

# **GRAPHO**

par Dan Beysson

**Manuel d'utilisation**

Editeur C.T.S. France

## INTRODUCTION A GRAPHO

Ce manuel vous présente l'utilisation de Grapho. Il est en fait composé de 2 parties correspondant aux deux produits qui composent Grapho. La première partie concerne l'éditeur graphique avec lequel vous allez produire vos dessins et planches d'animation. La seconde partie concerne les R.S.X de Grapho qui vous permettront d'animer vos dessins à partir de programmes très simplifiés que vous aurez écrits en langage BASIC. Un R.S.X est une instruction supplémentaire incorporée au BASIC et dont le traitement est confié à un sous programme écrit en langage machine (assembleur) par le créateur du R.S.X. Votre BASIC sera donc enrichi par Grapho par plus de 40 instructions supplémentaires. Grapho est destiné à fonctionner sur l'AMSTRAD CPC 6128 sans aucune extension. Dans l'introduction qui suit, les titres des chapitres sont précédés de petits symboles: = ††, ou X. Ces symboles représentent le degré de difficulté pour aborder le chapitre (un peu à l'image des balises des pistes de ski.)

**Piste verte:** = signifie que le chapitre est à l'usage de ceux d'entre vous qui ne sont pas des informaticiens distingués (mais qui le deviendront...)

**Piste rouge:** †† plus technique, les termes usuels de l'informatique sont employés

**Croisement:** X intérêt général.

**Note:**

Dans ce document il est fait référence à des couleurs, si vous possédez un moniteur vidéo monochrome, les couleurs seront remplacées par des dégradés de gris.

**Terminologie employée**

Commençons tout d'abord par définir la terminologie employée dans ce document afin d'éviter les répétitions dans le texte. Celle qui nous intéresse en premier chef concerne la terminologie que nous utiliserons pour le graphisme et l'animation.

**Un motif:** un motif est un dessin isolé sur votre écran qui normalement à une durée de visualisation limitée. C'est à dire qu'un motif est destiné à évoluer dans un décor ou à être remplacé par un autre motif.

L'éditeur de Grapho vous permet outre la création du motif, de l'enregistrer sous un numéro. Ce même numéro sera utilisé par la suite dans le R.S.X pour le faire évoluer. Un motif peut appartenir à une séquence.

**Une séquence:** Une séquence est constituée de plusieurs motifs qui décomposent un mouvement quelconque: C'est la technique du dessin animé. Le nombre de motifs à définir dans une séquence pour donner l'illusion d'un mouvement est directement dépendant de l'effet désiré. A titre d'exemple, avec trois motifs dans une séquence on obtient une illusion de mouvement généralement adaptée aux jeux. Le nombre maximum de motifs dans une séquence est de treize. L'illusion du mouvement est également tributaire (dans notre contexte) d'une part de la dimension des motifs et d'autre part de la distance, qui sépare sur votre écran, un motif, du motif précédant. La dimension et la distance s'expriment en nombre de points écran. Plus la distance entre deux motifs sera grande, plus votre mouvement paraîtra rapide et moins il sera réaliste. Quant à la dimension d'un motif, elle influence directement le temps nécessaire à son traitement par le R.S.X. Plus un motif est important, plus il demandera de temps pour être géré et moins le mouvement sera rapide. A titre indicatif Grapho s'accommode bien de motifs d'environ 400 points.

(ex: 40 points de haut sur 10 points de large) avec un déplacement de 1 à 2 points écran à chaque fois, ceci nous amène naturellement à définir ce qui compose un motif sur votre écran.

**Pinceaux:** Les pinceaux de Grapho vous permettent de dessiner en couleurs (ou en dégradés de gris) sur l'écran. Il y en a 15 utilisables dans l'éditeur numérotés de 0 à 14. Pour dessiner il faut théoriquement tracer des suites de points dans les couleurs désirées en sélectionnant des pinceaux. Chaque pinceau contient la couleur (il y en a 27 possibles) que vous lui aurez affectée. Si vous changez la couleur d'un pinceau après l'avoir déjà utilisé, tous les points déjà tracés avec le pinceau prendront instantanément la nouvelle couleur. La notion de pinceau est très importante dans notre cas. En effet, dans vos animations, la présence d'un point sur une position donnée de votre écran s'exprimera par la présence ou l'absence d'un numéro de pinceau qui normalement définit ce point.

**Exemple:** Supposons que vous vouliez faire avancer un personnage jusqu'à ce qu'il rencontre un mur de briques. Votre personnage avançant de 1 point à chaque fois dans votre programme, il vous faut avant de commander son déplacement, demander si, à la position que votre personnage devra occuper il y a présence ou absence d'un pinceau qui constitue le mur de briques. Si le mur de briques a été dessiné avec les pinceaux 13 et 14 vous détecterez sa présence sur les valeurs 13 ou 14 et l'absence sur les valeurs non (13 et 14).

Autrement dit: Un point sur votre écran est un numéro de pinceau, c'est réellement l'information qui est écrite à l'emplacement du point dans la mémoire de l'écran ( ne l'oubliez pas) A toute fin utile, le terme pinceau

est appelé stylo dans le manuel d'utilisation de votre AMSTRAD.

### = Couleur de fond.

Lorsque vous dessinez sur votre feuille de papier cette feuille est initialement d'une couleur donnée (blanche, bleue etc...) et vous utilisez un crayon dont la couleur est évidemment différente de celle du papier. Sur votre écran c'est le même principe. La couleur initiale de votre écran est appelée couleur de fond. En fait, dans votre mémoire écran, le fond est également un numéro de pinceau auquel on a donné une couleur. Alors que peut-on mettre d'autre que des numéros de pinceaux dans la mémoire écran?

Réponse: Rien d'autre, chaque point est un numéro de pinceau et ce que l'on appelle le fond est un numéro de pinceau également.

### = Comment ce système fonctionne-t'il?

Succinctement, vous savez certainement que l'affichage sur un tube cathodique est le déplacement (balayage) d'un spot électronique dont l'intensité est variable en fonction de l'effet de visualisation désiré. Dans votre AMSTRAD ce spot est contrôlé par un système électronique (VGA+CRTC) qui synchronise le spot sur le tube avec le balayage de votre mémoire d'écran en réglant l'intensité du spot en fonction de la couleur mémorisée par le numéro du pinceau sur lequel est le balayage. Dans vos programmes BASIC vous mémorisez la couleur d'un pinceau par l'instruction INK numéro de pinceau, numéro de couleur et vous écrivez un numéro de pinceau par l'instruction PLOT (par exemple).

### × Localisation d'un motif dans Grapho

Pour dire à Grapho qu'à tel endroit il y a un motif il faut tout d'abord localiser l'endroit sur l'écran (écran où mémoire d'écran sont équivalents dans ce qui suit).

L'éditeur de Grapho vous fournit un cadre rectangulaire dont vous pouvez changer les dimensions, il vous faut ensuite déplacer ce cadre pour y inscrire le motif désiré. L'endroit du motif étant alors localisé par le cadre, l'ensemble de la surface incluse dans le cadre est considérée par Grapho comme étant un motif.

De quoi est composée la surface du motif encadré? de numéros de pinceaux qui dessinent le motif mais il se peut également que le numéro de pinceau de la couleur du fond de l'écran fasse partie de cette surface.

Exemple d'un motif représentant un triangle.

Cadre définissant  
le motif - Triangle -



N° pinceau pour la couleur du fond  
N° pinceau pour la couleur du triangle

Lorsque vous souhaiterez dans Grapho soit afficher ce motif, soit le déplacer, vous aurez 3 possibilités en sachant que Grapho ne connaît (donc n'affiche) que le contenu de cadres rectangulaires.

- Possibilité 1: afficher le cadre rectangulaire (tous les numéros des pincaux contenus dans le cadre)

- Possibilité 2: n'afficher que les pincaux qui ne sont pas du numéro de pincau du fond.

- Possibilité 3: n'afficher que les pincaux qui ne sont pas des numéros pincaux du fond à condition qu'à l'endroit où le pincau courant doit être affiché il y ait déjà un pincau de la couleur du fond. En considérant que l'écran contient déjà un dessin avant l'affichage du motif (décor) ces trois possibilités vous permettent d'obtenir les effets suivants:

- Possibilité 1: superposition d'un motif et de la couleur de son fond sur un décor.

- Possibilité 2: incrustation du motif uniquement devant le décor.

- Possibilité 3: incrustation du motif uniquement derrière le décor.

Pour obtenir ces effets, les R.S.X de Grapho jouent sur le pincau du fond du décor et du motif qui peuvent être différents. Dans Grapho vous devez définir le numéro de pincau du fond d'un motif ainsi que le numéro du pincau du fond d'un décor (ils peuvent ne pas être identiques)

Dans votre AMSTRAD le numéro de pincau du fond qui est pris par défaut est le numéro zéro auquel l'AMSTRAD a donné la couleur bleue (numéro1) ce qui est fait au moment de l'initialisation ou réinitialisation de votre ordinateur. Du même coup nous venons de définir: MOTIF - DECOR - AVANT-PLAN - ARRIERE-PLAN et vous verrez qu'à l'usage cela paraît plus évident qu'à décrire.

#### = Petit conseil (Grand service)

Avant de vous lancer dans la réalisation d'une animation écrivez au moins succinctement votre scénario, dessinez sur une feuille de papier ce que seront les décors et les motifs.

Marquez les numéros des pincaux que vous utiliserez (les couleurs ont peu d'importance à ce moment là)

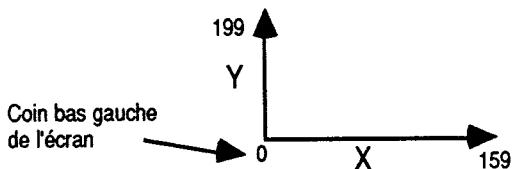
#### X Numéros des couleurs

Les numéros des couleurs que vous aurez à donner sont ceux qui figurent dans votre manuel AMSTRAD et sur le boîtier de votre lecteur de disquette (de 0 à 26).

