101010
30 POKE USR "S",BIN 01010101
40 NEXT X
```

l'instruction des les utiliser en alternance dans un caractère. Quand vous exécutez le programme vous devez voir un carré tacheté. Si vous utilisez la même technique dans un programme qui contient des mots-clés de couleur, le carré tacheté produira une couleur qui sera un mélange de la couleur du papier du programme et de la couleur de l'encre.

Simplifiez les graphiques à l'aide de READ et de DATA

Il est plus facile de créer vos caractères machine en utilisant des nombres décimaux avec READ et DATA. Commencez par remplacer les huit nombres binaires par des nombres décimaux en entrant PRINT BIN suivi du nombre, par exemple:

PRINT BIN 00111100

Le Spectrum affiche 60, l'équivalent décimal de 00111100. Pour l'araignée, les huit nombres décimaux sont 60, 126, 219, 255, 189, 165, 165 et 36.

Utilisez maintenant READ et DATA. Ces deux mots-clés vous permettent d'introduire facilement des tas de valeurs, des nombres par exemple, dans les variables d'un programme. READ est suivi d'une variable — une ou plusieurs lettres s'il s'agit de nombres ou une lettre suivie de \$ s'il s'agit de chaînes.

Lorsque votre Spectrum rencontre READ, il examine la première instruction DATA du programme. Cette instruction contient une liste de valeurs séparées par des virgules. L'ordinateur prend la première valeur de la liste et cette valeur est attribuée à la variable qui suit READ. Lorsque l'ordinateur arrive à READ pour la seconde fois, la seconde valeur est attribuée à la valeur, etc., le tout strictement par ordre.

Voici le nouveau programme qui produira l'araignée.

```

100000 FOR X=0 TO 7
100001 READ J
100002 POKE J,USR "S"+X.J
100003 NEXT X
100004 DATA 60,126,219,255,165,165,165,36

```

Le programme stockera huit nombres décimaux dans la mémoire pour créer un caractère. Remplacez "s" à la ligne 30 par la lettre que vous souhaitez et après DATA à la ligne 50, introduisez par le clavier les huit nombres, séparés par une virgule. Appuyez sur GRAPH et la lettre pour obtenir le caractère après avoir exécuté le programme.

Dessinez un échiquier

Voici un programme qui affiche un échiquier sur l'écran, puis dispose les pions qui vous permettront de jouer. Si vous voulez que des couleurs apparaissent dans un listage, utilisez les codes de contrôle des couleurs — voyez le paragraphe ci-dessous.

ECHQUIER

```

100000 PRINT "ORDER X=1 TO 8
100001 ORDER Y=1 TO 8
100002 FOR X=1 TO 8
100003   FOR Y=1 TO 8
100004     PRINT AT Y,X: "PAPER 1: CLS
100005   NEXT Y
100006 NEXT X
100007 RETURN

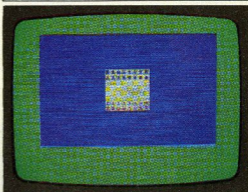
```

Les pions sont définis par un sous-programme commençant à la ligne 500.

```

500 DATA "P",0,0,16,56,56,16,12
400 DATA "R",0,64,124,56,56,124
410 DATA "N",0,16,56,120,24,56,
10040 DATA "B",0,16,40,60,100,56,
10040 DATA "K",0,16,56,16,56,68,5
600 DATA "Q",0,64,40,16,108,124
124

```



Utilisez les codes de contrôle des couleurs

Au lieu d'utiliser des mots-clés comme INK et PAPER, vous pouvez introduire des codes de contrôle dans les instructions PRINT après le premier guillemet. Les caractères qui

seront affichés changent de couleur dans le listage et apparaissent sur l'écran dans cette couleur.

Pour obtenir les codes, appuyez sur EXTEND MODE, puis sur une

touche numérique avec ou sans CAPS SHIFT. Le tableau ci-dessous vous montre comment sélectionner une couleur.

Appuyer sur

1 2 3 4 5 6 7 8 9 0

EXTEND MODE CAPS SHIFT

BLUE INK RED INK MGNTA INK GREEN INK CYAN INK YELLOW INK WHITE INK FLASH OFF FLASH ON BLACK INK

EXTEND MODE

BLUE PAPER RED PAPER MGNTA PAPER GREEN PAPER CYAN PAPER YELLOW PAPER WHITE PAPER BRIGHT OFF BRIGHT ON BLACK PAPER

ANIMATION

Les caractères machine sont mis en valeur lorsque les caractères ou les lignes bougent sur l'écran et il n'est pas très difficile d'animer les caractères sur votre Spectrum. Il vous suffit de continuer à changer la position à laquelle un caractère est imprimé ou une ligne tracée. La meilleure méthode est d'utiliser une ou plusieurs boucles FOR NEXT.

Mouvement vertical et horizontal

Introduisez ce programme par le clavier et exécutez-le. Si vous n'avez pas remis votre Spectrum à l'état initial ou si vous ne l'avez pas débranché depuis que vous avez produit le graphique de l'araignée de la page 32, vous n'avez pas besoin d'entrer les lignes 10 à 50. Le graphique sera encore dans la mémoire, sous "s".

ARAIGNEE SUSPENDUE A SON FIL

```

5 BORDER 0: PAPER 5: CLS
10 FOR X=0 TO 9
20 READ Y
30 POKE USR "s"+X,Y
40 NEXT X
50 DATA 60,120,210,255,180,185
105,30
60 FOR X=0 TO 7
70 READ Y
80 POKE USR "t"+X,Y
90 NEXT X
100 DATA 16,16,16,16,16,16,16,1
110 FOR I=0 TO 20
120 PRINT AT I,3,"INK 0:"I"
130 PRINT AT I+1,3,"INK 2:""
140 NEXT I

```

Chaque fois que vous passez le programme, vous verrez l'araignée descendre le long de son fil.

Dans le programme, les lignes 60 à 100 produisent un autre caractère graphique ("t") correspondant au fil. L'animation se produit aux lignes 110 à 140 qui constituent une boucle FOR NEXT dans laquelle I (le numéro de ligne) passe de 0 à 20. Chaque fois que la boucle recommence, une longueur de fil est imprimée à une position et l'araignée est imprimée à la position en-dessous. La fois suivante, l'araignée est remplacée par une autre longueur de fil, puis réapparaît en dessous. De cette manière, l'araignée descend ainsi rapidement le long de son fil jusqu'au moment où elle atteint la ligne 21, c'est-à-dire le bas de la zone d'affichage.

Pour ralentir l'action, introduisez la ligne suivante.

135 PAUSE 10

L'ordinateur attend alors un cinquième de seconde avant d'imprimer l'araignée à la position

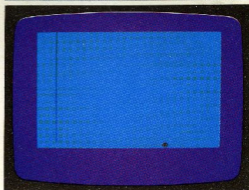
suivante. Essayez de changer 10 par d'autres valeurs et notez les changements de vitesse. Ajoutez maintenant les lignes suivantes.

ARAIGNEE EN FUITE

```

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

```



L'araignée file sur le côté dès qu'elle touche le bas de l'écran. Les lignes supplémentaires forment une autre boucle FOR NEXT qui contrôle "c", la position de la colonne. Notez qu'un espace est imprimé et que l'araignée est ensuite affichée à la position de colonne suivante. C'est ainsi que l'araignée disparaît à une position pour réapparaître à la suivante, déplaçant la position vers la droite. Il vaut toujours mieux effacer un caractère avant de l'imprimer à la position suivante. Ceci empêche ou réduit le tremblement des graphiques animés.

EXERCICES

Dans beaucoup de jeux machine, il arrive souvent que deux formes en mouvement entrent en collision ou qu'un objet soit touché par un rayon.

Il n'est pas difficile de prévoir les collisions. Si deux caractères sont imprimés aux positions I, c (ligne et colonne) et aux positions v, h (vertical et horizontal), et si I=v et c=h, les deux caractères doivent être à la même position. Vous pouvez écrire ceci comme une instruction, par exemple

```
160 IF I=v AND c=h THEN PRINT "BOUM!"
```

Une autre façon de prévoir une collision est d'utiliser les couleurs. Enlevez le programme "araignée" en entrant NEW. Puis entrez le programme complet de la page 31 ou bien chargez le programme si vous l'avez sauvegardé sur bande. Vous pouvez maintenant le faire évoluer et le combiner à votre araignée (qui se trouve encore dans la mémoire sauf si vous avez remis l'ordinateur à l'état initial ou si vous l'avez débranché) pour produire un autre programme.

Ajoutez les lignes suivantes qui produisent une explosion du graphique appelée "e"

```

100 FOR x=0 TO 7
101 READ v
102 POKe USR "e"+x.v
103 NEXT v
104 DATA 145,92,44,121,150,52,7
4,137

```

Effacez la ligne 190 et ajoutez ou changez les lignes suivantes

ARAIGNEES ET PYRAMIDES

```

114 LET h=RND*31
115 FOR v=0 TO 20
117 PRINT AT v,h: " " ; AT v+1,h:
118 INK 4: "S"
119 NEXT v
120 PRINT AT 21,h: FLASH 1: INK
121 PAPER 6: "E"
122 GO TO 114

```



Les étoiles n'apparaissent plus, mais sont remplacées par des araignées qui tombent du ciel et dévorent les pyramides et le sol. Ce qui s'est passé est ce que vous avez ajouté une boucle FOR NEXT dans laquelle v et h donnent la position de l'araignée. La variable h est aléatoire, c'est pourquoi les araignées apparaissent à plusieurs endroits sur l'écran. Ajoutez ensuite les lignes suivantes.

EXPLOSION D'ARAIGNEES

```

190 IF ATTR (v+1,h)=14 THEN GO
191 TO 500
192 PRINT AT v+1,h: FLASH 1: PA
193 PER 2: "E"
194 PAUSE 100
195 GO TO 114

```



Lorsque les rayons laser touchent les araignées, elles deviennent jaunes pendant quelques instants. Lorsque la ligne produite par DRAW pénètre dans la position d'un caractère de l'araignée, l'encre prend la couleur de la ligne qui est jaune. A la ligne 190, ATTR détecte si l'araignée est devenue jaune et envoie l'ordinateur au sous-programme "explosion" à la ligne 500.

Faites rebondir une balle

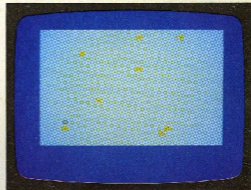
Beaucoup de programmes contiennent des graphiques qui rebondissent sur les bords de l'écran. Ce programme vous explique ce que vous devez faire. Les variables v et h ont la même fonction que dans le programme "explosion d'araignées", mais +1 ou -1 est ajouté à v ou h pour faire monter et descendre la balle ou pour la faire aller de droite à gauche. SCREENS vérifie s'il y a un X aux positions v, h.

BALLE QUI REBONDIT

```

10 BORDER 1
20 FOR z=1 TO 10
30 LET v=INT (RND*25): LET w=1
40 PRINT INK 2: PAPER 6: FLASH
41 AT v,h: "X"
50 NEXT z
60 LET x=1: LET y=1
70 PRINT AT v,h:
80 LET v=v+w: LET h=h+x
90 IF h=0 OR h=31 THEN LET x=-
x: BEEP .2,24
100 IF v=0 OR v=21 THEN LET y=-
y: BEEP .2,12
110 IF SCREENS (v,h)="X" THEN P
111 PRINT INK 1: PAPER 5: AT v,h: "+"
112 STOP
120 PRINT AT v,h: "O"
130 PAUSE 20
140 GO TO 20

```



Utilisez des attributs

Le mot-clé ATTR détecte les "attributs" à une position déterminée sur l'écran. Les attributs sont la couleur de l'encre et du papier et le fait qu'une position est brillante ou clignotante. ATTR permet de détruire l'araignée lorsqu'elle devient jaune. Jaune est alors la couleur de l'encre (numéro 6) et bleu est la couleur du papier (numéro 1). Ceci signifie que ses attributs sont 14 au total. L'entrée du guide du programmeur vous explique comment l'on obtient ces nombres.

COMMENT PRODUIRE DES EFFETS SONORES ET MUSICAUX

Le ZX Spectrum + possède un synthétiseur de sons qui animera vos programmes grâce à des musiques variées et des effets sonores spéciaux. Il est facile à utiliser même si vous ne connaissez pas grand-chose à la musique. Le synthétiseur produit un signal sonore qui va jusqu'au haut-parleur intérieur du Spectrum.

Programmez des sons

Pour produire des sons sur votre Spectrum, vous n'utilisez qu'un mot-clé — BEEP. Il est suivi de deux nombres ou de deux variables représentant ces nombres. Le premier dit à l'ordinateur combien de temps (en secondes) le son doit durer et le second l'informe de la hauteur du son. Cette hauteur est mesurée en demi tons. La hauteur prend la valeur 0 pour l'ut moyen, 1 pour ut dièse, — 1 pour B (ut bémol), etc. Exécutez ce programme.

```
10 FOR P=60 TO -60 STEP -1
20 BEEP 0.05,P
30 PRINT "UT",P;" " ; AT 0.01 P
40 NEXT P
```

Le Spectrum exécute la gamme complète de notes, de la plus haute (-69) à la plus basse (-60). Vous verrez que les notes les plus hautes

sont pratiquement inaudibles et que les plus basses ressemblent à des "clics".

Le schéma du bas de la page vous donne les hauteurs d'une gamme de note ce qui vous permettra de transformer une partition en un programme Spectrum.

Effets sonores

Votre Spectrum peut produire toutes sortes d'effets sonores, en règle générale en plaçant BEEP dans une boucle qui change rapidement la hauteur des notes. Essayez ces programmes et notez que la durée est très courte, pas plus d'un centième de seconde. Appuyez sur BREAK pour arrêter les programmes bouclés.

SONS PERLES

```
10 LET P=INT (RND*40)-20
20 BEEP 0.05,P: BEEP 0.05,P+7:
30 GO TO 10
```

Ce programme joue sans arrêt un groupe de trois notes, à des tons choisis au hasard. Les tons sont nombreux, mais vous pouvez les changer en modifiant les valeurs de la ligne 10.

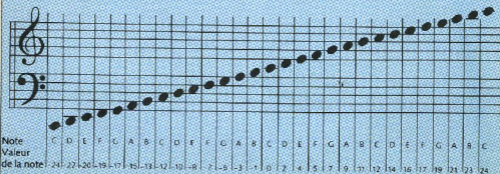
MACHINE

```
10 FOR X=12 TO 36
20 BEEP .01,X
30 BEEP .01,24-X
40 NEXT X
50 GO TO 10
```

Ce programme produit deux sons, un dont le son monte pendant que celui de l'autre descend. C'est parce que les deux instructions BEEP produisent sans arrêt deux notes à des hauteurs différentes, séparées par un centième de seconde seulement.

Hauteurs des notes produisant de la musique

Voici les hauteurs des notes, du bas | hauteur de la note pour une note en haut de la portée. Ajoutez 1 à la | laigüe, otez 1 pour une note basse.



DECOLLAGE

```
10 FOR P=1 TO 48 STEP 0.2
20 BEEP .01,P: BEEP .01,P-.6
30 NEXT P
```

Ce programme est similaire au programme machine, mais les deux notes montent ensemble, séparées par six demi-tons. En outre, les hauteurs changent de 0,02 — un cinquième de demi-ton à chaque fois. Le son devient ainsi progressivement de plus en plus aigu. Essayez de modifier légèrement la hauteur des notes en changeant la valeur STEP.

CONVERTISSEUR DU CLAVIER

```
10 LET P=CODE INKEY$
15 IF P=0 THEN GO TO 10
20 BEEP .04,(P-30)/2
30 GO TO 10
```

Ce programme vous permet d'appuyer sur n'importe quelle touche. A ce moment-là, chaque touche produit un son différent. Notez qu'appuyer sur CAPS SHIFT tout en maintenant une autre touche enfoncée produit un son plus grave. Ce programme fonctionne parce que CODE INKEY\$ attribue une valeur différente à p chaque fois que vous appuyez sur une nouvelle touche. La seconde ligne empêche l'ordinateur d'émettre un son si aucune touche n'est pressée. Vous pouvez voir les valeurs que CODE renvoie dans le tableau des caractères page 51.

Son et vision

Les effets sonores produits par votre Spectrum servent d'accompagnement idéal à l'action que vous voyez sur l'écran. Pour voir comment ajouter le son à des programmes, consultez le programme complet "araignée en fuite" de la page 34.

Souvenez-vous que vous avez introduit une instruction PAUSE à la ligne 135 pour ralentir l'action. Au lieu de ralentir le programme de cette manière, vous pouvez programmer une pause qui produit un son. Remplacez la ligne 135 par

135 GOSUB 500

Ajoutez maintenant les lignes suivantes au programme.

```
200 STOP
300 FOR P=40-1 TO 38-1 STEP -.1
510 BEEP 0.02,P
520 NEXT P
530 RETURN
```

Exécutez le programme: l'araignée produit maintenant un son mélodieux en descendant le long de son fil. Le sous-programme joue trois notes très rapidement, notes qui deviennent de plus en plus graves au fur et à mesure que l'araignée descend d'une position sur l'écran. Essayez d'ajouter des notes en changeant la ligne 500 et d'accélérer ou de ralentir les notes en changeant 0,02 à la ligne 510.

Comment amplifier votre Spectrum

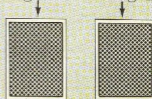
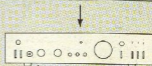
Pour amplifier le son produit par votre Spectrum, vous pouvez connecter l'embase EAR aux écouteurs ou à un amplificateur ou à un haut-parleur.

La façon la plus simple de procéder est d'utiliser le cordon du magnétophone pour connecter l'embase EAR ou MIC à l'embase MIC d'un magnétophone à cassette. Si nécessaire, enlevez la cassette, mettez en route le magnétophone à cassette et appuyez sur PLAY, REWIND

(REVERSE) ou FAST FORWARD (CUE). Manoeuvrez le bouton de contrôle du volume et vous devriez entendre le son produit par l'ordinateur provenant du haut-parleur du magnétophone. Si vous le souhaitez, vous pouvez également utiliser les écouteurs avec le magnétophone à cassette.

Si vous voulez faire beaucoup de bruit, vous pouvez aussi connecter votre Spectrum à un système haute fidélité. Vous aurez besoin d'un cordon spécial muni d'un jack de

3,5mm s'adaptant au Spectrum et d'un jack que vous introduirez dans l'embase d'entrée de l'amplificateur de votre système haute fidélité. Le Spectrum produit un signal linéaire similaire à celui produit par des lecteurs de cassette ou des magnétophones, de telle sorte que l'embase REPLAY ou LINE IN de l'amplificateur devrait faire l'affaire.



SAUVEGARDER VOS PROGRAMMES

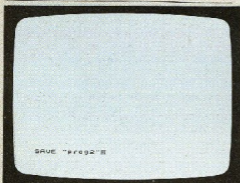
Vous aurez bientôt envie de stocker vos programmes sur bande magnétique. Pour cela,

vous devez connecter un magnétophone à cassette à votre Spectrum et sauvegarder le programme qui se trouve dans l'ordinateur. Chaque fois que vous avez besoin du programme, vous le transférez de la bande magnétique à l'ordinateur en suivant la procédure de chargement décrite aux pages 14 et 15. Ces deux pages vous expliquent comment sauvegarder vos programmes et comment vérifier s'ils ont été correctement sauvegardés.

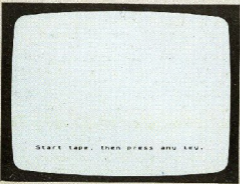
Enregistrez vos propres programmes

- 1 Connectez votre Spectrum à un magnétophone à cassette à l'aide du cordon à magnétophone, comme décrit à la page 14, mais assurez-vous que seule l'embase MIC du Spectrum est connectée au magnétophone à cassette.
- 2 Si le magnétophone à cassette est muni d'un bouton de réglage du volume ou du niveau d'enregistrement, réglez-le à environ deux tiers du maximum. S'il n'y a pas de bouton de réglage, ne vous en faites pas, le niveau d'enregistrement se réglera automatiquement.
- 3 Introduisez SAVE, puis le nom du programme entre guillemets, par exemple SAVE "prog 2".

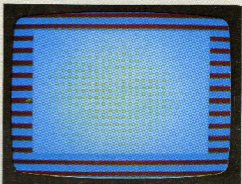
Sauvegarder "prog 2"



Vous pouvez utiliser n'importe quelle combinaison de lettres et de nombres ne dépassant pas dix. La ligne SAVE disparaîtra, remplacée par le mode d'emploi du magnétophone.

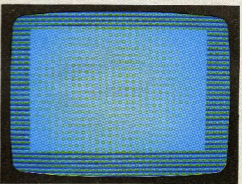


- 4 Mettez le magnétophone à cassette sur "enregistrement", ce qui se fait habituellement en appuyant en même temps sur RECORD et PLAY. Ensuite, appuyez sur n'importe quelle touche du Spectrum.
- 5 Attendez que le Spectrum ait terminé de sauvegarder le programme. Vous verrez d'abord des bandes bleues et rouges se déplaçant lentement sur l'écran.



Ensuite, des raies bleues et jaunes clignoteront sur l'écran. Elles signifient que le Spectrum a envoyé le nom du programme à la bande.

- 6 Après un bref intervalle, des bandes bleues et rouges réapparaissent. Elles sont suivies de raies bleues et jaunes qui indiquent que le Spectrum est en train d'envoyer le programme à la bande. Un long programme peut prendre plusieurs secondes.



- 7 Une fois que le programme a été envoyé à la bande, l'état OK,0,1 apparaît. Arrêtez la bande. Le programme est maintenant sauvegardé. Si vous le souhaitez, vous pouvez maintenant vérifier qu'il a bien été sauvegardé.

Vérifiez votre programme

Bien que l'ordinateur ait envoyé le programme au magnétophone à cassette, vous n'êtes pas certain que le programme a bien été enregistré sur la bande. Heureusement, votre Spectrum peut vérifier la chose pour vous.

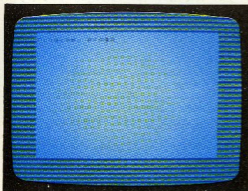
Cette procédure s'appelle la vérification. Rembobinez la bande jusqu'au début du programme, puis connectez l'embase EAR du Spectrum à l'embase EAR du magnétophone. (Vous pouvez laisser les embases MIC connectées.) Ensuite, introduisez par le clavier VERIFY, suivi du nom du programme entre guillemets. Puis, appuyez sur ENTER et mettez la bande en route. Vous devriez voir la même séquence de bandes bleues et rouges et de raies bleues et jaunes. Le nom du programme apparaîtra et restera sur l'écran jusqu'à ce que la vérification soit terminée.



Lorsque les secondes raies bleues et jaunes représentées ci-dessous disparaissent, l'état

OK, 0:1

devrait apparaître. Cela signifie que votre Spectrum a comparé le programme qui se trouve sur la bande à celui qui se trouve dans sa mémoire et s'est rendu compte qu'ils étaient identiques.



Le programme a été vérifié positivement. Vous pouvez être certain que votre programme est en sécurité sur la bande.

Sauvetage du logiciel: quelques petits trucs.

1. Lorsque vous sauvegardez un programme, inscrivez son nom sur l'étiquette ou la carte de la cassette. Utilisez les minuscules et majuscules qui apparaissent sur l'écran. Si le magnétophone à cassette a un compteur, repérez l'emplacement du programme et inscrivez le chiffre qui apparaît sur le compteur à côté du nom du programme.
2. Avant de sauvegarder un programme, introduisez le nom du programme dans le programme à l'aide d'une instruction REM, par exemple

