

# SHARP®

## ORDINATEUR DE POCHE

MODELE **PC-1350**

### MODE D'EMPLOI



# TABLE DES MATIERES

	Page
INTRODUCTION . . . . .	4
CHAPITRE 1 UTILISATION DE CETTE BROCHURE. . . . .	5
CHAPITRE 2 PRESENTATION DU PC-1350 . . . . .	7
Description des touches . . . . .	8
Description de l'écran. . . . .	12
Bouton ALL RESET . . . . .	14
Remplacement des piles . . . . .	16
CHAPITRE 3 UTILISATION DU PC-1350 COMME CALCULATRICE. . .	19
Démarrage. . . . .	19
Mise hors tension. . . . .	19
Mise hors tension automatique. . . . .	19
Quelques conseils utiles . . . . .	20
Calculs simples . . . . .	21
Rappel des données d'entrée . . . . .	22
Erreurs . . . . .	26
Calculs en série . . . . .	27
Nombres négatifs. . . . .	28
Calculs composés et parenthèses. . . . .	29
Utilisation de variables dans les calculs. . . . .	30
Calculs en chaîne. . . . .	31
Notation scientifique . . . . .	32
Limites . . . . .	33
Fonction dernière réponse. . . . .	33
Longueur des formules . . . . .	34
Calculs scientifiques. . . . .	35
Priorité dans les calculs manuels. . . . .	37
Impression des calculs manuels. . . . .	38
Erreur de calcul. . . . .	38
CHAPITRE 4 CONCEPTS ET TERMES BASIC. . . . .	40
Constantes alphanumériques . . . . .	40
Nombres hexadécimaux . . . . .	40
Variables . . . . .	41
Variables fixes . . . . .	42
Variables simples. . . . .	44
Variables tableaux . . . . .	44
Variables de format A ( ). . . . .	48
Expressions . . . . .	49
Opérateurs numériques. . . . .	50

Expressions alphanumériques. . . . .	50
Expressions de relation. . . . .	51
Expressions logiques . . . . .	52
Parenthèses et priorité des opérateurs . . . . .	54
Mode RUN. . . . .	54
Fonctions . . . . .	54
CHAPITRE 5 COMMENT PROGRAMMER AVEC LE PC-1350. . . . .	56
Les programmes . . . . .	56
Les instructions BASIC. . . . .	56
Les numéros de ligne . . . . .	56
Les verbes BASIC . . . . .	57
Les commandes BASIC. . . . .	57
Les modes . . . . .	58
Début de la programmation sur le PC-1350. . . . .	58
Exemple 1 – Entrée et exécution d'un programme . . . . .	59
Exemple 2 – Correction d'un programme . . . . .	59
Exemple 3 – Utilisation de variables. . . . .	62
Exemple 4 – Programmation plus complexe. . . . .	65
Enregistrement des programmes dans la mémoire du PC-1350 . . . . .	66
Fonctions graphiques. . . . .	66
Fonctions d'entrée-sortie série . . . . .	71
CHAPITRE 6 QUELQUES METHODES RAPIDES. . . . .	73
La touche DEF et les programmes identifiés par une lettre . . . . .	73
Mode RESERVE. . . . .	74
Réglettes . . . . .	76
CHAPITRE 7 UTILISATION DE L'INTERFACE IMPRIMANTE/ CASSETTE . . . . .	77
Utilisation de l'imprimante . . . . .	77
Utilisation de l'interface cassette . . . . .	79
Remarque sur les bandes. . . . .	84
CHAPITRE 8 UTILISATION DE LA CARTE RAM EN OPTION. . . . .	85
Insertion de la carte RAM. . . . .	85
Retrait de la carte RAM . . . . .	87
Utilisation de la carte RAM. . . . .	89
Sauvegarde de données sur une carte RAM. . . . .	90
Précautions lors de l'utilisation d'une carte RAM. . . . .	91
CHAPITRE 9 REFERENCES BASIC . . . . .	93
Commandes . . . . .	97
Verbes. . . . .	113
Fonctions . . . . .	163
Pseudo variables . . . . .	163
Fonctions numériques . . . . .	167

Fonctions alphanumériques . . . . .	171
CHAPITRE 10 DETECTION ET CORRECTION DES INCIDENTS . . . . .	207
Fonctionnement de l'ordinateur . . . . .	207
Mise au point des programmes BASIC . . . . .	208
CHAPITRE 11 MAINTENANCE DU PC-1350 . . . . .	213
ANNEXES . . . . .	214
Annexe A: Messages d'erreur . . . . .	214
Annexe B: Tableau des codes caractères . . . . .	217
Annexe C: Mise en forme des données en sortie . . . . .	220
Annexe D: Calcul des expressions et priorité des opérateurs . . . . .	224
Annexe E: Fonction des touches en BASIC . . . . .	226
Annexe F: Signaux utilisés avec le terminal d'entrée-sortie série . . . . .	231
Annexe G: Spécifications . . . . .	233
Annexe H: Utilisation de programmes mis au point pour le PC-1245, le PC-1250/1251, le PC-1260/1261 ou le PC-1401 . . . . .	235
EXEMPLES DE PROGRAMMES . . . . .	241



# INTRODUCTION

Vous faites maintenant partie des heureux détenteurs d'un ordinateur SHARP!

Peu d'industries de nos jours connaissent la croissance foudroyante et les progrès rapides qui ont été fait dans le domaine des ordinateurs personnels. Les ordinateurs qui, il n'y a pas encore si longtemps, occupaient une pièce énorme, nécessitaient les compétences d'une personne diplômée pour leurs programmes et coûtaient des milliers de dollars, tiennent maintenant dans la paume de la main, sont faciles à programmer et coûtent si peu qu'ils sont presque à la portée de chacun.

Votre nouvel ordinateur PC-1350 de SHARP a été conçu de façon à vous offrir les progrès les plus récents de cette révolution en matière d'ordinateurs et comporte de nombreuses fonctions d'avant-garde:

- \* Sauvegarde de la mémoire — le PC-1350 se rappelle les programmes et les variables enregistrés même si vous le mettez hors fonction.
- \* Fonctionnement sur piles pour une portabilité totale.
- \* Fonction de mise hors tension automatique qui permet d'économiser les piles et met l'ordinateur hors tension si aucune activité n'a lieu pendant un laps de temps déterminé.
- \* Version étendue BASIC qui permet la mise en forme des données en sortie, des tableaux à deux dimensions, des chaînes de longueur variable et bien d'autres fonctions avancées.
- \* Une interface imprimante/cassette en option (modèle CE-126P) permet d'obtenir des copies imprimées des programmes et des données. L'interface cassette permet de connecter un enregistreur afin d'enregistrer les programmes et les données sur cassettes.

Votre ordinateur vous fait entrer dans un monde passionnant et excitant. Nous sommes sûrs que cet achat vous paraîtra l'un des plus sages que vous ayez jamais fait. L'ordinateur PC-1350 de SHARP est un outil puissant conçu pour répondre à vos besoins particuliers dans des domaines divers: mathématiques, sciences, technique, gestion et également dans le domaine domestique. Avec le PC-1350 de SHARP, vous pouvez commencer **DES MAINTENANT** à rechercher les solutions dont vous aurez besoin demain!

# CHAPITRE 1

## UTILISATION DE CETTE BROCHURE

Cette brochure a pour but de vous familiariser avec les possibilités et les fonctions de votre PC-1350. Elle sera pour vous un outil de référence précieux. Que vous soyez novice en la matière ou que vous ayez déjà l'habitude des ordinateurs, nous vous conseillons de lire attentivement les chapitres 2 à 6 pour vous familiariser avec votre PC-1350.

- \* Le Chapitre 2 décrit les caractéristiques physiques du PC-1350.
- \* Le Chapitre 3 montre l'emploi du PC-1350 comme calculatrice scientifique.
- \* Le Chapitre 4 définit certains termes et certaines notions indispensables pour programmer en BASIC. Vous y trouverez également quelques remarques sur l'emploi de ces notions avec le PC-1350.
- \* Le Chapitre 5 vous montre comment programmer en BASIC sur le PC-1350 et vous indique comment entrer, corriger et exécuter vos programmes.
- \* Le Chapitre 6 présente quelques méthodes rapides qui facilitent l'emploi de votre nouvel ordinateur et le rendent encore plus attrayant.

Les programmeurs ayant déjà une expérience du BASIC peuvent ensuite lire le chapitre 9 pour savoir comment les fonctions spécifiques du BASIC sont utilisées sur le PC-1350. BASIC est quelque peu différent suivant l'ordinateur sur lequel il est utilisé; par conséquent lisez ce chapitre au moins une fois avant de vous lancer dans la programmation.

Le Chapitre 9 pourra servir d'outil de référence; il comporte tous les verbes, toutes les commandes et toutes les fonctions de BASIC regroupés par ordre alphabétique.

Si vous n'avez jamais programmé en BASIC auparavant, nous vous conseillons d'acheter un manuel de programmation en BASIC pour débutant ou de suivre un cours avant de tenter de lire ces chapitres car cette brochure n'a pas pour but de vous apprendre à programmer.

Le reste de la brochure comporte:

- \* Le Chapitre 7 — Renseignements essentiels sur l'interface Imprimante/Cassette CE-126P en option.
- \* Le Chapitre 8 — Utilisation de la carte RAM en option CE-201M/202M.
- \* Le Chapitre 10 — Guide de résolution des problèmes qui vous permettra de résoudre certains problèmes d'utilisation et de programmation.
- \* Le Chapitre 11 — Maintenance de votre nouvel ordinateur.

Des annexes détaillées, à la fin de cette brochure, vous fournissent des tableaux et des explications particulières sur l'emploi du PC-1350.

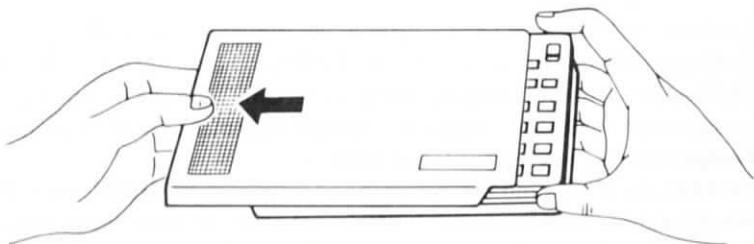
## UTILISATION DU COUVERCLE RIGIDE

Lorsque vous n'utilisez pas votre ordinateur, remplacez le couvercle rigide sur le panneau des touches.

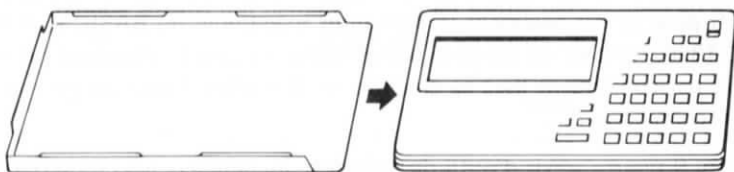
- Lorsque vous voulez utiliser votre ordinateur:

Pour retirer le couvercle rigide, procédez comme indiqué à la figure ci-dessous.

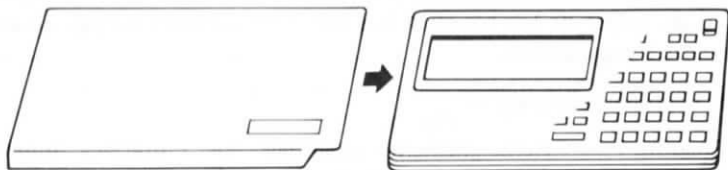
Etape ①



Etape ②



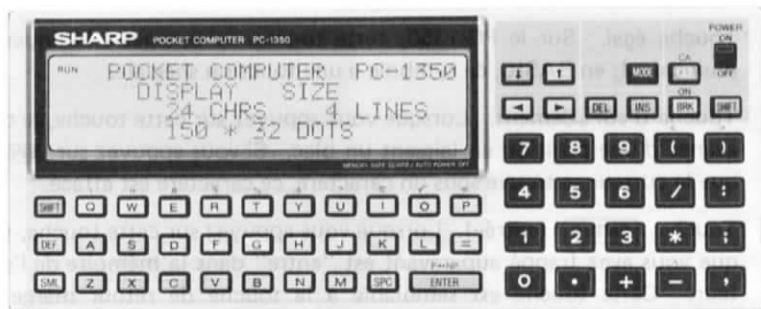
- Lorsque vous n'utilisez plus votre ordinateur.



## CHAPITRE 2 PRESENTATION DU PC-1350

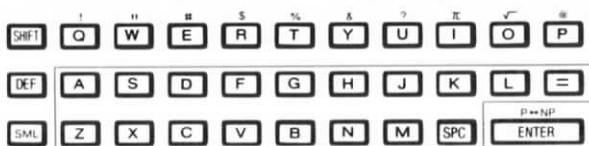
Le PC-1350 de SHARP se compose de:

- \* Un clavier de 62 touches
- \* Un écran de 4 lignes de 24 caractères ou un écran graphique de 32 x 150 points.
- \* Un processeur CMOS 8 bits (SC-61860 SHARP)
- \* Une mémoire morte (ROM) de 40KO.
- \* Une mémoire vive (RAM) de 5KO (standard)
- \* Une interface imprimante/cassette CE-126P en option.



Pour vous familiariser avec l'emplacement et les fonctions des différentes parties du clavier du PC-1350, nous allons maintenant étudier chacune de ces parties. Pour l'instant, contentez-vous de repérer les touches et de lire leur description. Nous commencerons à utiliser votre nouvel ordinateur au chapitre 3.

## DESCRIPTION DES TOUCHES

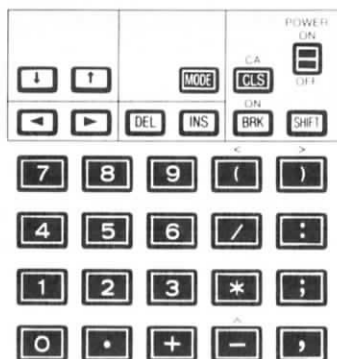


- A ~ Z** Le clavier du PC-1350 de SHARP comporte 26 lettres disposées de la même manière que sur la plupart des machines à écrire. Les lettres sont en majuscule normalement (à l'inverse des machines à écrire). Ceci est plus pratique car le PC-1350 ne reconnaît les instructions et les commandes que lorsqu'elles sont en majuscule. Les lettres peuvent être frappées en minuscule: il faut pour cela appuyer sur la touche **SML** avant d'appuyer sur la lettre voulue.
- =** Touche égal. Sur le PC-1350, cette touche ne sert pas à indiquer la fin d'un calcul; en BASIC, ce symbole a une fonction spéciale.
- SPC** Touche d'eSPaCement. Lorsque vous appuyez sur cette touche, le curseur avance d'une position en laissant un blanc. Si vous appuyez sur **SPC** alors que le curseur se trouve sous un caractère, ce caractère est effacé.
- P ↔ NP**  
**ENTER** Touche **ENTER** (entrée). Lorsque vous appuyez sur cette touche, tout ce que vous avez frappé auparavant est "entré" dans la mémoire de l'ordinateur. Cette touche est semblable à la touche de retour marge d'une machine à écrire. Pour que le PC-1350 puisse utiliser les données alphanumériques en provenance du clavier, vous devez appuyer sur **ENTER**. Si vous appuyez sur **SHIFT** avant d'appuyer sur cette touche, le PC-1350 imprimera ou arrêtera d'imprimer les données sur l'imprimante (suivant qu'il était déjà en mode impression ou non).
- DEF** Touche **DEF**. Il s'agit d'une touche spéciale utilisée pour l'exécution des programmes BASIC.
- SHIFT** La touche **SHIFT** a deux fonctions importantes. Tout d'abord, si cette touche est utilisée avant l'utilisation d'une touche comportant une seconde fonction imprimée au-dessus de la première, c'est cette seconde fonction qui sera appelée (par exemple si la touche **SHIFT** est utilisée avant la touche **CLS** c'est la fonction CA (effacement total qui sera appelée). Deuxièmement, si la touche **SHIFT** est utilisée avant la touche **MODE**, l'ordinateur passera en mode Réserve. D'ordinaire, la touche **MODE** sert à sélectionner alternativement le mode RUN (exécution) ou le mode PROGRAM. La touche **SHIFT** permet de choisir un troisième

mode. (Pour plus de détails, se reporter aux explications de la touche **MODE** .)

**SML** La touche **SML** permet d'imprimer toutes les lettres en minuscules. Si vous appuyez une seconde fois sur la touche **SML** , vous revenez au mode majuscules.

! " # Ces symboles se trouvent sur la rangée supérieure du clavier. Pour  
 \$ % & afficher l'un de ces symboles, appuyez sur la touche **SHIFT** puis sur  
 ?  $\pi$   $\sqrt{\quad}$  la touche alphabétique sur laquelle se trouve ce symbole.  
 @



POWER



Ces interrupteur permet de mettre le PC-1350 en fonction et hors fonction (ON et OFF).

SHIFT

Touche **SHIFT** . Voir page 8.

CA  
CLS




















La touche rouge d'effacement **CLS** permet d'effacer l'écran. Si vous appuyez sur la touche **SHIFT** avant d'appuyer sur la touche **CLS** c'est la fonction secondaire d'effacement total qui sera en virgule. Cette fonction permet de restaurer l'ordinateur et efface l'écran et la pile mémoire. REMARQUE: La fonction d'effacement total n'efface ni les programmes, ni la mémoire de réserve, ni les variables. Pour effacer tous les programmes, il faut se trouver en mode PROGRAM et frapper **NEW** puis appuyer sur la touche **ENTER** . Pour effacer les positions de la mémoire de réserve, il faut se trouver en mode RESERVE et frapper **NEW** puis appuyer sur la touche **ENTER** . Pour effacer toutes les variables et les remettre à zéro, il faut frapper **CLEAR** et appuyer sur **ENTER** .

MODE

Lorsque vous mettez votre PC-1350 sous tension, regardez sur l'écran dans quel mode il se trouve (RUN, PROGRAM ou RESERVE). Pour passer du mode RUN au mode PROGRAM et vice versa, appuyez sur la touche **MODE** . Pour passer en mode RESERVE, appuyez sur la touche **SHIFT** puis sur la touche **MODE** . Pour sortir du mode RESERVE, appuyez sur la touche **MODE** . Tous les calculs manuels (lorsque l'ordinateur est utilisé comme une calculatrice) sont effectués en mode RUN et tous les programmes sont exécutés en mode RUN. Tous les programmes sont écrits et édités (corrigés) en mode PROGRAM. Le mode RESERVE permet d'enregistrer les fonctions fréquemment utilisées en les affectant à une touche déterminée et d'enregistrer les menus qui permettent d'identifier rapidement ces fonctions.

ON  
BRK

Cette touche interrompt temporairement un programme en cours d'exécution. Lorsque cette touche est utilisée après une mise hors tension automatique, elle remet l'ordinateur sous tension.

-  Touche d'insertion. Lorsque vous appuyez sur cette touche, l'ordinateur insère un espace juste devant le caractère sous lequel se trouve le curseur. Vous pouvez alors INSérer de nouveaux caractères dans cet espace.
-  Touche de suppression. Cette touche permet de supprimer le caractère qui se trouve à l'emplacement du curseur.
-  Touche de défilement vers le bas. Cette touche permet d'afficher la ligne suivante.
-  Touche de défilement vers le haut. Cette touche permet d'afficher la ligne précédente.
-  Touche de rappel arrière. Cette touche permet de déplacer le curseur vers la gauche sans effacer les caractères frappés précédemment.
-  Touche d'avance. Cette touche permet de déplacer le curseur vers la droite sans effacer les caractères frappés auparavant.
-  Touches numériques. La disposition de ces touches est semblable à celle d'une calculatrice standard.
-  Touche de division. Appuyez sur cette touche pour insérer l'opérateur de division dans vos calculs.
-  Touche de multiplication. Appuyez sur cette touche pour insérer l'opérateur de multiplication dans vos calculs.
-  Touche de soustraction. Appuyez sur cette touche pour insérer l'opérateur de soustraction dans vos calculs. Si vous appuyez sur la touche  puis sur cette touche, le symbole "puissance" s'affichera indiquant qu'un nombre doit être élevé à une puissance déterminée.
-  Touche d'addition. Appuyez sur cette touche pour insérer l'opérateur d'addition dans vos calculs.
-  Touche parenthèse gauche. Si vous appuyez sur  puis sur cette touche, le caractère "inférieur à" s'affiche.
-  Touche parenthèse droite. Si vous appuyez sur  puis sur cette touche, le caractère "supérieur à" s'affiche.
-  Touche deux points
-  Touche point virgule.
-  Touche virgule.