## X Nombre de points et coordonnées d'un point

Votre écran peut-être considéré comme une matrice de points. Cette matrice est constituée de 200 lignes (horizontalement) dont la première de numéros zéro est en bas de votre écran (la dernière en haut de l'écran à l'adresse 199) et de 160 colonnes (verticalement) numérotées de 0 (à gauche sur votre écran) à 159.

Il existe donc  $200 \times 160 = 32000$  points accessibles par Grapho. Chaque point a l'adresse de ses coordonnées (X et Y)



Exemple

le point le plus en bas à gauche:

Y=0

X=0

le plus en haut à gauche :

Y=199

X=0

le plus en bas à droite :

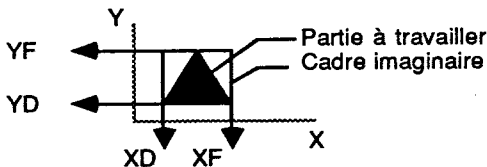
Y=0

X=159

## X : Coordonnées d'un MOTIF dans Grapho

Comme nous l'avons vu un motif peut-être identifié par un cadre dans lequel il est inscrit. Vous pouvez par la suite, donner un numéro à ce cadre et rappeler le motif par son numéro. Cependant les R.S.X de Grapho ne travaillent pas toujours sur des motifs catalogués. Lorsque vous souhaitez travailler sur une portion d'écran qui n'est pas un motif, il faudra indiquer ses coordonnées rectangulaires.

Exemple:



Coordonnées rectangulaires à fournir pour définir la partie à travailler.

Y D = coordonnée Y de début (bas du cadre imaginaire)

Y F = coordonnée Y de fin (haut du cadre imaginaire)

X D = coordonnée X de début (gche du cadre imaginaire)

X F = coordonnée X de fin (droite du cadre imaginaire)

où  $Y F \gg Y D$  et  $X F \gg X D$

Ces notations Y D, Y F, X D, X F seront utilisées par la suite

Exemple: un motif qui se déplace horizontalement du bord gauche de votre écran au bord droite a ses coordonnées x qui varient de 0 à 159 à chaque déplacement alors que ses coordonnées y sont invariables.

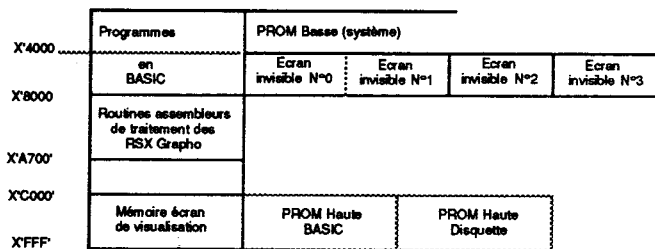
### ⇓ Points, pixels, adresses mémoire

En mode 0 dans lequel travaille GRAPHO, un point est constitué de 4 pixels horizontalement et de 2 verticalement. Comme il y a 16 numéros de pincesaux définissables (0 à 15) la mémoire d'écran serait de 32000 points divisée par deux soit 16000 octets (2 pincesaux possibles par octet) en fait cette mémoire est découpée en 8 blocs de 2 Koctets, la taille réelle est donc de 8x2k soit 16384 octets (les 48 derniers octets de chaque bloc ne sont pas utilisés)

L'écran qui est visualisé par Grapho a sa mémoire qui débute à l'adresse hexadécimale X'C000' jusqu'à X'FFFF'. Les 4 autres bancs de 16k de votre 6128 peuvent recevoir les images par l'intermédiaire de RSX qui assurent la commutation des bancs de mémoire.

Ces 4 mémoires supplémentaires pourront servir à recevoir des décors et (ou) des planches de motifs, elles seront appelées par la suite « écrans invisibles »

### Structure mémoire du système avec Grapho



Il faut environ 50 millisecondes au RSX pour transférer un écran invisible vers l'écran visible, ce qui autorise environ 20 transferts par seconde.

### = : Ecran visible / écran invisible

La mémoire de votre CPC 6128 fait 128 kilo octets compte tenu des possibilités d'adressage du microprocesseur Z 80, ce dernier ne peut accéder qu'à un maximum de 64 k. (K. signifiant kilo) octets. Dans la première partie de la mémoire nous trouverons: vos programmes écrits en BASIC, l'éditeur de Grapho, les programmes assembleurs de gestion des RSX de grapho, ainsi que la mémoire de votre écran qui sera appelée «écran visible» par la suite

Les 64 K.octets supplémentaires de votre CPC 6128 sont utilisés par Grapho pour y stocker des écrans complets destinés à supporter motifs et décors.

Les RSX de Grapho assurent la gestion de cette mémoire supplémentaire. Il est donc possible de stocker l'équivalent de 4 écrans dans la mémoire supplémentaire. Ces 4 blocs supplémentaires seront appelés par la suite «ECRANS INVISIBLES» et seront numérotés de 0 à 3.

Nous allons maintenant prendre un exemple simple d'animation et expliquer les étapes que vous aurez à programmer.

**SCENARIO:** Un personnage animé par une séquence de 3 motifs doit se déplacer du bord gauche de votre écran vers le bord droite. le personnage évoluera dans un décor.

1<sup>ere</sup> étape: sous l'éditeur de grapho.

- Créer les 3 motifs du personnage et les cataloguer
- Ecrire le fichier des 3 motifs catalogués sur la disquette
- Créer le fichier décor
- Ecrire le fichier décor sur la disquette
- Quitter l'éditeur de Grapho.

2<sup>eme</sup> étape: écrire votre programme d'animation en BASIC. Ce programme devra se servir des RSX de Grapho et sera construit comme suit

- Charger le fichier des motifs dans par exemple l'écran invisible numéro zéro (2 RSX à exécuter)
- Charger le fichier du décor dans par exemple l'écran invisible numéro 1 en demandant également à ce que les pinceaux du fichier Décor soient chargés (2 RSX à exécuter)
- Dire à Grapho quel est l'écran de Motif et quel est le numéro de pinceau du fond des motifs (1 seul RSX à exécuter)
- Faire la même chose pour dire à Grapho où est le fichier Décor
- Transférer le fichier Décor vers l'écran visible (1 seul RSX à exécuter)
- Définir la séquence (1 seul RSX à exécuter)
- Commander le déplacement de la séquence (1 seul RSX à exécuter pour le déplacement total)

Vous voyez la simplicité du programme BASIC qui devrait théoriquement ne pas dépasser les cinq lignes de programme. Nous touchons là à l'objectif même de Grapho qui est de prendre en charge toutes les difficultés techniques (et elles sont nombreuses) pour laisser libre cours à votre imagination artistique qui est sans fin n'en doutons pas.



## 5 : Liste des termes que nous venons de découvrir

- éditeur grapho
- RSX
- motif
- séquence
- pinceaux, couleurs, numéros de pinceaux
- localisation d'un motif
- superposition
- incrustation en avant plan
- numéros des couleurs
- couleur du fond
- incrustation en arrière plan
- décor
- coordonnées (YD YF XD XF) d'un motif
- structure mémoire
- point pixels
- écrans visibles
- écrans invisibles
- exemple d'étapes d'animation

## 1<sup>ere</sup> PARTIE

# MANUEL D'UTILISATION DE L'EDITEUR DE GRAPHO

### 1. Introduction

Puisqu'il faut bien commencer par dessiner avant de pouvoir animer, regardons en détail les possibilités de l'éditeur. Pour mieux assimiler le fonctionnement de ce programme, nous vous conseillons de vous placer devant votre AMSTRAD préféré et d'exécuter les commandes au fur et à mesure de votre lecture.

Mais avant tout un peu d'organisation.

#### Organisation de la disquette que vous venez d'acquérir

La disquette est composée de deux faces

**Face 1:** Cette face contient l'éditeur de Grapho et les programmes nécessaires à son exécution. L'Editeur est protégé et ne peut être recopié sur une autre disquette. Cette face ne vous servira qu'à charger l'éditeur dans votre ordinateur. Une fois chargé, l'éditeur vous demandera de mettre à la place votre disquette de travail (ne pas essayer de créer des fichiers sur cette face)

**Face 2:** Elle contient le nécessaire de base pour constituer une disquette de travail. Nous vous conseillons de recopier cette face sur une de vos disquettes et de conserver l'originale comme référence de base.

Voici les fichiers qui vous seront nécessaires sur votre disquette de travail.

Nom du fichier	Utilisation
AIDE. GRA	Utilisé par l'éditeur pour la commande H.
MMOO. GRA	Utilisé par l'éditeur en début d'exécution et par la commande DEL+Q (lettre L). Ce sont les motifs de base que l'éditeur vous propose
RPIXEL. OBJ	Fichier des RSX en langage machine. A utiliser par vos programmes d'animations et par "SIMUL"
SIMUL. BAS	Générateur d'animation.

Les fichiers suivants ne sont pas nécessaires mais peuvent vous être utiles.

SPI2. BIN	C'est une planche de motifs que vous pouvez utiliser.
DEC. BIN	Décor du programme "DEMOCTS"

#### Note sur le programme SIMUL:

Ce programme est largement interactif et vous posera les questions nécessaires. Sachez toutefois qu'il vous permet de charger une planche de motifs de constituer jusqu'à 8 séquences d'animations (8 sens) et de les faire évoluer dans 1 ou 2 fichiers de décor.

Au début vous pouvez utiliser SPI2. BIN comme planches de motifs (pinceau du fond = 0) et DEC. BIN comme fichier de décor (pinceau du fond = 11). Les pinceaux sont à prendre dans "SPI2" (SPI2. PIN)

Pour lancer ce programme: RUN "SIMUL"

SIMUL vous sera utile pour tester l'effet de vos animations et pour prendre vos "mesures" écran avant de programmer vos scénarios.

#### Exécution de l'éditeur

- 1 Placez la face 1 dans votre lecteur de disquette.
- 2 Entrez : RUN "GRAPHO" (RETURN)
- 3 Après chargement une animation de bienvenue s'exécute. Vous pouvez l'écouter en appuyant sur la barre ESPACE lorsque le personnage commence à avancer.
- 4 Suite à l'animation, la deuxième partie de l'éditeur se charge en mémoire
- 5 Une fois chargée, l'éditeur vous demande de placer votre disquette de travail dans le lecteur et de faire "RETURN" ensuite
- 6 Le message «GRAPHO PRET» apparait. Appuyez sur une touche pour l'arrêter.