```
5 REM SPIDER program version 3
```

L'ordinateur ignore toutes les instructions REM pendant que le programme passe et vous pouvez utiliser REM pour introduire des remarques ou des rappels dans le programme chaque fois que vous le souhaitez.

Si vous n'obtenez pas cet état, il y a un problème quelque part. Commencez par jeter un coup d'oeil à la page 16, "Chargement du logiciel: dépannage". Il se peut que le programme soit correctement sauvegardé sur la bande, mais ne se recharge pas dans l'ordinateur pour être vérifié. Si c'est là que le problème se situe, corrigez-le, rebobinez la bande et vérifiez à nouveau le programme. Si l'ordinateur ne vérifie toujours pas le programme, consultez "Sauvetage du logiciel: dépannage" à la page suivante. N'appuyez pas sur NEW, ne remettez pas l'ordinateur à l'état initial et ne le débranchez pas sous peine de perdre le programme qui se trouve dans sa mémoire, sans en avoir une copie valable sur bande.

Mise en route automatique d'un programme

Vous pouvez faire suivre SAVE du nom du programme, puis de LINE 1, par exemple

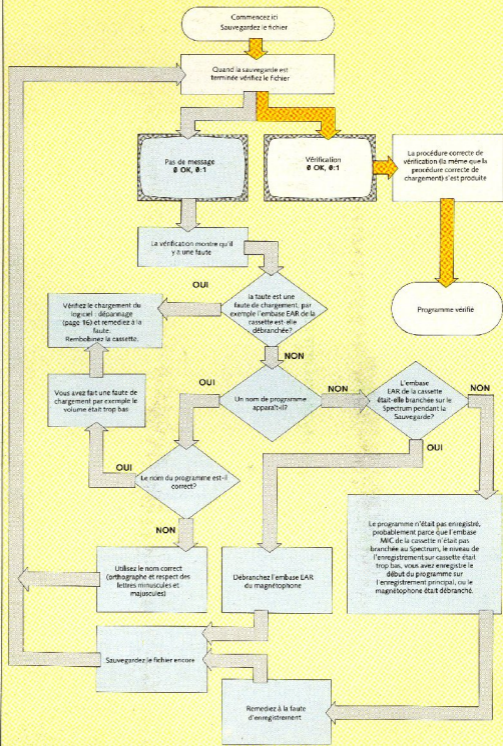
SAVE "SPIDER" LINE 1

La procédure de sauvetage est la même que précédemment, mais lorsque vous effectuez la vérification, n'ajoutez pas LINE 1 à la suite de VERIFY et du nom du programme. Les programmes sauvegardés avec LINE 1 se mettront automatiquement en route lorsque vous les chargez. Il n'y a pas besoin d'utiliser RUN (mais n'oubliez pas d'arrêter la bande lorsque le programme se met en route).

Ce qui se passe, c'est que le programme commence à la ligne 1 et que s'il n'y a pas de ligne 1, l'ordinateur saute la première ligne du programme. Remplacer 1 par un autre numéro signifie que le programme commence automatiquement à la ligne portant ce numéro.

Sauvetage de CODE, SCREEN\$ et DATA

Vous pouvez maintenant être sûr que votre programme est bien sauvegardé sur la bande. SAVE peut également être utilisé avec CODE ou SCREEN\$ pour stocker une section de la mémoire du Spectrum et avec DATA pour stocker un ensemble de données. Voir les entrées sur SAVE CODE, SAVE SCREEN\$ et SAVE DATA dans le guide du programmeur.



FAMILIARISEZ-VOUS AVEC VOTRE ZX SPECTRUM +

Ce chapitre vous emmène à l'intérieur de votre ZX Spectrum + et vous explique comment les divers composants qui se trouvent en-dessous du clavier fonctionnent et comment ils sont reliés les uns aux autres pour que l'ordinateur puisse fonctionner. Il vous explique également comment utiliser les "périphériques" — les éléments à ajouter qui transforment votre Spectrum en un système informatique complet. Enfin, il vous permettra de vous familiariser avec l'aspect technique de votre ordinateur — y compris l'organisation de sa mémoire et la fiche technique du Spectrum.



QUE CONTIENT-IL?

Pour le savoir, lisez les pages qui suivent — sans essayer d'ouvrir votre ZX Spectrum + pour voir comment il fonctionne. Votre garantie ne serait alors plus valable et vous risqueriez de l'amimer sérieusement.

A l'intérieur du boîtier se trouvent deux connecteurs flexibles qui relient le clavier au reste des composants du Spectrum. Tous les composants sont montés sur une même carte de circuit imprimé. Sur la carte se trouvent des éléments électriques tels que des résistances et des condensateurs, mais les éléments les plus importants sont les microchips noirs rectangulaires, isolés ou disposés en blocs.

Que contient un chip?

La partie active d'un chip est en fait beaucoup plus petite que le boîtier en plastique qui le contient. Le boîtier est essentiellement destiné à servir de support aux connexions que le chip exige et qui lui permettent d'être branché dans les embases de la carte de circuit imprimé. Le chip en lui-même est une mince plaquette en silicium contenant plusieurs milliers de connexions électriques. Chaque connexion sert de contact arrêtant, passant ou stockant les signaux qui l'atteignent. Bien que la procédure soit simple, il y a tellement de connexions qui agissent ensemble qu'elles peuvent produire des signaux qui stockeront et traiteront les informations avec une rapidité et une précision étonnantes.

Comment les chips sont-ils reliés les uns aux autres?

En gros, votre Spectrum est donc un circuit électrique d'une immense complexité. Des signaux codés se composent d'impulsions électriques clignotent sans arrêt le long des parcours à l'intérieur des chips et composants et entre eux pour permettre à l'ordinateur de fonctionner.

Comment se fait-il que l'ordre règne et que le signal voulu arrive à l'endroit voulu au moment voulu? Dissimulée dans l'un des chips se trouve l'horloge de l'ordinateur. Elle fonctionne en émettant des impulsions électriques — 3,5 millions par seconde. Ces impulsions se déplacent régulièrement dans les circuits et produisent les signaux codés qui contrôlent chacun des éléments et synchronisent le tout.

L'intérieur de votre ZX Spectrum +
Sur cette illustration de la carte de circuit du Spectrum, les deux connecteurs flexibles et le clavier ont été enlevés.

Lorsque le Spectrum fonctionne, appuyer sur une touche met en contact une paire de fils se trouvant sous le clavier. Ceci envoie un signal codé à l'unité centrale.

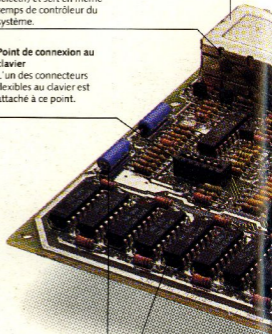
Logique non affectée (ULA)

Ce chip produit le contenu de l'écran à partir d'informations stockées dans la RAM (mémoire à accès sélectif) et sert en même temps de contrôleur du système.

Sortie TV
Ceci produit le signal qui va au téléviseur.

Point de connexion au clavier

L'un des connecteurs flexibles au clavier est attaché à ce point.



Codeur TV

Il transforme les signaux produits par les circuits de l'ordinateur en un signal de télévision en couleurs.

Mémoire à accès sélectif (RAM)

Ces chips contiennent le programme introduit dans l'ordinateur et les informations particulières nécessaires au programme, par exemple les valeurs attribuées aux variables. Le contenu des 48K de la RAM peut être changé à partir du clavier ou peut être effacé en remettant l'ordinateur à l'état initial ou en l'arrêtant.

Embases du magnétophone

Elles servent à transmettre les informations et les programmes de la mémoire à une bande et à les réintroduire dans la mémoire à partir de la bande.

Chips logiques

Ces chips servent d'interface lors de l'échange d'informations entre l'unité centrale et la RAM.

Unité centrale (CPU)

Le "cerveau" de l'ordinateur. L'unité centrale est un microprocesseur Z80. Il effectue tous les calculs de la machine et contrôle le fonctionnement global du Spectrum.

Embase 9 VDC

Elle permet de brancher l'ordinateur à l'alimentation électrique.

Connecteur latéral

Il permet de connecter le Spectrum à des machines extérieures, comme une imprimante.

Mémoire inaltérable (ROM)

C'est la section 16K de la mémoire contenant les instructions inaltérables nécessaires à l'unité centrale. Entre autres, ces instructions transforment les programmes BASIC de façon à ce que l'unité centrale puisse les comprendre. Il n'est pas possible de modifier le contenu de cette mémoire à partir du clavier.

Point de connexion au clavier

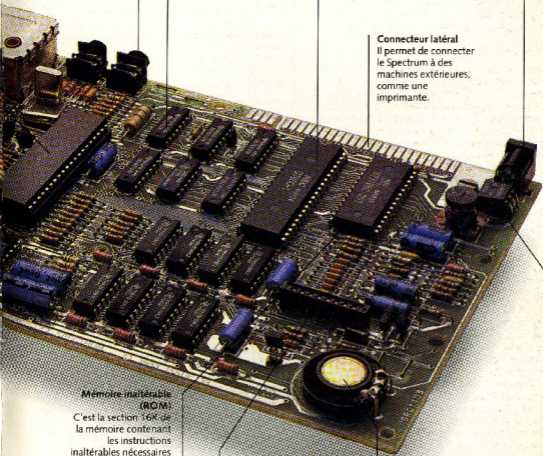
L'un des connecteurs flexibles au clavier est attaché à ce point.

Haut-parleur

Il transmet le son au moment voulu.

Régulateur de tension

Ce composant empêche les variations de tension d'affecter l'ordinateur.



COMMENT VOTRE ZX SPECTRUM + FONCTIONNE-T-IL ?

Comme d'autres micro-ordinateurs, le ZX Spectrum + se compose essentiellement de quatre parties. Il s'agit des *unités d'entrée*, tel le clavier, qui introduisent des informations ou un programme dans l'ordinateur; des *mémoires* temporaire et permanente qui stockent les informations, les programmes et les consignes d'exploitation; de l'*unité centrale* qui exécute les instructions d'un programme sur les informations et des *unités de sortie* qui produisent les résultats.

Entrez et exécutez un programme

Que se passe-t-il à l'intérieur du Spectrum lorsque vous entrez et exécutez un programme extrêmement simple? Voici un exemple d'une ligne

10 PRINT 6+2

Tout, d'abord, vous manœuvrez le clavier. En-dessous des touches se trouve un treillis de fils. Chaque fois que vous appuyez sur une touche, une paire de fils établissent un contact et transmettent un signal codé à l'unité centrale. L'unité centrale transmet alors le code à la RAM où il sera stocké.

Lorsque vous exécutez le programme, l'unité centrale extrait les codes stockés dans la RAM, un par un dans l'ordre du programme. En premier lieu, elle reçoit le code PRINT qui lui dit de lire un code d'exécution particulier de la ROM. Ce code d'exécution est transmis à l'unité centrale qui s'apprête à exécuter les opérations à la suite desquelles une valeur sera affichée sur l'écran. Ensuite, la RAM transmet la valeur 6 à l'unité centrale. Cette valeur se présente également sous la forme d'un code que l'unité centrale stocke dans une petite mémoire intérieure appelée un registre. Vient ensuite le code d'addition et une fois de plus, la ROM transmet le code d'exécution nécessaire à l'unité centrale. En dernier lieu, l'unité centrale lit le code pour 2 de la RAM. Il ajoute ce code à la valeur du registre pour obtenir le résultat (8). L'unité centrale transforme alors le résultat en une autre série de codes et le numéro 8 apparaît sur l'écran.

Stockez un programme

Si vous demandez au Spectrum de sauvegarder le programme sur bande, l'unité centrale lit les codes de la RAM. Mais au lieu d'effectuer un traitement sur ces codes, elle les transmet à un convertisseur qui les transforme en signaux sonores. Ces signaux sont alors transmis au magnétophone à cassette et enregistrés sur bande.

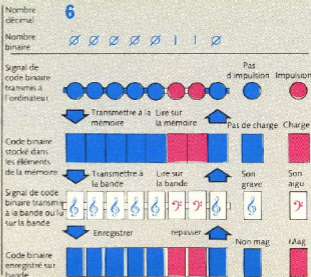
Plus tard, lorsque vous chargez le programme, les signaux sonores du magnétophone à cassette sont de nouveau transformés en codes machine par le convertisseur. L'unité centrale les retransmet à la RAM où ils restent stockés jusqu'au moment où ils seront utilisés.

Codes binaires

Tous les codes qui font fonctionner votre Spectrum se présentent sous forme binaire. Ils sont appelés binaires parce qu'ils ne sont composés que de deux types de signaux. Ils peuvent être représentés comme des nombres binaires, c'est-à-dire des nombres ne contenant que deux chiffres — 0 et 1. Le nombre binaire pour 6 par exemple est 00000110.

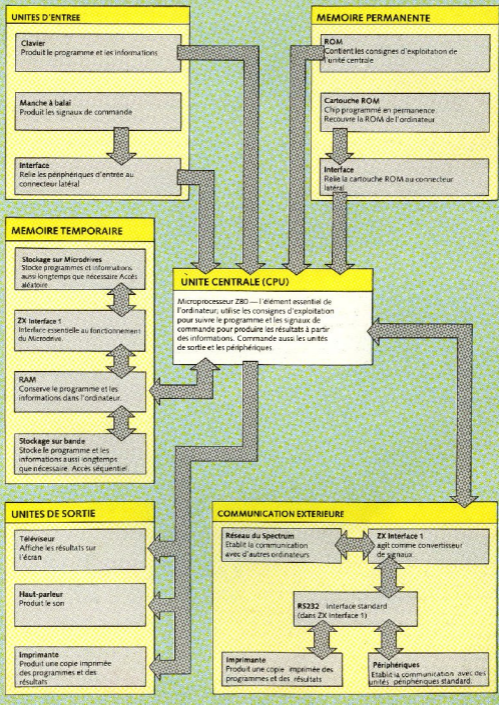
À l'intérieur de votre Spectrum, les codes se composent de séries d'impulsions électriques rapides. Lorsqu'une impulsion arrive à un point quelconque, cette soudaine pointe d'électricité représente un 1 dans le nombre binaire. Si une impulsion n'arrive pas dans un délai déterminé, elle représente un 0. En codes machine, 6 est donc *off-off-off-off-off-on-on-off*. Le Spectrum traite huit codes en même temps.

Le tableau vous montre comment l'ordinateur utilise différents types de codage binaire pour transférer des informations d'un endroit à un autre.



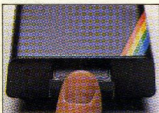
Le tableau montre comment des informations codées passent des unités d'entrée, le clavier par exemple, à travers le système de traitement du Spectrum pour arriver aux unités de sortie comme l'écran de télévision. Les flèches à une pointe représentent les

parcours qui ne fonctionnent que dans un sens. Les flèches à deux pointes indiquent les parcours qui peuvent fonctionner dans les deux sens.



COMMENT CONNECTER LES PERIPHERIQUES

Grâce à des périphériques Sinclair et Spectrum compatibles, vous pouvez transformer votre ZX Spectrum + en une puissante installation complète. L'élément essentiel de ce système est la ZX Interface 1 qui vous permet de connecter les Microdrives grâce auxquels le traitement des programmes et des données est chose facile et rapide; elle peut aussi être connectée à plusieurs autres périphériques, y compris d'autres Spectrum. Grâce à cette interface, vous pouvez connecter votre Spectrum à des imprimantes standard et connecter l'ordinateur à un modem. Il y a également d'autres interfaces qui connectent les cartouches ROM enfichables à l'ordinateur, ce qui permet de charger instantanément un programme. Grâce à elles, vous pouvez également fixer des manches à balais qui rendent les jeux plus agréables.



Chargez un Microdrive

Les cartouches des Microdrives sont introduites dans la fente se trouvant à l'avant du support.



Enfichez une cartouche ROM

Il suffit d'introduire la cartouche dans l'embase de l'interface. Lorsque l'ordinateur est branché, le programme est chargé automatiquement, ignorant la ROM qui se trouve à l'intérieur de l'ordinateur.

Imprimantes compatibles avec le Spectrum

Certaines imprimantes se branchent directement dans le connecteur latéral du Spectrum. Si vous avez une imprimante Sinclair ZX, par exemple, vous pouvez la connecter à l'ordinateur sans utiliser d'interface. Toutefois, pour utiliser les imprimantes qui exigent une sortie RS232, vous devez vous servir de l'embase D de la ZX Interface 1.

ZX Interface 1

La ZX Interface 1 est attachée à la base du Spectrum à l'arrière de celui-ci. Elle connecte votre Spectrum à jusqu'à huit Microdrives et 63 autres ordinateurs Spectrum et par l'intermédiaire de son interface standard RS232, à un large éventail de périphériques standard.

Pour stocker les programmes et les données, des Microdrives et des cartouches de Microdrive remplacent le magnétophone à cassette et les bandes. Une fois des cartouches de Microdrive introduites, vous pouvez sauvegarder, vérifier et charger des programmes en quelques secondes. Chaque cartouche peut stocker jusqu'à 85K de

Connexion des périphériques

La ZX Interface 1 est enfichée en permanence dans le connecteur latéral et se trouve donc en-dessous et à l'arrière de l'ordinateur. L'illustration représente le système avant que l'ordinateur ne soit connecté.



Microdrive
Huit de ces unités de stockage peuvent être attachées à un Spectrum.

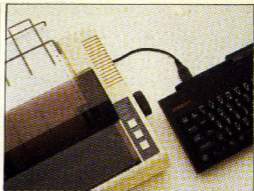
Flexible
Il relie le Microdrive à l'ordinateur, via la ZX Interface 1.

Le ZX Spectrum + possède deux pieds qui peuvent être utilisés pour incliner le clavier. Vous n'avez pas besoin d'utiliser ces pieds quand on se sert de l'Interface 1 ZX.

données et si huit Microdrives sont utilisés, votre Spectrum a une capacité de stockage en direct de 680K! N'importe quel programme est localisé automatiquement, le temps d'accès typique étant de 3,5 secondes.

A l'aide du câble de réseau dont est munie l'interface, vous pouvez connecter votre ordinateur à un autre — soit un ZX Spectrum ou un autre ZX Spectrum +. Ce réseau peut englober un maximum de 63 autres Spectrum. L'information est échangée entre eux à raison de 10 000 caractères à la seconde.

La ZX Interface 1 comprend également une interface RS232 munie d'une embase D à neuf directions. Un câble d'interface standard est également disponible.



Les interfaces standard sont connectées par l'intermédiaire de la ZX Interface 1.

Les cartouches et les manches à balai ROM

Des interfaces comme la ZX Interface 2 vous permettent de connecter des cartouches et manches à balai ROM. Les cartouches ROM permettent le chargement immédiat une fois que l'ordinateur est branché et produisent des programmes dont le chargement à partir d'une bande magnétique prendrait beaucoup de temps.

Interface de
cartouche/manche à balai
ROM

Connecteur latéral

Les périphériques sont connectés à l'ordinateur par l'intermédiaire de ce connecteur.



Attention

Les périphériques doivent toujours être connectés avant de brancher l'appareil.

LA TOPOGRAPHIE MÉMOIRE ZX SPECTRUM +

Si vous jetez un coup d'oeil à la photo de l'intérieur du Spectrum, pages 42-43, vous verrez qu'il y a un chip ROM et 16 petits chips RAM. Ces chips constituent la mémoire du Spectrum. La mémoire se compose de 65536 unités contenant chacune un octet (un nombre entre 0 et 255). Chaque unité est identifiée par un numéro que l'on appelle son *address*.

ROM signifie *mémoire inaltérable*. Cette partie de la mémoire contient les consignes d'exploitation provenant de l'unité centrale. Il s'agit d'une ROM de 16K, c'est-à-dire content 16 × 1024 (16384) octets ou adresses. Les octets ne peuvent être lus qu'à partir de cette mémoire et ne peuvent donc pas être modifiés. (Dans le cas contraire, l'ordinateur s'arrêterait de fonctionner.) Vous pouvez obtenir l'octet à une adresse à l'aide de PEEK.

RAM signifie *mémoire à accès sélectif* et elle contient les programmes et les informations introduits dans l'ordinateur. Le Spectrum a une RAM de 48K, c'est-à-dire contenant 48 × 1024 (49152) octets ou adresses. A accès sélectif signifie qu'un octet à une adresse peut être changé à l'aide de POKE.

Les adresses de la mémoire vont de 0 à 65535, un quart étant des adresses ROM et le reste des adresses RAM.

Les variables système

La colonne de droite vous montre comment la mémoire du Spectrum est organisée, autrement dit où se trouvent les diverses parties qui commandent l'ordinateur; les limites de ces dernières sont données par les *variables système*.

Les variables système du Spectrum ne sont pas identiques à celles utilisées en BASIC. Un nom a été donné à certaines valeurs utiles se trouvant à des adresses ou à des points d'implantation précis de la mémoire. Un nom vous aide à vous souvenir de la signification de la valeur stockée à un point particulier. Par exemple, la variable système RAMTOP est l'adresse supérieure de la RAM pouvant contenir un programme BASIC et les valeurs de ses variables. L'adresse de la RAMTOP est 23730.

Topographie mémoire du Spectrum

UDG Graphiques utilisateur		
Pile GOSUB	RAMTOP	
De réserve		
Pile de la calculatrice	STKEND	
Zone de travail temporaire	STKBOT	
Données INPUT		
Commande ou ligne en cours d'édition	WORKSP	
Variables	E LINE	
Programme BASIC	VARS	
Informations aux canaux	PROG	
Topogrammes du micro-support	CHANS	
Variable système	23734	
Tampon de l'imprimante	23552	
Attributs	23296	
Fichier d'affichage	22528	
	16384	
16K ROM		

FAMILIARISEZ-VOUS AVEC LE SINCLAIR BASIC

Ce chapitre décrit entièrement la BASIC Sinclair. Vous y trouverez un résumé de la façon dont chaque mot-clé est utilisé, et d'autres détails sur le fonctionnement du BASIC Sinclair. L'information donnée s'étend de la plus simple à la plus avancée des programmations en BASIC. Ceci n'est pas un chapitre que vous devez lire depuis le début jusqu'à la fin. Mais c'est le dictionnaire du programmeur qui vous permettra de faire le meilleur usage des capacités du Spectrum.



GUIDE DES MOTS-CLES BASIC SINCLAIR POUR LE PROGRAMMEUR

Classes de mots-clés

Les mots-clés entrent dans une ou plusieurs classes et il y a quatre classes en tout.

Commande

Un mot-clé produisant une intervention et pouvant être utilisé comme commande directe. Elle est exécutée dès son entrée. Exemples RUN, LOAD.

Instruction

Un mot-clé produisant une intervention et pouvant être utilisé dans une ligne de programme. Elle n'est exécutée que lorsque le programme est passé. Exemples - DRAW, INPUT.

Fonction

Un mot-clé produisant une valeur quelconque. Il fait partie d'une commande ou d'une instruction. Exemples - RND, INT.

Opérateur logique

Un mot-clé utilisé pour exprimer de la logique dans une instruction ou une commande. Il peut déterminer ou altérer la véracité de certaines conditions. Le Spectrum a trois opérateurs logiques - AND, OR et NOT.

De guide contient une description des tous les mots-clés BASIC disponibles sur le ZX Spectrum +. Chaque rubrique comprend:

- L'emplacement sur le clavier
- La classe du mot-clé
- La raison d'être du mot-clé
- L'utilisation du mot-clé
- La structure de programmation

Les détails relatifs à l'emplacement, la raison d'être, l'utilisation et le code du jeu de caractères se passent d'explications. La classe et la structure sont des aspects plus complexes et si vous voulez utiliser efficacement votre guide, vous devriez commencer par lire attentivement les informations se trouvant sur cette page.

Nombres et variables

Nombres

Stockés jusqu'à une valeur de 9 ou 10 digits. Valeurs extrêmes du domaine de définition de 10^{38} à environ 4×10^{39} .

Variables acceptées