## DESCRIPTION DE L'ECRAN


RUN  
PRO


SML  
SHIFT  
DEF

Le PC-1350 comporte un écran programmable à cristaux liquide matriciel de 32 x 150 points. Cet écran peut afficher 24 caractères par lignes et 4 lignes au total. Chaque caractère occupe 5 x 7 points.

Pour les graphiques, la totalité de l'écran peut être utilisée comme une matrice de 32 x 150 points. Chaque point de l'une des 150 colonnes peut être excité de façon à créer des graphiques, des figures ou des symboles spéciaux.

L'écran comporte:

 Symbole d'attente d'entrée au clavier qui apparaît lorsque l'ordinateur attend des données d'entrée. Lorsque vous frappez des données, ce symbole disparaît et est remplacé par le curseur.

 Le curseur. Ce symbole (le souligné) vous indique l'emplacement du caractère que vous allez frapper. Lorsque vous commencez à frapper des données, le curseur remplace le guide-opérateur. Le curseur permet également de placer l'ordinateur sur certains caractères lors de fonctions d'insertion et de suppression.

**RUN** Indicateur RUN (exécution). Cet indicateur vous avertit que votre PC-1350 se trouve en mode RUN.

**PRO** Indicateur PROgramme. Cet indicateur vous avertit que votre PC-1350 se trouve en mode programmation.

**REMARQUE:** Si ni RUN ni PRO n'est affiché à l'écran, votre ordinateur se trouve en mode RESERVE (RSV).

**SML** Indicateur de mode minuscules. "SML" est affiché lorsque la touche **SML** est utilisée. Lorsque cet indicateur est affiché, les lettres de l'alphabet sont entrées en minuscules. Lorsque la touche **SML** est utilisée alors que "SML" est affiché, l'indicateur disparaît et l'ordinateur repasse en mode majuscules.

**SHIFT** Indicateur SHIFT. Cet indicateur s'allume lorsque la touche **SHIFT** a été enfoncée. N'oubliez pas qu'il faut relacher la touche **SHIFT** avant d'appuyer sur une autre touche.

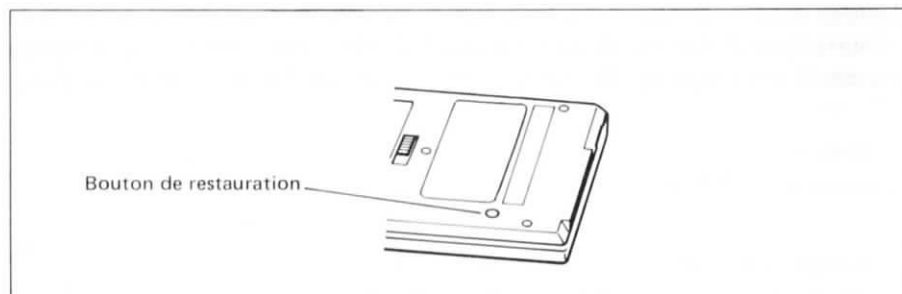
**DEF** Indicateur de mode définition. Ce symbole s'allume lorsque vous appuyez sur la touche **DEF** .

Pour les affichages dépassant 4 lignes

L'unité d'affichage du PC-1350 comporte 4 lignes (24 caractères par ligne). Les données entrées au clavier ou les résultats calculés sont affichés à partir de la première ligne de l'écran. Si les caractères à afficher dépassent 4 lignes, l'affichage sera décalé d'une ligne vers le haut (la première ligne affichée disparaîtra de l'écran).

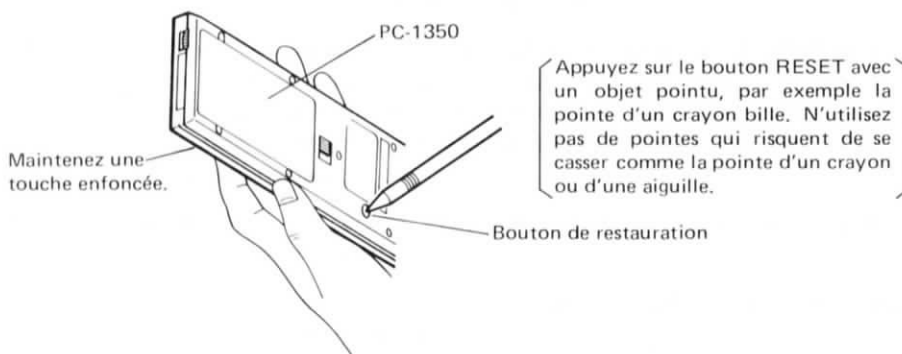
## ALL RESET

**ALL RESET:** Bouton de restauration. Ce bouton permet de restaurer l'ordinateur lorsque les touches d'effacement ( **CLS** ) ou d'effacement total ( **CA** ) ne suffisent pas à corriger un incident.

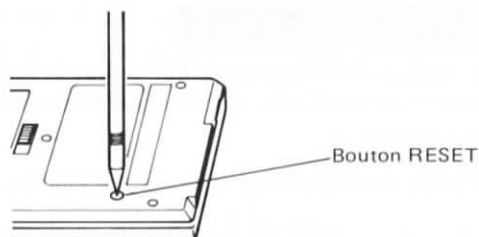


### REMARQUE

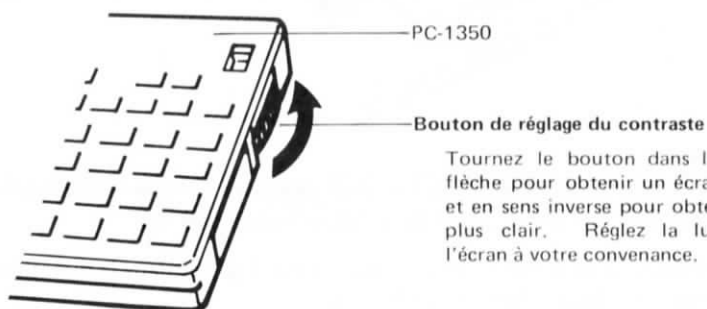
Pour restaurer le PC-1350, maintenez l'une des touches du clavier enfoncée et appuyez en même temps sur le bouton RESET situé au dos de l'ordinateur. Tous les programmes, toutes les variables et la mémoire de réserve seront ainsi préservés.



Si vous n'obtenez aucune réponse des touches même après l'opération ci-dessus, appuyez sur le bouton RESET sans appuyer sur aucune touche. Cette opération efface les programmes, les données et le contenu de la mémoire de réserve. Par conséquent, n'appuyez sur le bouton RESET seul qu'en cas de besoin.



Si l'ordinateur ne fonctionne toujours pas correctement, retirez les piles. Après avoir attendu 10 secondes, insérez à nouveau les piles et appuyez sur le bouton ALL RESET.



Tournez le bouton dans le sens de la flèche pour obtenir un écran plus foncé et en sens inverse pour obtenir un écran plus clair. Réglez la luminosité de l'écran à votre convenance.

## REMPACEMENT DES PILES

Le PC-1350 fonctionne sur piles au lithium uniquement. Lorsqu'il est connecté au CE-126P en option, le PC-1350 peut également être alimenté à partir du CE-126P si celui-ci a suffisamment de courant et si la puissance de ses propres piles est en baisse. Il est ainsi possible d'économiser les piles.

Lorsque vous remplacez les piles, tenez compte des instructions ci-dessous. Vous éviterez ainsi bien des problèmes:

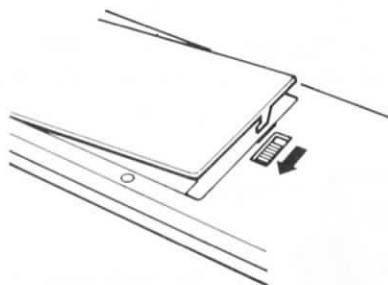
- Remplacez toujours les deux piles en même temps.
- Ne mettez pas ensemble une pile neuve et une pile usagée.
- N'utilisez que des piles au Lithium (type CR-2032). Il en faut deux.

## INSTALLATION DES PILES

Si l'écran est sombre, difficile à lire même lorsque vous tournez le bouton de réglage du contraste au maximum, cela veut dire que les piles sont trop usées. Dans ce cas, remplacez-les rapidement.

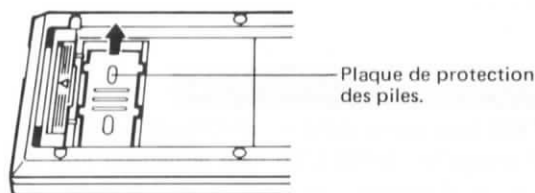
- (1) Mettez l'ordinateur hors tension en plaçant l'interrupteur sur la position OFF.
- (2) Retirez la plaque arrière de l'ordinateur en glissant le bouton de verrouillage dans la direction de la flèche comme indiqué à la figure ci-dessous.

Fig. 1



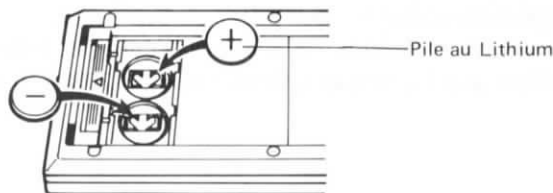
- (3) Si la carte de mémoire RAM en option se trouve dans la fente réservée à cette carte, retirez-la en suivant les procédures indiquées à la page 88.
- (4) Retirez la plaque de protection des piles en la faisant glisser dans la direction de la flèche comme indiqué à la figure ci-dessous.

Fig. 2

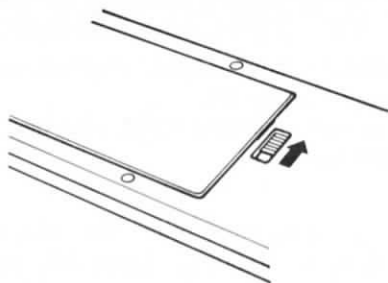


- (5) Remplacez les deux piles (Fig. 3).

Fig. 3

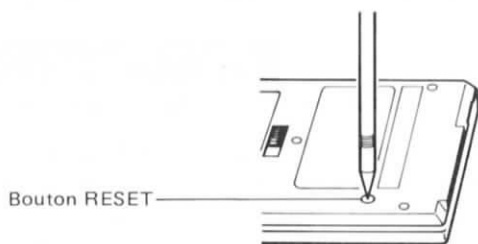


- (6) Remplacez la plaque de protection des piles en la faisant glisser dans le sens inverse de la flèche représentée à la figure 2.
- (7) Remplacez la plaque arrière de l'ordinateur et verrouillez-la correctement.



**Remarque:** Remplacez bien le loquet en position de verrouillage. Le PC-1350 ne peut fonctionner si le loquet n'est pas dans cette position. S'il est mis sous tension alors que le loquet n'est pas en position de verrouillage, remplacez le loquet verrouillage, mettez l'ordinateur hors tension puis à nouveau sous tension.

- (8) ① Mettez l'ordinateur sous tension en plaçant l'interrupteur en position ON et appuyez sur le bouton RESET pour réinitialiser l'ordinateur.



- ② Entrez les suivants: CALL &EF88  POKE &6EAF, 0

L'écran devrait se présenter comme ceci:

```

RUN  RUN MODE
      CALL &EF88
      POKE &6EAF, 0
      >
    
```

## Presentation

- ③ Si l'affichage à la page précédente n'est pas obtenu, exécutez à nouveau les étapes ① et ② .

CALL &EF88 initialise la fonction d'entrée-sortie serie. POKE &6EAF, 0 initialise la fonction d'affichage. En omettant ceci des problems pourraient survenir.

- (9) Si vous avez retiré la carte mémoire RAM à l'étape (3), remplacez-la. Pour installer cette carte, reportez-vous à la page 85.

(Avant de replacer la carte mémoire RAM, vous devez vérifier que les piles ont été correctement installées.)

### REMARQUE:

Si vous laissez des piles usées dans votre ordinateur, vous risquez de l'abîmer en raison des fuites qui peuvent se produire. Lorsqu'une pile est usée, retirez-la aussitôt.

**ATTENTION:** Gardez les piles hors de la portée des enfants.

## CHAPITRE 3

# UTILISATION DU PC-1350 COMME CALCULATRICE

Maintenant que vous vous êtes familiarisé avec la présentation et les différents composants de votre PC-1350, nous allons commencer à étudier les possibilités étonnantes de votre nouvel ordinateur.

Votre PC-1350 vous offre des possibilités de calcul très étendues auxquelles s'ajoutent des fonctions de programmation en BASIC (utiles pour les calculs plus complexes); c'est pourquoi on l'appelle souvent une calculatrice "intelligente", ce qui fait évidemment de vous un utilisateur "intelligent"!

(Avant d'utiliser le PC-1350, vérifiez que les piles sont correctement installées).

### DEMARRAGE

Pour mettre le PC-1350 en fonction, placez l'interrupteur sur la position ON. Pour l'utiliser comme une calculatrice, le PC-1350 doit être en mode RUN. Pour choisir ce mode, utilisez la touche **MODE** et faites apparaître l'indicateur RUN à la gauche de l'écran.

Lorsque l'ordinateur est en fonction, le guide-opérateur (>) s'affiche à l'écran.

### MISE HORS TENSION

Pour mettre le PC-1350 hors tension, placez l'interrupteur sur la position OFF.

Lorsque vous mettez l'ordinateur hors tension, vous effacez l'écran. Cependant, le PC-1350 se souvient de tous les programmes et le contenu des touches de réserve quand même il est mis hors tension. Tous ces éléments sont à nouveau en vigueur lorsque l'ordinateur est à nouveau mis sous tension.

Lorsque la commande CLOAD est exécutée, appuyer sur la touche **BRK** pour arrêter son exécution et placez l'interrupteur sur la position OFF.

### MISE HORS TENSION AUTOMATIQUE

Pour éviter l'usure des piles, le PC-1350 se met automatiquement hors tension lorsqu'aucune touche n'est utilisée pendant environ 11 minutes. (Remarque: le PC-1350 ne se met pas hors tension pendant l'exécution d'un programme.)

Pour remettre l'ordinateur en marche après une mise hors tension automatique, appuyez sur la touche **ON BRK**. L'ordinateur se retrouvera dans l'état dans lequel il était juste avant la mise hors tension automatique.



## QUELQUES CONSEILS UTILES

Tant que vous n'êtes pas habitué à votre nouvel ordinateur, vous allez faire des erreurs en entrant vos données. Par la suite, nous vous apprendrons plusieurs méthodes pour corriger ces erreurs. Pour l'instant, si un message d'erreur s'affiche, appuyez sur la touche rouge d'effacement ( **CLS** ) et reprenez vos données. Si l'ordinateur reste en attente – s'il ne répond pas du tout – appuyez sur le bouton ALL RESET (voir au Chapitre 2).

Le SYMBOLE  $>$  d'attente d'entrée au clavier vous indique que votre PC-1350 attend l'entrée des données. Lorsque vous entrez les données, le symbole disparaît et le CURSEUR ( **\_** ) se déplace vers la droite pour indiquer la position libre suivante sur l'écran.

Les flèches droite **▶** et gauche **◀** permettent de déplacer le curseur à l'intérieur d'une ligne.

Appuyez sur **ENTER** pour avertir le PC-1350 que vous avez fini d'entrer des données et qu'il peut exécuter les opérations indiquées. **VOUS DEVEZ APPUYER SUR LA TOUCHE ENTER A LA FIN DE CHAQUE LIGNE D'ENTREE SINON L'ORDINATEUR N'EFFECTUERA AUCUNE OPERATION SUR CES DONNEES.**

Lors de l'exécution d'opération numériques, l'entrée figure à la gauche de l'écran et le résultat à la droite de l'écran.

Lorsque la touche **SHIFT** est utilisée avec une autre touche (par exemple pour demander une racine carrée) appuyez sur **SHIFT** puis relâchez cette touche et appuyez sur l'autre touche. L'effet de la touche **SHIFT** ne dure que pendant l'utilisation de l'autre touche.

N'utilisez ni signes dollar ni virgules pour l'entrée de vos opérations. Ces caractères ont une signification spéciale en langage BASIC.

Dans cette brochure, nous utilisons  $\emptyset$  pour indiquer zéro de façon à faire la différence entre le chiffre ( $\emptyset$ ) et la lettre (O).

Pour vous aider à entrer correctement vos données au départ, nous allons vous montrer quelles touches utiliser pour frapper les calculs données en exemple. Lorsque la touche **SHIFT** est utilisée, nous représenterons à sa suite le caractère voulu. Par exemple, le fait d'appuyer sur **SHIFT** et **Q** permet d'obtenir le caractère "!" . Nous représenterons les touches de la manière suivantes: **SHIFT** **!** .

Essayez d'appuyer sur la touche **CLS** (effacement) après chaque opération (sauf si vous faites des calculs en série). Cette touche permet d'effacer l'écran et restaure la condition d'erreur. Elle n'efface rien de ce qui est enregistré dans la mémoire de l'ordinateur.

## CALCULS SIMPLES

Le PC-1350 exécute des calculs avec une précision de dix chiffres. Si vous ne l'avez pas encore fait, mettez votre ordinateur sous tension (ON) et placez-le en mode RUN (exécution). Maintenant, essayez d'exécuter ces exemples d'opérations arithmétiques simples.

Entrée

5 0 + 5 0 ENTER

1 0 0 - 5 0 ENTER

6 0 \* 1 0 ENTER

Ecran

50+50	100.
-------	------

50+50	100.
100-50	50.

100-50	50.
60*10	600.

L'écran du PC-1350 comporte 4 lignes de 24 caractères chacune. Les données entrées au clavier ou les résultats obtenus par calcul sont affichés à partir de la première ligne. Si les caractères à afficher dépassent 4 lignes, l'affichage se décale d'une ligne vers le haut (la première ligne affichée remontera vers le haut de l'écran et disparaîtra).

Entrée

3 0 0 / 5 ENTER

1 0 SHIFT ^ 2 ENTER

Ecran

60*10	600.
300/5	60.

300/5	60.
10^2	100.

2 \* SHIFT  $\pi$  ENTER

$10^2$	100.
$2 * \pi$	6.283185307

SHIFT  $\sqrt{\quad}$  6 4 ENTER

$2 * \pi$	6.283185307
$\sqrt{64}$	8.

4 E 3 ENTER

$\sqrt{64}$	8.
4E3	4000.

## RAPPEL DES DONNEES D'ENTREE

Vous pouvez éditer (modifier) vos dernières données d'entrée même si le PC-1350 a déjà affiché le résultat de votre opération. Pour l'édition, utilisez les touches  $\leftarrow$  et  $\rightarrow$ .

La flèche vers la gauche  $\leftarrow$  permet de placer le curseur après le dernier caractère.

La flèche vers la droite  $\rightarrow$  permet de placer le curseur "au-dessus" du premier caractère.

N'oubliez pas que ces touches permettent également de déplacer le curseur à l'intérieur une ligne. Ces touches sont très utiles pour l'édition des données d'entrée et évitent de refrapper la totalité d'une expression.

Vous allez vous familiariser avec l'emploi de ces touches dans les exemples ci-après. Maintenant, supposez que vous soyez un manager et exécutez les calculs que nous allons vous expliquer.

En tant que chef du personnel d'un service de marketing d'une société américaine, c'est à vous qu'il incombe de prévoir le congrès annuel des vendeurs. Vous attendez 300 personnes pour ce congrès de 3 jours. Pendant une partie de ce temps, les vendeurs se réuniront par petits groupes. Vous pensez que six serait un bon nombre. Combien y aura-t'il donc de groupes?

Entrée







Ecran

300/6	50.
-------	-----

Après réflexion, vous décidez que les groupes comportant un nombre de participants impair sont plus efficaces. Rappelez vos dernières données d'entrée à l'aide de la touche .

Entrée

Ecran

300/6_
--------

Pour calculer le nouveau nombre de groupes, vous devez remplacer le chiffre six par un nombre impair. Cinq semble mieux convenir que sept. Etant donné que vous avez rappelé vos données d'entrée à l'aide de la touche , le curseur se trouve à la fin des données affichées. Utilisez la touche  pour ramener le curseur d'une position vers la gauche.

Entrée

Ecran

300/6
-------

Remarquez que, une fois le curseur déplacé, il se présente sous la forme d'un rectangle clignotant. Chaque fois que vous placez le curseur "au-dessus" d'un caractère existant, il se présente sous forme d'un rectangle clignotant.