- 7 Grapho attend vos commandes

Pour quitter l'éditeur, appuyer deux fois sur la touche ESC et répondez au message

Avant d'entrer dans le vif du sujet, arrêtons nous quelques instants sur le clavier de VOTRE AMSTRAD

## = Le clavier du CPC 6128

Il existe deux types de clavier sur AMSTRAD 6128 l'un se nomme AZERTY parce que les premières touches alphabétiques commencent par les 6 lettres A,Z,E,R,T,Y, l'autre se nomme QUERTY pour les mêmes raisons.

Les claviers AZERTY sont adaptés à la langue française alors que les QWERTY à la langue anglaise, Grapho fonctionne avec les deux types de claviers. A l'utilisation la différence entre les deux types de claviers réside dans le fait que certaines touches sont en "SHIFT" dans un cas et non dans l'autre. A ce propos, dans l'éditeur vous aurez certaines touches à commander en "SHIFT" ou en "CONTROLE"

Voici le mode d'emploi

- Lorsqu'une touche doit être en "SHIFT" ou "CONTROLE" vous devez appuyer simultanément sur la touche "SHIFT" ou "CONTROLE" et la touche désignée.

- Un autre point sur le clavier: le pavé de touches où se trouvent les 4 flèches (à droite sur le clavier) s'appelle le «clavier des numériques réduits» l'autre partie est le clavier principal.

Le symbol I devant les RSX correspond à la touche @ en shift

## 2 LES COMMANDES DE GRAPHO

Il y a dans Grapho deux types de commandes,

1 les commandes élémentaires qui s'initialisent en n'appuyant que sur la touche concernée

2 Les macro-commandes pour les lesquelles vous devez appuyer sur la touche DEL avant d'appuyer sur la commande concernée.

Dans les deux cas lorsqu'une commande est terminée vous pouvez l'annuler en appuyant sur la touche CLR pour retrouver l'écran dans l'état qu'il avait avant l'exécution de la dernière commande. Cette facilité vous permet de tester une commande et de l'annuler si vous n'en êtes pas satisfait. Commençons par expliquer les commandes élémentaires

Pour annuler une commande avant son exécution, appuyez sur la touche DEL

## 2 -1 Commandes élémentaires

### LES PINCEAUX (sélection) TOUCHES NUMERIQUES

Vous devez être familiarisé avec la notion de pinceau. Pour connaître la couleur que contient un pinceau, appuyez sur le chiffre du clavier principal correspondant au numéro du pinceau désiré (touches de 0 à 9) et pour les pinceaux 10,11,12,13, et 14, appuyez sur les touches 0,1,2,3,4, du clavier des numériques réduits. La réponse de Grapho est un rectangle de la couleur du pinceau qui s'affiche temporairement en bas et à gauche de votre écran. Le numéro de la couleur "C" est également indiqué dans ce rectangle

**IMPORTANT** Suite à cette commande, le pinceau que vous avez sélectionné devient le pinceau courant pour les commandes à venir

### Rappel du pinceau courant **TOUCHE ESPACE**

Pour connaître le pinceau sélectionné (si vous l'avez oublié) appuyez sur la touche "ESPACE" la couleur du pinceau apparaît temporairement avec le numéro de pinceau courant (P=n). Cette touche fait également apparaître un rectangle qui clignote noir et blanc. Ce rectangle est le cadre de Grapho. Ce cadre peut être déformé, déplacé, agrandi, ou diminué par des commandes spécifiques. Il vous permet de travailler par surface. Vous vous en servez pour le remplir par des couleurs ou par des motifs. A la première commande le cadre se retrouve matérialisé par quatre points clignotants. La vitesse de clignotement du cadre peut être changée par la macro-commande «Paramètres de Grapho»

### SELECTION du DRIVE: touches **CONTROLE A** ou **B**

Cette commande permet de choisir le lecteur de disquette pour vos fichiers.

**CONTROLE A** sélectionne le lecteur de votre clavier alors que **CONTROLE B** sélectionne votre deuxième lecteur éventuel.

### MODE CADRE: Touche **C**

Cette commande initialise le mode de déplacement du cadre. Dans ce mode l'image de l'écran reste fixe et le cadre peut être déplacé au moyen des 4 flèches du clavier (Ce mode est pris par défaut en début de l'exécution de l'éditeur)

### MODE ECRAN: Touche **E**

Dans ce mode lorsque vous utilisez les flèches du clavier le cadre reste en position fixe et l'écran se déplace dans le sens des flèches.

### AGRANDIR ou DIMINUER le cadre par ses 4 côtés: Touches **Z** et **Controle Z**

Les 4 côtés du cadre se trouvent agrandis par la commande **Z** et diminués par la commande **CONTROLE Z**

## **AGRANDIR ou DIMINUER le cadre par un de ses côtés**

En utilisant les flèches en "SHIFT " vous agrandirez le côté du cadre correspondant à la flèche, en utilisant les flèches en "CONTROLE" vous diminuerez les côtés du cadre correspondant à la flèche

## **FORME INITIALE du cadre: Touche Q**

Cette commande ramène le cadre à la position et à la forme qu'il avait au moment où vous avez accédé à Grapho, la surface retrouvée est de 8x8 points

## **REMPLISSAGE par un pinceau, une alternance de pinceaux où un motif 8x8 Touche TAB**

Le contenu du cadre ou d'un arc de cercle peuvent être remplis par:

- Soit un motif 8x8 si un motif a été sélectionné (voir plus loin)
- Soit une alternance de pinceaux si cette commande a été sélectionnée (voir après)
- Par défaut, par le pinceau qui est sélectionné

## **ALTERNANCE de pinceaux: Touche A**

Cette commande vous permet d'effectuer un remplissage avec deux pinceaux. Le pinceau sélectionné plus un autre que vous devez sélectionner après commande (attendez bien la fin du message avant de faire la 2<sup>ème</sup> sélection)

Après la deuxième sélection: 2 couleurs apparaissent successivement au bas de votre écran, suite à cette sélection vous devez utiliser la Touche TAB

**IMPORTANT:** La commande A n'est valable que pour un seul remplissage. Vous avez trois formules d'alternances possibles: horizontale, verticale, ou mixte cette sélection est à faire par la commande « Paramètres de GRAPHO » Lettre A

## **TRACER les côtés du cadre: Touche T**

Les 4 côtés du cadre sont tracés par le pinceau sélectionné

## **COPIE d'une partie de l'écran: Touches COPIE et CONTROLE C**

Ces commandes vous permettent de copier une partie de votre écran visible vers une autre partie de votre écran visible.

### **La Touche COPIE:**

Cette touche indique à l'éditeur la partie qui doit être copiée. Cette partie est celle qui est incluse dans le cadre rectangulaire.

### **La Touche: CONTROLE C**

Elle indique à l'éditeur l'emplacement de votre écran qui doit recevoir la

copie. Cet emplacement est dans le cadre rectangulaire. Vous pouvez changer d'écran entre ces deux commandes.

Au moment de la recopie, l'éditeur vous demandera la conversion éventuelle à apposer

1) Message:

FORME	1	recopie de dimension identique à la source
	2	recopie doublée par rapport à la source
	3	recopie diminuée de moitié par rapport à la source

2) Message

INVERSION

0	pas d'inversion
1	effet miroir latéral
2	effet miroir vertical
3	effets: 1+ 2

Note sur la copie:

Le pinceau numéro zéro de l'image source n'est jamais recopié à la position nouvelle. Cette particularité vous permet d'incruster une image dans un décor.

## CATALOGUE d'un motif pour les RSX: Touche M

Avec les RSX vous pouvez identifier un motif par un numéro. La commande M dans l'éditeur vous permet d'affecter ce numéro

Procédure

- Encadrez sur votre écran la partie constituant le motif (cadre rectangulaire)
- Appuyez sur la touche M
- Donnez un numéro (0 à 39) faire RETURN

Lorsque vous demanderez à l'éditeur d'écrire l'écran sur votre disquette de travail, les numéros de motifs seront transmis avec l'écran ( les informations font partie de la première et dernière ligne de votre écran, c'est pour cette raison que des petits points de couleur apparaîtront à l'usage de cette commande). On appellera par la suite « écran de motifs» un écran dans lequel la fonction M à été utilisée. Ces écrans sont destinés aux RSX de Grapho et seront normalement stockés dans les mémoires non visibles. Dans les RSX vous identifierez un motif par son numéro dans la mémoire non visible. Si vous souhaitez utiliser les 4 mémoires non visibles vous pouvez avoir au maximum 4x40 motifs (160) à votre disposition dans vos programmes d'animation.

Note: Vous pouvez également créer un motif avec un RSX particulier sans avoir utilisé la commande M de l'éditeur

## RECHERCHE D'UN MOTIF SUR L'ECRAN TOUCHE ?

Si vous voulez retrouver un motif dans votre écran utilisez la commande ?

- Un message vous demandera le numéro du motif à retrouver.
- Entrez les numéros (0 à 39) faire RETURN.
- Si le motif existe, le cadre l'entourera

## TRACER une ligne: TOUCHE "L"

Pour tracer une ligne avec le pinceau qui a été sélectionné appuyez sur la touche: "L". Un message vous demandera de placer le curseur "point" à la position origine de la ligne à tracer. Utilisez les 4 flèches du clavier pour positionner le curseur point. Appuyez ensuite sur la touche "ESPACE" Un 2<sup>ème</sup> message vous demandera de placer le curseur point à la position fin de la ligne à tracer. Effectuez la même opération que précédemment

IMPORTANT: Si avant d'appuyer sur la barre "ESPACE" vous appuyez sur la touche ? vous obtenez les coordonnées du point curseur Y, X avec également le numéro du pinceau à cet emplacement (P=) Appuyez sur "ESPACE" pour arrêter ce message et continuez l'opération initialisée.