Numériques: tout longueur, commençant par une lettre. Les espaces ne sont pas pris en compte et toutes les lettres sont transformées en lettres minuscules. Les lettres minuscules et les lettres majuscules ne sont pas distinguées.

Chaîne: toute simple lettre suivie par \$. Les lettres majuscules et les lettres minuscules ne sont pas distinguées.

Tableaux: pour les tableaux de variables et d'indices voir entrée de DIM.

Structure des mots-clés

La structure d'un mot-clé exprime sa syntaxe - c'est-à-dire la combinaison correcte du mot-clé et d'autres éléments comme les valeurs ou des

variables. Les abréviations ci-dessous sont utilisées pour donner les structures.

Abréviation	Explication	Exemple
const-num	Une constante numérique (un nombre)	24.5
var-num	Une variable numérique (une variable pouvant contenir un nombre)	somme
expr-num	Une expression numérique (n'importe quelle combinaison de constantes et de variables numériques et de mots-clés donnant un nombre)	somme*24.5 RND*7
int-num-const int-num-var int-num-exp	Une constante variable ou expression numérique dont la valeur tourne autour du nombre entier le plus proche.	
const-ch	Une constante de chaîne ou une chaîne (n'importe quelle combinaison de caractères <i>entre guillemets</i>)	"ZX Spectrum +"
var-ch	Une variable de chaîne (une variable pouvant contenir une chaîne).	
expr-ch	Une expression de chaîne (n'importe quelle combinaison valide de constantes et de variables de chaîne et de mots-clés donnant une chaîne). Peut être constituée uniquement d'une constante ou d'une variable de chaîne.	a\$ + "ZX Spectrum +" a\$(GT0B)
lettre	N'importe quelle majuscule ou minuscule	Y x
lettre\$	N'importe quelle majuscule ou minuscule suivie de \$	B\$ a\$
cond	Une condition ou une sous-condition de condition	x = 10 AND t < 10
Instruction	N'importe quelle instruction BASIC qui est valide lorsqu'elle est utilisée avec une autre instruction	IF T > 10 THEN STOP PRINT INK 2,x
[]	Un élément en option pouvant être répété	

Les signes du BASIC Sinclair

Signe	Émplacement	Action/Use
\$	4	Variable de chaîne
'	7	Début d'une nouvelle ligne
(8	Ouvre une parenthèse
)	9	Ferme une parenthèse
<=	Q	Est inférieur ou égal à
<>	W	N'est pas égal à
>=	E	Est supérieur ou égal à
<	R	Est inférieur à
>	T	Est supérieur à
↑	H	Elève à une puissance
-	J	Soustraction, nombre négatif
+	K	Addition, nombre positif
		Enchaînement de chaîne

=	L	Est égal à
:	Z	Sépare les instructions dans une ligne de programme
/	V	Division
*	B	Multiplication
.	Propre touche	Décimale
:	Propre touche	Produit à la colonne suivante des instructions séparées d'une instruction de programme
'	Propre touche	Ouvre et ferme une chaîne
:	Propre touche	Produit à la colonne 0 ou 16 les valeurs séparées qui suivent les mots-clés

Jeu de caractères du ZX Spectrum +

	0	1	2	3	4	5	6	7	8	9
0							PRINT virgule	EDIT	Curseur gauche	Curseur droit
10	Curseur abaissé	Curseur levé	DELETE	ENTER	Nombre		INK commande	PAPER commande	FLASH commande	BRIGHT commande
20	Commande INVERSE	Commande OVER	AT contrôle	TAB contrôle						
30		Espace	!	"	#	\$	%	&	'	
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	INKEYS	PI	FN	POINT
170	SCREENS	ATTR	AT	TAB	VALS	CODE	VAL	LEN	SIN	COS
180	TAN	ASN	ACS	ATN	LN	EXP	INT	SQR	SGN	ABS
190	PEEK	IN	USR	STRS	CHRS	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	LLIST	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 valeur ABSolue *CF*

Emplacement sur le clavier
MODE ETENDU

G

Fonction

ABS donne la grandeur absolue d'une valeur numérique, c'est-à-dire la valeur sans signe positif ou négatif.

Comment utiliser ABS

ABS est suivie d'une valeur numérique. Une expression doit être *entre parenthèses*, par exemple

50 LET x=ABS(y-z)

ABS renvoie la valeur absolue de la valeur numérique

Exemple

La commande

PRINT ABS-34.2

produit sur l'écran 34.2

Structure

ABS constante-numérique

ABS variable-numérique

ABS (expression-numérique)

ACS Arc CoSinus *B4*

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT W

Fonction

ACS calcule la valeur d'un angle à partir de son cosinus.

Comment utiliser ACS

ACS est suivie d'une valeur numérique. Une expression doit être *entre parenthèses*, par exemple

60 LET x=ACS(y.z)

La valeur qui suit ACS ($y.z$ ci-dessus) est le cosinus de l'angle qui peut varier entre -1 et 1 . ACS convertit alors la valeur de l'angle en radians. Pour convertir les radians en degrés, multipliez la valeur convertie par ACS par 180/PI.

Exemple

La commande

PRINT 180/PI * ACS 0.5

affiche 60, l'angle en degrés ayant un cosinus de 0.5

Structure

ACS constant-numérique

ACS variable-numérique

ACS (expression-numérique)

AND *B8*

Emplacement sur le clavier
SYMBOL SHIFT Y

Opérateur logique/fonction

AND sert d'opérateur logique utilisé pour vérifier la véracité d'une combinaison de conditions. La combinaison globale n'est vraie qu'à condition que toutes les conditions soient vraies. AND sert également de fonction exécutant des opérations binaires sur deux valeurs numériques ou deux valeurs d'une chaîne.

Comment utiliser AND

En tant qu'opérateur logique, AND unit deux conditions dans une instruction dont la véracité doit être vérifiée, par exemple

90 IF x=y+z AND temps < 10 THEN PRINT "Correct"

L'ordinateur n'affiche Correct que si les deux conditions ($x=y+z$ et temps < 10) sont vraies. Si l'une des conditions ou si les deux conditions sont fausses, la combinaison toute entière est fautive et dans notre exemple, le programme passe à la ligne suivante.

AND en tant que fonction

En tant que fonction, AND peut traiter deux valeurs numériques, par exemple

50 LET x=y AND z

AND renvoie la première valeur (y) si la seconde (z) n'est pas égale à 0 et renvoie 0 si la seconde valeur (z) est 0.

AND peut également traiter une valeur de chaîne à condition qu'elle précède AND. Une valeur numérique doit toujours suivre AND, par exemple

50 LET a\$=b\$ AND z

AND renvoie la première valeur (b\$) si la seconde (z) n'est pas égale à zéro et une chaîne vide (" ") si la seconde valeur (z) est zéro.

Notez que le ZX Spectrum + affecte la valeur 1 à une condition vraie et la valeur 0 à une condition fautive; il identifie n'importe quelle valeur autre que zéro comme vraie et zéro comme fautive. Il n'évalue pas les combinaisons de valeurs numériques conformément aux tables de vérité ordinaires.

Structure

cond AND cond

expression-numérique AND

expression-numérique

chaîne-expression AND

expression-numérique

ASN Arc SiNus *B3*

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT Q

Fonction

ASN calcule la valeur d'un angle à partir de son sinus.

Comment utiliser ASN

ASN est suivie d'une valeur numérique. Une expression doit être *entre parenthèses*, par exemple

60 LET x=ASN(y.z)

La valeur qui suit ASN ($y.z$ ci-dessus) est le sinus de l'angle qui peut varier entre -1 et 1 . ACS convertit alors la valeur de l'angle en radians. Pour reconvertir les radians en degrés, multipliez la valeur convertie par ASN par 180/PI.

Exemple

La commande

PRINT 180/PI * ASN 0.5

affiche 30, l'angle en degrés ayant sinus de 0.5

Structure

ASN constant-numérique

ASN variable-numérique

ASN (expression-numérique)

AT *B10*

Emplacement sur le clavier
SYMBOL SHIFT J

Voir INPUT; LPRINT; PRINT

ATN Arc TaNgente *B5*

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT E

Fonction

ATN calcule la valeur d'un angle à partir de sa tangente.

Comment utiliser ATN

ATN est suivie d'une valeur numérique. Une expression doit être *entre parenthèses*, par exemple

60 LET x=ATN(y.z)

La valeur qui suit ATN ($y.z$ ci-dessus) est la tangente de l'angle. ATN convertit la valeur de l'angle en radians. Pour reconvertir les radians en degrés, multipliez la valeur convertie par ATN par 180/PI.

Exemple

La commande

PRINT 180/PI * ATN 1

affiche 45, l'angle en degrés ayant une tangente de 1.

Structure

ATN constante-numérique

ATN variable-numérique

ATN (expression-numérique)

ATTR ATTRibuts *B11*

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT L

Fonction

ATTR donne les attributs de la position d'un caractère spécifié sur l'écran. Ces attributs sont la couleur de l'encre et du papier, la brillance et le scintillement du caractère à la position où il se trouve.

Comment utiliser ATTR

ATTR est suivie de deux valeurs numériques, séparées par une virgule et entourées de parenthèses, par exemple

150 IF ATTR (v,h)=115 THEN GOSUB 2000

La première valeur après ATTR (v ci-dessus) peut varier entre 0 et 23 et c'est le numéro de ligne d'une position sur l'écran. La seconde valeur (h) peut varier entre 0 et 31 et c'est le numéro de colonne de la position. ATTR renvoie alors un nombre entre 0 et 255. Ce nombre est la somme des attributs à la position spécifiée et il est constitué des éléments suivants.

Couleur de l'encre	Code couleur (0 à 7)
Couleur du papier	8 fois code couleur
Brillance	64
Scintillement	128

Exemple
Si un caractère se trouvant à la position 11, 16 est affiché dans la couleur d'encre 3 (magenta), la couleur de papier 6 (jaune) et est brillant, mais ne scintille pas, la commande

PRINT ATTR (11,16)
affiche 115(3+8x6+64+0)

ATTR sous forme binaire
ATTR convertit l'octet dans lequel le bit 7 (le plus significatif) est 1 pour le scintillement et 0 pour normal, le bit 6 est 1 pour brillant et 0 pour normal, les bits 5 à 3 sont la couleur du papier (en binaire) et les bits 2 à 0 sont la couleur de l'encre.

Structure
ATTR (expression-numérique, expression-numérique)

BEEP D3

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT Z

Instruction/Commande

Après BEEP, le haut-parleur produit une note d'une certaine hauteur et d'une certaine durée.

Comment utiliser BEEP
BEEP peut être utilisée pour former une instruction de programme ou une commande directe. Elle est suivie de deux valeurs numériques, séparées par une virgule, par exemple

80 BEEP x,y

La première valeur (x) peut varier entre 0 et 10 et elle définit la durée de la note en secondes. La seconde valeur (y) peut varier entre -60 et 69 et elle définit la hauteur de la note en demi tons en-dessous de l'ut moyen si elle est négative et au-dessus de l'ut si elle est positive.

Exemple
La commande

BEEP 0 5,1

fait jouer le dièse au-dessus de l'ut moyen pendant un demi-seconde.

Structure
BEEP expression-numérique, expression-numérique

BIN Nombre BINaire D7

Emplacement sur le clavier
MODE ETENDU
B

BIN convertit un nombre binaire en un nombre décimal.

Comment utiliser BIN
BIN est suivie d'un nombre binaire composé de 1 et de 0, 16 au

maximum, par exemple
50 POKE USR "a", BIN 10101010

BIN reconvertit le nombre décimal en un nombre binaire. Elle est couramment utilisée en conjonction avec POKE et USR comme ci-dessus pour produire des graphiques utilisateur, 1 signifiant alors un pixel de la couleur de l'encre et 0 un pixel de la couleur du papier.

Exemple
La commande
PRINT BIN 1111110

affiche 254, la valeur décimale du nombre binaire.

Structure
BIN [1][0]

BORDER D7

Emplacement sur le clavier
B

Instruction/Commande
BORDER définit la couleur de la bordure de l'écran.

Comment utiliser BORDER
BORDER peut être utilisée comme une commande directe ou comme une instruction de programme. Elle est suivie d'une valeur numérique, par exemple
30 BORDER RND,7

La valeur qui suit BORDER est arrondie au nombre entier le plus proche et définit la couleur de la bordure comme suit:

- 0 noir
- 1 bleu
- 2 rouge
- 3 magenta
- 4 vert
- 5 cyan
- 6 jaune
- 7 blanc

Notez que BORDER définit également la couleur du papier dans la partie inférieure de l'écran. A la différence de INK et de PAPER, une instruction BORDER ne peut pas être intercalée (insérée) dans une instruction PRINT.

Exemple
La commande

BORDER 2

produit une bordure rouge autour de l'écran.

Structure
BORDER nombre entier-nombre-expression

BRIGHT D7

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT B

Instruction/Commande

Avec BRIGHT, les caractères apparaissent sur l'écran dans des couleurs plus brillantes que les couleurs normales.

Comment utiliser BRIGHT
BRIGHT peut être utilisée comme une commande directe, mais est habituellement utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple
80 BRIGHT 1

La valeur qui suit BRIGHT est arrondie au nombre entier le plus proche, si nécessaire et peut alors être 0, 1 ou 8. Si la valeur est 1, tous les caractères qui seront par la suite affichés par les instructions PRINT ou INPUT apparaîtront dans une couleur d'encre et de papier plus brillante et si la valeur est 8, les positions de caractère brillantes restent brillantes et les positions de caractère normales restent normales lorsque de nouveaux caractères sont imprimés. BRIGHT suivie de 0 annule à la fois BRIGHT 1 et BRIGHT 8 et tous les caractères qui sont affichés par la suite sont normaux.

BRIGHT peut également être intercalée (insérée) dans des instructions de l'écran formées par PRINT, INPUT, PLOT, DRAW ou CIRCLE. BRIGHT suit le mot-clé, mais précède les données ou les paramètres affichés; elle est suivie des mêmes valeurs et d'un point-virgule, par exemple

50 PRINT BRIGHT 1; "WARNING"

BRIGHT n'a qu'un effet local et ne concerne que les caractères affichés, le point tracé ou la ligne tirée par l'instruction d'affichage. Notez que BRIGHT 1 rend plus brillante la couleur du papier de la position de 8x8 pixels toute entière si un des pixels de la position est tracé dans l'une des couleurs de l'encre.

Structure
BRIGHT nombre entier-nombre-expression [:]

CAT CATalogue

Commande de prise en charge de fichier sur micro-support. Voir manuel du micro-support et de l'interface 1.

CHR\$ Chaîne de caractères B9

Emplacement sur le clavier
MODE ETENDU
U

Fonction

Les caractères et les mots-clés du clavier, plus n'importe quel caractère graphique utilisateur constituent le jeu de caractères de Spectrum. En utilisant CHR\$ et un numéro de code, chaque jeu de caractère peut être obtenu sous forme de chaîne. Le jeu de caractères contient également plusieurs codes de commande qui affectent l'affichage des caractères sur l'écran. L'on peut faire intervenir

ces codes et afficher des caractères en utilisant PRINT avec CHR\$. Le jeu de caractères complet et les numéros de code se trouvent à la page 51.

Comment utiliser CHR\$

CHR\$ est suivi d'une valeur numérique, par exemple

80 PRINT CHR\$ x

Une expression doit être entre parenthèses. La valeur qui suit CHR\$ (x) doit être arrondie au nombre entier le plus proche. Si elle se situe entre 32 et 255, CHR\$ reconstruit un caractère de clavier, un caractère graphique utilisateur ou un mot-clé en une chaîne. Le Spectrum utilise ASCII pour les valeurs entre 32 et 95 et entre 97 et 126. Si la valeur 65 est affectée à x, l'instruction ci-dessus affiche A, par exemple

CHR\$ codes de commande

Les valeurs entre 1 et 31 reconstruisent les codes de commande ou ne sont pas utilisées. CHR\$6 (PRINT virgule), 8 (espacement arrière) et 13 (nouvelle ligne ou ENTER) affectent le contenu de l'écran s'ils font partie d'une instruction PRINT. CHR\$ peut être suivie de la valeur du code et d'un point-virgule, par exemple

60 PRINT "A"; CHR\$6; "B"

Cette instruction affiche

A

B

CHR\$ codes de commande peuvent également être utilisés pour former une chaîne composée qui les contiendra. L'instruction

60 PRINT "A" + CHR\$6 + "B"

a exactement le même effet que l'exemple précédent.

Les codes 16 à 23 affectent la couleur et la position et chacun d'entre eux peut être utilisé dans une chaîne composée en même temps que CHR\$ suivi d'un code de couleur dont la valeur varie entre 0 et 7 pour CHR\$16 (commande INK) et pour CHR\$17 (commande PAPER) ou bien est 0 ou 1 pour CHR\$18 à CHR\$21 (commandes FLASH, BRIGHT, INVERSE et OVER). La commande

PRINT CHR\$16 + CHR\$3 + CHR\$17 + CHR\$18 + CHR\$1 + ZX Spectrum +

affiche ZX Spectrum + clignotant en rouge et jaune. Ou bien, comme précédemment chaque signe plus (+) peut être remplacé par un point-virgule.

CHR\$22 (commande AT) est suivie de deux valeurs CHR\$ indiquant le numéro de la ligne et de la colonne. La commande

PRINT CHR\$22 + CHR\$11 + CHR\$16 + CHR\$42

affiche une étoile au milieu de l'écran.

CHR\$23 (commande TAB) est elle aussi suivie de deux valeurs. La seconde valeur est habituellement 0

et la première valeur donne la position TAB. La commande

PRINT CHR\$23 + CHR\$16 + CHR\$0 + CHR\$42

affiche une étoile au milieu de l'écran.

Notez que seules ces commandes sont disponibles. Si PRINT CHR\$ est utilisée avec un mot-clé d'une valeur supérieure à 164, seul le mot-clé est affiché sur l'écran et il ne devient pas opérationnel.

Structure

CHR\$ nombre entier-constante-numérique [:] [+]

CHR\$ nombre entier-variable-numérique [:] [+]

CHR\$ (nombre entier-expression-numérique) [:] [+]

CHR\$ (nombre entier-expression-numérique) [:] [+]

CIRCLE

e8

Emplacement sur le clavier

MODE ETENDU
SYMBOL SHIFT H

Instruction/Commande

CIRCLE trace un cercle sur l'écran.

Comment utiliser CIRCLE

CIRCLE est suivie de trois valeurs numériques, séparées par une virgule, par exemple

80 CIRCLE x,y,z

Si nécessaire, les trois valeurs sont arrondies au nombre entier le plus proche. CIRCLE trace alors un cercle sur la grille du graphique à haute résolution dans la couleur d'encre alors utilisée. Les deux premières valeurs (x,y) définissent les coordonnées verticale et horizontale du centre et la troisième valeur (z) définit la longueur du rayon. Les dimensions du cercle ne doivent pas dépasser la zone d'affichage.

CIRCLE est affectée par les instructions ou commandes de couleur et peut englober des instructions de couleur intercalées ayant les mêmes effets que PLOT et DRAW.

Exemple

La commande

CIRCLE 128,88,87

trace un cercle couvrant la plus grande partie de la zone d'affichage.

Structure

Instruction CIRCLE, nombre entier-expression-numérique,

nombre

entier-expression-numérique,

nombre

entier-expression-numérique.

CLEAR

D4

Emplacement sur le clavier

X

Instruction/Commande

CLEAR annule les valeurs courantes de toutes les variables, libérant l'espace mémoire que les valeurs

occupaient et l'espace jusqu'à RAMTOP, l'adresse supérieure du système BASIC. CLEAR peut aussi être utilisée pour remettre RAMTOP à l'état initial.

Comment utiliser CLEAR

CLEAR peut être utilisée comme une commande directe ou bien constituer une instruction de programme. Elle n'exige pas de paramètres, par exemple

50 CLEAR

CLEAR annule alors les valeurs dont sont alors affectées toutes les variables, y compris les tableaux. Elle exécute également CLS et RESTORE pour effacer le contenu de l'écran et remet l'indicateur de données sur le premier élément des données. En outre, la position PLOT est remise dans le coin inférieur gauche de l'écran et la pile GOSUB est effacée.

Notez que CLEAR n'est pas nécessaire avant que les tableaux ne soient à nouveau déterminés dans la mesure où DIM supprime un tableau existant du même nom. Notez aussi que RUN exécute CLEAR.

CLEAR et RAMTOP

CLEAR peut également être suivie d'une valeur numérique, par exemple

CLEAR 65267

CLEAR exécute alors CLEAR comme ci-dessus et positionne également RAMTOP, l'adresse la plus haute du système BASIC, à la valeur donnée. RAMTOP est positionnée à 65367 dans le ZX Spectrum + et se trouve en-dessous de la zone réservée aux graphiques utilisateur. NEW remet la mémoire à zéro jusqu'à RAMTOP, si bien qu'il utilise CLEAR pour abaisser RAMTOP (de 100 octets dans l'exemple ci-dessus) donne davantage de mémoire à l'abri de NEW. Relever RAMTOP donne plus d'espace pour BASIC aux dépens des graphiques utilisateur. Notez que la pile GOSUB est alors placée à RAMTOP, à condition que RAMTOP demeure au-dessus de la pile de la calculatrice.

Pour tracer RAMTOP, entrez
PRINT PEEK 23730 + 256; PEEK 23731

Structure

CLEAR [expression-numérique]

CLOSE

A7

Commande de prise en charge de fichier sur micro-support. Voir manuel du micro-support et de l'interface 1.

CLS Effacer l'écran

D6

Emplacement sur le clavier

V

Instruction/Commande

CLS efface tout le texte et les graphiques de l'écran, le laissant vide dans la couleur du papier (arrière-plan alors utilisée).

Comment utiliser CLS

CLS peut être utilisée comme une commande directe ou peut constituer une instruction de programme. Elle n'exige pas de paramètres, par exemple