Frappez un 5 à la place du 6. Attention avant de remplacer des caractères! Lorsque vous frappez un nouveau caractère sur un caractère existant, le caractère initial est perdu à jamais! Vous ne pouvez rappeler une expression par dessus laquelle vous avez frappé d'autres données.

Entrée



Ecran

300/5_	60.
--------	-----

## Utilisation comme Calculatrice

Soixante semble un nombre de groupes raisonnable; par conséquent, vous décidez que chaque petit groupe aura cinq participants.

Le rappel des données est également utile pour vérifier vos dernières données d'entrée, notamment lorsque vos résultats semblent ne pas convenir. Par exemple, supposez que vous ayez effectué cette opération:

Entrée

**CLS** **3** **0** **/** **5** **ENTER**

Ecran

**30/5**

**6.**

Même un directeur fatigué et surmené comme vous l'êtes, réalise que six n'est pas un résultat possible alors que vous avez des centaines de personnes. Rappelez votre entrée à l'aide de la touche **▶**.

Entrée

**▶**

Ecran

**30/5**

Etant donné que vous avez rappelé vos données d'entrée à l'aide de la touche **▶** le curseur clignotant se trouve maintenant au-dessus du premier caractère de l'écran. Pour corriger ces données, vous voulez insérer un zéro supplémentaire. A l'aide de la touche **▶** amenez le curseur jusqu'à ce qu'il se trouve sur le zéro. Pour effectuer une INSertion, vous devez placer le curseur clignotant au-dessus du caractère avant lequel vous désirez placer les caractères à insérer.

Entrée

**▶**

Ecran

**30/5**

Utilisez la touche **INS** afin de faire de la place pour le caractère voulu.

Entrée

**INS**

Ecran

**3 0/5**

Lorsque vous appuyez sur la touche **INS**, tous les caractères à la droite du curseur se décalent d'un espace vers la droite et un espace entre crochets horizontaux est inséré. Le curseur clignotant se trouve maintenant à cet endroit, indiquant l'emplacement du prochain caractère frappé. Frappez votre zéro. Une fois les données d'entrée corrigées, affichez votre nouveau résultat.

Entrée

0

ENTER

Ecran

300/5

60.

Supposez d'autre part que vous ayez entré cette opération:

Entrée


CLS

3 0 0 0 / 5 ENTER

Ecran

3000/5

600.


Le résultat semble trop important. Si vous n'avez que 300 personnes à cette conférence, comment pouvez vous avoir 600 "petits groupes"? Rappelez vos données d'entrée à l'aide de la touche .

Entrée



Ecran

3000/5

Le curseur clignotant se trouve maintenant au-dessus du premier caractère de l'écran. Pour corriger ces données, il faut supprimer un des zéros. A l'aide de la touche , amenez le curseur sous le premier zéro (ou sous un autre zéro). Pour supprimer un caractère, placez le curseur "au-dessus" du caractère à supprimer.

Entrée



Ecran

3000/5

Maintenant, utilisez la touche DEL (suppression) pour éliminer un des zéros.

Entrée

DEL

Ecran

300/5

Lorsque vous appuyez sur la touche DEL, tous les caractères se décalent d'un espace vers la gauche. Le caractère "au-dessus" duquel se trouve le curseur est supprimé ainsi que la place qu'occupait ce caractère. Le curseur clignotant reste au même endroit indiquant la position d'entrée suivante. Etant donné que vous n'avez pas d'autres modifications à faire, terminez vos calculs.

Entrée

ENTER

Ecran

300/5

60.

(Remarque: La touche d'eSPaCement, alors que le curseur se trouve au-dessus d'un caractère, ce caractère est remplacé par un blanc alors que la touche DEL supprime le caractère et la position qu'il occupe).

## ERREURS

Il est indispensable de rappeler vos dernières données d'entrée lorsque le message ERROR s'affiche. Supposons que, par inadvertance vous ayez frappé ces données d'entrée dans votre PC-1350:

Entrée

CLS  
3 0 0 / / 5 ENTER

Ecran

300//5  
ERROR 1

Naturellement, vous êtes surpris lorsque ce message s'affiche! ERROR 1 est en fait la manière dont l'ordinateur vous dit: "Je ne sais pas ce que vous voulez que je fasse ici".

A ce stade, lorsque vous appuyez sur la touche ◀ le curseur clignotant apparaît à l'endroit où se trouve l'erreur.

Entrée

◀

Ecran

300//5

Pour corriger cette erreur, utilisez la touche DEL.

Entrée

DEL ENTER

Ecran

300/5

60.

Si, après avoir rappelé vos données d'entrées à la suite de l'affichage de ERROR 1, vous découvrez que vous avez oublié un caractère, utilisez la touche INS pour effectuer vos corrections.

Lorsque vous utilisez le PC-1350 comme calculatrice, c'est le message ERROR 1 (erreur de syntaxe) qui apparaîtra le plus souvent. Pour la liste complète des messages d'erreurs, reportez-vous à l'annexe A.

## CALCULS EN SERIE

Le PC-1350 vous permet d'utiliser les résultats d'une opération dans l'opération suivante.

Dans les prévisions de ce congrès, vous devez également établir un budget détaillé à soumettre à l'approbation du service financier. Vous savez que vous disposez de 150,00F pour chaque participant. Calculez votre budget total:

Entrée

CLS

3 0 0 \* 1 5 0 ENTER

Ecran

300\*150

45000.

Vous comptez utiliser 15% de cette somme pour la soirée finale. Lorsque vous effectuez des calculs en série, il n'est pas nécessaire de refrapper le résultat précédent. Dans ce cas, n'utilisez pas la touche CLS entre les entrées. Quel est donc le budget de cette soirée?

Entrée

\* . 1 5

Ecran

45000.\*.15\_

Remarquez que lorsque vous frappez la seconde opération (\* . 15), l'ordinateur affiche automatiquement le résultat de votre première opération à la gauche de l'écran et l'insère dans la nouvelle opération. Dans les calculs en série, l'entrée doit commencer par un opérateur. Comme toujours, il faut appuyer sur la touche **ENTER** pour indiquer la fin des données d'entrée:

**REMARQUE:** La touche  $\frac{\%}{\square}$  ne peut être utilisée dans les opérations. Cette touche ne peut être utilisée que comme caractère.

Exemple: 45000 \* 15 SHIFT % → ERROR 1

Entrée

ENTER

Ecran

6750.



Poursuivons avec l'affectation de votre budget. L'hôtel fournit les repas pour 4000 F.

Entrée

Ecran

6750. -4000\_

2750.

Les décorations coûteront 1225 F:

Entrée

Ecran

1525.

Enfin, vous devez affecter 2200 F pour l'animation de la soirée:

Entrée

Ecran

-675.

Il est évident qu'il va vous falloir changer soit vos plans soit l'affectation de vos ressources!

## NOMBRES NEGATIFS

Vous voulez que la soirée finale soit particulièrement réussie; par conséquent, vous décidez de grader ce que vous avez prévu et de dépenser plus d'argent. Cependant, vous vous demandez quel pourcentage du budget total sera pris par ce poste. Tout d'abord, changez le signe de la somme restante:

Entrée

Ecran

-675. \* -1\_

675.

Maintenant, vous ajoutez ce résultat à votre budget initial:

Entrée

Ecran

7425.

Le chiffre affiché divisé par 45000 vous donne le pourcentage du budget total qu'il représente:

Entrée

/ 4 5 0 0 0 ENTER

Ecran

0.165

C'est cela, vous décidez d'affecter 16,5% de votre budget à la soirée finale.

## CALCULS COMPOSES ET PARENTHESES

Dans les calculs ci-dessus, vous auriez pu combiner plusieurs opérations en une seule étape. Par exemple, vous auriez pu frapper ces deux opérations sur une même ligne:

$$675 + 6750 / 45000$$

Les calculs composés doivent cependant être entrés avec beaucoup de soin:

$675 + 6750 / 45000$  peut être interprété comme:

$$\frac{675 + 6750}{45000}$$

ou

$$675 + \frac{6750}{45000}$$

Lorsque vous effectuez des calculs composés, le PC-1350 obéit à des règles précises de calculs d'expression et de priorités d'opérateur (voir ANNEXE D). Vérifiez que vous obtenez bien les calculs voulus en utilisant des parenthèses afin de clarifier vos expressions:

$$(675 + 6750) / 45000 \quad \text{ou} \quad 675 + (6750 / 45000)$$

Pour voir les différences obtenues en fonction de l'emplacement des parenthèses, calculez ces deux exemples:

Entrée

( 6 7 5 + 6  
7 5 0 ) / 4  
5 0 0 0 ENTER  
6 7 5 + ( 6  
7 5 0 / 4 5 0 0  
0 ) ENTER

Ecran

0.165

675.15

## UTILISATION DE VARIABLES DANS LES CALCULS

Le PC-1350 peut enregistrer jusqu'à 26 variables numériques simples représentées par les touches alphabétiques A à Z. Si le concept des variables ne vous est pas familier, celles-ci sont expliquées de manière plus complète au Chapitre 4. Pour définir une variable, il faut utiliser une instruction d'affectation:

$$A = 5$$

$$B = -2$$

Vous pouvez également attribuer la valeur d'une variable (à droite du signe =) à une autre variable (à gauche du signe =):

$$C = A + 3$$

$$D = E$$

Une variable peut être utilisée à la place d'un nombre dans certaines opérations.

Maintenant que vous avez prévu votre soirée finale, il faut que vous en finissiez avec le budget de votre conférence. Vous voulez affecter le reste de votre budget par pourcentage également. Tout d'abord, vous devrez trouver combien il vous reste d'argent. Affectez une variable (R) qui sera le montant restant:

Entrée

R = 4 5 0 0 0  
 - 7 4 2 5  
 ENTER

Ecran

R=45000-7425\_  
 37575.

Lorsque vous appuyez sur la touche **ENTER** le PC-1350 exécute l'opération et affiche la nouvelle valeur de R. Vous pouvez afficher la valeur en cours d'une variable en entrant le caractère alphabétique sous lequel elle est enregistrée:

Entrée

R ENTER

Ecran

37575.

Vous pouvez alors utiliser cette variable pour effectuer des calculs. La valeur de (R) ne sera modifiée que lorsque vous lui attribuerez une nouvelle valeur.

Vous désirez affecter 60% de l'argent restant pour la location des chambres:

Entrée

R \* . 6 0  
 ENTER

Ecran

R\*.60\_  
 22545.

De même, vous voulez affecter 25% du budget restant pour les séminaires de formation:

Entrée

R \* . 2 5 ENTER

Ecran

9393.75

Les variables conservent la valeur qui leur a été attribuée même si l'ordinateur est mis hors tension soit par l'utilisateur, soit automatiquement. Les variables ne sont perdues que dans les cas suivants:

- \* Vous attribuez une nouvelle valeur à la même variable.
- \* Vous frappez CLEAR **ENTER** (attention, il ne s'agit pas de la touche **CLS**).
- \* Vous effacez la mémoire de l'ordinateur à l'aide du bouton ALL RESET.
- \* Vous changez les piles.

Il existe certaines limites à l'affectation de variables, et certaines procédures de programmation peuvent les modifier. Pour tous renseignements sur l'affectation des variables, se reporter au Chapitre 4. Pour tous renseignements sur l'emploi des variables en programmation, se reporter au Chapitre 5.

## CALCULS EN CHAÎNE

Le PC-1350 vous permet non seulement de combiner plusieurs opérations en une seule mais également d'exécuter plusieurs opérations les une à la suite des autres — sans avoir à appuyer sur **ENTER** pour passer à la suivante. Dans ce cas, vous devez séparer les équations par des virgules. Seul le résultat du calcul final s'affiche. (Rappelez-vous également que la longueur maximale de la ligne est de 80 caractères, y compris **ENTER**.)

Vous vous demandez de quelle somme vous auriez pu disposer pour les chambres si vous aviez gardé votre affectation initiale de 15% pour la réception finale:

Entrée

R = . 8 5 \* 4 5 0 0 0 , R \* . 6 0 ENTER

Ecran

R=.85\*45000, R\*.60

Bien que l'ordinateur effectue toutes les opérations à la chaîne, il n'affiche que le résultat final:

Entrée

ENTER

Ecran

R=.85\*45000, R\*.60

22950.

Pour trouver la valeur de R utilisée dans ce calcul, entrez R:

Entrée

Ecran

R ENTER

38250.

## NOTATION SCIENTIFIQUE

Les personnes qui ont à utiliser des nombres très grands ou très petits se servent souvent d'un format spécial appelé notation exponentielle ou scientifique. En notation scientifique, les nombres se divisent en deux parties.

La première partie se compose d'un nombre décimal normal compris entre 1 et 10. La seconde partie indique la grandeur de ce nombre en puissance de 10.

Comme vous le savez, en notation décimale, le premier chiffre à la gauche du point ou de la virgule décimale indique les unités, le second les dizaines, le troisième, les centaines et le quatrième le nombre de milles. Ce sont en fait des puissances de 10:

$$10^0 = 1, 10^1 = 10, 10^2 = 100, 10^3 = 1000, \text{ etc.}$$

La notation scientifique divise les nombres décimaux en deux parties: la première partie indique ce qu'est ce nombre, et la seconde sa distance, à droite ou à gauche, par rapport à la virgule ou au point décimal. Par exemple:

1234 devient 1.234 fois  $10^3$  (troisième position à droite)  
 654321 devient 6.54321 fois  $10^5$  (cinquième position à droite)  
 .000125 devient 1.25 fois  $10^{-4}$  (4 positions à gauche)

La notation scientifique permet d'aller plus vite. En effet, il faudrait beaucoup de temps pour écrire 1.0 fois  $10^{87}$  — un 1 et 87 zéros! En notation scientifique, ce nombre se présente comme ceci:

$$1.0 \times 10^{87} \quad \text{ou} \quad 1.0E 87$$

Le PC-1350 utilise la notation scientifique chaque fois qu'un nombre devient trop grand pour être affiché en notation décimale. Cet ordinateur utilise la lettre majuscule E pour indiquer "à la puissance fois 10":

12345678900000 est affiché sous la forme 1.23456789 E 12  
 .000000000001 est affiché sous la forme 1. E - 12

Les personnes qui ne connaissent pas bien ce type de notation devraient prendre maintenant le temps d'entrer des nombres très grands et très petits et de voir la façon dont ils s'affichent.



(Exemple) Utilisez le résultat (6.25) de l'opération  $50 \div 8$ , pour calculer  $12 \times 5 \div 6.25 + 24 \times 3 \div 6.25 = :$

Entrée

50  $\div$  8 **ENTER**

12  $\times$  5  $\div$   $\uparrow$  (ou  $\downarrow$ )

$\pm$  24  $\times$  3  $\div$   $\downarrow$  (ou  $\uparrow$ )

**ENTER**

**CLS**  $\downarrow$

Ecran

50/8 6.25

Dernière réponse  $\uparrow$

12\*5/6.25\_

$\uparrow$  Dernière réponse rappelée

12\*5/6.25+24\*3/6.25\_

$\uparrow$  Dernière réponse rappelée

**21.12**

21.12\_

La dernière réponse est remplacée par le résultat du calcul précédent obtenu manuellement en utilisant la touche **ENTER**.

Comme le montre cet exemple, la dernière réponse peut être rappelée à tout moment et en n'importe quel emplacement, mais elle sera remplacée par la dernière nouvelle réponse résultant de la dernière opération.

La dernière réponse n'est pas effacée lorsque vous appuyez sur **CLS** ou sur une autre touche.

- La dernière réponse ne peut pas être rappelée lorsque l'ordinateur n'est pas en mode RUN, lorsque l'exécution d'un programme est temporairement suspendue ou lorsque le mode Traçage est sélectionné.

## LONGUEUR DES FORMULES

La longueur des formules que vous pouvez entrer dans votre ordinateur est limitée. Avec le PC-1350, il est possible d'utiliser jusqu'à 79 caractères pour entrer une seule formule (sans compter la touche **ENTER**). Si vous tentez d'entrer un 80ème caractère, le curseur ( ) commencera à clignoter à l'endroit de ce caractère, indiquant qu'il n'est pas valide.

## CALCULS SCIENTIFIQUES

Le PC-1350 comporte les fonctions principales indiquées ci-dessous. Il convient de remarquer que la notation des fonctions en langages BASIC peut différer des notations mathématiques classiques.

Fonctions	Notations classiques	Touches utilisées	Remarques
Fonctions trigonométriques	sin cos tan	SIN COS TAN	
Fonctions trigonométriques inversées	$\sin^{-1}$ $\cos^{-1}$ $\tan^{-1}$	ASN ACS ATN	
Logarithme decimal	log	LOG	$\text{Log}_{10} x$ (logarithme à base 10).
Logarithme neperien	ln	LN	$\log_e x$ (logarithme à base e)
Fonction exponentielle	$e^x$	EXP	$e \approx 2.718281828$
Élévation à une puissance		$\wedge$	$A^B$ for $A \wedge B$
Racine carrée	$\sqrt{\quad}$	$\sqrt{\quad}$ ou SQR	
Conversions de degrés (décimaux) en degrés (degrés, minutes, secondes)		DMS	Conversion d'angle (vous ne devez pas omettre le 0 et écrire DEG. 5 à la place de DEG 0.5)
Conversion de degrés (degrés, minutes, secondes) en degrés (décimaux)		DEG	
Entier		INT	INT(x) permet d'obtenir l'entier le plus grand inférieur ou égal à x.
Valeur absolue	X	ABS	ABS(x) permet d'obtenir la valeur absolue de x.
Signe		SGN	Donne 1 lorsque $x > 0$ , -1 lorsque $x < 0$ et 0 lorsque $x = 0$ pour SGN(x).
Pi	$\pi$	$\pi$ ou PI	$\pi \approx 3.141592654$
Notation hexadecimal →	Notation décimale	&	Convertit X en un nombre à base 10 pour &X.



Unité d'angle	Commande	Description
Degré	DEGREE	Représente un angle droit comme $90 [^\circ]$
Radian	RADIAN	Représente un angle droit comme $\pi/2$ rad
Grade	GRAD	Représente un angle droit comme $100 [g]$

Ces instructions permettent de spécifier des unités d'angle dans un programme. Pour vous exercer, utilisez ces instructions pour spécifier des unités d'angle dans les exemples de calculs suivants.

(Exemple)  $\sin 30^\circ =$

(Opération) DEGREE **ENTER** (Spécifie "degré" comme unité d'angle)

SIN 30 **ENTER** | **0.5** |

(Exemple)  $\tan \frac{\pi}{4} =$

(Opération) RADIAN **ENTER** (Spécifie "radian" comme unité d'angle).

TAN (PI/4) **ENTER** | **1.** |

(Exemple)  $\cos^{-1}(-0,5) =$

(Opération) DEGREE **ENTER** (Spécifie "degré" comme unité d'angle).

ACS -0.5 **ENTER** | **120.** |

(Exemple)  $\log 5 + \ln 5 =$

(Opération) LOG 5 + LN 5 **ENTER** | **2.308407917** |

(Exemple)  $e^{2+3} =$

(Opération) EXP (2 + 3) **ENTER** | **148.4131591** |

(Exemple)  $\sqrt{4^3 + 6^4} =$

(Opération)  $\sqrt{(4 \wedge 3 + 6 \wedge 4)}$  **ENTER** | **36.87817783** |

(Exemple) Convertir 30 deg. 30 mm de notation 60 en notation décimale.

(Opération) DEG 30.30 **ENTER** | **30.5** |  
(30.5 degré)

(Exemple) Convertir 30.755 deg. de notation décimale en notation 60.

(Opération) DMS 30.755 **ENTER** | **30.4518** |  
(30 deg. 45 min. 18 sec.)

(Exemple) Convertir CF8 dans son équivalent décimal.

(Opération) &CF8 **ENTER** | **3320.** |

## PRIORITE DANS LES CALCULS MANUELS

Vous pouvez frapper vos formules dans l'ordre dans lequel elles sont écrites, y compris parenthèses ou fonctions. L'ordre de priorité des calculs et le traitement des résultats intermédiaires sera pris en charge par l'ordinateur.

L'ordre interne de priorité des calculs est le suivant:

- 1) Rappel des variable ou  $\pi$
- 2) Fonction (sin, cos, etc)
- 3) Elévation à la puissance ( $\wedge$ )
- 4) Signes (+, -)
- 5) Multiplication ou division (\*, /)
- 6) Addition ou soustraction (+, -)
- 7) Comparaison de grandeurs (>, >=, <, <=, <>, =)
- 8) AND, OR (ET, OU) Logiques.