## 2.2 - LES MACRO-COMMANDES DE L'EDITEUR

Comme déjà dit, appuyez sur la touche "D E L" avant d'utiliser les touches décrites ci-après. Un message apparaîtra suite à l'appui sur "D E L"

## CHARGEMENT D'UN FICHER "IMAGE" à partir de la disquette: Touche COPIE

Un nom de fichier peut-être constitué de 2 parties: "le préfixe et le suffixe" les deux parties étant séparées par un point (Ex: SPI2. BIN). Lorsque vous créez un fichier "IMAGE" avec l'éditeur nous vous conseillons de ne jamais donner de suffixe. Laissez l'éditeur gérer ses suffixes. En effet, l'éditeur crée 2 fichiers pour un écran, le premier aura le suffixe BIN et contiendra l'écran, le second aura le suffixe PIN et contiendra les pinceaux avec leurs couleurs.

Chargement:

- Appuyez sur la touche COPIE (après DEL)
- Donnez le nom du fichier à charger et faire "RETURN" (si vous faites simplement "RETURN" sans nom, la commande est annulée) si le suffixe de votre fichier est BIN ce n'est pas nécessaire de l'indiquer.
- Un message vous demandera alors, si vous souhaitez charger le fichier dans l'écran visible (E) où dans l'écran mémoire (écran invisible) (M). Répondez E ou M
- Un autre message vous demandera si vous souhaitez charger les



pinceaux également, répondez O ou N (oui ou non) plus RETURN

NOTE: Ecran visible/Ecran mémoire

Sous l'éditeur vous pouvez avoir 2 écrans image en mémoire. L'un est votre écran visible, l'autre est un écran invisible. En début d'exécution, l'éditeur charge un fichier de motifs standards 8x8 dans l'écran mémoire. Vous pouvez par la suite charger autre chose dans cet écran. Une autre macro commande (L) vous permet de recharger l'écran motifs 8x8 dans votre écran mémoire

- Chaque écran possède son propre jeu de pinceaux

- Pour vous faire la main, vous pouvez charger les fichiers SPI2 et DEC

### **ECHANGER L'ECRAN visible et L'ECRAN mémoire TOUCHE CLR**

Cette commande vous permet d'échanger le contenu des 2 écrans ainsi que leurs pinceaux respectifs.

### **ECRIRE L'ECRAN visible sur la disquette TOUCHE CONTROLE S**

Seul l'écran visible peut-être créé sur la disquette. Donnez le nom du fichier lorsque le message vous demandera (un simple "RETURN" sans nom de fichier annule cette commande) L'écran et ses pinceaux sont stockés sur la disquette en 2 fichiers.

### **MODIFICATION de la forme du Cadre Touche: ↓ (peu importe laquelle)**

Cette commande vous permet de modifier la forme du cadre qui initialement est un rectangle.

Procédure: Le cadre est matérialisé par 4 points curseurs clignotants.

- Déplacez le premier point au moyen des 4 flèches du clavier. Pour connaître la position du point appuyez sur la touche? (voir commande élémentaire L)

- Validez la position du point par la barre "ESPACE" un autre point est maintenant sélectionné par l'éditeur.

**Ordre de sélection des points:** Bas gauche - Haut gauche - Haut droite - Bas droite

L'éditeur cycle indéfiniment dans cet ordre jusqu'à ce que vous appuyiez sur la touche D E L pour terminer la commande (vous pouvez ensuite appuyer sur la touche "ESPACE" pour matérialiser la forme de votre cadre). Modifier le cadre vous permettra de le remplir par un motif 8x8, une alternance de pinceaux, le pinceau sélectionné ou bien de tracer sa forme.

### **MODE POINT Touche « . » (point)**

Dans ce mode, un point curseur apparaît à la place du cadre. Vous pouvez le déplacer au moyen des 4 flèches du clavier. Si vous déplacez

le point en appuyant simultanément sur la touche "shift" et une flèche, le point est remplacé par le pinceau sélectionné. Dans ce mode vous pouvez changer de pinceau courant en sélectionnant un autre numéro de pinceau, la commande D E L termine cette commande, la touche « ? » est disponible dans ce mode (voir commande L).

Dans ce mode la dimension du pinceau peut être changée en appuyant sur contrôle ↑ et contrôle ➡ . Pour retrouver une dimension de un point appuyer sur la touche P

## ZOOM DU CONTENU DU CADRE TOUCHE Z

Le contenu du cadre peut-être "zoomé" (il s'agit plutôt d'une fonction loupe). Le grossissement est par défaut x4 mais vous pouvez changer ce grossissement par la commande "paramètres de grapho" et il peut-être également x1, x2, x8. Une fois l'écran "zoomé" si vous faites D E L vous terminez la commande si vous appuyez sur la touche E vous rentrez en mode point (voir précédemment) pour le grossissement courant. Cette commande est très pratique pour figurer un motif.

## MODE TEXTE TOUCHE « ' » (apostrophe)

Un cadre ligne apparaît. Vous pouvez déplacer ce cadre d'une ligne à la fois par l'usage des flèche ↓ où ↑

- Sur la ligne du bas du cadre, un point curseur peut-être déplacé au moyen des flèches ➡ où ⬅

- La touche "RETURN" replace le point curseur en début de ligne.

- Dans ce mode, tout caractère entré au clavier est affiché à l'emplacement du point curseur dans la couleur du pinceau qui est sélectionné. Le fond du caractère est celui où se trouve le point curseur.

- La touche "D E L" termine cette commande.

## DECALAGE DU CONTENU DU CADRE Touche / (barre de fraction)

- Le contenu du cadre est décalé (façon circulaire) dans le sens des 4 flèches du clavier. Très utile pour cadrer un motif dans l'écran. La touche "D E L" termine cette commande.

## CHANGER LA COULEUR d'un pinceau touche S

	Procédure	
Répondez:	1 <sup>er</sup> message :	O (oui)
		N (non) fin de commande
		"RETURN" fin de commande

2<sup>ème</sup> message      donnez le numéro du pinceau qui doit être rempli par une nouvelle couleur (0 à 14)

3<sup>ème</sup> message      donnez le numéro de la couleur à mettre dans le pinceau (0 à 26) voir sur votre

boîtier de lecteur de disquette si vous souhaitez avoir la palette des couleurs disponibles répondez à ce message par ?. N'oubliez pas de faire "RETURN" à chaque réponse

**IMPORTANT:** Le dernier pinceau qui a été modifié devient maintenant le pinceau sélectionné pour la suite de vos opérations.

### **CHANGER DANS LE CADRE un numéro de pinceau par un autre Touche: CONTROL "W"**

- Cette commande est très pratique lorsque vous vous apercevez, comme je l'ai moi même expérimenté, qu'il y aura une certaine ambiguïté, au stade de l'animation, dans l'attribution des pincesaux (d'où l'expression s'emmêler les pincesaux!)

- Procédure:

- Encadrez l'image à modifier (avant cette commande)

- Donnez le numéro du pinceau à changer

- Donnez le nouveau numéro du pinceau

Ne pas confondre: changer de couleur avec changer de numéro de pinceau sinon reportez vous au début de ce document.

### **TRACER UN ARC de cercle (3 possibilités)**

**Touche O:** Arc de cercle à rayon constant

**Touche A:** Arc de cercle à rayon variable, circulaire ou elliptique.

Procédure      - 1<sup>er</sup> message      si touche A: circulaire ou elliptique? Répondez C ou E

- 2<sup>ème</sup> message Arc plein (O/N) ?

N (non) seul l'arc sera tracé

O (oui) la surface délimitée par l'arc et

sa corde sera remplie par un motif (si sélectionné) une alternance de pinceau où le pinceau sélectionné.

Le sens de l'arc est celui inverse du sens des aiguilles d'une montre.

L'arc est toujours tracé par le pinceau qui est sélectionné. La touche "D E L" avant l'exécution annule la commande.

### **SELECTION d'un motif standard 8X8 Touche "M"**

L'écran mémoire est pris comme support des motifs 8x8 si il contient le fichier de l'éditeur ( chargé initialement) l'usage des flèches permet de choisir un motif 8x8 si il contient un de vos fichiers la sélection a lieu

également mais les flèches ne font déplacer le cadre que de 1 point à chaque fois. Lorsque le cadre est sur la position choisie, appuyez sur la barre "ESPACE" le contenu est alors sélectionné comme motif 8x8 pour l'usage de la touche "T A B" où pour le remplissage d'un arc de cercle. Attention cette sélection n'est valable que pour un seul remplissage. Le motif sélectionné sera reproduit pour la forme et la dimension de votre cadre, en mode sans raccords (par défaut) ou un mode avec raccords (mode relatif). Ce mode de raccordement doit être choisi avant par la commande "Paramètre de Grapho" . lettre "M".

NOTE: Le terme 8x8 signifie que les motifs font 8 points par 8 points. Cette correspondance existe dans vos programmes "BASIC" en faisant un : PRINT CHR\$(N).

### CHANGER les paramètres de GRAPHO Touche "Q"

- Un menu vous est présenté. Entrez la lettre correspondant à la modification souhaitée puis faites "RETURN". Un simple "RETURN" termine cette commande.

## 2. 3 LISTE DES COMMANDES ELEMENTAIRES

TOUCHE	FONCTION
0 à clavier principal	- selection d'un pinceau
0 à clavier numérique réduit	
ESPACE	- rappel du pinceau sélectionné
A	- alternance de pinceau pour un remplissage
TAB	- remplissage du cadre
E	- mode écran
C	- mode cadre
↓ ← ↑ → (flèches)	- déplacement du cadre entier
SHIFT + 1 flèche	- déplacement du coté du cadre correspondant à la flèche dans le sens de la flèche
Contrôle + 1 flèche	- déplacement du coté du cadre correspondant à la flèche dans le sens opposé à la flèche
L	- tracer une ligne
CLR	- effacement dernière commande
Copie Contrôle C	- copie contenu du cadre
T	- tracer le cadre
Z Contrôle Z	- agrandir / diminuer la forme complète du

	cadre
Q	- cadre à l'origine
M	- catalogue d'un motif
?	- visualisation d'un motif

## 2. 4 LISTE DES MACRO commandes (après avoir appuyé sur la Touche DEL)

TOUCHE	FONCTION
une flèche	- modifier la forme du cadre
O/A	- tracer un arc de cercle
. (point)	- mode point
' (apostrophe)	- mode texte
/	- décalage du contenu du cadre
Q	- paramètres de grapho
Z	- Zoom du contenu du cadre
S	- changer la couleur d'un pinceau
copie	- chargement à partir de la disquette
contrôle S	- écriture sur disquette
C L R	- échanger écran visible en écran mémoire
CONTROLE W	- changer un numéro de pinceau dans le cadre
M	- sélection d'un motif 8x8

## 2eme PARTIE

### UTILISATION DES RSX DE GRAPHO

#### Installation des RSX de Grapho dans BASIC:

Nous allons aborder maintenant la programmation des RSX de Grapho. C'est un domaine passionnant puisqu'il vous offre des possibilités nouvelles que seuls les programmeurs confirmés en langage machine peuvent s'octroyer. Qu'est-ce qu'un RSX ? c'est tout simplement une instruction nouvelle qui est incorporée au BASIC. Il y a donc une première étape à effectuer dans vos programmes BASIC pour cataloguer les nouvelles instructions RSX au vocabulaire du BASIC.