```
250 IF a$ = "NON" THEN CLS
```

La zone d'affichage (mais pas la bordure) prend alors la couleur sélectionnée par l'instruction ou la commande PAPER précédente ou bien devient blanche, la couleur implicite du papier.

Notez que CLS doit être utilisée après PAPER et avant PRINT ou n'importe quelle autre instruction d'affichage pour produire un arrière-plan coloré sur toute la zone d'affichage.

Structure

CLS

CODE

810

Emplacement sur le clavier

MODE ETENDU

I

Fonction

CODE donne le numéro de code d'un caractère du jeu de caractères du Spectrum (voir page 51).

Comment utiliser CODE

CODE est suivie d'une valeur de chaîne, par exemple

```
90 IF CODE a$ < 65 OR CODE a$ > 90 THEN GOTO 80
```

Une expression de chaîne doit être entre parenthèses. CODE renvoie le numéro de code du premier caractère de la chaîne. S'il s'agit d'une chaîne vide (""), CODE renvoie 0.

SAVE CODE et LOAD CODE

Avec SAVE et LOAD, CODE est utilisée d'une manière différente. Voir SAVE CODE et LOAD CODE.

Structure

CODE chaîne-constante

CODE chaîne-variable

CODE (chaîne-expression)

CONTINUE

05

Emplacement sur le clavier

C

Commande

Si un programme s'arrête, CONTINUE peut être utilisée pour le relancer au point où il s'est arrêté. Si le programme s'est arrêté à la suite d'une erreur, celle-ci doit être corrigée avant que CONTINUE ne fasse redémarrer le programme.

Comment utiliser CONTINUE

CONTINUE est utilisée comme une commande directe lorsqu'un programme s'est arrêté. Elle n'exige pas de paramètres. Après CONTINUE, un programme est

normalement relancé à l'instruction à laquelle il s'est arrêté. S'il y avait une erreur, il faut entrer une commande qui corrigera cette erreur et CONTINUE fera redémarrer le programme à partir de l'instruction où il est arrêté. Si le programme s'est arrêté à une instruction STOP, produisant le message 9 ou s'il s'est arrêté parce que l'on a appuyé sur la touche BREAK, produisant le message L, CONTINUE relance le programme à l'instruction suivante. Si nécessaire, vous pouvez d'abord entrer une commande rectificatrice.

Si CONTINUE est utilisée pour relancer une commande directe, elle entre dans une boucle si la commande s'est arrêtée à sa première instruction. Le contenu de l'écran disparaît, mais tout rentre dans l'ordre si l'on appuie sur BREAK. CONTINUE donne le message 0 si la commande s'est arrêtée à la seconde instruction et signale N à la troisième instruction et aux suivantes.

Structure

CONTINUE

COPY

D3

Emplacement sur le clavier

Z

Commande

Avec COPY, les imprimantes Sinclair produisent une copie du contenu de l'écran.

Comment utiliser COPY

COPY est utilisée comme une commande directe lorsqu'un programme est terminé ou s'est arrêté. Elle n'exige pas de paramètres. Après COPY et à condition qu'une imprimante soit connectée, une copie des 22 premières lignes de l'écran est imprimée. Notez que toutes les couleurs de l'encre (avant-plan) sont imprimées en noir; les couleurs du papier (arrière-plan) ne sont pas imprimées. Vous pouvez arrêter l'imprimante en appuyant sur BREAK.

Si un listing de programme apparaît sur l'écran, il peut être imprimé à l'aide de COPY à condition qu'il ait été produit par une commande ou une instruction LIST. Notez qu'un listing apparaîtra sur l'écran si vous appuyez sur ENTER après qu'un programme est terminé ou s'est arrêté, mais ce listing "automatique" ne peut pas être imprimé par l'intermédiaire de COPY.

Structure

COPY

COS

COSinus B4

Emplacement sur le clavier

MODE ETENDU

W

Fonction

COS donne le cosinus d'un angle.

Comment utiliser COS

COS est suivi d'une valeur numérique, par exemple

```
140 LET X=COSy
```

Une expression doit être entre parenthèses. La valeur qui suit COS est l'angle en radians. COS reconvertit alors le cosinus de l'angle. Les degrés peuvent être convertis en radians en multipliant par $\pi/180$.

Notez que COS reconvertit une valeur négative pour des angles de 90 à 270 degrés et une valeur positive pour des angles de 0 à 90 et de 270 à 360 degrés.

Exemple

La commande

```
PRINT COS(60*PI/180)
```

affiche 0.5, le cosinus de 60 degrés.

Structure

COS constante-numérique

COS variable-numérique

COS (expression-numérique)

DATA

C5

Emplacement sur le clavier

MODE ETENDU

D

Instruction

DATA fournit une liste d'éléments de données d'un programme. Ces éléments peuvent être des valeurs de variables ou des chaînes à afficher par exemple. Chaque élément est affecté à une variable par une instruction READ.

L'affectation est effectuée dans l'ordre dans lequel les éléments de données apparaissent dans le programme, mais RESTORE peut être utilisée pour faire commencer une affectation au premier élément d'une instruction DATA donnée.

Comment utiliser DATA

DATA ne peut être utilisée que pour constituer une instruction de programme. Elle est normalement suivie d'une liste de constantes numériques ou en chaîne, séparées par une virgule, par exemple

```
50 DATA 31, "JAN", 28, "FEV"
```

Chaque constante est alors affectée d'une variable par d'une instruction READ qui lit les DATA. L'instruction DATA peut-être positionnée n'importe où dans le programme. Le nombre, le genre (numérique ou en chaîne) et l'ordre des constantes doit correspondre au nombre de fois que l'instruction READ est exécutée et au genre ainsi qu'à l'ordre des variables de l'instruction READ. La liste des données peut être divisée en plusieurs instructions DATA successives si une instruction contient un trop grand nombre d'éléments.

Exemple

Le programme ci-dessous

```
10 FOR n=1 TO 2
20 READ x,a$
30 PRINT a$,x; " jours"
40 NEXT n
50 DATA 31, "JAN", 28, "FEV"
```

affiche

```
JAN          31 jours
FEV          28 jours
```

Utilisation de DATA avec des variables

Les éléments de données dans une instruction DATA peuvent se composer de variables numériques ou de variables de chaîne ou encore d'expressions, à condition que des valeurs aient auparavant été affectées aux variables. Dans l'exemple ci-dessus, l'instruction DATA peut devenir

```
50 DATA d,m$,d-3 "FEV"
```

Si d a été auparavant affecté de la valeur 31 et m\$ de la valeur "JAN", le contenu de l'écran reste le même.

Structure

DATA expression-numérique

[expression-numérique]

[chaîne-expression]

DATA chaîne-expression

[expression-numérique]

[chaîne-expression]

DEF FN

A3

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT I

Instruction

DEF FN permet à l'utilisateur de définir une fonction n'existant pas en tant que mot-clé. De nombreux paramètres peuvent être attribués à la fonction dans une instruction FN qui appelle la fonction et peut produire comme résultat une valeur numérique ou une valeur de chaîne.

Comment utiliser DEF FN

DEF FN ne peut être utilisée que comme instruction de programme. Si une fonction numérique doit être définie, DEF FN est suivie d'une lettre quelconque, puis d'une ou plusieurs variables numériques, séparées par une virgule et entourées de parenthèses, par exemple DEF FN r(x,y). Ceci est suivi d'un signe égale et ensuite, d'une expression numérique contenant les variables, par exemple

```
000 DEF FN r(x,y)=
  SQR(x↑2+y↑2)
```

La lettre qui suit DEF FN (r) est un nom identifiant la fonction. Les variables peuvent elles aussi n'être qu'une lettre. Notez que dans les deux cas, le Spectrum ne fait pas de différence entre les minuscules et les majuscules.

L'expression qui suit le signe égale utilise les variables (x et y ci-dessus) pour définir la fonction.

Une instruction DEF FN peut être placée n'importe où dans le

programme. Pour appeler la fonction qu'elle définit, une instruction FN est utilisée. Ceci est suivi de l'initiale du nom de la fonction et d'une liste de valeurs numériques, séparées par une virgule et entourées de parenthèses, par exemple

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

Les valeurs entre parenthèses sont attribuées à la fonction dans le même ordre que les variables de l'instruction DEF FN. Donc, dans notre exemple, x est affecté de la valeur 3 et y de la valeur 4. FN évalue l'expression et renvoie la valeur.

DEF FN peut également être suivi d'une lettre ou d'uniquement deux parenthèses, par exemple

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

La valeur alors affectée à la variable (x ci-dessus) est attribuée à la fonction lorsque celle-ci est appelée par FN. Dans ce cas, FN r() renvoie la valeur alors attribuée à x, arrondie au nombre entier supérieure.

DEF FN et les chaînes

DEF FN et FN peuvent être utilisées de la même manière pour définir et appeler une fonction de chaîne.

Dans ce cas, le nom de la fonction est une lettre suivie de \$ et l'une ou plusieurs des variables de

l'instruction est une lettre suivie de \$.

Une expression de chaîne correspondante constitue la définition, par exemple

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

Dans cet exemple, l'expression de chaîne suivant le signe égale découpe la chaîne et x et y sont les premier et dernier caractères d'une section de b\$. FN doit être suivi du nom de la fonction et, entre parenthèses, d'une valeur de chaîne et de tous les autres paramètres qui sont attribués à la fonction. Dans ce cas, la commande

```
PRINT FN AS
```

```
("FONDAMENTALE", 1,3)
```

affiche "FUN" et la commande

```
PRINT FN a$
```

```
("FONDAMENTALE", 5,8)
```

```
AFFICHE AMEN.
```

Structure

DEF FN lettre([lettre],[lettre]) =

expression-numérique

DEF FN lettre\$([lettre\$][lettre]

[lettre],[lettre\$]) =

chaîne-expression

FN lettre ([expression-numérique]

[expression-numérique])

FN lettre \$([chaîne-expression]

[expression-numérique]

[expression-numérique] [,

chaîne-expression])

DIM DIMension

15

Emplacement sur le clavier

D

Instruction

DIM est utilisée pour établir les dimensions (constituer) un tableau d'un nombre donné de variables numériques ou de variables de chaîne. Un tableau contient une série de variables du même nom différenciées par des indices (valeurs identifiant chacune des variables ou chacun des éléments d'un tableau).

Comment utiliser DIM sans tableaux numériques

DIM est utilisé pour former une instruction de programme. Elle est suivie d'une lettre donnant son nom au tableau et d'une ou plusieurs valeurs numériques, séparées par une virgule et entourées de parenthèses, par exemple

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

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

Dans le premier cas, un tableau numérique à une dimension est produit, contenant dix éléments munis d'indices de 1 à 10. Le tableau est appelé x et les variables à indice sont x(1) à (10)

inclusivement. Tout autre tableau existant du même nom est effacé et la valeur 0 est affectée à chacune des variables. Notez qu'en établissant les dimensions d'un tableau, le Spectrum ne différencie pas les majuscules et les minuscules - la variable x(2) est donc identique à la variable X(2). Toutefois, de simples variables numériques ayant la même lettre que le nom d'un tableau (x ou X) peuvent exister en même temps et peuvent être utilisées séparément si nécessaire.

Le nombre de valeurs entre parenthèses est égal au nombre de dimensions constituées dans un tableau numérique. Le second exemple établit un tableau à deux dimensions de 100 éléments, avec 20 éléments dans la première dimension et 5 dans la seconde. Ces éléments sont numérotés de w(1,1) à z(20,5).

Des tableaux de n'importe quel nombre de dimensions peuvent être établis.

Les éléments d'un tableau numérique peuvent par la suite être identifiés par le nom du tableau, suivi d'une valeur numérique entre parenthèses, par exemple

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

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

DIM et les tableaux de chaîne

DIM est utilisée comme avec les tableaux numériques, sauf qu'une lettre suivie de \$ est utilisée pour le nom du tableau.

En outre, une valeur supplémentaire doit être ajoutée à la valeur des dimensions entre parenthèses afin de définir la longueur de chaque chaîne, par exemple

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

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

La première instruction établit un tableau de 20 éléments, chacun contenant une chaîne de 5 caractères. Les variables à indice

sont appelées a\$(1) à a\$(20) inclusivement et une chaîne vide (" ") leur est initialement affectée. Tout tableau existant du même nom est effacé et, à la différence des tableaux numériques, une variable de chaîne simple du même nom ne peut pas exister en même temps.

Le second exemple établit un tableau de chaînes à deux dimensions de 100 éléments avec 20 éléments dans la première dimension et 5 dans la seconde. La longueur de tous les éléments est de 10 caractères.

Lorsque les tableaux de chaînes sont par la suite affectés de valeurs de chaînes, ils sont complétés par des espaces à l'extrémité de la chaîne ou si nécessaire, coupés à la longueur définie.

Les éléments d'un tableau de chaînes sont identifiés par le nom du tableau suivi, entre parenthèses, par une ou plusieurs valeurs numériques donnant le numéro ou les numéros de l'indice. Par exemple, l'élément a\$(2) peut être "SMITH" et l'élément b\$(12,4) "Derbyshire". Toutefois, une valeur supplémentaire peut être ajoutée pour définir un caractère particulier de la chaîne. Dans ces exemples, a\$(2,2) serait "M" (le second caractère de "SMITH") et b\$(12,4,5) serait "Y".

Tableaux de chaînes sans dimension Il est possible d'établir un tableau de chaînes sans dimension en n'utilisant qu'une valeur entre parenthèses, par exemple

10 DIM c\$(15)

Ce tableau ne compte qu'un élément, à savoir c\$ et sa longueur est celle de la valeur définie (15 caractères).

Structure

DIM lettre (expression-numérique [, expression-numérique])

DIM lettres\$ (expression-numérique [, expression-numérique])

DRAW

B4

Emplacement sur le clavier
W

Instruction/Commande

DRAW est utilisée pour tracer des droites et des courbes sur l'écran.

Comment utiliser DRAW

DRAW est normalement utilisée pour former une instruction de programme. Si l'on veut tracer une ligne droite, elle est suivie de deux valeurs numériques séparées par une virgule, par exemple

40 DRAW x,y

Une droite est alors tracée sur la grille des graphiques à haute résolution, à partir de la position définie par l'instruction PLOT précédente ou de la position précédente par l'instruction DRAW

précédente. Si nécessaire, les deux valeurs qui suivent DRAW sont arrondies au nombre entier le plus proche. La première valeur (x) définit la distance horizontale à partir de la position et la seconde valeur (y), la distance verticale. Ces valeurs sont négatives si la droite est orientée vers la gauche et vers le bas et sa position d'arrivée doit se trouver dans la zone d'affichage.

Si aucune instruction DRAW ou PLOT n'a été exécutée auparavant, le point de départ de DRAW est 0,0 (le coin inférieur gauche de l'écran).

DRAW est affectée par les instructions et les commandes de couleur et peut englober des instructions intercalées ayant le même effet que PLOT et CIRCLE.

Dessiner des courbes

Si DRAW est suivie d'une troisième valeur, vous pouvez l'utiliser pour dessiner une partie de cercle, par exemple

40 DRAW x,y,z

La troisième valeur (z) définit l'angle de rotation (en radians). Si cette valeur est positive, la rotation se fait à gauche, si elle est négative, la rotation se fait à droite. Les valeurs Pi et -Pi produisent un cercle.

Exemple

Le programme ci-dessous dessine un triangle

10 PLOT 127, 150

20 DRAW 70, -100

30 DRAW -140, 0

40 DRAW 70, 100

Si vous ajoutez 1 ou -1 à l'instruction DRAW, les côtés s'arrondissent respectivement vers la droite et la gauche.

Structure

DRAW [instruction:] nombre entier-expression-numérique, nombre entier-expression-numérique [expression-numérique]

ERASE

A9

Commande de prise en charge de fichier sur micro-support. Voir manuel du micro-support et de l'interface 1.

EXP

D4

Emplacement sur le clavier
X

Fonction

EXP est une fonction mathématique qui élève l'exposant e à une puissance donnée

Comment utiliser EXP

EXP est suivi d'une valeur numérique, par exemple

60 LET y=EXPx

Une expression doit être entre parenthèses. EXP produit l'exposant 3 élevé à la puissance de l'argument (x).

Exemple

La commande

PRINT EXP 1

affiche 2.7182818, la valeur de e.

Structure

EXP constante-numérique

EXP variable-numérique

EXP (expression-numérique.)

FLASH

D6

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT V

Instruction/Commande

FLASH fait scintiller la position des caractères, c'est-à-dire que les couleurs de l'encre et du papier alternent à une vitesse constante.

Comment utiliser FLASH

FLASH peut être utilisée comme une commande directe, mais normalement, elle est utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple

50 FLASH 1

La valeur qui suit FLASH est arrondie au nombre entier le plus proche, si nécessaire et peut alors être 0, 1 ou 8. Avec 1, tous les caractères affichés par la suite par FLASH ou INPUT scintillent. Avec 8, les positions de caractères qui scintillent restent scintillantes et les positions de caractères normales restent normales lorsque de nouveaux caractères sont imprimés. Suivie de 0, FLASH annule FLASH 1 et FLASH 8, de telle sorte que tous les caractères affichés par la suite sont normaux.

FLASH peut également être intercalée (insérée) dans des instructions d'affichage formées par PRINT, INPUT, PLOT, DRAW, et CIRCLE. FLASH suit le mot-clé, mais précède les données et les paramètres d'affichage; elle est suivie des mêmes valeurs et d'un point-virgule, par exemple

120 PRINT FLASH 1; INK 2; PAPER 6; "ATTENTION"

FLASH a alors un effet local et ne concerne que les caractères affichés, le point tracé ou la ligne dessinée par l'instruction d'affichage. Notez que FLASH fait scintiller le carré 8x8 tout entier si un des pixels est tracé dans une des couleurs de l'encre.

Structure

FLASH nombre entier-expression-numérique[.]

FN

A4

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT 2

Fonction

FN appelle une fonction utilisateur. Elle est toujours utilisée en

conjonction avec DEF FN qui définit la fonction qui sera appelée.

Comment utiliser FN

Si une fonction numérique va être appelée, FN est suivi d'une lettre et de deux parenthèses. Si des paramètres doivent être attribués à la fonction, ces paramètres sont séparés par un virgule et entourés de parenthèses, par exemple

170 LET x=FN r(3,4)

Les paramètres (3 et 4) sont alors attribués à la fonction r. FN produit alors le résultat. Les deux paramètres doivent être utilisées même si aucun paramètre n'est attribué à la fonction, par exemple

70 PRINT FN r()

Dans ce cas, la fonction utilise les valeurs affectées à ce moment-là à ses variables.

FN appelle une fonction de chaîne de la même façon, sauf qu'il faut ajouter \$ après la lettre qui donne son nom à la fonction. Pour plus de détails, voir DEF FN.

Structure

FN lettre ([expression-numérique])

[expression-numérique])

FN lettre\$ ([chaîne-expression])

[expression-numérique]

[expression-numérique]

[chaîne-expression]

FOR

Emplacement sur le clavier

Instruction/Commande

FOR est toujours utilisé avec les mots-clés TO et NEXT pour produire une boucle FOR NEXT. Cette structure permet de répéter une section de programme un nombre donné de fois.

Comment utiliser FOR

Accompagnée de TO, FOR constitue toujours une instruction. FOR est suivie d'une lettre, d'un signe égale et deux valeurs numériques séparées par TO, par exemple

60 FOR a=1 TO 9

La lettre (a) est une variable de contrôle. Les instructions qui seront répétées suivent et normalement, une ou plusieurs de ces instructions utilisent la variable de contrôle. La boucle se termine par une instruction NEXT, dans laquelle NEXT est suivie de la variable de contrôle, par exemple

90 NEXT a

Lors de l'exécution, FOR efface toutes les variables ayant le même nom que la variable de contrôle et affecte à cette dernière une valeur initiale égale à la valeur qui précède TO (1 ci-dessus). Les instructions sont alors exécutées avec la variable de contrôle affectée de cette valeur. Lorsque NEXT est exécutée, la

valeur de la variable de contrôle est augmentée de 1. Si cette valeur est inférieure ou égale à la valeur qui suit TO (la valeur limite 9 ci-dessus), le programme opère un retour à l'instruction FOR et la boucle FOR NEXT est répétée. Si la valeur de la variable de contrôle est supérieure à la valeur limite, la boucle se termine et le programme exécute l'instruction qui suit NEXT.

Dans l'exemple ci-dessus, la boucle est répétée neuf fois et la variable de contrôle a passé de 1 à 9. Lorsque a sort de la boucle, il a la valeur 10.

Notez qu'en ce qui concerne le nom de la variable de contrôle, le Spectrum ne fait pas de différence entre les majuscules et les minuscules.

Utilisation de STEP dans une boucle FOR NEXT

STEP est un mot-clé pouvant être incorporé dans une instruction FOR si la variable de contrôle doit être augmentée d'une valeur autre que 1 ou doit être diminuée. STEP suit la valeur limite et est suivi d'une valeur numérique, par exemple

60 FOR a=1 TO 9 STEP 2

La variable de contrôle est augmentée de la valeur du pas (2) jusqu'à ce qu'elle soit supérieure à la valeur limite. La variable de contrôle prend successivement les valeurs 1, 3, 5, 7 et 9 et sort de la boucle avec la valeur 11.

Si le pas a une valeur négative, la valeur de la variable de contrôle diminue. Dans ce cas, la valeur initiale doit être supérieure à la valeur limite et la boucle se termine lorsque la valeur de la variable de contrôle est inférieure à la valeur limite, par exemple

60 FOR a=9 TO 1 STEP -1

La valeur de a passe de 9 à 1, laissant la boucle affectée de la valeur 0.

Boucles emboîtées

Une ou plusieurs boucles FOR NEXT peuvent être placées l'une dans l'autre, procédure qui s'appelle "emboîter" des boucles. L'ordre des variables de contrôle des instructions NEXT doit être l'inverse de l'ordre des variables de contrôle dans les instructions FOR. Les boucles FOR NEXT peuvent être emboîtées à n'importe quelle profondeur, autrement dit autant de boucles qu'il est nécessaire peuvent être placées l'une dans l'autre.

Structure

FOR lettre = expression-numérique
TO expression-numérique [STEP
expression-numérique] NEXT lettre

FORMAT

Commande de prise en charge de fichier sur micro-support. Voir manuel du micro-support et de l'interface 1.

GOSUB

Emplacement sur le clavier

H

Instruction/Commande

GOSUB branche le programme sur un sous-programme constituant une section séparée du programme. Ceci est très utile si un programme doit faire appel plusieurs fois à un sous-programme.

Comment utiliser GOSUB

GOSUB peut être utilisée comme une instruction ou une commande directe, elle est suivie d'une valeur numérique, par exemple

GOSUB 1000

Lors de l'exécution, la valeur qui suit GOSUB (1000 ci-dessus) est arrondie au nombre entier supérieur et le programme se branche sur le numéro de ligne ayant cette valeur. Utiliser une variable ou une expression permet au programme de se brancher sur un sous-programme à un numéro de ligne calculé. Notez que si le numéro de la ligne n'existe pas, le programme se branche tout de même et continue avec la première instruction qu'il rencontre.

Un sous-programme se termine avec RETURN et le programme se rebranche alors sur l'instruction qui suit l'instruction GOSUB. Les sous-programmes peuvent être emboîtés de façon à ce que l'un d'entre eux puisse être atteint à partir d'un autre, auquel cas RETURN renvoie le programme à l'instruction qui suit la dernière instruction GOSUB exécutée.

La pile GOSUB

Chaque fois que GOSUB est exécutée, son numéro de ligne est placé sur la pile GOSUB de la mémoire. Si deux GOSUB ou plus sont exécutés avant RETURN, leurs numéros de ligne s'empilent et le dernier numéro se trouve donc au sommet de la pile. RETURN prend toujours le numéro de ligne qui est au sommet de la pile et va à cette ligne pour continuer le programme. Notez que erreur 4 peut se produire (hors-mémoire) s'il n'y a pas assez d'instructions RETURN.

Structure

GOSUB nombre
entier-expression-numérique

GOTO

Emplacement sur le clavier

G

Instruction/Commande

GOTO branche un programme sur une ligne particulière.

Comment utiliser GOTO

GOTO peut être utilisée comme une commande directe pour exécuter un programme à partir d'un numéro de

ligne donné, sans que le contenu de l'écran doive être effacé. Elle peut également être utilisée pour former une instruction de programme. GOTO est suivie d'une valeur numérique, par exemple

60 GOTO 350

Lors de l'exécution, la valeur qui suit GOTO est arrondie au nombre entier supérieur et le programme se branche sur le numéro de ligne ayant cette valeur. Utiliser une variable ou une expression permet de brancher le programme sur un numéro de ligne calculé. Notez que si la ligne n'existe pas, le programme se branche tout de même et continue avec la première instruction qu'il rencontre.

Structure

GOTO nombre entier-expression-numérique

IF

89

Emplacement sur le clavier

Fonction

IF est toujours utilisée avec le mot-clé THEN pour dicter une décision qui influence les événements ultérieurs. L'ordinateur vérifie si quelque chose est vrai ou non. Si ce quelque chose est vrai, une ligne de conduite est adoptée et si ce quelque chose est faux, une autre ligne de conduite est adoptée.

Comment utiliser IF et THEN

IF et THEN constituent normalement une instruction. IF est suivie d'une valeur numérique ou d'une condition, puis de THEN et d'une ou plusieurs instructions BASIC valides, par exemple

80 IF x GOTO 250

240 IF a\$ = "NON" THEN PRINT "FIN": STOP

Une constante, variable ou expression (x ci-dessus par exemple) est considérée comme vraie si elle a une valeur différente de zéro. Dans ce cas, l'instruction qui suit THEN et toute autre instruction de la même ligne sont exécutées. Le programme passe alors à la ligne suivante. Si une constante, variable ou expression a la valeur 0 elle est considérée comme fautive. Les instructions qui suivent ne sont donc pas exécutées et le programme saute à la ligne suivante. Dans l'exemple, le programme n'ira pas à la ligne 250 si x=0.

Si une condition (a\$ = "NON") qui suit IF est vraie, les instructions qui suivent THEN sont exécutées. Si la condition est fautive, le programme passe à la ligne suivante. Dans l'exemple, si a\$ a la valeur "NON", "FIN" apparaît sur l'écran et le programme s'arrête. Si a\$ est affecté d'une autre valeur, le programme continue à partir de la ligne suivante.

Le Spectrum donne la valeur 1 à une condition vraie et la valeur 0 à une condition fautive. Il reconnaît toute valeur différente de zéro comme vraie et 0 comme faux. La valeur d'une condition peut être affectée à une variable par une instruction comme

70 LET x=a\$="NON"

Notez qu'à la différence de certains autres BASIC, THEN ne peut pas être omis devant GOTO.

Structure

IF expression-numérique **THEN**

instruction [:instruction]

IF condition **THEN** instruction

[:instruction]

IN

B10

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT I

Fonction

IN vérifie l'état du clavier et des autres dispositifs d'entrée et de sortie. Elle lit un octet provenant d'une adresse d'entrée déterminée indiquant l'état du dispositif connecté à l'entrée.

Comment utiliser IN

IN est suivie d'une valeur numérique, par exemple

150 LET x=IN y

La valeur qui suit IN peut varier entre 0 et 65535 et elle spécifie l'adresse de l'entrée qui doit être lue. IN renvoie alors l'octet lu à cette entrée.

Adresses du clavier

Le clavier compte huit adresses qui peuvent chacune contenir un octet parmi cinq en fonction de la touche sur laquelle vous avez appuyé. Les adresses sont 65278, 65022, 64510, 63486, 61438, 57342, 49150 et 32766. Les valeurs des octets à ces adresses peuvent être 175, 183, 187, 189 ou 190.

Cette ligne attend que vous appuyiez sur ENTER, puis réagit très rapidement.

80 IF IN 49150 < > 190 THEN GOTO 80

N'importe quelle autre touche peut être détectée en remplaçant l'adresse de l'entrée et la valeur de l'octet par celles qui figurent dans la table.

Structure

IN constante-numérique

IN variable-numérique

IN expression-numérique

INK

D4

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT X

Instruction/Commande

INK spécifie la couleur d'avant-plan dans laquelle les caractères apparaissent sur l'écran, les points

sont tracés et les droites et les courbes dessinées.

Comment utiliser INK

INK peut être utilisée comme une commande directe, mais est normalement utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple

70 INK x

La valeur qui suit INK est arrondie au nombre entier supérieur et peut varier entre 0 et 99. Vous obtenez alors les couleurs d'avant-plan ci-dessous.

0 noir

1 bleu

2 rouge

3 magenta (pourpre)

4 vert

5 cyan (bleu de prusse)

6 jaune

7 blanc

8 transparent

9 blanc et noir en contraste

INK B spécifie que la couleur existante reste inchangée quelle que soit la position de l'écran à laquelle INK est utilisée. INK 9 donne à l'encre la couleur blanche ou noire de façon à ce qu'elle ressorte sur la couleur du papier (arrière-plan).

Couleurs d'encre locales et globales

Lorsque INK forme à elle seule une instruction, comme ci-dessus, la couleur est globale et tout ce qui sera par la suite affiché sur l'écran le sera dans cette couleur d'avant-plan. INK peut également être intercalée (insérée) dans des instructions d'affichage formées par PRINT, INPUT, PLOT, DRAW ou CIRCLE. INK suit le mot-clé, mais précède les données ou les paramètres d'affichage; elle est suivie des mêmes valeurs et d'un point-virgule, par exemple

60 CIRCLE INK 4; 128,88,87

INK a alors un effet local et ne concerne que les caractères affichés, le point tracé ou la droite dessinée par l'instruction. Dans l'exemple ci-dessus, un cercle vert est dessiné. Après quoi, la couleur de l'encre redevient la couleur globale ou la couleur implicite noire.

Structure

INK nombre

entière-expression-numérique [:]

INKEY\$

Chaîne de touches d'entrée

Emplacement sur le clavier

MODE ETENDU

N

Fonction

INKEY\$ est utilisée pour détecter la pression exercée sur les touches du clavier.

Comment utiliser INKEY\$

INKEY\$ n'a pas d'argument et est généralement utilisée pour affecter un caractère à une variable de

chaîne ou pour vérifier un caractère particulier, par exemple