Remarques: \*

- \* Si des parenthèses sont utilisées dans une formule, l'opération indiquée à l'intérieur des parenthèses a la priorité la plus élevée.
- \* Les fonctions composées sont traitées de droite à gauche (sin cos<sup>-1</sup> 0.6).
- \* Les puissance chaînées (3<sup>4</sup><sup>2</sup> ou 3 <sup>4</sup> 2) sont traitées de droite à gauche.
- \* Pour les éléments 3) et 4), la dernière entrée a la priorité la plus élevée.

Exemple:  $-2 \wedge 4 \rightarrow -(2^4)$   
 $3 \wedge -2 \rightarrow 3^{-2}$

## IMPRESSION DES CALCULS MANUELS

Les résultats des calculs manuels sont en général affichés à l'écran. Cependant, les étapes de calculs et les résultats peuvent être imprimés lorsque l'imprimante en option (CE-126P) est connectée et mise en fonction et que les touches **SHIFT** et **P $\leftrightarrow$ NP** sont utilisées (mode impression).

Si aucune impression n'est requise, mettez l'imprimante hors fonction ou appuyez à nouveau sur les touches **SHIFT** et **P $\leftrightarrow$ NP** (mode sans impression).

### ATTENTION

#### ERREUR DE CALCUL

Voici les types d'erreurs qui surviennent avec les calculatrices ordinaires, les ordinateurs de poches et les ordinateurs personnels.

(1) Erreur due au traitement des chiffres les moins significatifs.

En général, le nombre maximum de chiffres pouvant être calculé sur un ordinateur est fixe. Par exemple,  $4 \div 3$  donne comme résultat 1.3333333333... Avec un ordinateur prenant en compte 8 chiffres au maximum, les 8 chiffres sont le chiffres significatifs. Les autres chiffres moins significatifs sont soit tronqués soit arrondis.

(Exemple) Ordinateur prenant en compte 10 chiffres significatifs

$$4/3 \quad \boxed{\text{ENTER}} \rightarrow \overbrace{1.3333333333}^{10 \text{ chiffres significatifs}} \dots$$

Tronqué, arrondi

Par conséquent, le résultat calculé diffère de la vraie valeur en raison du montant tronqué ou arrondi. (C'est cette différence qui constitue l'erreur.)

Dans le PC-1350, un résultat calculé comporte 12 chiffres. Ce résultat est arrondi et traité spécialement pour minimiser l'erreur dans la valeur affichée.

(Exemple)  $4 \div 3 \times 3$

①	$4/3 * 3$	$\boxed{\text{ENTER}}$	$\rightarrow 4.$	Calcul en série
②	$4/3$	$\boxed{\text{ENTER}}$	$\rightarrow 1.3333333333$	} Calcul séparés
	$* 3$	$\boxed{\text{ENTER}}$	$\rightarrow 3.9999999999$	

\* Dans le cas de calculs en série, le résultat de  $4 \div 3$  est obtenu dans l'ordinateur avec 12 chiffres significatifs puis il est utilisé pour le calcul suivant et ensuite arrondi.

Dans le cas de calculs séparés, c'est la valeur affichée (10 chiffres) qui est utilisée pour les calculs.

(2) Erreur due à des calculs successifs

Bien que l'unité exécute un certain nombre de traitement pour minimiser l'erreur de la valeur affichée, les erreurs s'additionnent lorsqu'une série de calculs de fonction sont exécutées et apparaissent en tant que telles dans le résultat affiché.

(Exemple)  $60^6 =$

$$60 \wedge 6 \quad \boxed{\text{ENTER}} \rightarrow 4.665599999E10$$

Bien que  $60^6$  donne  $4,6656 \times 10^{10}$ , l'ordinateur calcule ( $y^x$ ) en fonction de l'expression d'élévation à la puissance suivante.

$$y^x = 10^{x \times \log y}$$

Autrement dit,  $60^6$  est calculé comme  $10^{6 \times \log 60}$ .

Puisque la fonction elle-même emploie un algorithme d'approximation, l'erreur en ordinateur est légèrement supérieure à celle des calculs normaux (sans fonction). Par conséquent, lors de l'exécution d'une suite de calculs de fonction (tels que des calculs d'élévation à la puissance), les erreurs s'ajoutent et peuvent se voir à l'écran.

## CHAPITRE 4 CONCEPTS ET TERMES BASIC

Dans ce chapitre, nous allons étudier certains concepts et termes du langage BASIC.

### CONSTANTES ALPHANUMERIQUES (CHAINE DE CARACTERES)

Le PC-1350 de SHARP n'utilise pas seulement les nombres mais également les lettres et les symboles spéciaux dans de nombreuses combinaisons variées. Ces lettres, chiffres et symboles spéciaux sont appelés caractères. Voici les caractères qu'utilisent le PC-1350:

```

1 2 3 4 5 6 7 8 9 0
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
a b c d e f g h i j k l m n o p q r s t u v w x y z
! " # $ % & ( ) * + , - . / : ; < = > ? @ √ π ^

```

En langage BASIC, on appelle constante alphanumérique un ensemble de caractères. Pour que le PC-1350 puisse faire la différence entre une chaîne de caractères (ou constante alphanumérique) et d'autres parties d'un programme comme par exemple les verbes ou les noms de variables, les caractères doivent être écrits entre guillemets (""). Voici des exemples de constantes alphanumériques:

```

"HELLO"
"Au revoir"
"PC-1350 SHARP"

```

Les constantes alphanumériques ci-dessous ne sont pas valides:

```

"ORDINATEUR" Pas de guillemets de fin
"N"EST-CE PAS" Il n'est pas possible d'utiliser de guillemets à l'intérieur
d'une suite de caractères

```

### NOMBRES HEXADECIMAUX

Le système décimal n'est qu'un des nombreux systèmes qui permettent de représenter les nombres. Un autre système qui a pris de l'importance avec les ordinateurs est le système hexadécimal. Ce système est basé sur 16 au lieu de 10. Pour écrire des nombres en hexadécimal, vous utilisez les chiffres courants 0 à 9 et 6 autres "chiffres" qui sont A, B, C, D, E et F. Ces "chiffres" correspondent à 10, 11, 12, 13, 14 et 15. Lorsque vous voulez que le PC-1350 traite un nombre comme étant hexadécimal, placez le caractère "&" devant le chiffre.

```

&A    = 10
&10   = 16
&100  = 256
&FFFF = 65535

```

## VARIABLES

Les ordinateurs sont constitués par de multiples zones de mémoire minuscules appelées octets. Chaque octet peut être considéré comme un seul caractère. Par exemple, le mot octet requière 5 octets de mémoire car il comporte 5 caractères. Pour savoir de combien d'octets vous disposez, il suffit de frapper MEM **ENTER**. Le nombre affiché correspond au nombre d'octets disponibles pour l'écriture des programmes. Cette technique marche très bien pour les mots. Mais elle est tout à fait inefficace lorsque vous tentez d'enregistrer des chiffres. C'est pourquoi les chiffres peuvent être enregistrés sous forme de codes. Grâce à cette technique de codage, 8 codage, 8 octets suffisent pour enregistrer dans votre ordinateur des chiffres très élevés. Le chiffre le plus grand qui puisse être enregistré est +9.999999999E+99.

Le chiffre le plus petit est + 1. E-99. Vous avez ainsi une vaste gamme de chiffres à votre portée. Cependant, si le résultat d'un calcul dépasse ces limites, l'ordinateur vous l'indiquera en affichant un message d'erreur. Pour le message d'erreur, reportez-vous à l'Annexe A. Pour l'afficher dès maintenant, frappez :

9 E 99 \*9 **ENTER**

Pour remettre l'ordinateur en état de marche après une erreur, il suffit d'appuyer sur la touche **CLS**. Mais comment faire pour enregistrer en mémoire toutes ces informations? C'est vraiment facile. L'ordinateur aime utiliser des noms pour les différentes parties des données. Enregistrons le nombre 556 dans l'ordinateur. Vous pouvez donner à ce nombre le nom que vous désirez, mais pour cet exercice, nous allons utiliser la lettre R. L'instruction LET permet de demander à l'ordinateur d'attribuer une valeur à un nom de variable, mais uniquement dans une instruction de programme. Maintenant la commande LET n'est pas nécessaire et nous ne l'utiliserons pas. Frappez R = 556 et appuyez sur la touche **ENTER**. L'ordinateur a maintenant associé la valeur 556 à la lettre R. Les lettres utilisées pour enregistrer des informations sont appelées des variables. Pour voir le contenu de la variable R, appuyez sur la touche **CLS**, sur la touche R puis la touche **ENTER**. L'ordinateur répond en affichant la valeur 556, à droite sur votre écran. Cette possibilité peut être très utile lorsque vous écrivez des programmes et des formules.

Maintenant, utilisons la variable R dans une formule simple. Dans cette formule, la variable R indique le rayon d'un cercle dont nous désirons trouver la surface. La formule pour la surface d'un cercle est:  $A = \pi * R^2$ . Frappez R **SHIFT** **^** 2 **\*** **SHIFT** **π** **ENTER**. Le résultat est 971179,3866. Cette technique d'utilisation des variables dans les équations sera plus compréhensible lorsque nous passerons à l'écriture des programmes.

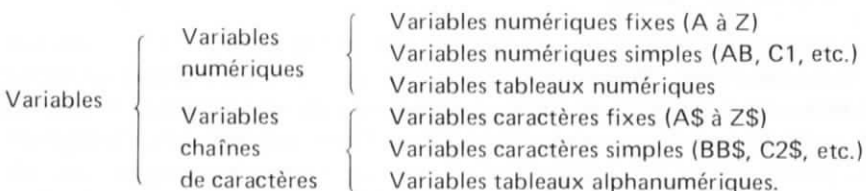
Jusqu'à présent, nous avons parlé des variables numériques. Mais comment enregistrer en mémoire des caractères alphabétiques! L'idée est la même mais pour que l'ordinateur fasse la différence entre les deux types de variables, on ajoute \$ au nom de la variable. Par exemple, enregistrons le mot OCTET dans la variable B\$. Remarquez le signe \$ après le B!

Celui-ci indique à l'ordinateur que le contenu de la lettre B est alphabétique ou alphanumérique.

Pour l'illustrer frappez B **SHIFT** **\$** = **SHIFT** **"** OCTET **SHIFT** **"** **ENTER** . La valeur OCTET est maintenant enregistrée dans la variable B\$. Pour vous en assurer, appuyez sur la touche **CLS** et frappez B **SHIFT** **\$** **ENTER** . L'écran affiche OCTET. Cette fois, l'affichage se trouve sur le côté gauche de l'écran et non plus à droite.

**Remarque:** Le contenu des chaînes de caractères ou variables alphanumériques est affiché à partir des positions de gauche de la première ligne.

Les variables traitées par le PC-1350 de SHARP se divisent en deux catégories:



## VARIABLES FIXES

La première catégorie, les variables fixes sont toujours utilisées par l'ordinateur pour l'enregistrement des données. Elles peuvent être considérées comme des emplacements pré-affectés aux variables. Autrement dit, quelle que soit la quantité de mémoire qu'utilise votre programme, vous aurez toujours au moins 26 variables parmi lesquelles choisir pour enregistrer vos données. Ces données peuvent être de deux types: NUMERIQUE ou ALPHANUMERIQUE. Ces emplacements de mémoire fixes comportent 8 octets et ne peuvent être utilisés que pour un type de données à la fois. Pour vous le montrer, frappez l'exemple suivant:

```
A = 123 ENTER
A$ ENTER
```

Vous obtenez le message:

```
ERROR 9
```

Ce message indique que vous avez placé des données numériques dans la zone de mémoire appelée A et que vous avez ensuite demandé à l'ordinateur de vous montrer la CHAÎNE de caractères enregistrée dans cette zone de mémoire. L'ordinateur ne sait plus où il en est et il vous indique une condition d'erreur. Appuyez sur la touche **[CLS]** pour restaurer la condition d'erreur. Prenez maintenant l'exemple suivant:

```

A$ = "ABC"  [ENTER]
A  [ENTER]

```

Là encore, l'ordinateur est perdu et affiche le message ERROR 9. Regardez la figure ci-dessous: le nom de variable A occupe en mémoire la même zone que le nom de variable A\$, le nom de variable B occupe la même position de mémoire que le nom de variable B\$, et la même chose peut être dite pour toutes les lettres de l'alphabet:

Figure:

A = A\$ = A(1)	= A\$(1)
B = B\$ = A(2)	= A\$(2)
C = C\$ = A(3)	= A\$(3)
D = D\$ = A(4)	= A\$(4)
E = E\$ = A(5)	= A\$(5)
F = F\$ = A(6)	= A\$(6)
G = G\$ = A(7)	= A\$(7)
H = H\$ = A(8)	= A\$(8)
I = I\$ = A(9)	= A\$(9)
J = J\$ = A(10)	= A\$(10)
K = K\$ = A(11)	= A\$(11)
L = L\$ = A(12)	= A\$(12)
M = M\$ = A(13)	= A\$(13)
N = N\$ = A(14)	= A\$(14)
O = O\$ = A(15)	= A\$(15)
P = P\$ = A(16)	= A\$(16)
Q = Q\$ = A(17)	= A\$(17)
R = R\$ = A(18)	= A\$(18)
S = S\$ = A(19)	= A\$(19)
T = T\$ = A(20)	= A\$(20)
U = U\$ = A(21)	= A\$(21)
V = V\$ = A(22)	= A\$(22)
W = W\$ = A(23)	= A\$(23)
X = X\$ = A(24)	= A\$(24)
Y = Y\$ = A(25)	= A\$(25)
Z = Z\$ = A(26)	= A\$(26)



## VARIABLES SIMPLES

Les noms de variables simples sont indiqués par deux caractères alphanumériques (ou plus), par exemple AA ou B1. Au contraire des variables fixes, les variables simples n'ont pas d'emplacement réservé en mémoire. Une zone est automatiquement réservée à une variable simple (dans la zone mémoire des programmes et des données) lorsqu'elle est utilisée pour la première fois.

Etant donné que les variables numériques simples et les variables alphanumériques simples occupent des zones de mémoire distinctes même si elles ont le même nom, les variables telles que AB et AB\$ par exemple peuvent être utilisées en même temps.

Les noms de variables simples peuvent comporter n'importe quel caractère alphanumérique mais le premier caractère de ce nom doit toujours être alphabétique. (Seuls les caractères alphabétiques majuscules sont reconnus). Si un nom de variable comporte plus de deux caractères, seuls les deux premiers caractères sont pris en compte.

- Remarque:**
- Les noms de fonctions ou d'instructions BASIC utilisées par le PC-1350 ne peuvent être utilisés comme noms de variable. (par exemple: PI, IF, TO, ON, SIN, etc.)
  - Chaque variable caractère simple peut comporter un maximum de 16 caractères ou symboles.

## VARIABLES TABLEAUX

Dans certains cas, il est utile que les nombres se présentent sous forme de groupes organisés, par exemple une liste de scores ou un tableau de taxes. En BASIC, ces groupes sont appelés des tableaux. Un tableau peut avoir une dimension (c'est le cas d'une liste), ou deux dimensions (c'est le cas des tableaux).

Pour définir un tableau, il faut utiliser l'instruction DIM (abréviation de dimension). Les tableaux doivent toujours être "déclarés" (définis) avant d'être utilisés (non pas comme les variables simples que nous avons utilisées). La forme de l'instruction DIM est la suivante:

DIM nom-variable-numérique (dimension)

Où:

Nom-variable -numérique est un nom de variable conforme aux règles classiques des noms de variables numériques que nous venons de voir.

Dimension est le nombre d'emplacements de mémoire; ce doit être un nombre compris entre 0 et 255. Notez que lorsque vous spécifiez un nombre pour la taille de votre tableau, vous obtenez une position de plus que le nombre spécifié.

Exemples d'instructions DIM numériques valides:

DIM X (5) DIM AA (24) DIM Q5 (0)
--

La première instruction crée un tableau X ayant 6 positions de mémoire. La seconde instruction crée un tableau AA ayant 25 positions de mémoire. La troisième instruction crée un tableau ayant une position de mémoire, ce qui est plutôt absurde (pour les nombres au moins) puisque cela revient au même que de déclarer une variable numérique simple.

Il est important de savoir qu'une variable-tableau X et une variable X sont des éléments bien distincts pour le PC-1350. Le premier X indique une série d'emplacements de mémoire numérique et le second un emplacement unique et distinct.

Maintenant que vous savez comment créer des tableaux, vous vous demandez peut-être comment fait-on pour indiquer chaque emplacement de mémoire. Etant donné qu'un groupe a un nom unique, pour se référer à une position distincte (appelée "élément"), il faut faire suivre le nom de groupe d'un chiffre entre parenthèses. Ce chiffre est appelé "indice". Ainsi, par exemple, pour enregistrer le chiffre 8 dans le cinquième élément de votre tableaux X (défini précédemment), il faut écrire:

X (4) = 8

Pourquoi le chiffre 4? Rappelez vous que la numérotation des éléments commence à zéro et se poursuit jusqu'au chiffre indiqué comme taille dans l'instruction DIM.

Les tableaux sont véritablement des outils très puissants car il est possible d'utiliser une expression ou un nom de variable comme indice.

Pour déclarer un tableau alphanumérique, il faut utiliser une instruction DIM légèrement différente:

DIM nom-variable-alphanumérique (dimension) \* longueur

Où:

Nom-variable-alphanumérique est un nom de variable conforme aux règles des variables alphanumériques classiques que nous avons étudiés précédemment.

Dimension indique le nombre de positions de mémoire; ce nombre doit être compris entre 0 et 255. Remarquez que lorsque vous spécifiez un nombre, vous obtenez une position de plus que le nombre spécifié.

\*Longueur est facultative. Si ce paramètre est utilisé, il indique la longueur des chaînes de caractères que comporte le tableau. La longueur est comprise entre 1 et 80. Si ce paramètre n'est pas utilisé, les chaînes ont une longueur par défaut de 16 caractères.

Exemples de définition de tableau alphanumérique valide:

```
DIM X$(4)
DIM NM$(10)*10
DIM IN$(1)*80
DIM R$(0)*26
```

Le premier exemple crée un tableau de cinq chaînes de caractère capables chacune d'enregistrer 16 caractères. La seconde instruction DIM définit un tableau NM comportant 11 chaînes de 10 caractères chacune.

L'indication de la longueur des chaînes, lorsque celle-ci est inférieure à la valeur par défaut, permet de ne pas gaspiller de place en mémoire. Le troisième exemple définit un tableau de deux éléments ayant des chaînes de 80 caractères et le dernier exemple définit une chaîne unique de 26 caractères.

En plus des tableaux simples que nous venons de voir, le PC-1350 accepte des tableaux à "deux dimensions". Par analogie, un tableau à une dimension est une liste de données disposées sous la forme d'une colonne unique. Un tableau à deux dimension est un véritable tableau comportant des lignes et des colonnes.

Pour définir un tableau à deux dimensions, il faut utiliser l'instruction:

DIM nom-variable-numérique (lignes, colonnes)

ou

DIM nom-variable-alphenumérique (lignes, colonnes) \* longueur

Où

Lignes indique le nombre de lignes que comporte le tableau. Ce doit être un nombre compris entre 0 et 255. Notez que vous obtenez une ligne de plus que le nombre indiqué.

Colonnes indique le nombre de colonnes que comporte le tableau. Ce nombre doit être compris entre 0 et 255. Notez que vous obtenez une colonne de plus que le nombre indiqué.

Le tableau suivant représente les emplacements de mémoire réservés à la suite de l'instruction DIM T (2, 3) ainsi que les indices (composés maintenant de deux chiffres) correspondant à chaque emplacement de mémoire:

	Colonne 0	Colonne 1	Colonne 2	Colonne 3
Ligne 0	T (0, 0)	T (0, 1)	T (0, 2)	T (0, 3)
Ligne 1	T (1, 0)	T (1, 1)	T (1, 2)	T (1, 3)
Ligne 2	T (2, 0)	T (2, 1)	T (2, 2)	T (2, 3)

**Remarque:** Des tableaux à deux dimensions peuvent rapidement occuper pratiquement toute la mémoire. Par exemple, un tableau de 25-lignes et 35 colonnes utilise 875 positions de mémoire!

Les tableaux sont des outils de programmation très puissants.

Le tableau ci-dessous indique le nombre d'octets utilisé pour définir chaque variable et le nombre d'octets utilisés par chaque instruction du programme.