Cette étape est constituée par la ligne BASIC ci-dessous qui généralement doit être la première de votre programme.

**10 MEMORY &7FFF:**

**LOAD "RPIXEL.OBJ", &8000: CALL &8000**

**Que fait cette ligne?**

**MEMORY & 7FFF:** - Partitionne la mémoire de votre ordinateur pour y installer le programme de RSX.

**LOAD "RPIXEL.OBJ",&8000:** - Charge le programme de RSX à l'adresse spécifiée.

**Call &8000:** - Appel du programme de RSX pour qu'il incorpore ses instructions dans le vocabulaire du BASIC.

Voilà, maintenant vous pouvez utiliser les RSX.

**Important:** cette ligne provoque le passage en mode Zéro de votre écran. La disquette contenant le fichier "RPIXEL.OBJ" doit-être dans votre lecteur.

#### Comment est constituée la syntaxe d'une instruction RSX?

Voici typiquement la syntaxe:

**I nom-du-RSX, paramètre(s).** Le premier caractère doit-être I (shift @). Il indique à l'interpréteur BASIC que l'instruction qui suit est une instruction RSX. Si vous oubliez ce caractère vous obtiendrez "SYNTAX ERROR" à l'exécution.

**nom-du-RSX:** c'est le nom du RSX (à découvrir ci-après)

**paramètre(s):** ce sont les valeurs qui seront transmises au programme RSX pour qu'il puisse effectuer la fonction demandée. Chaque paramètre doit être précédé d'une virgule.

Exemple d'un RSX à 3 paramètres: I TOTO,A,B,C. Si le nombre de paramètres est incorrecte, le RSX ne fera rien.

### Que sont les paramètres dans un RSX?

Vous avez les mêmes possibilités que dans une instruction BASIC à savoir:

- un nombre, exemple: I TOTO,1,2
- un nom de variable, exemple: A=1 : I TOTO,A
- une chaîne de caractère, exemple: A\$="TTTT": I TOTO, A\$ ou I TOTO, "TTTT"
- une expression arithmétique, exemple: I TOTO,(1+A)/2

**Important:** Si vous employez un nom de variable numérique, la variable doit être impérativement du type numérique entier, c'est à dire que son nom doit se terminer par le symbole % à moins que vous l'ayez déclarée par l'instruction BASIC: DEFINT. Comme en général, dans vos programmes d'animation vous n'aurez pas à travailler sur de très grandes ou très petites valeurs, nous vous conseillons de toujours avoir comme deuxième ligne BASIC :

```
20 DEFINT A-Z
```

De cette façon toutes vos variables numériques seront du type entier et vous n'aurez pas à faire terminer leur nom par le symbole %.

Autre point important, lorsque vous utilisez un nom de variable comme paramètre dans un RSX, cette variable doit toujours avoir été utilisée (initialisée) par une instruction BASIC avant de pouvoir l'utiliser dans un RSX. La raison est que tant qu'une variable n'a pas été utilisée en BASIC, elle n'existe pas. Dans le cas contraire, le BASIC n'accepterait pas d'appeler le RSX et vous donnerait le message d'erreur suivant: "IMPROPER ARGUMENT".

### Paramètres spéciaux:

Pour certains RSX vous verrez le symbole @ (a enroulé) apparaître devant un paramètre. Pour ce type de paramètre vous ne pouvez employer qu'un nom de variable à l'exclusion du reste et votre nom de variable devra être précédé du symbole @.

exemple : I TOTO, @ A

Nous pouvons maintenant décrire les RSX de Grapho. Toutefois vous devez être familiarisé avec les termes suivants:

- YD, YF, XD, XF

Motif, Séquence, Décor, Pinceaux

- Pinceau du fond

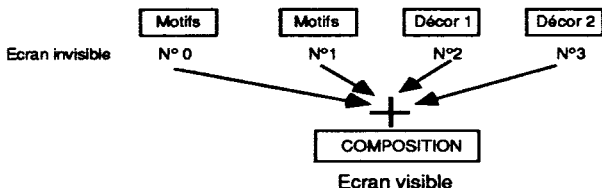
- Ecran visible/ écran invisible.

Si ce n'est pas le cas, reportez vous à l'introduction de ce document.

**PRINCIPE:** voici un principe que vous n'êtes pas obligé de suivre tant les possibilités sont grandes. Néanmoins, lorsque l'on débute, ce principe vous aidera à imaginer votre programmation:

- vos écrans invisibles vont vous servir à stoker vos planches de motifs et les décors de votre scénario.
- votre programmation composera, à l'aide du BASIC et des RSX, l'animation dans l'écran visible. Il vous sera donc possible de restituer n'importe quelle image même si vous avez altéré l'écran visible.

**EXEMPLE:**



## DESCRIPTION DES RSX

**Chargement d'un fichier issu de l'éditeur dans l'écran visible:**

RSX: I CHARGE, F, nom du fichier

Paramètres: F

si F = 1, seuls les pinceaux du fichier correspondant sont chargés et initialisés dans votre ordinateur.

si F = 2, seul le fichier image est chargé dans votre écran visible

si F = 3, correspond à F = 1 et F = 2

Nom du fichier: si le suffixe est .BIN, ce n'est pas utile de l'indiquer.

**Transfert de l'écran visible vers un écran invisible**

RSX: I SECRAN, ne

paramètre: ne = numéro de l'écran invisible devant recevoir l'écran visible.

ne = 0,1,2 ou 3

**Transfert d'un écran invisible vers un l'écran visible**

RSX: I RECRAN, ne

pour "ne" voir I SECRAN.



Exemple: chargement du fichier "SPI2" et transfert dans l'écran invisible numéro 0

```
10 MEMORY &7FFF : LOAD "RPIXEL.OBJ",&8000: CALL&8000
```

```
20 DEFINT A-Z
```

```
30 I CHARGE, 3, "SPI2": I SECRAN, 0
```

A la fin de ce programme, le fichier SPI2 se trouve et dans l'écran visible et dans l'écran invisible numéro 0.

Note dans les exemples qui suivront les lignes 10 et 20 ne seront plus mentionnées. La ligne 30 aurait pu s'écrire aussi:

```
30 I= 3: NOM$ = "SPI2" : I CHARGE, I, NOM$:I SECRAN,0
```

### Manipulation des motifs

Considérons maintenant que vous avez créé sous l'éditeur une planche de motifs et que vous l'avez transférée, dans votre programme BASIC, dans un écran invisible. Sous l'éditeur de Grapho vous avez donné un numéro (de 0à 39) à chacun de ces motifs. Vous souhaitez maintenant regrouper des numéros de motif en plusieurs séquences d'animation et exécuter ces animations. Pour chaque pas d'animation vous pouvez indiquer si le motif de la séquence doit être ou superposé au décor ou incrusté devant le décor ou derrière le décor. Voici les RSX de base que vous allez commander. Considérons aussi que le décor a été chargé lui aussi dans un écran invisible et dans l'écran visible. L'écran visible va nous servir à constituer l'animation.

**Premier point :** indiquer à Grapho où se trouve le décor

```
RSX: I EDEC, NEC,PFM, PFD
```

NEC= numéro d'écran invisible (0à3) où se trouve le décor

PFM= numéro du pinceau du fond des motifs (0 à 15)

PFD= numéro du pinceau du fond du décor (0 à 15)

**Définition d'une séquence de motifs:**

```
RSX: I SEQ, NOS,ECM,N0...N12
```

où: ECM= numéro de l'écran invisible où sont les motifs (0à3)

NOS= numéro de référence que vous donnez à cette séquence (0à15)

N0 à N12= numéros des motifs qui composent la séquence (numéros de 0à39 que vous avez donné sous l'éditeur).

Par la suite, les motifs se succéderont suivant l'ordre que vous avez donné dans cette instruction en recyclant sur le premier dès que le dernier aura été affiché. Vous pouvez indiquer de 1à13 numéros de motifs dans une séquence.

**Initialisation d'une séquence:**

```
RSX: I INITS, NOS, P, YD, XD
```

cette instruction commande l'affichage du premier motif de la séquence NOS aux coordonnées YD, XD indiquées.

P= 0 : affichage du motif sans tenir compte du décor. Le fond du motif n'est pas affiché

P= 1 : affichage du motif devant le décor

P= 2 : affichage du motif, derrière le décor

YD, XD : coordonnées du coin bas gauche ou sera affiché le motif.  
(rappel: YD de 0 à 199 , XD de 0 à 159)

### Lancement d'une séquence

RSX: I MOUV, @C, NOS, P, SENS, INCREMENT

A chaque fois que vous effectuerez cette instruction, le motif suivant dans la séquence NOS indiquée, sera affiché dans l'écran visible dans le sens indiqué et à la distance incrément (en nombre de points)

P= type d'affichage en fonction du décor (voir P dans l'instruction I INITS)

SENS=

- 1 vers le haut
- 2 vers le bas
- 4 à gauche
- 8 à droite
- 5 diagonale haute/gauche
- 6 diagonale basse/gauche
- 9 diagonale haute/droite
- 10 diagonale basse/droite

Ces valeurs sont identiques à celles que vous pouvez obtenir dans l'instruction BASIC: JOY ou par le RSX : I KBDJ

INCREMENT: nombre de points à respecter entre chaque affichage d'un motif de la séquence. Si Zéro les motifs s'afficheront toujours au même emplacement.