```
70 LET a$=INKEY$
130 IF INKEY$="N" THEN STOP
```

Lors de l'exécution, INKEY\$ renvoie le caractère produit par la touche sur laquelle vous venez d'appuyer. Si vous n'appuyez pas sur une touche, le résultat est une chaîne vide (" "). Notez que INKEY\$ fait la différence entre les majuscules et les minuscules et les caractères appelés ou non par la touche SHIFT. (Utilisez IN pour détecter n'importe quelle touche sans distinguer entre les caractères.)

A la différence de INPUT, INKEY\$ n'attend pas, mais passe immédiatement à l'instruction suivante. C'est pourquoi elle est normalement placée à l'intérieur d'une boucle qui se répète jusqu'à ce que vous ayez appuyé sur la touche voulue. Cette ligne attend que vous appuyiez sur la touche Y (sans CAPS SHIFT ou CAPS LOCK).

```
60 IF INKEY$ <> "Y" THEN
GOTO 60
```

Structure
INKEY\$

INPUT

Emplacement sur le clavier

Instruction/Commande

INPUT permet d'entrer des données pendant qu'un programme est en cours d'exécution.

Comment utiliser INPUT

INPUT forme normalement une instruction de programme et s'utilise presque comme PRINT. Dans sa forme la plus simple, elle est suivie d'une variable numérique ou d'une variable de chaîne, par exemple

```
60 INPUT X
90 INPUT a$
```

L'ordinateur attend alors qu'un nombre ou une chaîne soient entrés. La valeur est affichée au début de la ligne inférieure au moment où celle-ci est introduite par le clavier. Lorsque vous appuyez sur ENTER, la valeur est affectée à la variable désignée et le programme continue.

Une instruction INPUT peut inclure plus d'une variable et affichera des caractères formant un message guide-opérateur. Tout se passe comme avec PRINT: le message est entre guillemets et des points-virgules ou des virgules séparent les éléments. Des instructions d'affichage comme INK, FLASH ou PAPER peuvent être intercalées, par exemple

```
80 INPUT INK2: "Comment vous appelez-vous?" n$. ("Quel âge avez-vous." + n$ + "?"). age
```

Les différences entre PRINT et INPUT sont les suivantes. Lorsqu'elle atteint une variable, INPUT attend, c'est pourquoi toutes les variables et expressions (comme

celle qui inclut n\$ ci-dessus) qui doivent être englobées dans les messages guide-opérateur doivent être *entre parenthèses*. L'affichage commence au début de la ligne inférieure et remonte sur l'écran si plus d'une ligne est utilisée. AT peut être utilisée dans une instruction INPUT comme elle le serait avec PRINT. AT 0,0 affiche les données au début de l'avant-dernière ligne de l'écran et les autres lignes, s'il y en a, sont affichées en remontant sur l'écran.

Comment arrêter INPUT

Si INPUT est suivie d'une variable numérique et si STOP est entrée, le programme s'arrête. Dans le cas d'une variable de chaîne, le premier guillemet qui apparaît peut être effacé, puis STOP doit être entrée pour arrêter le programme.

Utilisation de INPUT avec LINE

INPUT LINE ne peut être utilisée qu'avec des variables de chaînes. Normalement, INPUT avec une variable de chaîne provoque l'apparition de deux guillemets. Lorsque la chaîne est introduite par le clavier, elle apparaît entre les guillemets. Pour éliminer ces guillemets, utilisez INPUT LINE suivie de la variable de chaîne. Si un message guide-opérateur est nécessaire, il est placé entre INPUT et LINE, par exemple

```
70 INPUT "Comment vous appelez-vous?": LINE n$
```

Structure

```
INPUT [mess. guide-op.] [:] [:] ["]
variable-numérique
INPUT [mess. guide-op.] [:] [:] ["]
chaîne-variable
INPUT [mess. guide-op.] [:] [:] ["]
LINE chaîne-variable
[mess.
guide-op.] = [chaîne-constante]
[(chaîne-expression)] [AT
nombre
entier-expression-numérique,
nombre
entier-expression-numérique]
[instruction] [:] [:] ["]
Code caractère 238
```

INT nombre entier

Emplacement sur le clavier
MODE ETENDU
R

Fonction

INT transforme les nombres qui ne sont pas entiers en nombres entiers.

Comment utiliser INT

INT est suivie d'une valeur numérique, par exemple

```
70 LET x=INTy
```

Une expression doit être *entre parenthèses*. INT renvoie alors la valeur arrondie au nombre entier par défaut.

Exemple

La commande

```
PRINT INT 45.67, INT 7.66
```

affiche

Structure
INT constante-numérique
INT variable-numérique
INT (expression-numérique)

INVERSE

Emplacement sur le clavier
MODE ETENDU
M

Instruction/Commande

INVERSE permute les couleurs aux positions de caractères, la couleur de l'encre devenant la couleur du papier et vice-versa.

Comment utiliser INVERSE

INVERSE est normalement utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple

```
70 INVERSE 1
```

La valeur qui suit INVERSE est arrondie au nombre entier supérieur et est alors 0 ou 1. Avec INVERSE 1, tout ce qui est par la suite affiché par PRINT et INPUT est dans la couleur inverse. Avec INVERSE 0, l'encre et le papier reprennent leur couleur normale.

Notez que INVERSE peut être intercalée (insérée) dans des instructions d'affichage comme le serait INK. Toutefois, utilisée avec CIRCLE, PLOT ou DRAW, INVERSE 1 trace une ligne ou un point dans la couleur du papier, les faisant ainsi disparaître.

Structure

INVERSE nombre
entier-expression-numérique

LEN Longueur de chaîne

Emplacement sur le clavier
MODE ETENDU
K

Fonction

LEN donne la longueur d'une chaîne

Comment utiliser LEN

LEN est suivie d'une valeur de chaîne, par exemple

```
50 LET x=LEN a$
```

Une expression doit être *entre parenthèses*. LEN renvoie le nombre de caractères de la chaîne.

Exemple

La ligne suivante