Variable	Nom de variable	Données	
Variable numérique	7 octets	8 octets	
Variables	7 octets	Variable tableau	Nombre spécifié
		Variable simple (2 caractères)	16 octets

\* Par exemple, si DIM Z\$(2, 3) \* 10 est spécifié, 12 variables capables d'enregistrer chacune 10 caractères sont réservées. Il faut pour cela 7 octets (nom de variables) + 10 octets (nombre de caractères) x 12 = 127 octets.

Élément	Nombre de lignes	Instruction & fonction	Divers, <input type="button" value="ENTER"/>
Nombre d'octets utilisés	3 octets	1 octet	1 octet

## VARIABLES DE FORMATS A( )

Une zone de données de la mémoire de l'ordinateur réservée pour les variables fixes peut également être utilisée pour définir des variables indicées ayant la même forme que les variables tableaux.

Il y a 26 noms de variables fixes possibles: A à Z (A\$ à Z\$). Chacun de ces noms peut être indicé au moyen des chiffres 1 à 26, par exemple A(1)–A(26) ou A\$(1)–A\$(26). Cela signifie que la variable A(1) peut être utilisée à la place de la variable A, A(2) à la place de B, A(3) à la place de C, etc.

Cependant, si un tableau appelé A ou A\$ a déjà été défini par l'instruction DIM, il n'est pas possible de définir des variables indicées appelées A. Par exemple, si un tableau A est défini par DIM A(5), les positions de mémoires de A(0) à A(5) sont réservées dans la zone des programmes et des données. Si par conséquent vous spécifiez une variable A(2), elle ne se réfère pas à la variable fixe B mais à la variable tableau A(2) définie dans la zone des programmes et des données. Si vous spécifiez A(9), vous provoquerez une erreur étant donné que A(9) est en dehors des limites fixées par l'instruction DIM A(5).

Inversement, si des variables indicées sont déjà définies sous la forme A( ), il n'est pas possible de définir de tableaux A ou A\$ avec l'instruction DIM tant que la définition des variables indicées n'a pas été effacée par l'instruction CLEAR.

### \* Utilisation d'indices au-delà de 26:

Si des indices supérieurs à 26 sont utilisés pour des variables indicées A( ) alors qu'aucun tableau A n'a été défini par une instruction DIM, les positions correspondantes de la zone des programmes et des données sont réservées pour ces variables A( ). Par exemple, si vous exécutez A(35) = 5, les emplacements pour les variables A(27) à A(35) seront réservés dans la zone des programmes et des données.

Les variables indicées au-delà de 26 sont traitées comme des variables tableaux mais elles sont soumises aux règles suivantes:

- (1) Les emplacements d'un tableau du même nom doivent être contigus dans la zone des programmes et des données. Sinon, il se produira une erreur.

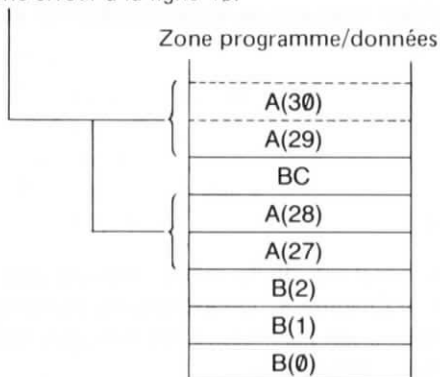
10 DIM B(2)

20 A(28) = 5

30 BC = 12

40 A(30) = 9

Si ce programme est exécuté, le tableau appelé "A" n'est pas défini dans des segments consécutifs de la zone des programmes et des données et il s'en suivra une erreur à la ligne 40.



- (2) Il n'est pas possible de définir en même temps des variables tableaux numériques et des variables tableaux caractères ayant le même indice. Par exemple, A(30) et A\$(30) ne peuvent être définis en même temps puisqu'elles occupent la même position dans la zone des programmes et des données.
- (3) Il n'est pas possible de définir des tableaux à deux dimensions pas plus qu'il n'est possible de spécifier la longueur des chaînes de caractères à insérer dans des variables tableaux. Par exemple, la longueur d'une chaîne de caractère pouvant être insérée dans la variable tableau caractère A\$( ) est limitée à 7 caractères ou moins.
- (4) Il n'est pas possible de définir des variables ayant pour indice zéro (0). Si A(0) ou A\$(0) est défini, une erreur se produira.
- (5) Lorsque des variables A ou A\$ ayant pour indice 27 ou un chiffre supérieur sont utilisées la première fois, 7 octets sont réservés pour le nom de la variable et 8 octets pour chaque variable.

## EXPRESSIONS

Une expression est une combinaison de variables, de constantes et d'opérateurs qui peuvent être calculés pour obtenir une valeur unique. Les calculs que vous avez entrés au Chapitre 3 étaient des exemples d'expressions. Les expressions font partie des programmes BASIC. Par exemple, une expression peut être une formule qui calcule la réponse d'une équation, un test pour déterminer la relation entre deux quantités ou un moyen de mettre en forme une série de chaîne de caractères.

## OPERATEURS NUMERIQUES

Le PC-1350 comporte cinq opérateurs numériques. Ce sont les opérateurs arithmétiques que vous avez utilisés lorsque vous vous êtes servis de votre PC-1350 comme calculatrice au Chapitre 3.

- + Addition
- Soustraction
- \* Multiplication
- / Division
- ^ Elévation à la puissance.

Une expression numérique se construit de la même manière que les calculs composés que vous avez entrés au Chapitre 3. Les expressions numériques peuvent comporter n'importe quelle combinaison significative de constantes numériques, de variables numériques et d'opérateurs numériques:

$(A * B) ^ 2$   
 $A(2, 3) + A(3, 4) + 5.0 - C$   
 $(A/B) * (C+D)$

## EXPRESSIONS ALPHANUMERIQUES (CHAINE DE CARACTERES)

Les expressions alphanumériques sont semblables aux expressions numériques à la différence près suivante: il n'y a qu'un seul opérateur pour les chaînes, l'opérateur de concaténation représenté par le même symbole que l'opérateur d'addition (+). Lorsqu'il est utilisé entre deux chaînes, ce symbole relie la seconde chaîne à la fin de la première de façon à ne former qu'une seule chaîne plus longue. Soyez prudent lorsque vous effectuez des concaténations ou d'autres opérations sur les chaînes plus complexes car la mémoire de travail utilisée par le PC-1350 pour les opérations sur les chaînes se limite à 80 caractères.

**Remarque:** Les quantités alphanumériques et les quantités numériques ne peuvent être combinées dans la même expression à moins que l'une utilise l'une des fonctions qui permet de convertir valeur alphanumérique en valeur numérique ou vice-versa:

"15" + 10            n'est pas possible  
 "15" + "10"        devient "1510" et non "25".

## EXPRESSIONS DE RELATION

Une expression de relation compare deux expressions et détermine si la relation mentionnée est Vraie ou Fausse. Les opérateurs de relation sont les suivants:

- > Plus grand que
- >= Plus grand que ou égal à
- = Egal à
- <> Non égal à
- <= Plus petit que ou égal à
- < Plus petit que

Voici des expressions de relations valides:

```
A < B
C(1, 2) >= 5
D(3) <> 8
```

Si A est égal à 10, B à 12, C(1, 2) à 6 et D(3) à 9, toutes ces expressions de relations sont Vraies.

Les chaînes de caractères peuvent également être comparées dans des expressions de relation. Deux chaînes peuvent être comparées caractère par caractère en fonction de leur valeur ASCII, en commençant au premier caractère (pour les valeurs ASCII, reportez vous à l'Annexe B). Si une chaîne est plus courte que l'autre, un  $\emptyset$  ou NUL sera utilisé pour les positions manquantes. Toutes les expressions de relation ci-dessous sont Vraies:

```
"ABCDEF" = "ABCDEF"
"ABCDEF" <> "ABCDE"
"ABCDEF" > "ABCDE"
```

Les expressions de relation aboutissent à une condition Vrai ou Faux. Le PC-1350 représente la condition Vrai par un 1 et Faux par un  $\emptyset$ . Dans un test logique, une expression qui aboutit à un 1 ou plus sera considérée comme Vrai alors qu'une expression qui aboutit à  $\emptyset$  sera considérée comme Fausse. Cependant, un programmeur expérimenté préférera utiliser une expression de relation explicite plutôt que de se fier à cette coïncidence.



## EXPRESSIONS LOGIQUES

Les expressions logiques sont des expressions de relation qui utilisent les opérateurs AND, OR et NOT. AND et OR sont utilisés pour relier deux expressions logiques. La valeur de l'expression combinée est représentée dans les tableaux ci-dessous:

A AND B		Valeur de A	
		Vrai	Faux
Valeur de B	Vrai	Vrai	Faux
	Faux	Faux	Faux

A OR B		Valeur de A	
		Vrai	Faux
Valeur de B	Vrai	Vrai	Vrai
	Faux	Vrai	Faux

(Remarque: la valeur de A et B doit être 0 ou 1).

- Les nombres décimaux peuvent être exprimés en notation binaire avec 16 bits comme suit:

NOTATION DECIMALE	NOTATION BINAIRE DE 16 BITS
32767	0111111111111111
⋮	⋮
3	0000000000000011
2	0000000000000010
1	0000000000000001
0	0000000000000000
-1	1111111111111111
-2	1111111111111110
-3	1111111111111101
⋮	⋮
-32768	1000000000000000

Le correspondant négatif (NOT) d'un nombre binaire 0000000000000001 est le suivant:

NOT 0000000000000001  
 (Nombre négatif) → 1111111111111110

Ainsi, pour obtenir le correspondant négatif d'un nombre, les uns sont remplacés par des zéros et les zéros par des uns pour chaque bit.

Lorsque 1 et NOT 1 sont additionnés, on obtient le résultat suivant:

$$\begin{array}{r} 0000000000000001 \quad (1) \\ +) \quad 1111111111111110 \quad (\text{NOT } 1) \\ \hline 1111111111111111 \quad (-1) \end{array}$$

Ainsi, tous les bits deviennent des 1. Suivant la liste des chiffres à la page précédente, les bits deviennent  $-1$  en notation décimale. Autrement dit,  $1 + \text{NOT } 1 = -1$ .

La relation entre la valeur numérique X et son correspondant négatif (NOT X) est:

$$X + \text{NOT } X = -1$$

On obtient l'équation  $\text{NOT } X = -X - 1$

c.a.d  $\text{NOT } X = -(X + 1)$

On peut déduire de cette équation les résultats suivants:

$$\text{NOT } 0 = -1$$

$$\text{NOT } -1 = 0$$

$$\text{NOT } -2 = 1$$

Il est possible de combiner plus de deux expressions relationnelles avec ces opérateurs. Il est conseillé d'utiliser des parenthèses pour que la comparaison voulue soit claire.

$$(A < 9) \text{ AND } (B > 5)$$

$$(C = 5) \text{ OR } (C = 6) \text{ OR } (C = 7)$$

Le PC-1350 utilise des opérateurs logiques fonctionnaires sur 16 bit et effectuant des opérations bit à bit. (Voir la remarque sur les expressions de relations et Vrai et Faux). Dans des conditions normales, ceci n'a aucune signification car le 1 et le 0 (vrai et faux) qui résultent d'une expression de relation n'utilisent qu'un bit unique. Si vous appliquez un opérateur logique à une valeur autre que 0 ou 1, cet opérateur s'appliquera à chaque bit séparément. Par exemple, si A est 17, et B 22,  $A \text{ OR } B$  est 23:

17 en notation binaire est représenté par 10001

22 en notation binaire est représenté par 10110

17 OR 22 est représenté par 10111 (1 si 1 dans les deux chiffres, sinon 0).

10111 correspond à 23 en notation décimale.

Si vous êtes un programmeur déjà expérimenté, ce type d'opération peut être très utile dans certaines applications. Les programmeurs débutants devraient en rester à des expressions de relation Vrai ou Faux simples et claires.

## PARENTHÈSES ET PRIORITÉ DES OPÉRATEURS

Pour calculer les expressions complexes, le PC-1350 se base sur des priorités pré-définies qui déterminent l'ordre dans lequel les opérateurs sont calculés. Ceci peut être très important:

$5 + 2 * 3$  pourrait être

$$5 + 2 = 7$$

$$7 * 3 = 21$$

ou

$$2 * 3 = 6$$

$$6 + 5 = 11$$

Les règles exactes de "Priorité des opérateurs" sont données à l'Annexe D.

Pour éviter d'avoir à se rappeler toutes ces règles et rendre votre programme plus clair, utilisez toujours des parenthèses pour déterminer l'ordre des calculs. L'exemple ci-dessus est plus clair si l'on écrit:

$$(5 + 2) * 3$$

ou

$$5 + (2 * 3)$$

## MODE RUN

En général, toutes les expressions ci-dessus peuvent être utilisées en mode RUN et également dans la programmation d'une instruction BASIC.

En mode RUN, une expression est calculée et immédiatement affichée. Par exemple:

Entrée

Ecran

$(5 > 3) \text{ AND } (2 < 6)$

1.

Le 1 signifie que l'expression est Vraie.

## FONCTIONS

Les fonctions sont des composants spéciaux du langage BASIC qui prennent une valeur et la transforment en une autre valeur. Les fonctions agissent comme des variables dont la valeur est déterminée par la valeur d'autres variables ou d'autres expressions. ABS est une fonction qui permet d'obtenir la valeur absolue de son argument:

ABS(-5) est 5

ABS(6) est 6

LOG est une fonction qui permet de calculer le logarithme base 10 de son argument.

LOG (100) est 2

LOG (1000) est 3

Une fonction peut être utilisée partout où une variable peut être utilisée. La plupart des fonctions ne nécessitent pas l'emploi de parenthèses:

LOG 100 revient au même que LOG (100)

Vous devez utiliser des parenthèses pour les fonctions qui comportent plusieurs arguments. L'emploi de parenthèses rend toujours les programmes plus clairs.

Pour la liste complète des fonctions possibles avec le PC-1350, reportez-vous au Chapitre 9.

## CHAPITRE 5 COMMENT PROGRAMMER AVEC LE PC-1350

Dans le chapitre précédent, nous avons étudié certains des concepts et des termes du langage de programmation BASIC. Dans ce chapitre, vous allez utiliser ces éléments pour créer des programmes sur votre PC-1350. Nous attirons cependant votre attention sur le fait que cette brochure n'est pas un manuel de programmation en BASIC. Ce chapitre a pour but de vous familiariser avec l'emploi du langage BASIC sur votre PC-1350.

### LES PROGRAMMES

Un programmes se compose d'une série d'instructions destinée à votre ordinateur. N'oubliez pas que le PC-1350 n'est qu'une machine. Il exécutera exactement ce que vous lui indiquez. C'est vous, en tant que programmeur, qui devez lui donner les ordres corrects.

### LES INSTRUCTIONS BASIC

Le PC-1350 interprète les ordres que vous lui donnez dans un format pré-déterminé que l'on appelle instruction. Il faut toujours entrer les instructions BASIC de la même manière. Les instructions doivent commencer par un numéro de ligne:

```
10: INPUT A
20: PRINT A * A
30: END
```

### LES NUMEROS DE LIGNE

Chaque ligne d'un programme doit avoir un numéro de ligne unique – un entier compris entre 1 et 65279. Les numéros de ligne servent de référence pour votre ordinateur. Ils indiquent au PC-1350 l'ordre dans lequel exécuter le programme. Il n'est pas nécessaire que vos lignes soient entrées dans l'ordre séquentiel (si vous êtes un programmeur débutant, vous vous y retrouverez sans doute mieux si vous les entrez dans cet ordre). L'ordinateur commence l'exécution du programme par la ligne ayant le numéro le moins élevé et poursuit en séquence croissante.

Lorsque vous écrivez un programme, il est conseillé de numéroter vos lignes de 10 en 10 ou de 20 en 20 par exemple (10, 20, 30, ... 10, 30, 50, etc). Ainsi, vous pourrez ajouter en cas de besoin des lignes supplémentaires.

**ATTENTION:** N'utilisez pas les mêmes numéros de ligne dans des programmes différents que vous prévoyez de fusionner. Si vous utilisez le même numéro de ligne, la ligne la plus ancienne sera supprimée et remplacée par la ligne la plus récente.

L'affichage des lignes dans un ordre non séquentiel n'affecte pas l'exécution ni l'impression du programme.

## LES VERBES BASIC

Toutes les instructions BASIC doivent comporter un verbe qui indique à l'ordinateur l'action à exécuter. Les verbes se trouvent toujours dans les programmes et, par conséquent, ne provoquent pas une action immédiate de la part de l'ordinateur.

Certaines instructions doivent obligatoirement ou peuvent facultativement comporter un opérateur:

```
1Ø: INPUT A
2Ø: PRINT A * A
3Ø: END
```

Les opérateurs fournissent des informations à l'ordinateur et lui indiquent quelles sont les données sur lesquelles le verbe doit agir. Les opérateurs sont obligatoires avec certains verbes et facultatifs avec d'autres. Ils sont même interdits avec certains verbes. (Pour la liste complète des verbes BASIC et leur emploi avec le PC-1350, reportez vous au Chapitre 9.)

**Remarque:** Les verbes, les commandes et les fonctions doivent être frappés en majuscules.

## LES COMMANDES BASIC

Les commandes sont des instructions destinées à l'ordinateur et entrées en dehors d'un programme. Les commandes indiquent à l'ordinateur d'exécuter une action déterminée dans le cadre de votre programme ou bien elles fixent les modes qui affectent la façon dont sont exécutés vos programmes.

A l'inverse des verbes, les commandes ont un effet immédiat — elles sont exécutées dès que vous avez fini de les entrer (à l'aide de la touche **ENTER**), les commandes ne sont pas précédées de numéros de lignes:

```
RUN
NEW
RADIAN
```

Certains verbes peuvent également être utilisés comme des commandes. (Pour la liste complète des commandes BASIC et leur emploi avec le PC-1350, reportez vous au Chapitre 9).

## LES MODES

Vous vous rappelez que lorsque nous avons utilisé le PC-1350 comme calculatrice, nous l'avons placé en mode RUN.

Ce mode permet également l'exécution de vos programmes.

Le mode PROGRAMME permet d'entrer et de corriger vos programmes.

Le mode RESERVE vous permet de désigner et d'enregistrer des variables alphanumériques pré-définies; il est utilisé pour une programmation plus élaborée (Voir au Chapitre 6).

## DEBUT DE LA PROGRAMMATION SUR LE PC-1350

Après tous les exercices effectués en utilisant le PC-1350 comme calculatrice, vous devez maintenant avoir l'habitude du clavier. Désormais, lorsque nous indiquerons une entrée de données, nous n'indiquerons plus les touches à utiliser. N'oubliez pas d'utiliser **SHIFT** pour frapper les caractères situés sur la partie supérieure des touches et DE TERMINER CHAQUE LIGNE EN APPUYANT SUR LA TOUCHE **ENTER**.

Maintenant, vous êtes prêt à programmer !

Pour entrer les instructions d'un programme dans l'ordinateur, vous devez tout d'abord placer ce dernier en mode PROGRAMME à l'aide de la touche **MODE**. L'écran se présentera comme ci-dessous.

PRO	PROGRAM MODE
	>

Entrer la commande NEW.

Entrée

Ecran

NEW

>
---

La commande NEW efface tous les programmes et toutes les données existant dans la mémoire du PC-1350. Le message-guide s'affiche une fois que vous avez appuyé sur **ENTER** afin d'indiquer que l'ordinateur attend des données d'entrée.

**EXEMPLE 1 – ENTREE ET EXECUTION D'UN PROGRAMME**

Vérifiez que le PC-1350 se trouve en mode PRO et entrez le programme suivant:

EntréeEcran

10 PRINT "HELLO"

10: PRINT "HELLO"

Remarquez que lorsque vous appuyez sur la touche **ENTER**, le PC-1350 affiche vos données d'entrée et insère automatiquement deux point (:) entre le numéro de ligne et le verbe. Vérifiez que l'instruction a bien le format voulu.

Maintenant, changez de mode et passez en mode RUN:

EntréeEcran

RUN

HELLO

Etant donné que c'est la seule ligne du programme, l'ordinateur s'arrêtera à ce stade. Appuyez sur **ENTER** pour sortir du programme et repassez en mode RUN si vous désirez exécuter à nouveau le programme.