@C: code retour que vous pouvez tester suite à cette instruction

C= 0 : l'exécution a été correcte

C= 1 : l'affichage n'a pu être réalisé car le motif sortirait de l'écran

C= 255 : l'exécution a été correcte mais le motif affiché se trouve dans la zone définie par l'instruction I LIMIT (voir après)

C= 1 est prioritaire sur C= 255

IMPORTANT: Si l'instruction I INITS n'a pas été exécutée, pour la séquence indiquée, l'instruction I MOUV ne fera rien.

### Chainage de séquences

RSX/ I CHAINE, NOS,1, NOS2, NOS3 . . .

Cette instruction permet de chaîner plusieurs séquences, c'est à dire que les séquences chaînées seront affichées au même emplacement courant sur l'écran visible. L'initialisation d'une séquence chaînée équivaut à l'initialisation de l'ensemble des séquences qui sont chaînées.

Exemple d'utilisation:

```
30 I SEQ,1,0,3,4,5 " Séquence mouvement allant vers le haut"
40 I SEQ, 2,0,10,11,12 "Séquence mouvement allant vers le bas"
50 I SEQ,8,0,15,30,25 "Séquence mouvement allant vers la droite"
60 I SEQ,4,0,1,6,2 "Séquence mouvement allant vers la gauche"
70 I CHAINE,4,8,2,1 " Chainage des 4 séquences"
80 I INITS,2,1,100,80 "Initialisation d'une des séquences chaînées"
90 C= 0 : I KBDJ, `@C " Initialisation du Joystick"
100 v=0 : I KBDJ,@C,@V " Lecture clavier Joystick"
110 if c < > 2 THEN 100 " Ne lire que le Joystick N° 1"
120 I MOUV,@C,V,1,V,2
130 GOTO 100
```

L'instruction MOUV de la ligne 120 à son sens asservie par la valeur lue par KBDJ ainsi que son numéro de séquence.

### Sélection ou Insertion dynamique d'un motif dans une séquence

RSX: I PMOT,NOS,M

Cette instruction sélectionne le prochain motif à être affiché par MOUV dans la séquence NOS si vous souhaitez changer l'ordre séquentiel défini dans SEQ

M = numéro du prochain motif à prendre ( 0 à 39 )

Si le numéro du prochain motif à prendre n'a pas été défini dans l'instruction SEQ il est à ajouter à la séquence.

### Récupération du numéro du prochain motif à être affiché par MOUV

RSX : I DMOT,nos,@M

Cette instruction place dans M le numéro du prochain motif que l'instruction MOUV doit afficher pour la séquence NOS.

### Annulation de toutes les séquences

RSX: I REINIT

Ce RSX annule tous les ordres qui ont pu être donnés sur les séquences et réinitialise l'écran visible en mode 0.

### Récupération des coordonnées d'une séquence:

RSX: I COO, NOS, @Y début, @Y fin, @X début, @X fin.

Cette instruction vous donne les coordonnées rectangulaires du dernier motif affiché par l'instruction IMOUV pour la séquence NOS spécifiée.

### Exemple de programmation d'une séquence:

```
20 I SEQ, 1,3,0,1,2,3,4,9,8
30 YDE= 0; YFE= 0; XDE= 0; XFE=0; CR = 0
40 IINITS,1,1,100,0
50 WHILE CR= 0: IMOUV,@CR,1,1,8,2: WEND
60 ICOO,1,@YDE,@YFE,@XDE,@XFE: "motif en fin d'écran à droite"
70 END
Facile, non?
```

**Note sur I MOUV:** Cette instruction opère les actions suivantes:

- 1 - Sélection du motif suivant de la séquence
- 2 - Calcul de la nouvelle position d'affichage en fonction de "SENS" et "INCREMENT"
- 3 - Affichage du motif en fonction de "P" à la nouvelle position
- 4 - Restitution du décor à l'ancienne position
- 5 - Positionnement du code retour.

### Définition d'une zone à contrôler pour l'évolution d'une séquence

RSX: I LIMIT,NOS, YD,YF,XD,XF,

Cette instruction indique à l'instruction IMOUV de contrôler si un de ses motifs entre dans les limites de la zone indiquée. Si le motif est affiché par IMOUV dans cette limite le retour de IMOUV sera égal à 255. Pour annuler l'effet de cette instruction, écrire: I LIMIT, NOS sans les paramètres. Cette instruction ne peut être effectuée que si au moins une instruction IINITS a été effectuée pour la séquence NOS Considérée.

### Tester si une séquence (un motif d'une séquence) est dans un champ donné

RSX: IZONE,@CR,NOS,YD,YF,XD,XF.

Cette instruction teste immédiatement si une séquence se trouve dans le champ YD,YF,XD,XF spécifié. Suite à l'exécution de cette instruction le code retour CR aura les valeurs suivantes:

CR = 0                   Oui, la séquence est dans le champ spécifié  
CR = 255                Non

### Tester Si il y a collision entre deux séquences:

RSX: ICOL, @CR, NOS1,NOS2

NOS1                   numéro de la première séquence

NOS2                   numéro de la deuxième séquence

CR                     code retour à tester suite à cette instruction.

CR=0                   oui les deux séquences sont en collision

CR=255                non il n'y a pas de collision.

Nous venons de voir des instructions qui travaillent sur des séquences de motifs, voici maintenant des instructions qui ne travaillent que sur un seul motif

### Catalogue d'un motif dynamiquement dans votre programme:

sous l'éditeur de Grapho vous avez pu créer et cataloguer des motifs en leur donnant un numéro de référence. Vous pouvez également en créer et cataloguer avec les instructions suivantes dans votre programme. Le motif sera pris dans l'écran visible et catalogué dans l'écran invisible.

### Indiquer quel écran invisible doit être utilisé

RSX: IECSV, Nem paramètre. Nem: numéro (0 à 3) de l'écran invisible qui devra être utilisé par les instructions ISAUVE et IRESTIT que nous allons décrire:

### Catalogue d'un motif:

I SAUVE, NOMO,YD,YF,XD,XF le motif dont les coordonnées rectangulaires sont spécifiées est catalogué sous le numéro NOMO (0à39) dans l'écran invisible spécifié par l'instruction I ECSV.

### Rappel d'un motif dans l'écran visible:

RSX: I RESTIT, NOMO,YD,XD

ou: I RESTIT, P, NOMO,YD,XD

Rappel d'un motif catalogué par l'éditeur ou par l'instruction I SAUVE, dans l'écran visible. L'écran invisible où doit être pris le motif de numéro NOMO (0à39) a été donné par l'instruction I ECSV. Si le paramètre P n'est pas indiqué, l'ensemble du cadre rectangulaire définissant le motif sera affiché. Si le paramètre P est indiqué (voir les valeurs dans l'instruction I INIT) le motif sera soumis aux règles de décor, avant -plan, arrière-plan (voir I init). Si P est mentionné, l'instruction I EDEC doit être exécutée avant l'emploi de I RESTIT

### Diminution de la vitesse d'affichage pour les instructions I MOUV et I RESTIT

RSX: INET, V

paramètre: V: de 0 à 255

si V= 0: affichage à vitesse maxi

si V= 1 les RSX attendront le top de synchronisation du balayage écran avant d'afficher. Vous pouvez l'utiliser si vos motifs sont très petits. L'effet de mouvement sera plus lent mais mieux défini. Pour les autres valeurs de V plus la valeur sera grande plus la boucle d'attente sera longue.

## Instructions de décalage dans l'écran visible

RSX:

I GAU Décalage fermé ( les points qui sortent de l'écran sont restitués à l'opposé) à gauche de 2 points de tout l'écran visible.

I DRO Décalage identique mais à droite.

I D GAUCHE, NB, YD, YF, XD, XF

Décalable à gauche du nombre de points (NB) spécifié de la partie rectangulaire spécifiée dans l'écran visible. Attention XD doit être paire et XF impaire, sinon le RSX ajuste lui-même. Ce décalage est fermé ( voir I GAU).

I DGAUFIN, NB, YD, YF, XD, XF

Identique à I D GAUCHE mais sans les contraintes de parités.

I DROIFIN, NB, YD, YF, XD, XF

Identique à IDGAUFIN mais à droite.

I DECALHW, NB, YD, YF, XD, XF

Décalage vers le haut pour le nombre de points (NB) spécifiés vers le haut de l'écran visible (décalage fermé). Attention si NB est négatif le décalage est ouvert (les points qui sortent de l'écran sont perdus).

I DECALBW, NB, YD, YF, XD, XF

Identique à I DECALHW mais vers le bas de l'écran visible.

## Instructions permettant de travailler sur des points écran

Mettre un point (numéro de pinceau) dans un écran

RSX: I MPOINT, PI, YD, XD ou I MPOINT, ECC, PI, YD, XD

où PI= numéro du pinceau qui constitue le point (0à15). Si ECC est absent l'écran est l'écran visible sinon ECC = 0,1,2 ou 3

Lire un point dans un écran

RSX: IDPOINT, @PI, YD, XD ou I DPOINT, ECC, @PI, YD, XD

Le numéro du pinceau (0à15) à l'adresse écran YD, XD est placé dans pi.

Pour ECC voir I MPOINT

Donner la couleur d'un numéro de pinceau:

RSX: I COUL, @CO

Le numéro du pinceau pour lequel vous souhaitez connaître la couleur doit être chargé dans CO avant d'appeler cette instruction. A son retour CO contient la couleur qui est affectée à ce pinceau (0 à 26).

**Rechercher sur une ligne (horizontalement) la présence d'un numéro de pinceau:**

RSX: I PX, @PI, YD, XD, XF

A partir du point de départ (YD, XD ou bien YD, XF). Cette instruction recherche point après point la présence d'un numéro de pinceau contenu avant l'appel dans PI (0à 15). Cette instruction s'arrête si elle trouve un numéro de pinceau identique à celui demandé en vous fournissant dans PI la coordonnée X du point trouvé. Si le point n'est pas trouvé dans les limites spécifiées PI contiendra la valeur 255. Attention le sens de la recherche est conditionné par les valeurs XD et XF. Si XD est plus petit que XF la recherche s'effectuera de XD vers XF, sinon de XF vers XD.