```
120 INPUT a$: IF LEN a$ > 9
THEN GOTO 120
```

ne passe que des chaînes contenant 9 caractères maximum.

Structure

LEN chaîne-constante
LEN chaîne-variable
LEN (chaîne-expression)

LET

Emplacement sur le clavier
L

Instruction/Commande

LET est utilisée pour affecter une valeur à une variable. En BASIC, LET ne peut pas être omise d'une instruction d'affectation.

Comment utiliser LET

LET constitue normalement une instruction de programme, mais peut être utilisée comme une commande directe. Elle est suivie d'une variable numérique ou d'une chaîne, d'un signe égale et d'une valeur. La valeur peut être numérique ou une valeur de chaîne en fonction de la variable qui précède LET, par exemple

60 LET X=X+1

80 LET A\$="Correct"

La valeur est alors affectée à la variable.

Notez que de simples variables ne sont pas définies avant que des valeurs ne leur soient affectées par LET, READ ou INPUT. Les variables d'un tableau sont toutefois initialisées à 0 ou à une chaîne vide (voir DIM).

Structure

LET

variable-numérique=expression-numérique

LET

chaîne-variable=chaîne-expression

LINE**Emplacement sur le clavier**

SYMBOL SHIFT 3

Voir **SAVE**

LIST**Emplacement sur le clavier**

K

Commande/Instruction

LIST produit un listage du programme se trouvant alors dans la mémoire.

Comment utiliser LIST

LIST est normalement utilisée comme une commande directe, mais peut constituer une instruction de programme. Pour lister un programme complet, elle est utilisée seule. Après la commande directe

LIST

la première page du listage apparaît et les pages suivantes remonteront sur l'écran si vous appuyez sur n'importe quelle touche sauf N, la barre d'espacement et STOP.

LIST peut également être suivie d'un numéro de ligne sous forme numérique, par exemple

LIST 100

La valeur qui suit LIST est alors arrondie au nombre entier le plus proche et le listage commence à cette ligne. Si aucune ligne ne correspond à ce numéro, le listage commence à la ligne suivante.

Structure

LIST [nombre
entier-expression-numérique]

LLIST**Emplacement sur le clavier**

MODE ETENDU

V

Commande/Instruction

LLIST fait produire un listage imprimé du programme se trouvant alors dans la mémoire par les imprimantes Sinclair.

Comment utiliser LLIST

LLIST est utilisée exactement comme le serait LIST (pour plus de détails, voir LIST). Notez que pendant que le listage est imprimé, le contenu de l'écran ne change pas.

Structure

LLIST [nombre
entier-expression-numérique]

LN Logarithme (Naturel)**Emplacement sur le clavier**

MODE ETENDU

Z

Fonction

LN donne le logarithme naturel (le logarithme en base e) d'une valeur. Elle est l'inverse d'EXP.

Comment utiliser LN

LN est suivie d'une valeur numérique, par exemple

60 LN X=LN Y

Une expression doit être entre *parenthèses*. La valeur qui suit LN doit être supérieure à 0. LN renvoie alors le logarithme naturel de cette valeur.

Structure

LN constante-numérique
LN variable-numérique
LN (expression-numérique)

LOAD**Emplacement sur le clavier**

J

Commande/Instruction

LOAD charge un programme complet enregistré sur bande dans la mémoire.

Comment utiliser LOAD

LOAD est normalement utilisée comme une commande directe, instruction de programme si un nouveau programme doit être chargé. LOAD est suivie d'un nom de fichier qui est une valeur de chaîne d'une longueur de dix caractères maximum, par exemple

LOAD "nom de fichier"

Lors de l'exécution, le programme se trouvant alors dans la mémoire et toutes les valeurs de ses variables sont effacées. Le Spectrum recherche alors le programme "baptisé" et le charge lorsqu'il a

localisé. Notez que dans les noms de programmes, l'ordinateur fait la différence entre les minuscules et les majuscules.

Si une chaîne vide suit LOAD, comme dans la commande

LOAD ""

le Spectrum charge le premier programme complet qu'il rencontre.

Notez que LOAD est utilisée différemment lorsqu'un micro-support est connecté. Pour plus de détails, voir le manuel du micro-support et de l'interface 1.

Structure

LOAD chaîne-expression

LOAD CODE**Emplacement sur le clavier**

MODE ETENDU

Commande/Instruction

LOAD CODE est utilisée pour charger des informations enregistrées sur bande dans une section de la mémoire. Les informations sont des séries d'octets qui sont envoyés à une série d'adresses de la mémoire. LOAD CODE peut également être utilisée pour charger le contenu d'un écran ou des informations relatives à des graphiques utilisateur.

Comment utiliser LOAD CODE

LOAD CODE peut être utilisée en tant que commande directe ou constituer une instruction de programme. LOAD est suivie d'un nom de fichier qui est une valeur de chaîne, puis de CODE, par exemple

LOAD "donnée CODE

Le nom de fichier qui suit CODE est le nom des informations à charger et est soumis aux mêmes restrictions que les noms de programmes (voir LOAD). LOAD CODE cherche alors les informations "baptisées" et lorsqu'elle les a trouvées, affiche les octets suivis du nom. Le Spectrum charge alors les octets dans la mémoire aux adresses où est intervenue leur sauvegarde. Toutes les informations existantes sont recouvertes.

CODE peut également être suivie d'une ou deux valeurs numériques, séparées par une virgule, par exemple

LOAD "image" CODE

16384.6912

Les valeurs qui suivent CODE sont arrondies au nombre entier le plus proche et définissent ensuite l'adresse de départ (16384 ci-dessus) à laquelle les informations "baptisées" doivent être transférées et le nombre d'octets à transférer aux points d'implantation commençant à cette adresse. Si le numéro est incorrect, un message d'erreur de chargement est produit. Si CODE n'est suivi que d'une valeur, celle-ci définit l'adresse de

début d'où tous les octets doivent être localisés.

L'exemple ci-dessus peut aussi être exécuté à l'aide des mots-clés **LOAD SCREEN\$**.

Pour plus de détails à propos du stockage des octets, voir **SAVE CODE**.

Structure

LOAD chaîne-expression **CODE**
[nombre
entier-expression-numérique] [,
nombre
entier-expression-numérique]

LOAD DATA

Emplacement sur le clavier

MODE ETENDU

Instruction/Commande

LOAD DATA est utilisée pour transférer les tableaux d'une bande magnétique. Les tableaux sont enregistrés à l'aide de **DATA SAVE**.

Comment utiliser LOAD DATA
LOAD DATA peut être utilisée pour former une instruction de programme ou comme une commande directe. **LOAD** est suivie d'un nom de fichier qui est une valeur de chaîne. Elle est ensuite suivie de **DATA** et d'une lettre ou d'une paire de parenthèses vides, par exemple **278 LOAD "nombres"**

DATA n()

300 LOAD "noms" DATA n\$ ()

Le nom de fichier qui suit **LOAD** est le nom donné au tableau sur bande magnétique et il est soumis aux mêmes restrictions que les noms de programmes utilisés avec **LOAD**. La lettre ou la lettre\$ qui suit **DATA** est le nom qui sera donné au tableau dans le programme lorsqu'il est chargé et utilisé.

Lors de l'exécution, le Spectrum cherche le tableau "baptisé" sur la bande magnétique. Lorsqu'il l'a trouvé, le message tableau de nombre: ou tableau de caractère: suivi du nom apparaît et le tableau est chargé. Tout tableau se trouvant alors dans la mémoire et "baptisé" de la même lettre (n ou n\$ ci-dessus) est effacé et un nouveau tableau ayant la même lettre et les mêmes valeurs est produit. Notez qu'en ce qui concerne les tableaux de caractères, toute variable de chaîne se trouvant alors dans la mémoire et "baptisée" de la même lettre est elle aussi effacée.

Structure

LOAD chaîne-expression **DATA**
lettre (\$) ()

LOAD SCREEN\$

Emplacement sur le clavier

MODE ETENDU
SYMBOL SHIFT K

Instruction/Commande

LOAD SCREEN\$ permet de transférer directement de la bande le contenu d'un écran. Elle transfère les informations de la bande à la section de la mémoire qui contrôle le contenu de l'écran, produisant ainsi l'image.

Comment utiliser LOAD SCREEN\$

LOAD SCREEN\$ peut être utilisée pour former une instruction de programme ou comme une commande directe. **LOAD** est suivie d'un nom de fichier qui est une valeur de chaîne, puis de **SCREEN\$** par exemple

LOAD "image" SCREEN\$

Le nom de fichier qui suit **LOAD** est le nom donné aux informations de l'écran enregistrées sur la bande et il est soumis aux mêmes restrictions que les noms de programmes utilisés avec **LOAD**. Le Spectrum cherche alors les informations "baptisées" et lorsqu'il les a trouvées, les charge d'abord dans le fichier d'affichage, puis dans la section "attributs" de la mémoire. L'image se construit lentement dans la couleur d'encre et de papier alors utilisée, puis les attributs (couleurs véritables, etc) sont ajoutés.

Pour plus de détails sur le stockage des informations de l'écran, voir **SAVE SCREEN\$**.

Structure

LOAD chaîne-expression **SCREEN\$**

LPRINT Imprimante Ligne PRINTe

Emplacement sur le clavier **MODE ETENDU PRINT B/2**

Instruction/Commande

Tout comme **PRINT** fait apparaître un élément de données sur l'écran, **LPRINT** fait imprimer cet élément par les imprimantes Sinclair.

Comment utiliser LPRINT

LPRINT peut constituer une instruction de programme ou une commande directe. Elle est suivie d'éléments de données qui peuvent être séparés par des points-virgules, des virgules ou des apostrophes, par exemple

**60 LPRINT "Nombre": x'
"Nom", n\$, "Age", a**

Lorsqu'ils sont transmis à l'imprimante, les éléments sont imprimés dans le format dans lequel **PRINT** les ferait apparaître sur l'écran. Une instruction ou une commande **LPRINT** peut également inclure des instructions **TAB**, certaines commandes **CHR\$**, des instructions **INVERSE** et **OVER** et des codes de contrôle ayant le même effet que s'il s'agissait de **PRINT**. Une instruction **AT** peut également être présente, mais le numéro de ligne est ignoré et l'élément de données est imprimé à la position de colonne donnée à la même ligne.

Structure

LPRINT [TAB nombre

entier-expression-numérique;)]

[AT nombre

entier-expression-numérique,

nombre

entier-expression-numérique;]

[CHR\$ nombre

entier-expression-numérique;]

[instruction;]

[expression-numérique]

[chaîne-expression] [,] [,] [,]

MERGE

Emplacement sur le clavier

MODE ETENDU
SYMBOL SHIFT T

Instruction/Commande

MERGE fusionne deux programmes.

Comment utiliser MERGE

MERGE peut être utilisée pour former une instruction de programme ou comme une commande directe. Elle est suivie d'un nom de fichier qui est une valeur de chaîne, par exemple

500 MERGE "prog2"

Le nom de fichier qui suit **MERGE** est le nom du programme qui sera fusionné avec celui qui se trouve déjà dans la mémoire. Ce nom est soumis aux mêmes restrictions que les noms de programmes utilisés avec **LOAD**. **MERGE** charge alors le nouveau programme sans supprimer le programme existant. Toutefois, le nouveau programme recouvre les lignes du programme existant qui ont le même numéro de ligne que les lignes du nouveau programme; les variables ayant le même nom sont elles aussi recouvertes.

Structure

MERGE [chaîne-expression]

MOVE

Commande de prise en charge de fichier sur micro-support, voir le manuel du micro-support et de l'interface 1.

NEW

Emplacement sur le clavier

A

Commande/Instruction

NEW remet à zéro la zone **BASIC** de la mémoire (jusqu'à **RAMTOP**), supprimant tout programme se trouvant alors dans cette partie de la mémoire.

Comment utiliser NEW

NEW est normalement utilisée comme une commande directe, mais peut constituer une instruction de programme. Elle s'utilise seule. Lors de l'exécution, le programme et les variables sont effacés. La mémoire est remise à zéro jusqu'à **RAMTOP**, de façon à ce que les graphiques utilisateur stockés au-dessus de **RAMTOP** ne soient pas affectés.

NEXT *DS*

Emplacement sur le clavier

N

Instruction/Commande

NEXT est toujours utilisée en conjonction avec FOR pour produire une boucle FOR NEXT.

Comment utiliser NEXT

NEXT est normalement utilisée pour former une instruction de programme qui complètera une boucle FOR NEXT. Elle est suivie d'une lettre qui est la variable de contrôle de la boucle, par exemple

90 NEXT a

Dans le langage Sinclair BASIC, la variable de contrôle doit être présente.

Pour plus de détails sur les boucles FOR NEXT, voir FOR.

Structure
NEXT lettre

NOT *CA*

Emplacement sur le clavier

SYMBOL SHIFT S

Opérateur logique/Fonction

NOT est utilisée pour donner la version contraire d'une condition vraie de telle façon qu'une vérité fausse devienne vraie et vice versa.

Comment utiliser NOT

NOT est suivie par une condition ou par une valeur numérique, par exemple

**90 IF NOT x=y+z THEN
PRINT "Wrong"**

**90 LET correct=x=y+z:IF
NOT correct THEN PRINT
"Wrong"**

Quand NOT est suivie par une condition ($x=y+z$ ci-dessus) le Spectrum donne en premier lieu une valeur de 1 à une condition vraie et de 0 à une condition fausse, NOT agit ensuite comme une fonction, inversant la valeur produite de telle façon que le contraire de la condition puisse être essayé.

Notez qu'une condition devrait être entre parenthèses si elle comprend AND ou OR.

Si NOT est suivie par une valeur numérique, il retourne 0 si la valeur suivante est non zero et 1 si la valeur suivante est 0. Ainsi dans les exemples ci-dessus le Spectrum affiche "Wrong" si $x < y+z$ ou si correct a une valeur de 0.

Structure

NOT condition

NOT expression-numérique

OPEN# *AG*

Commande de prise en charge de fichier sur micro-support. Voir le manuel du micro-support et de l'interface 1.

OR *B10*

Emplacement sur le clavier

SYMBOL SHIFT U

Opérateur logique/Fonction

OR sert d'opérateur logique utilisé pour vérifier la véracité d'une combinaison de conditions. La combinaison globale n'est vraie qu'à condition qu'une ou plusieurs des conditions soient vraies. OR sert également de fonction exécutant des opérations binaires sur deux valeurs numériques.

Comment utiliser OR

En tant qu'opérateur logique, OR unit deux conditions dans une instruction dont la véracité doit être vérifiée, par exemple

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

Si l'une des conditions est vraie ou si les deux conditions sont vraies, la combinaison toute entière est vraie. Dans la ligne ci-dessus, l'une des conditions ($INKEY$="N"$ et $INKEY$="n"$) devient vraie dès que vous appuyez sur la touche N, indépendamment du fait que CAPS SHIFT ou CAPS LOCK soient utilisées. La combinaison toute entière est alors vraie et le programme s'arrête.

OR en tant que fonction

Le ZX Spectrum + affecte la valeur numérique 1 à une condition vraie et 0 à une condition fausse. Il identifie n'importe quelle valeur autre que zéro comme vraie et 0 comme fausse. OR peut donc être suivi ou précédé d'une valeur numérique, par exemple

40 LET x=y OR z

La variable x est alors affectée de la valeur 1 si z est autre que zéro ou une condition vraie, ou bien de la valeur de y si z=0 ou est une condition fausse.

Notez que le Spectrum n'évalue pas les combinaisons de valeurs numériques conformément aux tables de vérité ordinaires.

Structure

condition or condition
expression-numérique OR
expression-numérique

OUT *B11*

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT O

Instruction/Commande

OUT envoie un octet à une adresse d'accès entrée/sortie pour commander un dispositif de sortie.

Comment utiliser OUT

OUT peut être utilisée pour former une instruction de programme ou comme une commande directe. Elle est suivie de deux valeurs numériques séparées par une virgule, par exemple

40 OUT 254,3

Les deux valeurs sont arrondies au nombre entier le plus proche. La première valeur (254) peut varier entre 0 et 65535 et est l'adresse de l'accès. La seconde valeur (3) peut varier entre 0 et 255 et est l'octet qui sera transmis à l'adresse.

Les bits 0 à 2 de l'octet transmis à l'adresse 254 déterminent la couleur de la bordure; dans l'exemple ci-dessus, la bordure devient magenta. A cette adresse, le bit 3 commande l'embase MIC et le bit 4, le haut-parleur. L'adresse 255 commande l'imprimante et les accès 254, 247 et 239 sont utilisés avec d'autres périphériques.

Structure

OUT nombre
entier-expression-numérique,
nombre
entier-expression-numérique

OVER *DS*

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT N

Instruction/Commande

OVER est utilisée pour supprimer deux caractères. Elle peut également être utilisée pour tracer des points ou dessiner droites et des courbes dans une couleur de papier au lieu d'une couleur d'encre.

Comment utiliser OVER

OVER est normalement utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple

80 OVER 1

La valeur qui suit OVER est arrondie au nombre entier le plus proche et peut alors être 0 ou 1. OVER 0 qui est l'état implicite (prédéfini) remplace un caractère par un autre, effaçant le premier caractère qui se trouvait à la position. OVER 1 combine deux caractères affichés à la même position.

OVER peut être intercalée (insérée) dans une instruction PRINT ou INPUT comme le serait INK, de telle sorte qu'elle n'affecte que les caractères affichés par l'instruction. L'instruction souligne un mot, par exemple

**60 PRINT AT 11, 15; "Oui";
OVER 1 AT 11, 15; " "**

Toutefois, notez que les caractères sont combinés de façon à ce que l'on obtienne la couleur du papier lorsque celle-ci chevauche la couleur de l'encre.

OVER dans les graphiques à haute résolution

OVER peut être utilisée avec PLOT, DRAW et CIRCLE. Sans OVER, les droites et les courbes peuvent se chevaucher, mais elles doivent avoir la même couleur d'encre sinon cette couleur change la position toute entière du caractère lorsque le

droites et les courbes se rencontrent. Si l'on utilise OVER 1, les droites et les courbes produisent la couleur du papier là où elles chevauchent ou rencontrent d'autres caractères. Tracer des points ou dessiner des lignes et des courbes exactement à la même position avec OVER 1 les fait disparaître.

Structure
OVER nombre
entier-expression-numérique

PAPER

Emplacement sur le clavier
MODE ETENDU
SYMBOL SHIFT C

Instruction/Commande

PAPER est utilisée pour sélectionner la couleur de papier ou de l'arrière-plan qui sera utilisée sur l'écran. Cette couleur peut être celle de l'arrière-plan tout entier ou bien celle sur laquelle apparaissent des caractères, des points ou des droites individuels.

Comment utiliser PAPER

PAPER peut être utilisée pour former une instruction de programme ou comme une commande directe. Elle est suivie d'une valeur numérique, par exemple

80 PAPER x

La valeur qui suit PAPER est arrondie au nombre entier le plus proche et elle peut varier entre 0 et 9. Les couleurs de papier alors produites sont identiques à celles produites par INK. Tout comme les couleurs d'encre, les couleurs de papier peuvent être globales ou locales; elles deviennent locales lorsqu'elles sont intercalées (insérées) dans des instructions d'affichage. Pour plus de détails, voir INK.

Chaque fois que des caractères sont imprimés à la suite d'une instruction PAPER globale ou locale, l'arrière-plan de la position toute entière du caractère prend la couleur sélectionnée. La même chose se produit lorsque des points sont tracés ou que des droites et des cercles sont dessinés à l'aide d'une instruction PAPER intercalée, mais pas après une commande ou une instruction globales.

Pour produire un arrière-plan coloré sur toute la zone d'affichage, il faut utiliser CLS après une instruction PAPER. L'écran tout entier prend cette couleur qui demeure alors la couleur de l'arrière-plan.

Structure
PAPER nombre
entier-expression-numérique[.]

PAUSE

Emplacement sur le clavier
M

Instruction/Commande

PAUSE peut être utilisée pour que le programme attende pendant un temps déterminé ou indéterminé.

Comment utiliser PAUSE

PAUSE est normalement utilisée pour former une instruction de programme. Elle est suivie d'une valeur numérique, par exemple

130 PAUSE 100

La valeur qui suit PAUSE est arrondie au nombre entier le plus proche et peut varier entre 0 et 65535. La pause est définie comme le nombre de trames de la télévision, ce qui veut dire que 50 produit une pause d'une seconde dans les pays où la fréquence de trame est 50Hz.

Notez toutefois qu'une pause peut être raccourcie en appuyant sur n'importe quelle touche et que PAUSE 0 produit une pause indéterminée qui ne prend fin que lorsque vous appuyez sur une touche.

Structure
PAUSE nombre
entier-expression-numérique

PEEK

Emplacement sur le clavier
MODE ETENDU
O

Fonction

PEEK donne la valeur de l'octet stocké à une adresse particulière de la mémoire.

Comment utiliser PEEK
PEEK est suivie d'une valeur numérique, par exemple

80 LET x = PEEK(256*y)

Notez qu'une expression doit être entre parenthèses. Si nécessaire, la valeur qui suit PEEK est arrondie au nombre entier le plus proche et peut varier entre 0 et 65535 pour donner une adresse de la mémoire. PEEK renvoie alors la valeur de l'octet (un nombre entre 0 et 255) à l'adresse spécifiée.

Exemple

Le nombre de trames de la télévision qui ont été produites depuis que le Spectrum a été mis en marche est stocké aux adresses 23672 à 23674. Vu que les trames sont produites à un rythme régulier, obtenir ces points d'implantation à l'aide de l'instruction PEEK permet de mesurer le temps. La ligne suivante affiche le temps en secondes qui s'est écoulé depuis que le Spectrum a été mis en fonction (moins le temps passé à produire le son et à utiliser des périphériques comme le magnétophone à cassette et l'imprimante).

10 PRINT (PEEK 23672 + 256*PEEK 23673 + 65536*PEEK 23674)/50

Remarque Si la fréquence est 60Hz (comme en Grande-Bretagne), remplacez 50 par 60.

Structure

PEEK nombre
entier-constante-numérique
PEEK nombre
entier-variable-numérique
PEEK (nombre
entier-expression-numérique)

PI

Emplacement sur le clavier
MODE ETENDU
M

Fonction

PI donne la valeur de pi (π) qui sera utilisé dans les calculs.

Comment utiliser PI

Lorsque PI est utilisée dans une instruction ou une commande, elle n'a pas besoin de valeurs ou de variables, par exemple

DRAW 255 0, -PI

PI produit une valeur de 3.1415927 et la commande ci-dessus trace un grand demi cercle sur l'écran.

Structure
PI

PLOT

Emplacement sur le clavier
Q

Instruction/Commande

PLOT est utilisée dans les graphiques à haute résolution pour tracer un pixel ou point de couleur à une position particulière de l'écran.

Comment utiliser PLOT

PLOT est utilisée pour former une instruction de programme ou comme une commande. Elle est normalement suivie de deux valeurs numériques séparées par une virgule, par exemple

50 PLOT 128,87

Si nécessaire, les valeurs qui suivent PLOT sont arrondies à un nombre entier. La première valeur peut alors varier entre 0 et 255 et définit la coordonnée horizontale d'une position de l'écran. La seconde valeur peut varier entre 0 et 175 et définit une coordonnée verticale. Un pixel est alors tracé dans la couleur de l'encre alors utilisée à la position ainsi définie.

Notez les effets que les instructions ou les commandes de couleur ont sur PLOT. Après OVER 1, un point se trouvant à la même position prend la couleur du papier. Après INVERSE 1, le point est tracé dans la couleur de papier alors utilisée. Après BRIGHT 1 ou FLASH 1, la position toute entière du caractère sur l'écran à basse résolution sur lequel le pixel est tracé sera brillante ou scintillante.

Ces quatre mots-clés et INK peuvent aussi être intercalés (insérés) dans une instruction PLOT, tout comme le serait PRINT, par exemple

160 PLOT INK 2; x, y

Leur effet est identique, mais il est local et limité au pixel tracé par l'instruction. Si PAPER est intercalée dans une instruction PLOT, la couleur de la position toute entière autour du pixel prend la couleur donnée

Structure

PLOT [instruction;] nombre entier-expression nombre entier-expression

POINT

A10

Emplacement sur le clavier

MODE ETENDU
SYMBOL SHIFT 8

Fonction

POINT est utilisée pour déterminer si la couleur à une position particulière de l'écran à haute résolution est une couleur d'encre ou une couleur de papier.

Comment utiliser POINT

POINT est suivie de deux valeurs numériques *séparées par une virgule et entourées de parenthèses*, par exemple

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

Si nécessaire, les deux valeurs qui suivent **POINT** sont arrondies à un nombre entier. La première valeur peut varier entre 0 et 255 et définit la coordonnée horizontale d'un pixel sur l'écran. La seconde valeur varie entre 0 et 175 et définit une coordonnée verticale. **POINT** renvoie alors 1 si le pixel à la position définie est de la couleur de l'encre et 0 s'il est de la couleur du papier.

Structure

POINT (nombre entier-expression-numérique, nombre entier-expression-numérique)

POKE

B11

Emplacement sur le clavier

O
Instruction/Commande

POKE est utilisée pour changer la valeur de l'octet à une adresse particulière de la mémoire.

Comment utiliser POKE

POKE est utilisée pour former une instruction de programme ou comme une commande. Elle est suivie de deux valeurs numériques *séparées par une virgule*, par exemple

POKE 23609,255

Les deux valeurs qui suivent **POKE** sont, si nécessaire, arrondies au nombre entier le plus proche. La première valeur peut varier entre 0 et 65535 et est une adresse de RAM. La seconde valeur varie de 0 à 255 et est l'octet qui doit être inscrit à l'adresse définie.

Structure

POKE nombre entier-expression-numérique, nombre entier-expression-numérique)

PRINT

B12

Emplacement sur le clavier

P
Instruction/Commande

PRINT affiche des données sur l'écran. Les données peuvent être un caractère ou un jeu de caractère. Une instruction **PRINT** peut incorporer d'autres mots-clés définissant la couleur et la position des données.

Comment utiliser PRINT

PRINT peut être utilisée seule ou être suivie des données. Ces données peuvent être des expressions numériques ou des expressions de chaînes ou encore un mélange des deux.

Lorsque **PRINT** est utilisée avec des données, deux éléments ou plus de ces données doivent être séparés par un point-virgule ou une apostrophe.

Certains autres mots-clés peuvent être insérés dans n'importe quel ordre entre **PRINT** et les données, à condition que chaque instruction formée par le mot-clé se termine par un point-virgule. Ces mots-clés sont CHR\$, TAB, AT, INK, PAPER, FLASH, BRIGHT, INVERSE et OVER.

PRINT avec des chaînes

PRINT seule ou suivie d'une chaîne vide (" ") produit une ligne blanche ou déplace le curseur au début de la ligne suivante.

PRINT suivie d'une constante de chaîne (n'importe quel caractère entre guillemet.) affiche les caractères tels qu'ils apparaissent entre les guillemets. La commande

PRINT "3/542/76/21"

par exemple, affiche

3/542/76/21

PRINT suivie d'une variable ou d'une expression de chaîne affiche le(s) chaîne(s) qu'elles représentent.

PRINT avec des nombres

PRINT suivie d'une expression numérique affiche la valeur de l'expression. Les nombres apparaissent dans la notation décimale: il y a huit chiffres significatifs maximum, sans zéros à droite de la décimale.

Des nombres très grands et très petits apparaissent dans une notation scientifique abrégée, comme deux chiffres séparés par la lettre E. Ceci représente un nombre dont la première partie (la mantisse) est élevée à la puissance de la seconde (l'exposant). La commande

PRINT 3/542/76/21

par exemple, affiche

3.468079E-6

Mise en page avec PRINT et des signes de ponctuation

PRINT suivie d'éléments de données séparés par un point-virgule affiche les éléments les uns à côté des autres, sans espace entre eux. La commande

PRINT 1,2,3

affiche

123

PRINT suivie d'éléments de données séparés par une virgule affiche chaque élément au début ou au milieu d'une ligne; en fonction de la position du premier élément. La commande

PRINT 1,2,3

affiche

1

2

3

PRINT suivie d'éléments de données séparés par une apostrophe affiche l'élément après l'apostrophe au début de la ligne suivante. La commande

PRINT 1'2'3

affiche

1

2

3

Si une instruction ou une commande **PRINT** se termine par un point-virgule, une virgule ou une apostrophe, l'élément affiché par l'instruction **PRINT** suivante est affecté de la même manière.

PRINT et divers autres mots-clés

PRINT peut être suivie de TAB, d'une valeur numérique, d'un point-virgule, puis d'un élément de données, par exemple

60 PRINT TAB x; a\$

La valeur qui suit TAB (x) est, si nécessaire, arrondie au nombre entier le plus proche et est ensuite divisée par 32; le résultat produit est une valeur entre 0 et 31. L'élément de données est alors affiché à cette position de colonne, à la même ligne ou à la ligne suivante.

PRINT peut être suivie de AT et de deux valeurs numériques *séparées par une virgule*, d'un point-virgule et d'un élément de données, par exemple

50 PRINT AT 1, c; "données"

La première valeur (1) peut varier entre 0 et 21 et détermine le nombre de lignes ou de rangées sur lesquelles les données seront affichées. La seconde valeur (c) peut varier entre 0 et 31 et détermine le numéro de la colonne dans laquelle le premier caractère ou chiffre des données apparaîtra. Les valeurs qui ne sont pas des nombres entiers sont acceptées et arrondies au nombre entier le plus proche. La commande **PRINT AT 11,16; "** produit une étoile au milieu de l'écran.

PRINT peut également être suivie d'une ou plusieurs fonctions CHR\$,

Pour plus de détails, voir CHRS.

PRINT et les mots-clés couleur