**EXEMPLE 2 – CORRECTION D'UN PROGRAMME**



Supposez que vous désiriez modifier le message qu'affiche votre programme, autrement dit que vous désiriez corriger (éditer) votre programme. Avec un programme d'une seule ligne, il suffit de refrapper l'entrée mais lorsque vous écrivez des programmes plus complexes, leurs corrections tiendra une place importante dans votre travail de programmation. Corrigeons le programme qui vient d'être écrit.


Etes-vous encore en mode RUN? Dans ce cas, rapassez en mode PROgramme.

Il vous faut rappeler votre programme pour le corriger. Pour cela, utilisez la flèche de défilement vers le haut ( **↑** ). Si l'exécution de votre programme est terminée, la touche **↑** rappelle la dernière ligne du programme. S'il y a une erreur dans le programme, ou si vous avez utilisé la touche BREAK ( **BRK** ) pour en arrêter l'exécution, la touche **↑** rappellera la ligne où s'est produite l'erreur ou bien où vous avez appuyé sur la touche **↑**. Pour effectuer des modifications dans votre programme, utilisez la touche **↑** pour remonter dans votre programme (rappeler la ligne précédente) et la touche **↓** pour descendre dans votre programme (afficher la ligne suivante). Si vous maintenez ces touches enfoncées, vous obtiendrez un défilement vertical (autrement dit, les lignes du programme défileront vers le haut ou vers le bas).



## Comment Programmer

N'oubliez pas que pour déplacer le curseur sur la ligne de programme affichée sur la première ligne de l'écran, vous devez utiliser les touches  (flèche vers la droite) et  (flèche vers la gauche). Vous pouvez, à l'aide de ces touches, placer le curseur sur le premier caractère que vous désirez modifier.

**Remarque:** Même si plusieurs lignes d'un programme sont affichées à l'écran, le curseur ne peut être déplacé que dans la première ligne affichée. Pour corriger une ligne située plus bas, amenez cette ligne en haut de l'écran à l'aide de la touche , puis corrigez-la.

Entrée



Ecran

```
10: PRINT "HELLO"
```

```
10 PRINT "HELLO"
```

Remarquez que le curseur a pris maintenant la forme d'un rectangle clignotant afin d'indiquer qu'il se trouve "au dessus" d'un caractère existant. Frappez :

Entrée

```
GOODBYE!"
```

Ecran

```
10 PRINT "GOODBYE!"_
```

N'oubliez pas d'appuyer sur la touche  à la fin de la ligne. Passez en mode RUN.

Entrée



```
RUN 
```

Ecran

```
ERROR 1 IN 10
```

Voici un nouveau type de message d'erreur. Non seulement le type d'erreur est identifié (notre vieille amie, l'erreur de syntaxe) mais le numéro de ligne à laquelle s'est produite l'erreur est également indiqué.

Appuyez sur la touche  pour restaurer cette condition d'erreur et repassez en mode PRO.

Pour pouvoir effectuer des modifications dans un programme, vous devez vous trouver en mode PROgramme. Utilisez la touche  (ou ) pour rappeler la dernière ligne de votre programme.

Entrée

```
 (ou )
```

Ecran

```
10: PRINT "GOODBYE!"
```

Le curseur clignotant se trouve sur la zone erreurs. Au Chapitre 4, vous avez appris que lorsque vous entrez des constantes alphanumériques en BASIC, tous les caractères doivent figurer entre guillemets. Utilisez la touche DEL pour supprimer le "!" :

Entrée

Ecran

DEL

10 PRINT "GOODBYE" \_

Plaçons maintenant le ! à l'endroit voulu. Lorsque vous corrigez vos programmes, les touches DEL (suppression) et INS (insertion) sont utilisées exactement de la même manière que pour la correction des calculs (voir Chapitre 3). La touche ◀ permet de placer le curseur au-dessus du caractère suivant immédiatement l'insertion.

Entrée

Ecran

◀

10 PRINT "GOODBYE"

Appuyez sur la touche INS. Des crochets ( [ ] ) indiqueront l'endroit où seront insérées les nouvelles données :

Entrée

Ecran

INS

10 PRINT "GOODBYE [ ]"

Frappez le !. L'écran se présente comme ceci :

Entrée

Ecran


!

10 PRINT "GOODBYE!"







N'oubliez pas d'appuyer sur **ENTER** pour entrer la correction dans le programme.



- Remarque**
- Si vous désirez supprimer toute une ligne de programme, il suffit de taper le numéro de ligne pour que la ligne soit supprimée.
  - Lors de la modification d'un programme, ajouter la nouvelle commande avant d'annuler l'ancienne. Par exemple, pour remplacer la commande PRINT dans "30 PRINT 123" par une autre commande (ex. LPRINT), d'abord ajouter la commande avant d'annuler "PRINT". Si on annule d'abord la commande PRINT, le numéro de ligne "30" et le nombre "123" se combinent et il se peut que la ligne suivante soit affichée ou que seul le curseur clignotant soit affiché.

**Remarque 1:**



Si une erreur survient au cours de l'exécution d'un programme et si cette erreur se trouve à la fin d'une ligne, appuyez sur la touche  avant d'insérer le texte à l'endroit indiqué.

(Exemple)

			ERROR 1 IN 20
	R		20: S = $\pi * R * R$
			20: S = $\pi * R * R$  Curseur

Si vous n'avez pas utilisé la touche , le programme affiché risque de disparaître de l'écran lorsque vous frapperez le texte à ajouter. Si vous appuyez sur la touche , le programme ajouté s'affiche avec le texte déjà entré.

**Remarque 2:**

Au cours de l'exécution d'un programme, les écrans d'erreurs et d'interruption sont indiqués à la ligne suivant celle où sont entrées les données ou sur laquelle se trouve la commande PRINT. L'erreur ou l'interruption peut ne pas être affichée lorsque cette erreur ou cette interruption (exécution de la commande STOP ou interruption temporaire due à l'utilisation de la touche ) survient alors que la position de début des données affichées a été spécifiée à l'aide de la commande CURSOR. L'ordinateur se trouve cependant en mode erreur ou interruption. En mode erreur, les touches alphabétiques et numériques ne fonctionnent pas. Il faut appuyer sur la touche  pour restaurer la condition d'erreur.

**EXEMPLE 3 – UTILISATION DE VARIABLES DANS UN PROGRAMME**

Si l'emploi des variables numériques et des variables alphanumériques en BASIC ne vous est pas familier, relisez les sections correspondantes au Chapitre 4.

L'emploi de variables dans les programmes permet d'utiliser de manière encore mieux les possibilités de calculs du PC-1350.

Rappelez vous que vous que vous définissez des variables numériques simples à l'aide des lettres A à Z:

A = 5

Pour définir des variables alphanumériques, vous utilisez également une lettre, suivie d'un signe dollar. N'utilisez pas la même lettre pour désigner une variable numérique et une variable alphanumérique. Vous ne pouvez indiquer A et A\$ dans le même programme.

N'oubliez pas que les variables alphanumériques ne peuvent comporter plus de 7 caractères:

A\$ = "TOTAL"

Les valeurs attribuées à une variable peuvent changer au cours de l'exécution d'un programme et prendre les valeurs frappées en entrée ou calculées au cours de l'exécution de ce programme. Pour affecter des variables, on peut utiliser le verbe INPUT. Dans le programme ci-dessous, la valeur de A\$ changera en fonction des données frappées en réponse à l'interrogation "WORD?". Entrez le programme ci-dessous:

```

10 INPUT "WORD?"; A$
20 B = LEN (A$)
30 PRINT "WORD IS "; B; " LETTERS"
40 END

```

↑  
  espacement

Etant donné que la ligne 30 de ce programme dépasse 24 colonnes, le reste de la ligne est affiché à la ligne suivante.

Le second élément nouveau dans ce programme est l'emploi de l'instruction END pour signaler la fin d'un programme. END indique à l'ordinateur que le programme est terminé. Il est conseillé d'employer l'instruction END en fin de programme.

Au fur et à mesure que vos programmes deviendront plus complexes, vous voudrez parfois les revoir avant de les exécuter. Pour revoir un programme, utilisez la commande LIST. Cette commande ne peut être utilisée qu'en mode PROgramme; elle permet d'afficher les programmes à partir du numéro de ligne le moins élevé.

Tentez de lister ce programme:

Entrée



LIST

Ecran

```

10: INPUT "WORD?"; A$
20: B= LEN (A$)
30: PRINT "WORD IS ";B;"
   LETTERS"

```

Utilisez les touches  et  pour faire défiler à l'écran les lignes de programme jusqu'à ce que vous ayez revu la totalité de votre programme. Pour revoir une ligne qui comporte plus de 24 caractères, amenez le curseur à l'extrême droite de l'écran et les caractères supplémentaires apparaîtront. Après avoir vérifié votre programme, exécutez-le:

Entrée

RUN

HELP

**ENTER**

Ecran

**RUN  
WORD?\_**

**RUN  
WORD? HELP\_**

**RUN  
WORD? HELP  
WORD IS 4. LETTERS**

C'est la fin de votre programme. Vous pouvez évidemment l'exécuter à nouveau en frappant RUN. Cependant, ce programme serait plus intéressant si l'on pouvait entrer des données sans avoir à frapper RUN après chaque réponse. Nous allons donc le modifier dans ce sens.

Revenez en mode PRO et utilisez les flèches vers le haut ou vers le bas pour arriver à la ligne 40 (ou frappez LIST 40 et appuyez sur **ENTER** ).

Ecran

**40:END**

Vous pouvez frapper 40 pour supprimer la totalité de la ligne ou utiliser la touche ► pour placer le curseur sur le E de End. Modifiez la ligne 40 de la manière suivante:

40: GOTO 10

Maintenant, exécutez (RUN) le programme modifié.

L'instruction GOTO permet au programme d'effectuer une boucle (de répéter sans cesse la même opération). Etant donné que vous n'avez pas indiqué de limites pour cette boucle, elle se répétera indéfiniment (c'est une boucle "infinie"). Pour arrêter ce programme, appuyez sur la touche **BRK** .

Lorsque vous arrêtez un programme à l'aide de la touche **BRK** vous pouvez le reprendre à l'aide de la commande CONT. CONT signifie CONTinuer. Grâce à cette commande, le programme reprendra à la ligne en cours d'exécution lorsque la touche **BRK** a été enfoncée.

## EXEMPLE 4 – PROGRAMMATION PLUS COMPLEXE

Le programme ci-dessus calcule la factorielle de N (N!). Le programme commence à 1 et calcule N! jusqu'à la limite entrée par vous. Frappez ce programme.

```

100 F = 1: WAIT 118
110 INPUT "LIMIT?"; L
120 FOR N = 1 TO L
130 F = F * N
140 PRINT N, F
150 NEXT N
160 END

```

Ce programme comporte plusieurs nouveaux éléments. Le verbe WAIT à la ligne 100 contrôle le laps de temps pendant lequel les données seront affichées à l'écran avant que le programme poursuive. Les nombres et leurs factorielles sont affichés au fur et à mesure qu'ils sont calculés. La durée de leur affichage à l'écran est fixée par l'instruction WAIT à environ 2 secondes; vous n'avez donc pas à appuyer sur la touche **ENTER**.

A la ligne 100 également, remarquez qu'il y a deux instructions sur la même ligne, séparées par deux points (:). Vous pouvez placer autant d'instructions que vous le désirez sur une ligne, mais elles doivent être séparées par deux points et ne doivent pas dépasser 80 caractères, y compris **ENTER**. Lorsque plusieurs instructions figurent sur une même ligne, les programmes peuvent devenir difficiles à lire et à modifier; c'est pourquoi, il est conseillé de n'utiliser cette pratique que pour des instructions très simples et lorsqu'il y a des raisons particulières pour placer plusieurs instructions sur une même ligne.

Dans ce programme, vous avez également utilisé le verbe FOR à la ligne 120 et le verbe NEXT à la ligne 150 pour créer une boucle. Dans l'exemple 3, vous aviez créé une boucle "infinie" qui se répétait sans cesse jusqu'à ce que vous appuyiez sur la touche **BRK**. Avec cette boucle FOR/NEXT, le PC-1350 ajoute 1 à N chaque fois qu'il arrive au verbe NEXT. Il effectue également un test pour voir si N est supérieur à la limite L. Si N est inférieur ou égal à L, l'exécution reprend en haut de la boucle et les instructions sont à nouveau exécutées. Si N est supérieur à L, l'exécution se poursuit par la ligne 160 et le programme s'arrête.

Il est possible d'utiliser n'importe quelle variable numérique dans une boucle FOR/NEXT. Vous n'êtes pas également obligé de démarrer à 1 et vous pouvez ajouter un incrément quelconque à chaque étape. Pour plus de détails, reportez vous au Chapitre 8.

Nous avons donné à ce programme des numéros de ligne en commençant à 100. Le fait de donner à vos programmes des numéros de lignes différents vous permet

d'avoir plusieurs programmes en même temps en mémoire. Pour exécuter (RUN) ce programme et non pas celui de la ligne 10.

RUN 100

Vous pouvez également identifier vos programmes par une lettre alphabétique et déclencher leur exécution en appuyant sur la touche DEF (voir au Chapitre 6).

## ENREGISTREMENT DES PROGRAMMES DANS LA MEMOIRE DU PC-1350

Rappelez-vous que les modes, les touches de réserve et les fonctions restent dans l'ordinateur même après sa mise hors tension. Les programmes restent également dans la mémoire lorsque vous mettez le PC-1350 hors tension ou lorsqu'il passe automatiquement hors tension. Même si vous utilisez les touches **BRK**, CLS ou CA, les programmes restent en mémoire.

Les programmes ne sont perdus que dans les cas suivants:

- \* Vous entrez NEW avant de commencer à programmer.
- \* Vous initialisez votre ordinateur en appuyant sur le bouton ALL RESET.
- \* Vous créez un nouveau programme en spécifiant les mêmes numéros de ligne (SAME LINE NUMBERS) qu'un programme existant en mémoire.
- \* Vous remplacez les piles.

Cette brève introduction à la programmation sur le PC-1350 vous a montré des possibilités de votre nouvel ordinateur en matière de programmation. Pour d'autres exercices de programmation, reportez-vous aux Exemples de programmes.

## FONCTIONS GRAPHIQUES

L'écran du PC-1350 se compose de 150 points horizontaux et de 32 points verticaux. Ces points permettent de reproduire à l'écran des dessins simples.

Voici les 6 commandes qui vous permettront de tracer des figures.

- GPRINT:** Impression (PRINT graphique). Cette commande permet d'obtenir des configurations avec 8 points verticaux par unité.
- GCURSOR:** CURSEUR graphique. Cette commande permet de spécifier la position lorsque vous tracez un dessin à l'aide de GPRINT.
- PSET:** Définit un point. Cette commande permet de mettre en brillance plus intense ou en affichage inversé un point spécifié.
- PRESET:** Restauration (RESET) du point. Cette commande efface le point spécifié.
- LINE:** Cette commande permet de tracer une ligne ou un carré entre deux points spécifiés.

**POINT:** Cette commande permet de déterminer si le point spécifié est affiché ou non.

Le principe du dessin est décrit ici. Pour plus de détails sur les fonctions de chaque commande, se reporter à la description de chaque commande.

A la base, il y a deux façons de produire des dessins.

La première consiste à produire des dessins en combinant des configurations pré-déterminées. La seconde consiste à produire des dessins en spécifiant et en affichant chaque point.

Pour afficher des caractères alphanumériques et numériques avec les graphiques, il faut utiliser la commande graphique après l'affichage de ces caractères.

(1) Reproduction d'un dessin à l'aide de configurations pré-déterminées.

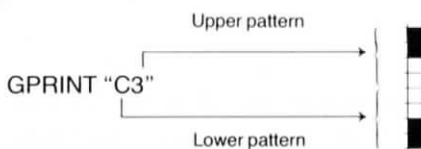
Il est possible de reproduire un dessin en combinant les 16 configurations représentées dans le tableau ci-dessous et en utilisant la commande GPRINT.

Hexadecimal character	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Pattern																

Les configurations de ce tableau utilisent toutes 4 points.

Cependant, dans la commande GPRINT, deux combinaisons sont combinées verticalement pour obtenir une configuration de 8 points comme le montre l'exemple ci-dessous. On peut reproduire un dessin en combinant et alignant un certain nombre de configuration à 8 points.

(Exemple)

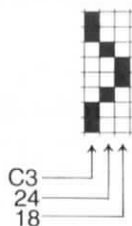


Deux nombres hexadécimaux sont utilisés avec la première configuration pour indiquer la configuration inférieure et avec la seconde pour indiquer la configuration supérieure.



(Exemple)

GPRINT "C32418"

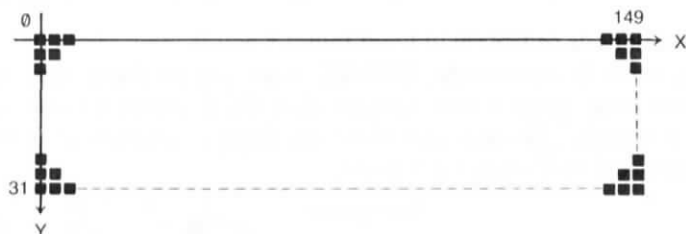


Les configurations à 8 points spécifiés par les chiffres pairs hexadécimaux sont disposées de façon à reproduire le dessin.

(2) Indication d'une position sur l'écran

Comme décrit, les configurations sont spécifiées à l'aide de la commande GPRINT. L'emplacement auquel cette configuration doit être affichée sur l'écran, peut être spécifié à l'aide de la commande GCURSOR. L'unité d'affichage (écran graphique) du PC-1350 se compose de 150 x 32 points.

Chaque point peut être représenté sous la forme "x, y" comme dans un système de coordonnées, "x" étant la position horizontale, et "y", la position verticale.



Il convient cependant de remarquer que dans les systèmes de coordonnées normaux, le point "y" ayant une valeur plus élevée se trouve plus haut alors qu'avec le PC-1350, ce point se trouve plus bas sur l'écran.

Les coordonnées des points de l'écran vont de 0 à 149 pour "x" et de 0 à 31 pour "y".

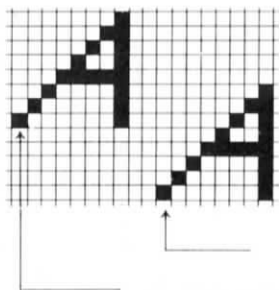
**Remarque:** Les valeurs spécifiées pour "x" et "y" peuvent aller de -32768 à +32767. Un emplacement virtuel (emplacement qui est fait pour exister temporairement mais n'existe pas réellement) sera spécifié si l'emplacement spécifié se trouve au-delà des limites de l'écran. Par conséquent, si un dessin est dessiné à partir d'un tel emplacement, rien ne sera affiché à l'écran. Ceci s'applique également à d'autres commandes décrites par la suite et pour laquelle sont spécifiées des coordonnées, par exemple, les commandes SET, RESET et LINE.

La commande GPRINT permet d'obtenir des dessins en alignant des groupes de configurations de 8 points. Si le point où le dessin doit être affiché a été spécifié à l'aide de la commande GCURSOR, la première configuration de 8 points est affichée à partir du point qui se trouve à l'emplacement spécifié et au-dessus de celui-ci.

(Exemple)

```
5 CLS
10 AA$="80402010181412FF"
20 GCURSOR (70, 15)
30 GPRINT AA$
40 GCURSOR (80, 20)
50 GPRINT AA$
```

\* L'exécution du programme permettra d'afficher la configuration suivante près du centre de l'écran.



(3) Création d'un dessin en spécifiant les points un par un.

Un dessin peut être obtenu en affichant ou en effaçant le point spécifié à l'aide des commandes PSET ou PRESET.

Chaque point est spécifié de la même manière qu'avec la commande GCURSOR.

(Exemple)

PSET (75, 15)

Affiche le point qui se trouve en (75, 15).

PSET (75, 15), X

Affiche le point qui se trouve en (75, 15) s'il est effacé et l'efface s'il est affiché

(Le fait de spécifier "X" affiche le point en image inversée.)

PRESET (75, 15)

Efface le point situé en (75, 15).

(Exemple)

```

100 WAIT 0 : DEGREE
110 FOR A = 180 TO -180
    STEP -2
120 B = SIN A * 12
130 C = COS A * 12
140 X = B + 80
150 Y = C + 14
160 PSET (X, Y)
170 NEXT A
180 WAIT : GPRINT
    
```

- \*1 Détermine les coordonnées du périmètre d'un cercle de rayon 12. Le centre se trouve en (0, 0).
- \*2 Déplace le centre du cercle en (80, 14).
- \*3 Affiche le point spécifié par (x, y).
- \*4 Continue à afficher le cercle dessiné.