**Exemple: rechercher la présence du pinceau 12**

PI= 12: I IPX,@PI, 100, 0, 255: IFPI =255 THEN non trouvé. La recherche ira sur la ligne 100 de la position 0 à 159 (gauche vers la droite)

PI= 12: I IPX, @PI, 100, 159, 0: IF PI = 255 THEN non trouvé de la droite vers la gauche.

**Rechercher sur une colonne (verticalement) la présence d'un numéro de pinceau:**

RSX: I PY, @PI, XD, YD, YF

Fonctionnement identique à I PX mais sur la colonne spécifique.

**Recherche sur une ligne (horizontalement) l'absence d'un numéro de pinceau.**

RSX: I IDX,@PI, YD, XD, XF

Fonctionnement identique à I PX mais en recherchant l'absence et non la présence d'un numéro de pinceau.

**Rechercher sur une colonne (verticalement) l'absence d'un numéro de pinceau**

RSX: I IDY,@PI,XD,YD,YF

Fonctionnement identique à I IDY mais en recherchant l'absence et non la présence d'un numéro de pinceau.

**Tracer une droite**

Vous avez 2 possibilités:

1<sup>ere</sup> possibilité:

Charger l'origine de la droite dans le curseur graphique

RSX: I ORGYX,YD,XD

Tracer la droite:

**RSX: I LIGNE,p,YF,XF**

La droite sera tracée avec le pinceau p de YD,XD jusqu'à YF,XF. Suite à cette instruction, le curseur graphique est à la position XF,YF  
2<sup>ème</sup> possibilité. Avec contrôle des valeurs pour rester dans les limites souhaitées

**RSX: I TRAIT, p,YD,XD,YF,XF**

Contrôle: Si XD est supérieur à XF le RSX ne fait rien également si YD est supérieur à YF

Si XF est supérieur à 159 alors XF aura la valeur 159

Si XD est inférieur à 0 alors XD aura la valeur 0

Si YF est supérieur à 199 alors YF aura la valeur 199

Si YD est inférieur à 0 alors YD aura la valeur 0

### Copie d'une ligne d'un écran vers un autre:

L'instruction suivante permet de copier une ligne d'un écran invisible vers l'écran visible ou bien de l'écran visible vers l'écran visible.

**RSX: I COPL,ECC,YD,XD,XF,YDE,XDE**

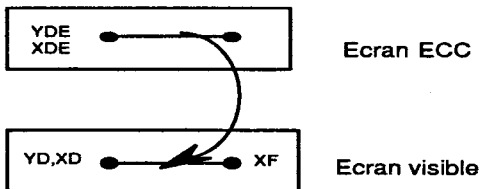
ECC= 0, 1, 2, ou 3, la copie s'effectue de ECC vers l'écran visible.

ECC = 4, la copie s'effectue de l'écran visible sur l'écran visible.

Coordonnées de destination de la copie: YD, XD, XF

Coordonnées d'origine de la copie: YDE, XDE

exemple

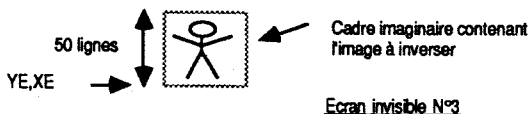


Note: XD et YDE doivent être pairs XF doit être impair. Si ce n'est pas le cas, Le RSX prendra le point pair inférieur et le point impair supérieur.



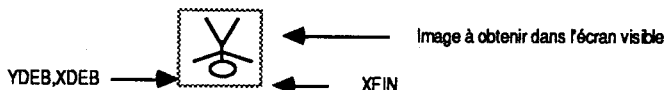
### Exemple de programmation:

Inversion d'une image contenue dans l'écran invisible 3:



Cette image doit être inversée dans l'écran visible aux coordonnées

YDEB = 30 et XDEB = 40



### Programmation:

```
10      YE=100: XE= 50: XDEB= 40: XFIN = 80: I = 50
20      FOR YDEB= 30 TO 80
30      I COPL, 3, YDEB, XDEB, XFIN, YE+I,XE
40      I= I -1
50      NEXT
```

### Recopie d'une partie d'écran dans un autre écran au même emplacement

RSX: I RGN,ECC, YD, YF, XD, XF ou I RGN, LECC, YD, YF, XD, XF  
ECC= 0, 1, 2 ou 3

La partie de l'écran ECC aux coordonnées spécifiées est recopiée aux mêmes coordonnées dans l'autre écran. Les coordonnées sont automatiquement ajustées si nécessaire pour rester dans les limites d'écran.

Si I = 0 ou si I est absent la copie s'effectue de l'écran invisible ECC vers l'écran visible. Si I = 1 la copie s'effectue de l'écran visible vers l'écran invisible ECC.

### Remplacement sélectif d'un numéro de pinceau sur une ligne donnée (horizontalement).

RSX: I CHINK, YD, XD, XF, APL, NPI

Sur la ligne spécifiée tous les pinceaux API seront remplacés par NPI.

### Changement inconditionnel de numéro de pinceau sur une ligne spécifiée:

RSX: I CHINK, YD, XD, XF, NPI

### Copie d'une partie d'écran vers un autre emplacement

Cette opération doit s'effectuer en deux temps

1<sup>er</sup> temps: indiquer l'origine de la copie

RSX: I ORIC,ECNO,YDO,YFO,XDO,XFO

où: ECNO = numéro de l'écran origine (0,1,2, ou 3 pour écrans invisibles, 4 pour l'écran visible )

YDO,YFO,XDO,XFO = coordonnées rectangulaires de la partie à copier

2<sup>eme</sup> temps, indiquer la destination de la copie et lancer l'opération:

RSX: I COPIE,ECND,YDD,XDD

Où: ECND = numéro de l'écran de destination(0,1,2,3 ou 4 voir I ORIC )

YDD,XDD = base de la destination.

Note XDO et XDD doivent être pairs, XFO doit être impair, sinon les adresses sont ajustées aux points pairs inférieurs et impairs supérieurs.

### Sélection d'un lecteur de disquette

RSX: I DRIVE,n

Où: n = 0 si Drive A (lecteur de l'AMSTRAD)

n = 1 si Drive B ( 2<sup>eme</sup> lecteur éventuel)

Cette instruction doit être évidemment utilisée avant I CHARGE

### Lecture du clavier et ou du joystick

L'instruction suivante peut remplacer les instructions BASIC: INKEY et JOY

Initialisation:

RSX: I KBDJ,@V

si v= 1 les joysticks ne seront pas utilisés

si v= 0 les joysticks et le clavier seront utilisés sans répétition de la touche TIR

si V = 2 identique à V= 0 mais avec répétition de la touche TIR

si V = 100 les joysticks et clavier seront utilisés sans répétition sur la touche TIR mais avec les positions diagonales du joystick ( si V = 0 les positions diagonales seront décomposées)

si V = 102 identique à V = 100 mais avec répétition sur la touche TIR

Note: si V= 1, le joystick sera remplacé par les 4 flèches du clavier et la touche TAB équivalente au bouton TIR du joystick.

### Lecture:

RSX: I KBDJ,@CR,@VAL

CR= code retour:

0 = rien n'a été lu

1= entrée à partir du clavier

2 = entrée à partir du joystick 1

22= entrée à partir du joystick 2

Si vous obtenez une entrée à partir du clavier (cr=1):

V contient alors la valeur binaire de la touche pressée. La page 21 du chapitre 7 de votre manuel AMSTRAD vous donne les valeurs binaires (exprimées en hexadécimale) des touches de votre clavier. Vous remarquerez que sur le dessin du clavier vous pouvez avoir jusqu'à 3 valeurs superposées par touche. La valeur du bas correspond à la valeur qui vous est fournie si la touche n'est pas en shift, la valeur du milieu est celle qui correspond au shift de cette touche et celle du haut correspond à la valeur fournie si la touche est en contrôle (certaines touches n'ont pas de contrôle).

Exemple de la touche TAB:

E1	valeur si contrôle
09	valeur si shift
09	valeur si non shift

Pour tester ces valeurs dans votre programme vous devez faire précéder la valeur par le symbole & pour indiquer que vous exprimez un nombre en hexadécimal (vous pouvez bien sûr le transformer en décimale)

Exemple:

```
10      I KBDJ,@C, @V
20      IF C=0 OR C=2 THEN 10
30      IF V=&09 (touche en shift ou non)
40      IF V=&E1 (touche en contrôle)
```

Si vous souhaitez transformer la valeur binaire fournie en représentation caractère:

```
A$= CHR$(V) : IF A$= "A"....
```

Ecrivez le petit programme suivant et testez les différentes possibilités avec les joysticks

```
10 MEMORY &7FFF: LOAD "RPIXEL.OBJ",&8000: CALL &8000
20 DEFINT A-Z
30 MODE 2: V = 0
40 C = 0 :I KBDJ,@C "initialisation"
50 I KBDJ,@C,@V: IF C < > 2 THEN 50
60 PRINT "C=";C;"V=" ";V
70 Goto 50
```

Testez successivement en changeant les valeurs de C à la ligne 40,

Testez avec C = 100

Testez avec C = 102 et C = 2

Comparez l'effet du manche en diagonale entre ces deux valeurs ainsi que l'usage de la touche TIR du joystick. Si vous avez deux joysticks, remplacer le IF de la ligne 50 par IF C < > 2 and C < > 22 THEN 50

**Si vous obtenez une entrée à partir du joystick (CR=2ou 22):**

ou bien si le joystick n'est pas utilisé mais que la touche pressée est soit une flèche du clavier ou la touche TAB.

Les valeurs fournies dans V seront :

V = 1 : en haut ou ↑ V = 5 diagonale haute /gauche

V = 2 : en bas ou ↓ V = 6 diagonale basse/gauche

V = 4 : à gauche ou ← V = 9 diagonale haute/ droite

V = 8 : à droite ou → V = 10 diagonale basse/droite

V = 16 : TIR (ou TAB)

exemple:

```
10      ! KBDJ, @C, @V
20      IF C = 0 OR C = 1 THEN 10
30      IF V = 1...
        IF V = 2
        IF V = 16...
```

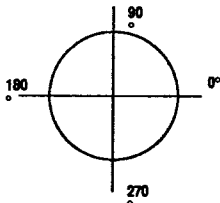
### **Instructions particulières:**

**Coordonnées d'un point vue sous un angle donné:**

RSX: I VUE, ANGLE, DISTANCE, @YD, @XD

Condition d'appel:

angle = angle sous lequel vous voyez le point (0° à 359°) à partir du centre



distance = nombre de points dont est distant le point vu. @YD, @XD = coordonnées du point d'observation (centre du cercle).