L'écran produit par PRINT est affecté par les instructions ou commandes de couleur données par INK, PAPER, FLASH, BRIGHT, INVERSE et OVER et en cours d'exécution. PRINT peut également être suivie d'une ou plusieurs de ces six instructions, séparées par un point-virgule précédant l'élément de données, par exemple

50 PRINT AT 11, 16; INK2; FLASH 1; "

L'élément de données est alors affiché avec les attributs spécifiés par let(s) mot(s)-clé(s) couleur. Ces attributs sont locaux et ne concernent que l'élément affiché. Une fois l'instruction PRINT exécutée, ils reviennent à leur valeur implicite ou à leur valeur globale précédente. PRINT obéira également aux codes de contrôle de couleur locale insérés en même temps que les données (voir page 33).

Structure

PRINT [TAB nombre
entier-expression-numérique;]

[AT nombre

entier-expression-numérique,
nombre

entier-expression-numérique;]

[CHRS (nombre

entier-expression-numérique);]

[instruction;]

[expression-numérique]

[chaîne-expression] [:] [:] [:]

RANDOMIZE

Emplacement sur le clavier

Instruction/Commande

RANDOMIZE, RAND sur le clavier, est utilisée en conjonction avec RND pour produire des séries de nombres aléatoires ou prévisibles.

Comment utiliser RANDOMIZE

RANDOMIZE est utilisée pour former une instruction de programme ou comme une commande. Elle est suivie d'une valeur numérique en option, par exemple,

RANDOMIZE 1

10 RANDOMIZE

La valeur qui suit RANDOMIZE est arrondie au nombre entier le plus proche et peut varier entre 0 et 65535. Une valeur supérieure à 0 positionne la variable de système SEED à cette valeur, à la suite de quoi RND génère toujours la même série de nombres (pour plus de détails sur les variables de système, voir page 48). La série obtenue dépend de la valeur de RANDOMIZE.

Si RANDOMIZE est suivie de 0 ou n'est suivie d'aucune valeur, SEED est alors affectée de la valeur d'une autre variable de système appelée FRAMES qui compte les trames de la

télévision depuis que le Spectrum a été mis en fonction. Vu que SEED change 50 ou 60 fois par seconde, la série de nombres générée par RND à la suite de RANDOMIZE ou de RANDOMIZE 0 est extrêmement aléatoire.

Structure

RANDOMIZE (nombre
entier-expression-numérique)

READ

Emplacement sur le clavier

MODE ETENDU
A

Instruction/Commande

READ est utilisée en conjonction avec DATA pour affecter des valeurs aux variables en utilisant les valeurs d'une instruction DATA.

Comment utiliser READ

READ est normalement utilisée pour former une instruction de programme. Elle est suivie d'une ou plusieurs variables numériques ou variables de chaîne séparées par une virgule, par exemple

20 READ A\$, X

Lorsque READ est exécutée pour la première fois, elle prend le même nombre de valeurs qu'il y a de variables depuis le début de la première liste de DATA et affecte les valeurs aux variables par ordre.

Lorsque READ est de nouveau exécutée, la série suivante de valeurs DATA est affectée aux variables "baptisées" dans l'instruction READ, et ainsi de suite. Pour plus de détails, voir DATA.

Structure

READ variable-numérique

[variable-numérique]

[chaîne-variable]

READ chaîne-variable

[variable-numérique]

[chaîne-variable]

REM

REMARQUE

Emplacement sur le clavier

Instruction

REM est utilisée pour introduire des remarques ou des rappels dans un programme. Il peut s'agir du titre ou de l'auteur du programme ou encore d'explications de lignes du programme, comme la raison d'être d'une variable. Les remarques n'interviennent pas dans l'exécution du programme et n'apparaissent que dans le listing.

Comment utiliser REM

REM constitue une ligne de programme à elle seule ou bien la dernière instruction d'une ligne. Elle est suivie de n'importe quelle remarque pouvant être introduite par le clavier, par exemple

80 INPUT N\$. REM n\$ est le nom

Lorsque l'ordinateur rencontre REM, il ignore tout ce qui vient à la

suite de REM sur cette ligne.

Structure

REM n'importe quel caractère

RESTORE

Emplacement sur le clavier

MODE ETENDU

S

Instruction/Commande

RESTORE est utilisée en conjonction avec READ et DATA pour faire prendre par READ les valeurs d'une instruction DATA particulière au lieu de la première ou seconde instruction DATA du programme.

Comment utiliser RESTORE

RESTORE est normalement une instruction de programme. Elle est suivie d'une valeur numérique en option, par exemple

100 RESTORE 800

La valeur qui suit RESTORE est arrondie au nombre entier le plus proche, si nécessaire et devrait alors être le numéro d'une ligne de programme contenant une instruction DATA. Après RESTORE, l'instruction READ suivante affectera les valeurs contenues dans cette instruction DATA. Si la ligne numérotée n'existe pas ou ne contient pas d'instruction DATA, READ passe alors à l'instruction DATA suivante, après cette ligne.

Si RESTORE est suivie de 0 ou n'est suivie d'aucune valeur, l'instruction READ suivante va à la première instruction DATA du programme.

Structure

RESTORE (nombre
entier-expression-numérique)

RETURN

Emplacement sur le clavier

Y

Instruction/Commande

RETURN est utilisée pour terminer un sous-programme et renvoyer l'ordinateur au programme principal ou à un sous-programme précédent.

Comment utiliser RETURN

RETURN est normalement utilisée pour former une instruction de programme. Elle est utilisée seule à la fin d'un sous-programme, par exemple

1000 RETURN

Lors de l'exécution, le programme se branche sur l'instruction qui suit la dernière instruction GOSUB exécutée.

Pour plus de détails, voir GOSUB.

Structure

RETURN

RND

Nombre aléatoire

Emplacement sur le clavier

MODE ETENDU

T

Fonction

RND est utilisée pour générer un nombre aléatoire.

Comment utiliser RND

RND est utilisée seule dans une instruction ou une commande, par exemple

ØØ LET X=RND

RND renvoie alors un nombre aléatoire inférieur à 1 et supérieur ou égal à 0.

Lorsque le Spectrum est mis en fonction ou remis à l'état initial ou lorsque vous utilisez NEW, les nombres sont alors renvoyés par RND dans la même séquence. La séquence est produite en élevant à 2 à diverses puissances (75, 75*75, 75*75*75, etc.), en divisant chaque puissance par 65537 et en n'utilisant que le résultat, en soustrayant 1 du résultat qui est alors divisé par 65536.

Si une séquence plus aléatoire ou une autre séquence fixe sont nécessaires, utilisez RANDOMIZE avant RND.

Nombres entiers aléatoires

N'importe quel nombre entier de 1 à x est produit par INT (RND*x)+1. Pour générer un nombre entier aléatoire entre 0 et x, utilisez INT (RND*x+0.5).

**Structure
RND****RUN**

86.

Emplacement sur le clavier

R

Commande/Instruction

RUN exécute un programme, normalement à partir de la première ligne.

Comment utiliser RUN

RUN peut être utilisée comme une commande directe ou elle peut constituer une instruction de programme. Elle est suivie d'une valeur numérique en option, par exemple

RUN 5Ø

Si RUN n'est suivie d'aucune valeur, le programme est exécuté à partir de la première ligne. S'il y a une valeur, elle est arrondie au nombre entier le plus proche si nécessaire et le programme est alors exécuté à partir de cette ligne. Si cette ligne n'existe pas, le programme est exécuté à partir de la ligne suivante.

Si un programme a été sauvegardé à l'aide de LINE, il est exécuté automatiquement dès qu'il a été chargé et RUN n'est donc pas nécessaire.

Structure

RUN [nombre entier-expression-numérique]

SAVE

C4

Emplacement sur le clavier

S

Commande/Instruction

SAVE transmet un programme au magnétophone à cassette qui le stockera sur bande magnétique.

Comment utiliser SAVE

SAVE est normalement utilisée comme une commande directe, mais peut constituer une instruction de programme. Elle est suivie d'un nom de fichier qui est une valeur de chaîne, par exemple

SAVE "nom de fichier"

Le nom de fichier contient un maximum de dix caractères. Lors de l'exécution, le message

Démarrez la bande, ensuite appuyez sur une touche

est affiché. Lorsque vous appuyez sur la touche, le programme est transmis au magnétophone à cassette et à la fin du processus, le message ØOK, Ø.1 apparaît

Exécution automatique

Si le programme stocké doit être exécuté automatiquement après avoir été chargé, SAVE doit être utilisée en conjonction avec LINE. Le nom du programme est suivi de LINE et d'une valeur numérique, par exemple

SAVE "nom de fichier" LINE1

La valeur qui suit LINE est arrondie, si nécessaire, au nombre entier le plus proche et devrait alors être 1 ou le numéro d'une ligne du programme. Le programme est alors enregistré sur la bande magnétique comme il le serait avec SAVE. Lors du chargement, le programme est exécuté automatiquement à partir de la ligne ayant le numéro défini ou si cette ligne n'existe pas, à partir de la ligne suivante.

Structure

SAVE chaîne-expression [LINE nombre entier-expression-numérique]

SAVE CODE**Emplacement sur le clavier**

S
MODE ETENDU
I

Commande/Instruction

SAVE CODE transmet une partie des informations de la mémoire au magnétophone à cassette qui les enregistrera sur bande magnétique. Les informations peuvent alors être remises dans la mémoire à l'aide de LOAD CODE.

Comment utiliser SAVE CODE

SAVE CODE peut être utilisée en tant que commande directe ou constituer une instruction de programme. SAVE est suivie d'un nom de fichier qui est une valeur de chaîne, puis de CODE qui est lui suivi de deux valeurs numériques séparées par une virgule, par exemple

**SAVE "image" CODE
16384, 6912**

Le nom de fichier qui suit CODE peut contenir dix caractères maximum. Si nécessaire, les deux valeurs qui suivent CODE sont arrondies au nombre entier le plus proche. La première donne l'adresse de début (16384 ci-dessus) des informations de la mémoire et la seconde (6912), le nombre d'octets qui seront stockés.

Les informations sont alors enregistrées sur la bande comme un programme le serait avec SAVE.

Structure

SAVE chaîne-expression CODE nombre entier-expression-numérique, nombre entier-expression-numérique

SAVE DATA**Emplacement sur le clavier**

S
MODE ETENDU
D

Instruction/Commande

SAVE DATA stocke un tableau sur bande. Le tableau peut ensuite être chargé à l'aide de LOAD DATA.

Comment utiliser SAVE DATA

SAVE DATA peut être utilisée pour former une instruction de programme ou comme une commande directe. SAVE est suivie d'un nom de fichier, puis de DATA, puis d'une lettre ou d'une lettre \$ et enfin, de deux parenthèses vides, par exemple

**45Ø SAVE "nombres" DATA n()
75Ø SAVE "noms" DATA n\$()**

Le nom de fichier du tableau peut contenir dix caractères maximum. La lettre ou la lettre \$ qui suit DATA est le nom du tableau dans le programme qui sera stocké sur bande. Le tableau est ensuite enregistré sur bande comme un programme le serait avec SAVE.

Structure

SAVE chaîne-expression DATA lettre \$()

SAVE SCREENS**Emplacement sur le clavier**

S
MODE ETENDU
SYMBOL SHIFT K

Commande/Instruction

SAVE SCREENS stocke le contenu de l'écran sur bande. Ce contenu peut par la suite être rechargé dans l'ordinateur à l'aide de LOAD SCREENS.

Comment utiliser SAVE SCREENS

SAVE SCREENS peut être utilisée comme une commande directe ou pour former une instruction de programme. SAVE est suivie d'un nom de fichier qui est une valeur de chaîne, puis de SCREENS, par

exemple

SAVE "image" SCREEN\$

Le nom de fichier peut contenir dix caractères maximum. Le contenu de l'écran est ensuite enregistré sur bande comme le serait un programme avec SAVE.

Structure

SAVE chaîne-expression SCREEN\$

SCREEN\$ *C10*

Emplacement sur le clavier

MODE ETENDU
SYMBOL SHIFT K

Fonction

SCREEN\$ détecte le caractère qui apparaît à une position particulière de l'écran.

Comment utiliser SCREEN\$

SCREEN\$ est suivie de deux valeurs numériques séparées par une virgule et entourées de parenthèses, par exemple

160 IF SCREEN\$(I,C)="X"
THEN PRINT "BOUM"

Si nécessaire, les valeurs qui suivent SCREEN\$ sont arrondies au nombre entier le plus proche. La première valeur (I) peut varier entre 0 et 21 et donne le numéro de ligne d'une position de l'écran. La seconde valeur (C) peut varier entre 0 et 31 et donne le numéro de colonne de la position. SCREEN\$ renvoie alors le caractère à cette position, en tant que constante de chaîne (le caractère entre guillemets, "X", par exemple). S'il n'y a pas de caractère à la position, SCREEN\$ renvoie une chaîne vide ("").

Structure

SCREEN\$ (nombre
entier-expression-numérique)
nombre
entier-expression-numérique)

SGN SiGNe *C6*

Emplacement sur le clavier

MODE ETENDU

F

Fonction

SGN indique si un nombre est positif, négatif ou égal à zéro.

Comment utiliser SGN

SGN est suivie d'une valeur numérique, par exemple

50 LET X=SGN Y

Une expression doit être entre parenthèses. SGN renvoie alors 1 si la valeur de l'argument (Y) est positive, -1 si elle est négative et 0 si elle est égale à zéro.

Structure

SGN constante-numérique
SGN variable-numérique
SGN (expression-numérique)

SIN SiNus *3*

Emplacement sur le clavier

MODE ATENDU

Q

Fonction

SIN donne le sinus d'un angle.

Comment utiliser SIN

SIN est suivie d'une valeur numérique, par exemple

80 LET X=SIN Y

Une expression doit être entre parenthèses. La valeur qui suit SIN est l'angle en radians et SIN renvoie le sinus de l'angle. Les degrés peuvent être convertis en radians en multipliant par $\pi/180$

Notez que SIN renvoie une valeur positive pour les angles entre 0 et 180 degrés et une valeur négative pour les angles entre 180 et 360 degrés.

Structure

SIN constante-numérique
SIN variable-numérique
SIN (expression-numérique)

SQR Racine carrée *28*

Emplacement sur le clavier

MODE ETENDU

H

Fonction

SQR donne la racine carrée d'un nombre.

Comment utiliser SQR

SQR est suivie d'une valeur numérique, par exemple

70 LET X=SQR Y

Une expression doit être entre parenthèses. La valeur qui suit SQR (Y) doit être supérieure à zéro et SQR produit alors sa racine carrée.

Structure

SQR constante-numérique
SQR variable-numérique
SQR (expression-numérique)

STEP *25*

Emplacement sur le clavier

SYMBOL SHIFT D

Voir FOR

STOP *23*

Emplacement sur le clavier

SYMBOL SHIFT A

Instruction/Commande

STOP arrête le programme à un point déterminé. Il est parfois nécessaire d'utiliser STOP pour arrêter la section principale d'un programme et placer les sous-programmes dans une section séparée. STOP est également très utile pour mettre au point un programme.

Comment utiliser STOP

STOP est normalement utilisée pour former une instruction de programme. Elle est utilisée seule, par exemple

650 STOP

Lors de l'exécution, le programme s'arrête et le message

9 STOP instruction

apparaît avec le numéro de ligne et d'instruction auquel le programme est arrêté.

Les procédures de mise au point comme l'affichage et la modification de la valeur des variables peuvent alors être entreprises. Si vous entrez CONTINUE, le programme est relancé à l'instruction suivante, avec les nouvelles valeurs.

Structure

STOP

STR\$ *B8*

Emplacement sur le clavier

MODE ETENDU Y

Fonction

STR\$ convertit un nombre en une chaîne.

Comment utiliser STR\$

STR\$ est suivie d'une valeur numérique, par exemple

90 LET A\$=STR\$ X

Une expression doit être entre parenthèses. STR\$ renvoie alors la valeur de son argument (X) en tant que constante de chaîne. Si la valeur 65 était affectée à X, l'instruction ci-dessus affecterait la valeur "65" à A\$.

Structure

STR\$ constante-numérique
STR\$ variable-numérique
STR\$ (expression-numérique)

TAB *B12*

Emplacement sur le clavier

MODE ETENDU

P

Voir L-PRINT, PRINT

TAN TANgente *B5*

Emplacement sur le clavier

MODE ETENDU

E

Fonction

TAN donne la tangente d'un angle.

Comment utiliser TAN

TAN est suivie d'une valeur numérique, par exemple

130 LET X=TAN Y

Une expression doit être entre parenthèses. La valeur qui suit TAN est l'angle en radians et TAN renvoie la valeur de l'angle. Les degrés peuvent être convertis en radians en les multipliant par $\pi/180$.

Structure

TAN constante-numérique
TAN variable-numérique
TAN (expression-numérique)

THEN *27*

Emplacement sur le clavier
SYMBOL SHIFT G

voir IF

TO *cl***Emplacement sur le clavier**
SYMBOL SHIFT F

Fonction

TO a deux utilisations différentes en Sinclair BASIC. Elle est utilisée en tant que conjonction avec FOR pour constituer une boucle NEXT FOR (pour plus de détails, voir FOR) et elle sert également à découper des chaînes (leur division en sous-chaînes plus petites).

Comment utiliser TO pour découper des chaînes

TO est utilisée pour définir le premier et le dernier caractère d'une sous-chaîne faisant partie d'une chaîne principale. TO est précédée d'une valeur de chaîne, d'une parenthèse, puis d'une valeur numérique en option. Elle est suivie d'une autre valeur numérique en option, puis d'une parenthèse, par exemple

80 PRINT a\$ (4 TO 7)

Une expression de chaîne doit toujours être *entre parenthèses*. La valeur de chaîne (a\$) est la chaîne qui sera découpée. Les deux valeurs numériques (4 et 7) déterminent les positions du premier et du dernier caractères de la sous-chaîne de la chaîne. TO renvoie alors la sous-chaîne (les caractères 4 à 7 de a\$).

La valeur implicite de la première valeur numérique est 1 et celle de la seconde est égale à la position du dernier caractère de la chaîne.

Structure

Chaîne-constante
([expression-numérique] TO
[expression-numérique])
chaîne-variable
([expression-numérique] TO
[expression-numérique])
(chaîne-expression)
([expression-numérique] TO
[expression-numérique])

USR Sous-programme utilisateur *cl***Emplacement sur le clavier**
MODE ETENDU

Fonction

USR est utilisée pour appeler un sous-programme machine rodé qui a été introduit dans la mémoire à une adresse spécifique. Elle est également utilisée pour introduire les données des graphiques utilisateur dans les lieux d'implantation qui leur sont réservés en haut de la mémoire.

USR et le code machine

Lorsque le code machine est utilisé, USR est suivie d'une valeur numérique, par exemple

80 PRINT USR 65000
100 RANDOMIZE USR 65000

Une expression doit être *entre parenthèses*. La valeur qui suit USR est arrondie au nombre entier le plus proche et est alors l'adresse de début de la mémoire à laquelle le code machine du sous-programme a été introduit. N'importe quelle instruction contenant USR appelle alors le sous-programme se trouvant à cette adresse et USR renvoie la valeur du contenu du registre bc.

USR et les graphiques utilisateur

Pour produire des graphiques utilisateur, USR est utilisée avec POKE. Elle suivie d'une constante ou d'une variable de chaîne donnant l'adresse d'une instruction POKE, par exemple

50 POKE USR "a", 255

La valeur de chaîne qui suit USR peut être une lettre allant de A à U ou de a à u, aucune différence n'étant faite entre les majuscules et les minuscules.

USR renvoie alors l'adresse de début d'une des 21 sections de la mémoire réservées aux graphiques utilisateur. Chaque section contient huit adresses auxquelles huit octets sont implantés à l'aide de POKE pour produire des graphiques utilisateur.

Structure

USR nombre
entier-constante-numérique
USR nombre
entier-variable-numérique
USR (nombre
entier-expression-numérique)
USR chaîne-constante
USR chaîne-variable

VAL VALeur *eg***Emplacement sur le clavier**
MODE ETENDU

Fonction

VAL convertit une chaîne accompagnée d'une valeur numérique en un nombre.

Comment utiliser VAL

VAL est suivie d'une constante ou d'une variable de chaîne, par exemple

70 LET x=VAL a\$

Les guillemets entourant la valeur de la constante ou de la variable de chaîne sont éliminés et la valeur doit donc être une valeur numérique. VAL l'évalue et la renvoie en tant que constante numérique.

Exemples

Si a\$ a la valeur "435", l'instruction ci-dessus affecte la valeur 435 à x. Toutefois, VAL peut également évaluer des expressions, par exemple

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

La valeur de chaîne qui est affectée à a\$ doit être une expression utilisant x, par exemple x*x. Une valeur numérique, 5 par exemple, est alors affectée à x. VAL élimine les guillemets de la valeur de chaîne pour produire x*x et évalue le résultat à l'aide de la valeur affectée x, 25 étant le résultat affiché.

Structure

VAL chaîne-constante
VAL chaîne-variable

VAL\$ VAL (chaîne) *eg***Emplacement sur le clavier**
MODE ETENDU
SYMBOL SHIFT J

Fonction

VAL\$ évalue une chaîne en tant qu'expression de chaîne.

Comment utiliser VAL\$

VAL\$ est suivie d'une variable de chaîne, par exemple

130 PRINT VAL\$ a\$

Les guillemets entourant la valeur de la variable de chaîne sont éliminés et la valeur doit donc être une expression de chaîne. VAL\$ évalue l'expression et renvoie la valeur en tant que constante de chaîne.

Exemple

Essayez ce programme

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

La valeur de chaîne affectée à a\$ doit être une expression utilisant x\$, par exemple "x\$+x\$". Une valeur de chaîne est alors affectée à x\$, par exemple "DO". VAL\$ supprime les guillemets entourant la valeur de a\$ pour produire x\$+x\$ et évalue le résultat à l'aide de la valeur affectée à x\$. DODO étant le résultat affiché.

Structure

VAL\$ chaîne-variable

VERIFY *86***Emplacement sur le clavier**
MODE ETENDU
SYMBOL SHIFT R

Commande/Instruction

VERIFY vérifie si un programme a été correctement stocké sur bande après SAVE.

Comment utiliser VERIFY

VERIFY est normalement utilisée comme une commande directe, comme le serait LOAD et est suivie d'un nom de programme, par exemple

VERIFY "nom de fichier"

Lorsque la bande a été mise en route, le nom de chaque programme rencontré est affiché et les programmes de la bande qui ont le même nom sont tous comparés au programme de la mémoire. Si deux programmes sont identiques, le message

⊗ OK, ⊗:1

apparaît. Notez que VERIFY™ compare le programme suivant de la bande à celui de la mémoire.

VERIFY CODE et VERIFY DATA

VERIFY CODE peut être utilisée exactement comme LOAD CODE pour vérifier si les informations d'une section de la mémoire ont été stockées sur bande. VERIFY DATA fonctionne comme LOAD DATA et vérifie si un tableau a été stocké sur bande. Pour plus de détails, voir LOAD CODE et LOAD DATA.

Structure

VERIFY chaîne-expression

VERIFY chaîne-expression **CODE**

[nombre entier-expression]

[nombre

entier-expression-numérique]

VERIFY chaîne-expression **DATA**

lettre [\$] ()

VERIFY CODE

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT R

MODE ETENDU

1

Voir VERIFY

VERIFY DATA

Emplacement sur le clavier

MODE ETENDU

SYMBOL SHIFT R

MODE ETENDU

D

Voir VERIFY

ETATS DE L'ECRAN SPECTRUM +

Quand le Spectrum arrête l'exécution du BASIC, un message apparaît au bas de l'écran. Ceci indique qu'une commande ou un programme a été complété ou qu'une erreur a été faite. Chaque message consiste en un nombre ou une lettre codés suivis d'un bref message puis du numéro de la ligne et de l'instruction à laquelle l'ordinateur s'est arrêté. Une commande est montrée comme une ligne Ø et à l'intérieur d'une ligne, l'instruction 1 est au début de la ligne, l'instruction 2 est après les premiers deux points (:) ou après THEN et ainsi de suite. CONTINUE amène normalement le programme à reprendre à l'instruction spécifiée dans le message.

Ø OK

Terminé avec succès ou tentative de saut à un numéro de ligne supérieur à ceux atteints dans le programme. CONTINUE ignore ce message et recommence à l'instruction spécifiée dans le message précédent.

1 NEXT without FOR

NEXT a été rencontré sans FOR correspondant et il y a une variable ayant le même nom que la variable de contrôle.

2 Variable not found

Une simple variable a été utilisée sans qu'une valeur lui ait été affectée ou que sa valeur ait été transférée de la bande. Une variable de contrôle a été utilisée avec NEXT sans avoir préalablement été constituée dans une instruction FOR. Une variable à indice a été utilisée avant que les dimensions du tableau n'aient été établies à l'aide de DIM ou que le tableau ait été transféré de la bande.

3 Subscript wrong

Une indice est supérieure aux dimensions du tableau.

4 Out of Memory

Il ne reste pas suffisamment d'espace mémoire pour terminer l'instruction ou la commande.

5 Out of screen

INPUT a généré plus de 23 lignes dans le bas de l'écran ou un nombre de lignes égal ou supérieur à 22 a été utilisé avec PRINT AT.

6 Number too big

L'ordinateur a essayé de produire un nombre supérieur à environ 1Ø.

7 RETURN without GOSUB

Il y a une instruction RETURN en plus que les instructions GOSUB.

8 End of file

Message de prise en charge du fichier du micro-support.

9 STOP statement

STOP a été utilisée pour arrêter le programme. CONTINUE reprendra à l'instruction suivante.

A Invalid argument

Un mauvais argument ou valeur ont été affectés à une fonction.

B Integer out of range

Une valeur a été arrondie au nombre entier le plus proche et est hors des limites acceptables.

C Nonsense in BASIC

Dans le contexte du programme, l'instruction n'a pas de sens en BASIC. Par exemple, une expression invalide a été utilisée avec VAL ou VAL\$.

D BREAK - CONT repeats

Vous avez appuyé sur BREAK. CONTINUE répétera l'instruction à laquelle l'ordinateur s'est arrêté.

E Out of DATA

READ a essayé de lire après la fin de l'instruction finale DATA du programme. RESTORE sera peut-être nécessaire.

F Invalid file name

SAVE a été utilisée avec un nom ne contenant aucun caractère ou en contenant plus de dix.

G No room for line

Il ne reste pas suffisamment d'espace mémoire pour entrer la nouvelle ligne du programme.

H STOP in INPUT

STOP a été entrée en réponse à INPUT ou se trouvait au début des données entrées. CONTINUE répète l'instruction INPUT.

I FOR without NEXT

Une boucle FOR NEXT n'a pas été exécutée parce que les limites ou la valeur de STEP étaient erronées (par exemple, FOR x=5 TO Ø sans STEP) et le NEXT correspondant n'a pas été trouvé.

J Invalid I/O device

Message de prise en charge du fichier sur micro-support.

K Invalid colour

La valeur spécifiée pour INK, PAPER, FLASH, BRIGHT, INVERSE ou OVER ou bien le caractère de contrôle correspondant est hors des limites.

L BREAK into program

Vous avez appuyé sur BREAK. Le message précise que la dernière instruction a été exécutée et CONTINUE reprend à l'instruction suivante.

M RAMTOP no good

La valeur affectée à RAMTOP est soit trop grande, soit trop petite.

N Statement lost

Un saut à une instruction qui n'existe plus a été tenté.

O Invalid stream

Message de prise en charge du fichier sur micro-support.

P FN without DEF

Une instruction FN a été utilisée sans l'instruction DEF correspondante.

Q Parameter error

Une instruction FN contient un mauvais nombre de valeurs à transmettre à la fonction ou bien une des valeurs n'est pas du type correct (une chaîne au lieu d'un nombre ou vice-versa).

R Tape loading error

La procédure de chargement, de classement ou de vérification a échoué.

AU-DELA DU BASIC

BASIC est un langage machine universel convenant très bien à la plupart des applications. Ce n'est toutefois pas le seul langage machine que vous pouvez utiliser avec le Spectrum. Il existe du logiciel vous permettant d'utiliser d'autres langages, par exemple le FORTH, le micro-PROLOG ou le LOGO. Ces langages ne travaillent pas de la même façon que le BASIC et ouvrent de nouvelles possibilités à votre ordinateur.

Le BASIC étant un langage universel, il peut être assez encombrant dans certaines applications et il est également relativement lent. D'autres langages offrent plus de souplesse d'emploi associée à une programmation simple et une exécution plus rapide. Le FORTH, par exemple, vous permet de définir vos propres mots et de les utiliser dans des instructions que l'ordinateur comprend et qu'il exécute presque dix fois plus vite que leurs commandes équivalentes en BASIC. Avec le micro-PROLOG, l'ordinateur comprend des phrases en anglais simple qu'il stocke dans sa mémoire et dont il se sert pour dialoguer avec l'utilisateur. LOGO est un langage machine mis au point pour les besoins de l'informatique éducative. Mais si vous voulez écrire des programmes que vous pourrez exécuter très rapidement sur votre Spectrum, il vous faut savoir comment programmer en code machine.

Le code machine

Le BASIC vous permet de donner des instructions à l'ordinateur dans une forme que vous comprendrez facilement. L'unité centrale du Spectrum — le puissant chip Z80A — ne comprend pas le BASIC, mais une section de la mémoire contient un programme permanent, appelé l'interprète BASIC qui convertit vos instructions BASIC en une séquence de signaux codés.

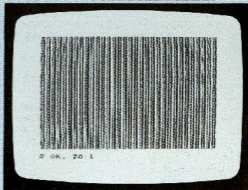
L'interprète met un certain temps pour traduire vos instructions BASIC dans les codes Z80A ou encore code machine. Toutefois, si vous le souhaitez, vous pouvez vous passer de l'interprète BASIC et envoyer le code machine directement au Z80A. Votre programme sera alors exécuté très rapidement. Le prix à payer est le temps mis à écrire le programme du code machine. A la différence du BASIC, ce n'est pas un langage machine "sympathique" en ce sens qu'il faut longtemps pour l'apprendre. La programmation du code machine est au delà des limites de ce guide, mais il existe de nombreux ouvrages traitant du code machine Spectrum de façon très approfondie. Ne serait-ce que pour vous faire une idée de la vitesse du code machine,

entrez le programme ci-dessous par le clavier et exécutez-le.

BANDES INSTANTANÉES