L'exécution de ce programme provoque le traçage d'un cercle ayant un rayon de 12 et dont le centre se trouve en (80, 14);

L'angle passe de +180 à -180 dans la boucle FOR-NEXT. A chaque angle, les coordonnées du périmètre du cercle sont déterminées et le point correspondant est affiché.

#### (4) Tracés de lignes et de carrés

Bien que les lignes et les carrés puissent également être tracés à l'aide de la commande PSET, ils peuvent être tracés facilement en spécifiant uniquement deux points dans la commande LINE.

(Exemple)

```
LINE (0, 0) - (149, 31)
```

Cette commande permet de tracer une ligne allant de (0, 0) à (149, 31).

```
LINE (30, 0) - (80, 31), B
```

Cette commande permet de tracer un carré dont la diagonale va de (30, 0) à (80, 31). Si B devient BF, l'intérieur du carré sera rempli.

```
LINE (30, 0) - (80, 31), X, BF
```

Un carré rempli est dessiné. Cependant, si un point à l'intérieur de ce carré était déjà affiché, il est effacé.

(Lorsque "X" est utilisé, les points sont affichés en image inversée.)

(Exemple)

```

200 "A" : WAIT 0
210 LINE (60, 0) - (100, 31)
    , X, BF
220 GOTO 210
    
```

L'exécution de ce programme permet de tracer un carré dont la diagonale va de (60, 0) à (100, 31), puis le remplit et l'efface. Laissez le dessin obtenu précédemment à l'aide des commandes GPRINT et PSET à l'écran et tentez d'exécuter le programme en appuyant sur **DEF** et **A**. L'intérieur du carré est inversé.

**Remarque:** L'écran graphique est plus large que l'écran alphabétique et numérique classique: il comporte 6 points de plus sur la gauche.

Le PC-1350 est conçu pour conserver l'image dessiné jusqu'à ce qu'elle soit effacée. Par conséquent, si le programme se termine ou s'il est arrêté, l'image graphique peut rester sur la partie gauche, ou en haut de l'écran.

(Pour effacer l'écran, il faut appuyer sur la touche **CLS**.)

**Remarque:** Les caractères et les chiffres frappés lorsque le programme est arrêté par une commande graphique risquent de ne pas être affichés. Effacez tout d'abord l'écran à l'aide de la touche **CLS** puis entrez à nouveau les caractères.

## FONCTION D'ENTREE-SORTIE SERIE

Le PC-1350 est équipé d'une interface d'entrée-sortie série. La fonction d'entrée-sortie permet de se connecter à un ordinateur personnel pour l'entrée ou la sortie de données.

**Remarque 1:** Pour la connection à l'ordinateur PC-1350 de l'interface RS-232C qui se trouve sur la plupart des ordinateurs, un adaptateur de niveau en option devient indispensable étant donné que le niveau de tension des signaux du RS-232C diffère. Un câble de connection est également nécessaire.

**Remarque 2:** Il convient d'être très prudent car l'envoi d'une tension dépassant les limites autorisées sur le PC-1350 au terminal d'entrée-sortie peut abîmer les différentes pièces internes à la machine.

### Utilisation de l'interface d'entrée-sortie série

Le circuit de l'interface d'entrée-sortie série est en général fermé. Dans ce cas, les données en provenance du terminal d'entrée-sortie série ne peuvent être envoyées et les données reçues ne peuvent être lues.

Par conséquent, il est tout d'abord nécessaire d'ouvrir ce circuit à l'aide de la commande **OPEN**. (S'il est déjà ouvert, le message **ERROR 8** s'affiche.)

D'autre part, les conditions permettant d'échanger des données avec l'ordinateur connecté au PC-1350 doivent être satisfaites. Autrement dit, les conditions relatives aux signaux doivent être identiques pour le PC-1350 et l'ordinateur personnel connecté. Sinon, les signaux (données) ne peuvent être lus correctement et provoquent des erreurs de données.

## Comment Programmer

La commande OPEN peut être utilisée pour définir et modifier les conditions d'entrée-sortie.

Une fois les conditions satisfaites des deux côtés et le circuit ouvert, les commandes suivantes permettent d'exécuter l'entrée-sortie de données ou de programmes.

LPRINT, LLIST, SAVE, LOAD, PRINT #1, INPUT #1

A la fin de l'entrée ou de la sortie d'un programme ou des données, le circuit de l'interface d'entrée-sortie série est fermé.

Bien que la commande CLOSE permette de fermer le circuit, celui-ci se ferme également lorsque le programme se termine (par exemple lorsque la commande END est exécutée) ou lorsque la commande RUN est exécutée.

Lors de l'écriture d'un programme qui utilise l'interface d'entrée-sortie série, le circuit doit être ouvert, l'opération d'entrée-sortie exécutée, puis le circuit doit être fermé comme décrit ci-dessus.

Lorsque plusieurs programmes sont chargés avec la commande MERGE, il n'est pas possible de les sauvegarder par ce RS-232C à l'aide de la commande SAVE.

**Remarque:** Le PC-1350 n'est pas conçu pour attendre pendant les commandes entrée-sortie destinées à l'interface d'entrée-sortie série; il n'est pas non plus équipé d'une fonction d'horloge pour interrompre les communications avec l'ordinateur connecté.

Par conséquent, si l'ordinateur connecté n'est pas prêt à communiquer (s'il est hors tension par exemple) alors que les commandes sont exécutées ou si la transmission avec l'ordinateur connecté est interrompue, le PC-1350 ne peut arrêter les commandes exécutées et celles-ci se poursuivront.

Dans ce cas, appuyez sur la touche **BRK** pour arrêter l'exécution des commandes.

## CHAPITRE 6 QUELQUES METHODES RAPIDES

Le PC-1350 comporte différentes possibilités qui facilitent encore la programmation en réduisant le nombre de touches à utiliser pour entrer des éléments répétitifs.

Parmi ces possibilités figure l'abréviation des verbes et des commandes (Voir au Chapitre 8).

Le présent chapitre explique deux fonctions supplémentaires qui peuvent éliminer des opérations de frappe: la touche DEF et le mode Reserve.

### LA TOUCHE DEF ET LES PROGRAMMES IDENTIFIES PAR UN LETTRE (LABEL)

Vous voudrez souvent enregistrer plusieurs programmes différents en même temps dans la mémoire du PC-1350. (N'oubliez pas que les numéros de ligne de chacun de ces programmes doivent être uniques). En principe, pour déclencher l'exécution d'un programme à l'aide d'une commande RUN ou GOTO, vous devez vous souvenir du premier numéro de ligne de chaque programme (voir au Chapitre 8). Mais il existe une méthode plus facile! Vous pouvez attribuer à chacun de vos programmes une lettre et déclencher leur exécution en n'utilisant plus que deux touches. Voici comment attribuer une lettre à un programme et l'exécuter à l'aide de la touche DEF:

**Remarque:** Indiquez sur la première ligne de chacun de vos programmes la lettre par laquelle vous le désignerez. Cette lettre doit être écrite entre guillemets suivie de deux points:

```
10: "A": PRINT "FIRST"
20: END
80: "B": PRINT "SECOND"
90: END
```

Vous pouvez utiliser l'un des caractères suivants:

A, S, D, F, G, H, J, K, L, =, Z, X, C, V, B, N, M, et SPC. Vous remarquerez que ce sont les touches des deux dernières rangées de votre clavier et qu'elles sont enfermées de la ligne blanche de façon à ce que vous vous en rappeliez.

**Remarque:** Pour lancer l'exécution du programme, au lieu de frapper RUN 80 ou GOTO 10, il suffit d'appuyer sur la touche **DEF** puis sur la lettre qui désigne le programme. Dans l'exemple ci-dessus, le fait d'appuyer sur **DEF** puis sur 'B' provoquerait l'affichage de 'SECOND' à l'écran.

Lorsque la touche DEF est utilisée pour exécuter un programme, les variables et les modes sont affectés de la même manière que lorsque c'est la commande GOTO qui est utilisée. Pour plus de détails, reportez-vous au Chapitre 8.

### MODE RESERVE

Le mode Reserve est une autre fonction du PC-1350 qui vous permettra de gagner du temps.

144 caractères de la mémoire du PC-1350 sont définis comme "mémoire de réserve". Cette partie de la mémoire peut servir à enregistrer les expressions fréquemment utilisées qui peuvent alors être rappelées en n'utilisant plus que deux touches.

**Remarque:** ● Les chaînes de caractères peuvent être enregistrées en mode ReSerVe et rappelées pour être utilisées en modes RUN et PROgramme.

- Le PC-1350 a une mémoire de réserve de 144 octets. Vous pouvez donc placer 144 octets y compris la touche RSV dans la mémoire de réserve. Une commande BASIC, une fonction, un chiffre ou une lettre de l'alphabet occupe un octet.

Exemple: A: S: + - 1 2 A B SIN COS INPUT RUN....

(1 octet chacun)

- La longueur d'une chaîne de caractères correspondant à une touche est d'au maximum 80 caractères, y compris la touche réserve et la touche **ENTER**.

Voici un exemple d'enregistrement d'une chaîne de caractères dans la mémoire de réserve et de rappel de cette chaîne.

Placez le PC-1350 en mode ReSerVe en appuyant sur les touches **SHIFT** et **MODE**. L'indicateur de mode "RUN" et "PRO" disparaît et le message "RESERVE MODE" s'affiche.

Frappez NEW et appuyez sur la touche **ENTER**. Les caractères enregistrés précédemment seront effacés (de même que NEW efface les programmes enregistrés lorsque l'ordinateur est en mode PROgramme).

Frappez **SHIFT** suivi de 'A':

Entrée

Ecran

**SHIFT** A

**A: \_**

Remarquez que le 'A' apparaît à l'écran à la gauche des deux points.

Entrez le mot 'PRINT' et appuyez sur la touche **ENTER** :

Entrée

Ecran

PRINT **ENTER**

**A: PRINT**

Un espace apparaît après les deux points signalant que 'PRINT' est maintenant enregistré dans la mémoire de réserve sous la lettre A.

Placez le PC-1350 en mode PROgramme. Frappez NEW et appuyez sur la touche **ENTER** pour effacer la mémoire des programmes. Frappez '10' comme numéro de ligne et appuyez sur la touche **SHIFT** puis sur la touche A:

Entrée

Ecran

10 **SHIFT** A

**10 PRINT\_**

**ENTER**

**10: PRINT**

Aussitôt le mot 'PRINT' apparaît à l'écran après le numéro de ligne. Il est possible d'enregistrer dans la mémoire de réserve n'importe quelle chaîne de caractères (valeur alphanumérique). Ces chaînes peuvent être rappelées à tout moment soit en mode PROgramme soit en mode RUN; il suffit pour cela d'appuyer sur la touche **SHIFT** puis sur la touche correspondant à la chaîne enregistrée. Les touches sous lesquelles peuvent être enregistrées des chaînes de caractères sont identiques à celles utilisées avec DEF, c'est-à-dire les touches enfermées de la ligne blanche.

Pour corriger une chaîne de caractères enregistrée, passez en mode réserve et appuyez sur la touche **SHIFT** et sur la touche correspondant à la chaîne de caractère voulue. Vous pouvez alors corriger cette dernière à l'aide des touches **◀** (flèche vers la gauche), **▶** (flèche vers la droite), DEL et INS de la même manière que dans les autres modes.

Lorsque le dernier caractère d'une chaîne enregistrée est un '@', il est interprété comme **ENTER** lorsque la chaîne est rappelée. Par exemple, si vous enregistrez la chaîne "GOTO 100@" sous la touche 'G', le fait d'appuyer sur les touches **SHIFT** et 'G' en mode RUN déclenche immédiatement l'exécution du programme



## Quelques Methodes Rapides

de la ligne 100. Si le caractère '@' ne figure pas, vous devez appuyer sur la touche **ENTER** après **SHIFT** et 'G' pour déclencher l'exécution.

Suppression des programmes réservés:

1. Comme vous le savez, la commande NEW suivie de **ENTER** efface toutes les positions de la mémoire de réserve lorsque l'ordinateur se trouve en mode RESERVE.
2. Pour effacer une position de la mémoire de réserve, utilisez la touche **SPC** ou **DEL** comme décrit ci-dessous:

Entrée	Ecran	Remarques
<b>SHIFT</b> S	S: _	Mode Reserve
A * A <b>ENTER</b>	S: A * A	
<b>CLS</b>	>	
<b>SHIFT</b> S	S: A * A	
<b>←</b> OU <b>→</b>	S: A * A	
<b>DEL DEL DEL</b>	S: _	Delete A * A
<b>ENTER</b>	>	

## REGLETTES

Deux réglettes vous sont fournies avec le PC-1350. Vous pouvez utiliser ces réglettes pour vous aider à vous rappeler les séries de touches fréquemment utilisées et enregistrées dans la mémoire de réserve ou les identifications des programmes. Après avoir désigné vos programmes ou créé des séquences de caractères, indiquez les sur les réglettes de façon à vous souvenir de la séquence ou du programme associé à chacune des touches. Vous pourrez alors relancer l'exécution de vos programmes ou rappeler vos séquences en n'utilisant plus que deux touches.

Par exemple, si vous avez un groupe de programmes que vous utilisez souvent en même temps, attribuez une lettre à chacun de vos programme et indiquez-les sur votre réglette de façon que vous puissiez facilement déclencher l'exécution de ces programmes en n'utilisant plus que deux touches. Vous pouvez également enregistrer les commandes et les verbes BASIC que vous utilisez fréquemment dans la mémoire de réserve et les indiquer sur une réglette pour accélérer l'entrée des programmes BASIC.

Exemple:

## CHAPITRE 7

# UTILISATION DE L'INTERFACE IMPRIMANTE/CASSETTE CE-126P

L'interface Imprimante/Cassette CE-126P en option vous permet d'ajouter une imprimante et de connecter un enregistreur de cassette à votre ordinateur SHARP PC-1350. Voici les caractéristiques de l'interface CE-126P:

- \* Imprimante thermique de 24 caractères par ligne.
- \* Alimentation du papier commode et barre de déchirement.
- \* Impression simultanée des calculs, si l'utilisateur le désire.
- \* Contrôle facile de la sortie sur écran ou sur imprimante en BASIC.
- \* Interface cassette incorporée avec contrôle à distance.
- \* Contrôle manuel et contrôle par programme de l'enregistrement des programmes et des données.
- \* Fonctionnement sur piles qui assure une totale indépendance.

Pour la connection du PC-1350 et du CE-126P, reportez vous au manuel d'instructions fourni avec le CE-126P.

### UTILISATION DE L'IMPRIMANTE

Si vous utilisez le PC-1350 pour des calculs manuels, vous pouvez utiliser l'interface CE-126P pour imprimer simultanément ces calculs.

#### ATTENTION:

Les résultats obtenus par le dispositif de calculs en direct en calcul manuel ne peut être imprimé.

Pour imprimer les calculs, il suffit d'appuyer sur la touche **SHIFT** puis sur la touche **ENTER** (P ↔ NP) lorsque l'ordinateur est en mode RUN. Après quoi, lorsque vous appuyez sur **ENTER** à la fin d'un calcul, le contenu de l'écran est imprimé sur une ligne et le résultat sur la ligne suivante. Par exemple:

Entrée

300/5 **ENTER**

Papier

300/50 6.

Si vous êtes dans des programmes BASIC, vous pouvez imprimer des données en sortie sur l'imprimante à l'aide de l'instruction LPRINT (pour plus de détails, reportez vous au Chapitre 8). LPRINT peut être utilisée sous la même forme que l'instruction PRINT. La différence entre ces deux instructions est la suivante: si vous utilisez PRINT pour afficher des données supérieures à 16 caractères, les caractères en trop ne pourront être affichés alors qu'avec LPRINT, les caractères en trop seront imprimés sur la seconde et éventuellement sur la troisième ligne en cas de besoin.

Les programmes écrits avec PRINT peuvent être convertis pour pouvoir utiliser l'imprimante; il faut pour cela insérer une instruction PRINT=LPRINT dans le programme (pour plus de détails, reportez vous au Chapitre 8). Toutes les instructions PRINT qui suivent cette instruction auront les mêmes effets qu'une instruction LPRINT. PRINT=PRINT restaurera cette condition à son état normal. Cette structure peut également être insérée dans l'instruction IF d'un programme afin de permettre un choix de sortie au moment où le programme est utilisé.

Vous pouvez également lister vos programmes sur l'imprimante à l'aide de la commande LLIST (pour plus de détails, reportez vous au Chapitre 8). Lorsqu'elle est utilisée sans numéro de ligne, LLIST listera toutes les lignes du programme actuellement en mémoire, par ordre croissant. Des limites peuvent également être indiquées avec LLIST de façon à limiter les lignes qui seront imprimées. Lorsque les lignes d'un programme dépassent 24 caractères, il faut deux ou plusieurs lignes pour imprimer cette ligne de programme. La seconde ligne et les suivantes seront en retrait de 4 ou 6 caractères de façon que le numéro de ligne identifie clairement chaque ligne de programme distinct. (Numéro de ligne 1 à 999: 4, au-delà de 999: 6)

### Attention:

- Si une erreur se produit (ERROR 8) en raison d'une mauvaise alimentation du papier, déchirez le papier et retirez la partie restée dans l'imprimante. Appuyez ensuite sur la touche **[CLS]** pour restaurer la condition d'erreur.
- Lorsque l'imprimante est exposée à des bruits électriques externes assez forts, elle risque d'imprimer des nombres au hasard. Dans ce cas, appuyez sur la touche **[BRK]** pour arrêter l'impression. Mettez l'interface CE-126P hors tension puis à nouveau sous tension et appuyez ensuite sur la touche **[CLS]** pour replacer l'imprimante dans des conditions normales.

Lorsqu'il s'est produit une mauvaise alimentation du papier ou lorsque l'imprimante est exposée à des bruits électriques externes importants pendant l'impression, elle risque de ne pas fonctionner correctement et seul le symbole "BUSY" sera affiché. Dans ce cas, appuyez sur la touche **[BRK]** pour arrêter l'impression. (Retirez le bourrage de papier.) Mettez l'interface CE-126P hors tension puis à nouveau sous tension et appuyez ensuite sur la touche **[CLS]**.

- Lorsque l'interface CE-126P n'est pas utilisé, mettez l'imprimante hors tension de façon à éviter de décharger les piles.

## UTILISATION DE L'INTERFACE CASSETTE

Cette interface vous permet d'enregistrer les programmes et les données de la mémoire de votre ordinateur sur cassettes (il est évident qu'il vous faut également un enregistreur de cassette comme celui que nous vendons pour cet ordinateur; l'enregistreur de cassette en option, modèle CE-152). Une fois sur bande, vous pouvez recharger ces programmes et ces données dans l'ordinateur grâce à une procédure simple.

## CONNEXION DE L'INTERFACE CE-126P A L'ENREGISTREUR DE CASSETTE

Seules, trois connexions sont nécessaires:

1. Connectez la fiche rouge dans la prise MICrophone de l'enregistreur de cassette.
2. Connectez la fiche grise dans la prise "EARphone" de l'enregistreur de cassette.
3. Connectez la fiche noire dans la prise "REMote" de l'enregistreur de cassette.

## ENREGISTREUR DE CASSETTE

Nous vous recommandons d'utiliser l'enregistreur de cassette en option CE-152 avec votre ordinateur de poche. Cet enregistreur est conçu pour enregistrer les programmes et les données de la mémoire du PC-1350 par l'intermédiaire de l'interface cassette CE-126P. Tout programme enregistré peut être recherché et rechargé dans le PC-1350.

Si vous utilisez un enregistreur autre que le CE-152, voici la description des spécifications minimales nécessaires pour qu'il puisse être connecté à l'interface CE-126P:

Elements	Conditions
1. Type d'enregistreur	N'importe quel enregistreur de cassette standard ou de microcassette sous réserve qu'il soit conforme aux exigences indiquées ci-dessous.
2. Plot d'entrée	L'enregistreur doit avoir un plot d'entrée désigné par les lettres "MIC". Ne jamais utiliser le plot "AUX".
3. Impédance en entrée	Le plot d'entrée doit avoir une impédance faible en entrée (200 ~ 1000 OHM)
4. Niveau d'entrée minimum	Au dessous de 3mV ou -50 dB.
5. Plot de sortie	L'un des plots désignés par les lettres "EXT (haut-parleur externe)", "MONITOR", "EAR" ou un plot équivalent.
6. Impédance en sortie	Doit se situer au-dessous de 10 OHM.
7. Niveau de sortie	Doit être supérieur à 1V (Sortie maximum correcte au dessus de 100mW)
8. Distortion	Doit être de 15% dans la gamme de 2KHz à 4KHz.
9. Pleurage et jeu	0,3% maximum (W.R.M.S.)
10. Autre	La vitesse du moteur de l'enregistreur doit être stable.