En retour: YD et XD contiennent les coordonnées du point qui est VU

**Exemples d'utilisation de cette instruction**

-Tracer un cercle :

Dans ce cas la distance = rayon du cercle, YD et XD en entrée sont les coordonnées du centre et l'angle varie de 0 à 360.

- Calculer l'adresse du point d'impact d'un tir de "LASER"
- etc ...

### Vibrations de l'écran

RSX: I VIBR : début de vibrations

I AVIBR : arrêt des vibrations

### LISTE DES RSX DE GRAPHO

I CHARGE	I MPOINT
I SECRAN	I DPOINT
I RECRAN	I COUL
I EDEC	I IPX
I SEQ	I IPY
I INTS	I IDX
I MOUV	I IDY
I COO	I ORGYX
I LIMIT	I LIGNE
I ZONE	I TRAIT
I COL	I COPL
I ECSV	I RGN
I SAUVE	I CHINK
I RESTIT	I KBDJ
I NET	I VUE
I GAU	I VIBR
I DRO	I AVIBR
I DGAUCHE	I ORIC
I DGAUFIN	I COPIE
I DROIFIN	I CHAINE
I DECALHW	I PMOT
I DECALBW	I DMOT
I REINT	

### Si un RSX ne donne pas le résultat escompté

- vérifiez que le nombre de paramètres du RSX est correct
- vérifiez que les valeurs des paramètres sont correctes
- si les RSX ont été installés, vous pouvez les utiliser directement à partir du clavier
- vérifiez que vos variables numériques sont bien de type "entier"
- n'avez vous pas oublié le symbole @ devant un paramètre?

## EN CAS DE PROBLEMES

Si vous sortez involontairement de l'éditeur de Grapho. Tout n'est pas perdu, votre dessin se trouve toujours dans l'écran invisible numéro 1. Ce petit programme vous permet de le récupérer après avoir réinitialisé votre ordinateur (touches SHIFT+ CONTROLE+ ESC)

La réinitialisation n'altère pas les écrans invisibles:

```
10      MEMORY &7FFF: LOAD"RPIXEL.OBJ", &8000: CALL
      &8000
20      DEFINT A-Z
30      MODE 2: INPUT"placez la disquette pour recevoir l'image
      et faites RETURN"; A$
40      INPUT "nom du fichier à créer"; NOM$
50      MODE 0: I RECRAN, 1
60      SAVE NOM$, B, 49152, 16384
70      END
```

Si vous avez un 2<sup>ème</sup> dessin dans l'éditeur, il se trouve dans l'écran invisible numéro 3. Adaptez la ligne 50 à l'écran 3

**Nous vous souhaitons de riches heures  
aux commandes de votre AMSTRAD.**

---

## Exemples de programmation

### 1- Motif circulaire

```
10 MEMORY &7FFF : LOAD "RPIXEL.OBJ", &8000 : CALL
    &8000
20 DEFINT A-Z : FORI = 1 TO15: INKI, I: NEXT
30 FOR L = 60 TO 0 STEP-2 "RAYON"
40 FOR A = B TO A+36 0 STEP 40 " ANGLE"
50 YD = 100: XD = 80 " CENTRE"
60 I ORGYX,YD,XD " CENTRE"
70 I VUE,A,L,@YD,@XD "ADRESSE DU POINT DARC"
80 I LIGNE,A,YD,XD
90 NEXT A
100 B=B+2
110 NEXT L
```

En modifiant les valeurs de L et des STEP vous obtiendrez des formes différentes.

### 2- Apparition d'un écran par ouverture progressive du diaphragme

```
10 MEMORY &7FFF: LOAD "PPIXEL.OBJ", &8000:CALL&8000
20 DEFINT A-Z
30 XD=80:XF=80:YF=100
40 I CHARGE,3,"DEC":I SECRAN,O: CLS
50 FOR YD=100 TO 0 STEP -1
60 I RGN,O,YD,YF,XD,XF
70 YF=YF+1: XD= XD-1: XF= XF+1
80 NEXT YD
```

Le fichier utilisé ici est le fichier "DEC" qui vous est fourni sur la disquette GRAPHO.

Si vous avez un deuxième lecteur de disquette et que "DEC" est sur ce lecteur rajoutez la ligne 35 I DRIVE,1 et la ligne 90 I DRIVE,0

### 3- Catalogue dynamique de 2 motifs et asservissement de 2 séquences de mouvements par 2 joysticks.

```
10 MEMORY &7FFF: LOAD "RPIXEL.OBJ",&8000: CALL &8000
20 DEFINT A-Z
30 I ECSV,O:V=O: INK 1,26:I NK2,24: GRAPHICS PEN 1
40 I ORGYX,7,0: TAG
50 PRINT CHR$(248); :I SAUVE,1,0,7,0,7: GRAPHICS
    PEN2:I ORGYX,7,0
60 PRINT CHR$(249); :I SAUVE,2,0,7,0,7
```

```

70  IREINIT: I EDEC,1,0,0: ISECRAN, 1
80  ISEQ,1,0,1,2 "SEQUENCE POUR LE JOYSTICK 1"
90  ISEQ,2,0,1,2 "SEQUENCE POUR LE JOYSTICK 2"
100 C=100: I KBDJ,@C "DIAGONALES JOYSTICKS VALIDEES"
110 IINITS,1,1,100,20 "INITIALISATION SEQUENCE 1"
120 IINITS,2,1,100,60 "INITIALISATION SEQUENCE 2"
130 I KBDJ, @C,@V: IF C< >2 AND C< >22 THEN 130
140 IF C=2 THEN I MOUV,@C,1,1,V,2 ELSE I MOUV,@C,2,1,V,2
150 I COL,@C,1,2 "TEST DE COLLISION ENTRE LES 2
      SEQUENCES"
160 IFC < > 0 THEN 130
170 SOUND 1,200,5,9: GOTO 130 "INDICATION DE COLLISION"

```

#### 4- Apparition/Disparition d'un motif

```

10  MEMORY &7FFF: LOAD "R PIXEL.OBJ", &8000: CALL&8000
20  DEFINT A-Z
30  I CHARGE,2,"DEC": ISECRAN,0 "ECRAN DECOR"
40  I CHARGE,3,"SP12": ISECRAN,1 "ECRAN MOTIF"
50  I EDEC,0,0,0: I RECRAN,0
60  EVERY 50,0 GOSUB 80
70  GOTO 70
80  S=S XOR 1
90  IF S=1 THEN I RESTIT,1,30,0,80 ELSE
      I RGN,0,0,40,80,90
100 RETURN

```

#### 5- Mouvements dans les 8 directions du joystick

```

10  MEMORY &7FFF: LOAD "RPIXEL.OBJ", &8000: CALL&8000
20  DEFINT A-Z: EM= 0: INK 1,26: GRAPHICS PEN 1
30  "CATALOGUE DES MOTIFS DES SEQUENCES"
40  I ECSV,EM "ECRAN DE MOTIFS"
50  I ORGYX,7,0:TAG: PRINT CHR$( 248); : I SAUVE,0,0,7,0,7
60  I ORGYX,7,0:TAG: PRINT CHR$( 249); : I SAUVE,1,0,7,0,7
70  I ORGYX,7,0:TAG: PRINT CHR$( 250); : I SAUVE,2,0,7,0,7
80  I ORGYX,7,0:TAG: PRINT CHR$( 251); : I SAUVE,3,0,7,0,7
90  "DEFINITION DES SEQUENCES"
100 ISEQ,4,EM,0,3 "SEQUENCE ALLANT A GAUCHE"
110 ISEQ,8,EM,0,2 "SEQUENCE ALLANT A DROITE"
120 ISEQ,1,EM,0,1 "SEQUENCE ALLANT EN HAUT"
130 ISEQ,2,EM,0,1 "SEQUENCE ALLANT EN BAS"

```



```

140 ISEQ,5,EM,0,3,1 "SEQUENCE DIAGONALE HAUTE/GAUCHE"
150 ISEQ,6,EM,0,3,1 "SEQUENCE DIAGONALE BASSE/GAUCHE"
160 ISEQ,9,EM,0,2,1 "SEQUENCE DIAGONALE HAUTE/DROITE"
170 ISEQ,10,EM,0,2,1 "SEQUENCE DIAGONALE BASSE /DROITE"
180 ICHAINE,4,8,1,2,5,6,9,10 "CHAINAGE DES SEQUENCES"
185 IINTS,4,1,100,80
190 CLS:ISECRAN,1:I EDEC,1,0,0 "ECRAN DECOR"
200 C=100,V= 0: I KBDJ,@C,@ V "INIT. JOYSTICK"
210 I KBDJ,@C,@V: IF C < > 2 THEN 210
220 I MOUV,@C,V,1,V,2
230 GOTO 210

```

Remarquez que les numéros des séquences correspondent au sens du joystick.

A la ligne 200 remplacez C= 100 par C= 0 et constatez la différence sur les diagonales.

---

## AVERTISSEMENT

- L'éditeur de Grapho se trouvant sur la face 1 de la disquette est protégé.  
La copie sur une autre disquette donne des résultats aléatoires dans le temps.
- Chargement de l'éditeur  
Le message "Placez votre disquette de travail" doit apparaître, si à sa place vous obtenez un message illisible, voici la marche à suivre:
  - 1) réinitialisez votre ordinateur (touches ESC+SHIFT+CONTROLE)
  - 2) lancez RUN "GRAPHO"

### CONSEIL:

Si lors de l'utilisation de votre logiciel vous rencontriez une difficulté de compréhension du système GRAPHO, écrivez à C.T.S à l'adresse figurant au dos du boîtier en joignant une enveloppe timbrée à votre adresse et en posant clairement votre question, nous essayerons de vous aider dans les plus brefs délais.