```

10 FOR X=0 TO 15
20 READ N: POKE 65000+X,N
30 NEXT X
40 DATA 33,255,63,1,1,24,22
50 DATA 55
60 DATA 35,11,120,177,200,114,
24,245
70 RANDOMIZE USR 65000
  
```



Essayez de remplacer 55 à la ligne 50 par n'importe quelle valeur entre 1 et 255 et vous verrez les bandes changer. Si vous utilisez une commande INK en premier lieu, vous produirez des bandes de couleur. Mais ce n'est pas là le but de l'exercice. L'important, c'est la rapidité avec laquelle l'image est produite à l'aide du code machine — pratiquement instantanément.

Ce programme peut être exécuté parce que les instructions DATA contiennent 16 codes stockés à l'adresse 65000 et suivantes de la mémoire par les lignes 10 à 30. La ligne 70 transmet les codes au Z80A et l'image est immédiatement produite.

Bon nombre des jeux destinés à votre Spectrum sont écrits en code machine précisément en raison de cette mise en activité ultra rapide. Pour vous aider à écrire le code machine, il existe des programmes appelés des assembleurs. Ceux-ci vous donnent des instructions que vous entrez par le clavier au lieu des nombres et c'est exactement ce dont le code machine a besoin. Ces instructions ne sont pas des mots anglais comme les mots-clés BASIC, mais des abréviations ou des symboles mnémoniques représentant les opérations que l'ordinateur doit effectuer. Avant d'utiliser le langage d'assemblage, vous devez donc comprendre le fonctionnement étape par étape de votre ordinateur.

LE JARGON INFORMATIQUE — SA SIGNIFICATION

Beaucoup des mots utilisés dans l'informatique sont utilisés dans la vie courante, mais avec une signification différente. Nous vous donnons ici l'explication de certains de ces mots qui apparaissent dans la brochure, accompagnés de certains termes spéciaux employés dans l'informatique. Les mots en italiques sont eux aussi expliqués. S'il y a un mot ou un terme de la brochure que vous ne comprenez pas et qui ne se trouve pas ici, consultez l'index.

Adresse Une unité de la *mémoire*. Il y a 65.536 adresses dans le ZX Spectrum +.

Argument Une *valeur* utilisée par une *fonction* pour produire un résultat.

Attributs Les codes donnant la couleur des *caractères*.

BASIC Le langage machine utilisé par le ZX Spectrum + et la plupart des autres micro-ordinateurs domestiques.

Bit Etat avec ou sans en *code binaire*. Abréviation de chiffre binaire.

Boucle La section d'un *programme* qui est répétée une ou plusieurs fois.

Byte Un ensemble de huit *parties* qui représente un nombre d'une valeur de 0 à 255. Chaque *adresse* dans la *mémoire* a une *byte*.

Caractère Toute lettre, chiffre (de 0 à 9), signe ou élément *graphique* pouvant être affiché ou imprimé.

Chaîne Un groupe d'un ou plusieurs *caractères* entre guillemets, les différenciant des nombres et des *variables numériques*.

Charger Introduire un *programme* ou des *données* dans l'ordinateur, provenant d'un dispositif de stockage comme une *cartouche* ou une cassette.

Code binaire Le genre de code utilisé par les ordinateurs. Il se compose de séries d'états avec ou sans — par exemple, avec-sans impulsions électriques.

Code machine Le langage compris par le ZX Spectrum +. Les programmes en BASIC sont traduits en code machine par l'ordinateur comme il les exécute.

Commande Une instruction exécutée par l'ordinateur ou une *commande directe*.

Commande directe Un jeu de une ou plusieurs instructions exécutées immédiatement après avoir été données à l'ordinateur.

Constante Un certain nombre de lettres ou un groupe de une ou plusieurs lettres ou n'importe quels autres *caractères*.

Curseur La position sur l'écran où quelque chose va être affiché. Elle peut être marquée par un signe clignotant indiquant le *mode* dans lequel l'ordinateur se trouve.

Défilement Le déplacement uniforme de *caractères* sur l'écran permettant d'afficher plus d'informations qu'un seul écran ne pourrait en contenir.

Données Les *informations* que l'ordinateur lit dans un *programme* ou qui sont introduites dans l'ordinateur pour produire des résultats.

Editer Changer les détails d'un *programme*.

Emboîtement La disposition de *boucles* à l'intérieur d'un programme, permettant d'exécuter une ou plusieurs boucles à l'intérieur d'une autre.

Enchaînement Une combinaison de chaînes qui s'ajoutent entre elles.

Ensemble Un groupe de *données* apparentées réunies dans une section de la *mémoire*.

Entrée Les *programmes* et les *données* introduits dans l'ordinateur.

Entrer Donner une instruction ou des informations complètes à l'ordinateur.

Équipement L'ordinateur lui-même et tous les autres dispositifs ou machines qui lui sont associés, par exemple les *périphériques*.

Etat Un message affiché par l'ordinateur pour rendre compte de ses actions.

Expression Une combinaison de *constantes*, de *variables* et de *mots-clés*.

Fonction Opération au cours de laquelle l'ordinateur prend une ou plusieurs *valeurs* (ou *arguments*) et les utilise pour produire un résultat qui constitue une autre valeur.

Graphiques La production d'images, comme des modèles, des tableaux ou des diagrammes, par l'ordinateur.

Imprimer Afficher les résultats ou les *graphiques* sur l'écran ou les imprimer sur une imprimante.

Informations Les mots, les nombres, les signes dans n'importe quelle combinaison que l'ordinateur doit traiter.

Instruction Soit un *mot-clé* utilisé pour constituer une instruction dans une *ligne* de programme ou l'instruction elle-même.

Interface Un élément reliant l'ordinateur et/ou les *périphériques* et qui assure qu'ils peuvent communiquer entre eux.

K Une mesure de la capacité de la *mémoire* d'un ordinateur. 1K est égal à un kilo-octet ou 1024 octets. La capacité de la mémoire en K est égale au nombre total d'*adresses* de la mémoire, chacune d'entre elles pouvant stocker un octet. Le ZX Spectrum + a une RAM de 48K et une ROM de 16K, donnant un total de 64K.

Ligne Une instruction ou une série d'instructions dans un programme. Elle est numérotée de façon à être exécutée au point correct dans une série d'autres lignes.

Listage Les lignes d'un programme listées par ordre.

Logique Le processus par lequel l'ordinateur décide si les résultats sont corrects ou faux ou si les états sont *vrais* ou *faux*.

Mémoire La partie de l'ordinateur où lorsque nécessaire, le *programme* et les *données* sont conservés, ainsi que les consignes d'exploitation permanentes.

Mode Dans le Spectrum, l'un des cinq états qui dictent les *mots-clés* et les *caractères* pouvant être produits par chacune des touches du clavier. Lors de la programmation, le mode est indiqué par une lettre clignotante à l'intérieur du *curseur*.

Mot-clé Une instruction machine en BASIC. Il se peut qu'il exige certaines *valeurs* pour fonctionner.

Octet Un jeu de huit *bits* représentant un nombre dont la valeur varie entre 0 et 255.

Opérateur Une instruction qui exécute de l'arithmétique ou de la *logique*.

Périphérique Tout dispositif connecté à l'ordinateur.

Pixel Le plus petit des points de couleur apparaissant sur l'écran. Abréviation de "picture cell" — élément d'image.

Programmer Une série d'instructions à exécuter par l'ordinateur.

RAM (mémoire à accès sélectif) La partie de la *mémoire* à laquelle on peut donner un *programme* ou des *données* et d'autres *valeurs* variables. Également connue sous le nom de mémoire volatile. Le contenu de la RAM est effacé lorsque l'ordinateur est débranché. Le ZX Spectrum + a une RAM de 48K.

Registre Une petite mémoire séparée de la *mémoire* principale. Les registres de l'unité centrale sont utilisés pour effectuer les calculs.

Résolution Le degré de détail qu'il est possible d'obtenir dans les *graphiques* machine.

ROM (mémoire inaltérable) La partie de la mémoire contenant les *programmes* et les instructions permanentes de l'ordinateur. Le ZX Spectrum + a une ROM de 16K.

Sauvegarder Stocker un *programme* ou des *données* dans un dispositif de stockage comme une *cartouche* ou cassette.

Sortie Les résultats de l'ordinateur

Syntaxe La séquence correcte de *mots-clés*, de *constantes*, de *variables* et d'*expressions* indispensables pour constituer une instruction BASIC valide.

Unité centrale La partie centrale de l'ordinateur qui effectue les calculs et commande les autres unités. Le ZX Spectrum + utilise un microprocesseur Z80.

Valeur Tout nombre ou chaîne qui peuvent être donné ou représenté par une constante, une variable ou une expression

Variable Une ou plusieurs éléments de la *mémoire* contenant une *constante* particulière qui sera utilisée par l'ordinateur. Un nom ou une lettre est attribuée à chaque variable pour pouvoir les identifier facilement. Le ZX Spectrum + fait la différence entre les *variables numériques* et les *variables de chaîne*.

Variable de chaîne Une *variable* contenant une *chaîne*. Les variables de chaîne se composent toujours d'une lettre unique et du signe \$.

Variable numérique Une variable contenant un nombre. Les variables numériques se composent d'une ou plusieurs lettres.

Vrai Tout état ou résultat dont l'ordinateur décide est *vrai* ou *correcte*. Vrai a une valeur numérique de 1.

Jeu de caractères Le jeu complet de *caractères* et certains caractères de commande utilisés par l'ordinateur.

INDEX

- Alimentation électrique 4, 5, 43
- Amplification du son 37
- Animation 34-5
- ATTR 35
- Bandes 12, 44
 - Entretien 12
 - Étiquetage 14
 - Son 12
 - Stockage 12
- Bandes magnétiques 12, 44, 45
 - Entretien 12
 - Étiquetage 14, 39
 - Son 12
 - Stockage 12
- Barre d'espace 19
- BASIC 18, 49-73
- BEEP 18, 36
- BIN 33
- Boucles 26-7, 30
- Bouton "reset" (remise zero) 5, 11, 12
- BREAK 19
- BRIGHT 31
- Calculs 22-3
- CAPS LOCK 18, 21
- CAPS SHIFT 8, 21, 18
- Caractères, création 32-3
 - Sélection 20
- Caractères utilisateur 32, 3, 80
- Cartouches Microdrive 12, 46
 - ROM 12, 46-47
- Cartouches ROM 12, 46-47
- Chaînes 22
- Chargement 13, 14, 15, 16
- Chips 42-3
- Chips logiques 43
- CIRCLE 28
- Clavier 18-19
 - Caractères graphiques 26
 - Modes 20-1
- Code binaire 44
- Code machine 75
- Couleur TV 42
- Collisions 34-5
- Commandes de curseur 19
- Connecteur latéral 5, 43, 47
- Connexions 5
 - Magnétophone à cassette 13
 - Mise sous tension 5
 - Télévision 4
- Couleurs 24-5
 - Codes 24
 - Codes de contrôle 33
 - Mélanges 25
 - Touches d'affichage 19
 - Mixtes 32
 - Essais 6, 24
- Couleur de la bordure 6, 24-5
- Couleur de l'encre 24-5
- Couleur du papier 24-5
- DATA 33
- Défilement 8
- DELETE 10
- Deux points 23, 51
- Diagrammes de Gantt 25
- DRAW 28-9
- EDIT 18, 21
 - Lignes de programme 21
- Effets aléatoires 30
- Effets sonores 36-7
- Embases 5
- Embase et câble d'antenne 4-5
- Embase EAR 5, 13, 37
- Embase MIC 5, 13, 37
- ENTER 9, 10, 11, 19
- Entrée de programmes 8-9
- Erreurs corrections 10, 21
 - Messages sur écran 74
- Exécution de programmes 8-9
- EXTEND MODE 8, 18, 21
- FLASH 31
- Flexible 46
- Formes, remplissage 29
- FOR NEXT 26-7, 29, 30, 31, 34
- FORTH 75
- Fonctions 50
- GOTO 23
- GRAPH 18, 21, 26
- Graphiques animation 34-5 et
 - Couleur 24-5
 - Création de caractères 32-3
 - Effets aléatoires 30
 - Haute résolution 26, 28-9
 - Basse résolution 26, 27
 - Motifs 30-1
 - Remplissage de formes 29
- Graphiques à haute résolution 26, 28-9
- Graphiques à basse résolution 26-7
- Grille, haute résolution 28, 80
 - basse résolution 26, 80
- Guillemet 23, 51
- Haut-parleur 43
- IF THEN 29
- Images, dessin 30-1
 - basse résolution 26-7
- Imprimantes 45, 47
- INPUT 23, 29
- Instructions 22, 50
- Interfaces 45, 46-7
- Interface 1.2X 45, 46-7
- Interfaces RS 232 45, 47
- Introduction par le clavier 8, 9
- INVERSE 31
- INV VIDEO 18
- Jeu de caractères 51
- LET 23
- Leviers de commande 45, 47
- Lignes 8
 - Suppression 21
 - Édition 21
- Lignes de programme, suppression 21
 - Édition 21
- LIST 21
- Listages 8, 21
- LOAD 14-16
- Logiciel 12
 - Chargement 14-16
- Prêt à être exécuté 12-13
- Convenances 12
- Types 12
- LOGO 75
- Machines, définition 12
- Magnétophone à cassette, bouton de réglage du volume 12, 14, 15
- Magnétophone à cassette, bouton de réglage de la tonalité 12, 14, 15
- Magnétophone à cassette servant d'amplificateur 37
 - Boutons de réglage de la tonalité 14, 15, 16
 - Boutons de réglage du volume 14, 15, 16
 - Chargement de programmes 14-16
 - Choix 12
 - Compteurs 14
 - Connexions 5, 13
 - Sauvegarde de programmes 38-40
- Mémoire 12, 43, 44-8
- Messages sur écran 74
- Microprocesseur Z80 43, 45, 75
- MICRO-PROLOG 75
- Microdrive 5, 46
- Cartouches 12, 45
 - Chargement 46
- Modes 20-1
- Mode étendu 20, 21
- Mode graphique 20, 21
- Mode lettre 20, 21
- Mode majuscules 20, 21
- Mode mot-clé 20
- Modems 46
- Mots-clés 9, 18, 19, 20-21, 50, 52, 73
 - Sélection 19, 20
- Modifications de programmes 9
- Musique 36-7
- NEW 11, 12, 18
- Nombres 50
- Nouveaux programmes 11
- Opérateurs arithmétiques 22
- Paquets RAM 4
- Parasites 4
- Parcours d'entrée-sortie 45
- Parenthèses 23
- Périphériques 45, 46-7
- Pixels 28
- PLOT 28
- Point 23, 51
- Point-virgule 23, 51
- POKE 48
- PRINT 22
- Prise 9VDC 5, 43
- Programmation 17-40
- Programme "arc-en-ciel" 26-7
- Programme "Balle en train de rebondir" 35
- Programme "Bannière étoilée" 29
- Programme "Carnet de croquis" 11
- Programme "Carrés" 30
- Programme "Cercles clignotants" 9
- Programme de relance 10
- Programme "Échiquier" 33
- Programme "Étoile" 28
- Programme "Formes" 9
- Programme "Formes symétriques" 30
- Programme "Mosaïque farfelue" 10
- Programme, modification 9
 - Chargement 12, 13, 14-15
- Commencer un nouveau programme 11
 - Correction d'erreurs 10
 - Entrée 8-9, 44
 - Exécution 8-9, 44
 - Relance 10
 - Sauvegarde 13, 38-40
 - Vérification 39
- Programme "Noms" 8
- Programme "Polyhédres" 10
- Programme "Pyramides" 31

Programmes "Reflets du soleil" 11
 Programme Robot ZX 27
 Programme "Table de multiplication" 23
 RAM (Mémoire à accès sélectif) 42, 45, 48
 RAM 16K ZX 4
 RAMTOP 48
 READ 33
 Réglage du téléviseur 6
 Régulateur de tension 43
 REM 39
 RND 26, 30
 ROM (mémoire inaltérable) 43, 45, 48
 SAVE 38-9
 Sauvegarde 13, 38-40
 Signes, calculs 22, 50
 sélection 19
 Signes de ponctuation 23, 51
 Sinclair BASIC 49-73
 Sous-programme 30-1
 STEP 29
 Stockage 44, 45
 SYMBOL SHIFT 8, 19, 21
 Symboles, sélection 20

Tableau logique non affecté 42
 Télévision, connexion 5
 convenance 4
 réglage 6
 Topographie mémoire 48
 Touches 18-19
 Manoeuvre 20-1
 Touches numériques 19
 Ton, musical 36
 TRUE VIDEO 18
 Unite Centrale (UC) 43, 44, 45, 48, 75
 Variables 22-3, 50
 Variable système 48
 Virgule 23, 51

First published 1984 by Dorling Kindersley Ltd, 9 Henrietta Street, London WC2E 8PS in association with Sinclair Research Ltd, 25 Willis Road, Cambridge

Copyright © 1984 by Sinclair Research Ltd and Dorling Kindersley Ltd, London
 Illustrations copyright © 1984 by Dorling Kindersley Ltd, London

sinclair ZX Spectrum +, ZX Microdrive and ZX Interface are Trade Marks of Sinclair Research Limited

Tous droits réservés. Aucune partie de cet ouvrage ne peut être reproduite, stockée dans un système d'extraction ou transmise sous une forme quelconque ou par un quelconque moyen, électronique, mécanique, photocopie, enregistrement ou autre, sans l'autorisation écrite préalable des propriétaires du copyright.

Rédacteur David Burnie
Rédacteur artistique Peter Luff
Dessinatrice Debra Lee
Photographie Trevor Melton
Photos des écrans Vincent Oliver
Rédacteur gérant Alan Buckingham

Composition par Express Typesetters Ltd, Aldershot, Angleterre.
 Reproduction par A. Mondadori, Vérone.
 Imprimé et relié en Italie par A. Mondadori, Vérone.

t

5
6
0
2
4
6
8

coordonnées verticales

LE LOGICIEL DU SPECTRUM

La gamme complète de logiciel destinée aux ordinateurs Spectrum (y compris tous les titres existants) est entièrement compatible avec votre nouveau ZX Spectrum +.

DORLING KINDERSLEY LTD.
en association avec
SINCLAIR RESEARCH LTD.