\* Si la prise fournie avec le CE-126P ne correspond pas au plot d'entrée-sortie de votre enregistreur, il existe sur le marché des prises de conversion spéciales.

**Remarque:** Certains enregistreurs ne conviennent pas en raison de spécifications différentes. De même, certains enregistreurs ayant déjà servi de nombreuses années risquent de ne pas donner les résultats escomptés en raison de modifications de leurs caractéristiques électriques.

## UTILISATION DE L'INTERFACE ET DE L'ENREGISTREUR DE CASSETTE

### REGISTREMENT (SAUVEGARDE) SUR BANDE MAGNETIQUE

Reportez-vous aux remarques sur les bandes magnétiques.

1. Mettez l'interrupteur REMOTE du CE-126P sur OFF.
2. Entrez un programme ou des données dans l'ordinateur.
3. Insérez une cassette dans l'enregistreur.  
Vous devez savoir à quel endroit de la bande vous voulez que soit enregistré le programme.
  - Lorsque vous utilisez une bande, vérifiez que celle-ci se trouve au-delà de l'entête d'effacement (partie en mylar non-magnétique).
  - Lorsque vous utilisez une bande qui comporte déjà des enregistrements, recherchez un endroit où rien n'est enregistré.
4. Insérez la fiche rouge de l'interface dans la sortie MIC de l'enregistreur et la fiche noire dans la sortie REM.
5. Mettez l'interrupteur REMOTE sur ON.
6. Appuyez en même temps sur les boutons RECORD et PLAY de votre enregistreur (pour le placer en mode enregistrement).
7. Entrez les instructions d'enregistrement (CSAVE et PRINT #) et appuyez sur la touche **ENTER** pour déclencher l'enregistrement.

Placez tout d'abord votre ordinateur en mode "RUN" ou "PRO" puis appuyez sur les touches suivantes:

**C S A V E** **SHIFT** " " nom de fichier **SHIFT** " " **ENTER** .

Pour écrire le contenu des données de la mémoire sur bande, appuyez sur les touches suivantes: **P R I N T** **SHIFT** **#** **ENTER** .

Exemple: **C S A V E** **SHIFT** " " **A A** **SHIFT** " " **ENTER**

Lorsque vous appuyez sur la touche **ENTER** la bande commence à se dérouler en laissant un blanc de 8 secondes environ. (La tonalité est enregistrée). Après quoi, le nom de fichier et son contenu sont enregistrés.

8. Une fois l'enregistrement fini, le symbole > s'affiche à l'écran et l'enregistreur s'arrête automatiquement. Votre programme se trouve sur bande (il se trouve également dans la mémoire de l'ordinateur).

Lorsque des données doivent être automatiquement enregistrées au cours de l'exécution d'un programme (Instruction PRINT #) et non par opération manuelle, exécutez les étapes 1 à 6 ci-dessus avant l'exécution du programme.

Pour vous aider à repérer vos programmes sur bande, utilisez le computer de votre enregistreur.

## VARIFICATION DU PROGRAMME SUR BANDE PAR COMPARAISON AVEC LE PROGRAMME EN ORDINATEUR

Reportez-vous aux remarques relatives aux bandes.

Après avoir rechargé un programme en mémoire à partir de la bande, ou transféré un programme de la mémoire sur la bande, vous pouvez vérifier que les deux programmes sont identiques (et ainsi vous assurer que tout est bien avant de continuer à programmer ou avant d'exécuter vos programmes).

1. Mettez l'interrupteur REMOTE sur OFF.
2. Utilisez les boutons de défilement de la bande pour placer la bande juste avant le nom du fichier à vérifier.
3. Insérez la fiche grise dans la sortie "EARphone" et la fiche noire dans la sortie REMote.
4. Remplacez l'interrupteur REMOTE sur ON.
5. Appuyez sur le bouton PLAY de votre enregistreur.
6. Entrez une instruction CLOAD? et appuyez sur la touche **ENTER** pour déclencher l'exécution. Procédez comme ci-dessous: Placez l'unité en mode "RUN" ou "PRO". Appuyez ensuite sur les touches suivantes

Nom de fichier utilisé précédemment

**C** **L** **O** **A** **D** **SHIFT** **?** **SHIFT** **"** **A** **A** **SHIFT** **"** **ENTER**

L'ordinateur de poche recherchera automatiquement le nom de fichier indiqué et comparera le contenu de la bande avec le contenu en mémoire.

Au cours de la vérification, la marque "\*" est affiché dans la position d'extrême droite de la dernière ligne de l'écran. Ce signe "\*" disparaîtra lorsque la vérification sera terminée. Ce signe n'est pas affiché pendant la recherche du nom de fichier mais uniquement lorsque la vérification commence.

(C'est la même chose qui se passe lorsque le premier programme est lu sans nom de fichier).

Si le contenu des programmes est identique, le symbole d'attente d'entrée au clavier > s'affichera à l'écran.

Si le contenu diffère, l'exécution sera interrompue et un code d'erreur 8 s'affichera. Dans ce cas, faites une nouvelle vérification.

## CHARGEMENT D'UN PROGRAMME A PARTIR D'UNE BANDE MAGNETIQUE

Reportez-vous aux remarques relatives aux bandes.

Pour charger, transférer ou lire des programmes et des données à partir d'une bande magnétique dans la mémoire de l'ordinateur, utilisez la procédure ci-dessous:

1. Mettez l'interrupteur REMOTE sur OFF.
2. Insérez la bande dans l'enregistreur. Placez la bande juste avant la partie à lire.
3. Insérez la fiche grise dans la sortie EAR de l'enregistreur et la fiche noire dans la sortie REM.  
(Si votre enregistreur n'a pas de sortie REM, appuyez sur le bouton PAUSE pour vous arrêter temporairement.)
4. Mettez l'interrupteur REMOTE sur ON.
5. Appuyez sur le bouton PLAY de votre enregistreur (pour passer en mode lecture).

Placez le bouton de réglage du VOLUME sur moyen ou maximum.

Placez le bouton du réglage de la tonalité sur les aigus.

6. Entrez les instructions de transfert (CLOAD, INPUT # et appuyez sur la touche **ENTER**) pour lancer l'exécution.

Placez l'ordinateur en mode "RUN". Puis appuyez sur les touches:

**C** **L** **O** **A** **D** **SHIFT** **"** nom de fichier **SHIFT** **"** **ENTER**.

(Pour charger le contenu de la mémoire de données, appuyez sur les touches suivantes: **I** **N** **P** **U** **T** **SHIFT** **#** **ENTER**.)

Exemple: **C** **L** **O** **A** **D** **SHIFT** **"** **A** **A** **SHIFT** **"** **ENTER**

Le nom de fichier indiqué sera automatiquement recherché et son contenu sera transféré dans la mémoire de l'ordinateur. Le signe "\*" s'affiche pendant le chargement du programme indiqué, de la bande dans la mémoire de l'ordinateur.

(C'est la même chose qui se passe lorsque le premier programme est lu sans nom de fichier.) Le signe "\*" disparaît lorsque le chargement est terminé.

7. Une fois le programme chargé en mémoire, l'ordinateur arrête automatiquement le défilement de la bande et le symbole > s'affiche à l'écran.

Pour transférer des données (instructions INPUT #) au cours d'un programme, reprenez les étapes 1 à 5 ci-dessus avant l'exécution du programme.

- Remarques:**
- S'il se produit une erreur (code d'erreur 8 affiché), recommencez au début. Si l'erreur persiste, mettez le volume légèrement plus fort ou plus faible.
  - Si le code d'erreur n'est pas affiché mais si la bande continue à défiler, le transfert des données ne s'effectue pas correctement.
  - Appuyez sur la touche **ON** **BRK** pour arrêter le défilement de la bande et reprenez les étapes ci-dessus.
  - Si l'erreur persiste ou si la bande continue à défiler après plusieurs tentatives pour corriger l'incident, essayez de nettoyer et de démagnétiser les têtes de lecture de l'enregistreur.



## REMARQUES SUR LES BANDES

- 1) Pour tout transfert ou toute vérification, reprenez l'enregistreur que vous aviez utilisé pour l'enregistrement. Si vous prenez un enregistreur différent, il n'est pas possible d'effectuer de transfert ni de vérification.
- 2) Utilisez toujours des bandes de la meilleure qualité possible pour l'enregistrement des programmes et des données (des bandes de qualité moyenne risquent de ne pas fournir les caractéristiques propres aux enregistrements digitaux).
- 3) Les têtes de lecture doivent toujours être propres de même que les autres parties de l'enregistreur – pour cela, utilisez une bande de nettoyage des têtes de lecture.
- 4) Réglage du volume – réglez le volume ou niveau moyen au maximum. Le volume peut être très important lors de la lecture des données à partir de l'enregistreur; faites de légers réglages, selon les besoins, pour que le transfert de données se fasse sans erreur. En réglant le volume légèrement plus fort ou plus faible, vous pouvez obtenir des résultats parfaits à tous les coups.
- 5) Vérifiez que toutes les connexions entre l'ordinateur de poche et l'interface cassette sont bien en place. Vérifiez également que les connexions entre l'interface et l'enregistreur sont bien en place et propres.
- 6) Si un incident survient lorsque vous utilisez le courant du secteur pour l'interface CE-126P, et/ou l'enregistreur, utilisez plutôt les piles (parfois lorsque l'enregistreur est branché sur secteur, des bruits s'ajoutent au signal, ce qui empêche un enregistrement digital correct).
  - Pour connecter l'adaptateur de cossant alternatif à l'interface CE-126P, mettez cette interface hors tension et connectez l'adaptateur.
- 7) Réglage de la tonalité – réglez la tonalité sur les aigus.
- 8) Lorsque vous enregistrez des programmes ou des données sur une bande déjà utilisée, effacez la partie que vous allez utiliser avant d'écrire dessus et d'exécuter la commande d'enregistrement. (Vérifiez que le programme précédent est complètement effacé et qu'il n'en reste pas un bout.)

## CHAPITRE 8 UTILISATION DE LA CARTE RAM EN OPTION

Le PC-1350 peut utiliser les cartes RAM en option CE-201M ou CE-202M. (Les cartes RAM CE-201M et CE-202M ont une capacité de 8 K octets et de 16 K octets respectivement.)

Une carte RAM permet d'accroître la zone des programmes et des données du PC-1350. De plus, même si la carte RAM est retirée de l'ordinateur, le programme qui y est enregistré peut être sauvegardé. Par conséquent, il est possible d'enregistrer différentes programmes sur des cartes RAM; il suffira de changer la carte RAM pour exécuter le programme voulu.

### 1. Insertion de la carte RAM

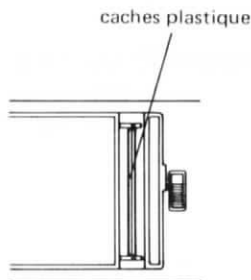
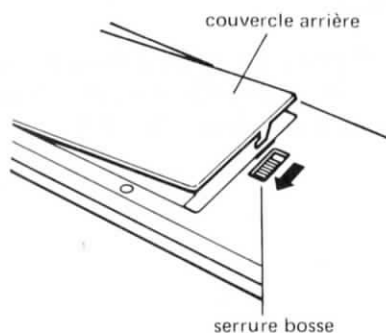
S'il s'agit d'une carte qui n'a jamais été utilisée, n'oubliez pas d'insérer les piles.

(1) Mettez l'ordinateur hors tension.

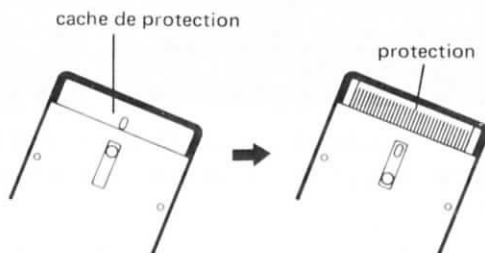
N'oubliez pas de mettre le PC-1350 hors tension avant d'installer ou de retirer une carte RAM.

(2) Faites glisser le couvercle arrière dans la direction de la flèche indiquée sur la figure ci-dessous et retirez le.

**Remarque:** Ne touchez pas les circuits, les différentes pièces ni les caches plastique etc qui se trouvent à l'intérieur du PC-1350.

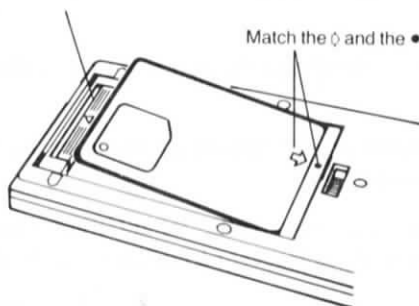


- (3) Faites coulisser le cache de protection de la carte RAM.



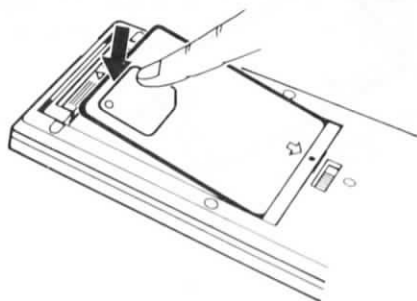
**Remarque:** Le cache doit être tiré au maximum de façon qu'il ne recouvre plus les connexions. Ne touchez pas non plus les connexions.

- (4) Insérez le bout de la carte RAM portant les connexions dans le PC-1350 comme indiqué à la figure ci-dessous.

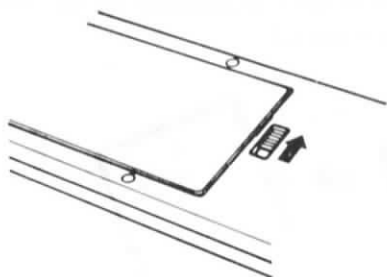


**Remarque:** N'insérez pas la carte RAM à l'envers. Ne l'insérez pas non plus lorsque le cache de protection n'est pas complètement tiré. Vous risqueriez d'abîmer les connexions et également le PC-1350.

Appuyez ensuite doucement sur la plaque de protection des piles. La carte se mettra en place dans son logement.



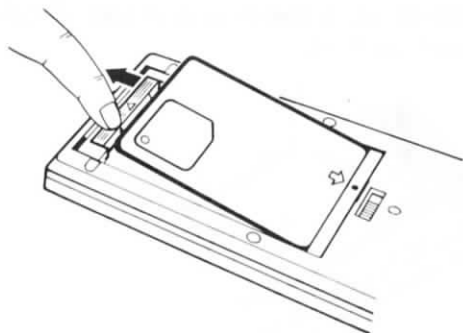
- (5) Remplacez le couvercle arrière et verrouillez le.



**Remarque:** Vérifiez que le couvercle arrière est bien verrouillé. Le PC-1350 ne peut fonctionner s'il n'est pas verrouillé. Si l'ordinateur est mis sous tension alors que le couvercle n'est pas verrouillé, verrouillez-le et mettez l'ordinateur hors tension puis à nouveau sous tension.

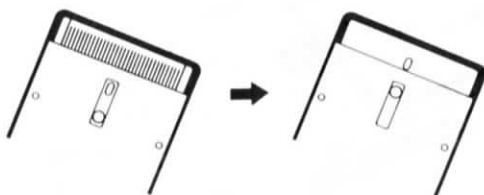
## 2. Retrait de la carte RAM

- (1) Mettez le PC-1350 hors tension.
- (2) Retirez le couvercle arrière du PC-1350 comme indiqué plus haut. Tirez le levier de montage/retrait de la carte dans la direction de la flèche comme indiqué sur la figure. La carte RAM se trouve libérée et peut être retirée.

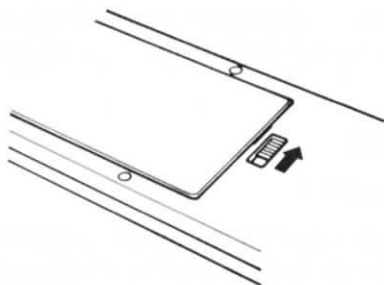


- (3) Refermez immédiatement le cache de protection après avoir retiré la carte RAM.

**Remarque:** N'oubliez pas de refermer complètement le cache (de façon que les connexions soient invisibles).



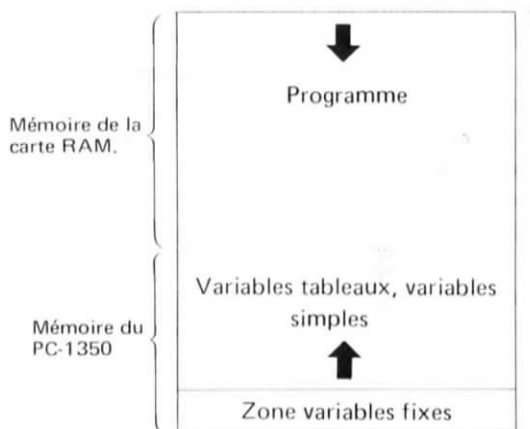
- (4) Remplacez le couvercle arrière du PC-1350 et verrouillez-le.



### 3. Utilisation de la carte RAM.

Lorsqu'une carte RAM sur laquelle rien n'est enregistré est insérée dans le PC-1350, la mémoire du PC-1350 et la mémoire de cette carte sont reliées de façon à ne former qu'une seule zone mémoire pour les programmes et les données.

Comme l'indique la figure ci-dessous, le programme est écrit dans la mémoire de la carte RAM alors que les variables tableaux et les variables simples sont envoyées dans la mémoire du PC-1350.



Le programme écrit sur la carte RAM est conservé (grâce à la pile insérée dans la carte RAM) même si cette carte est retirée de l'ordinateur. Par conséquent, ce programme peut être utilisé à nouveau lorsque la carte RAM est ré-installée dans le PC-1350.

Si vous écrivez sur une carte RAM un programme qui dépasse la capacité mémoire de cette carte, celui-ci sera détruit et ne pourra plus être utilisé une fois la carte retirée. Pour qu'un programme puisse être préservé, une fois la carte RAM retirée, il faut que sa taille ne dépasse pas la capacité de cette carte.

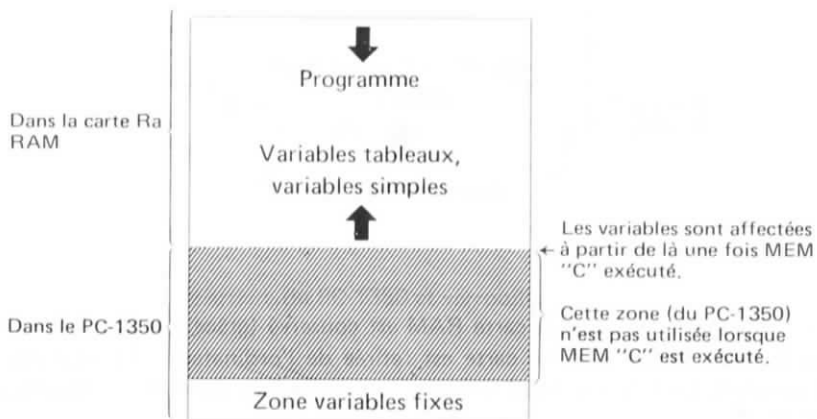
**Remarque:** Etant donné que la zone réservée se trouve dans la mémoire du PC-1350, son contenu ne peut être enregistré sur une carte RAM.

Capacité de la zone de programmes et des données (en octets)

PC-1350+CE-201M	CE-201M uniquement	PC-1350+CE-202M	CE-202M uniquement
11 262	8 142	19 454	16 334

#### 4. Sauvegarde des données sur une carte RAM

Comme l'indique la figure précédente, les variables (variables tableaux et variables simples) sont envoyées dans la mémoire du PC-1350. Par conséquent, si la carte RAM est retirée du PC-1350, les variables seront effacées. La commande MEM "C," peut être spécifiée de façon que les variables soient affectées dans la mémoire de la carte RAM. L'exécution de la commande MEM "C" permet d'affecter les variables dans la mémoire de la carte RAM. Les variables (données) sont conservées même si la carte RAM est retirée du PC-1350.



- \* Le mode MEM "C" est supprimé par la commande MEM "B". Cependant, si la commande MEM "C" est exécutée, les variables tableaux et les variables simples précédemment affectées sont effacées.

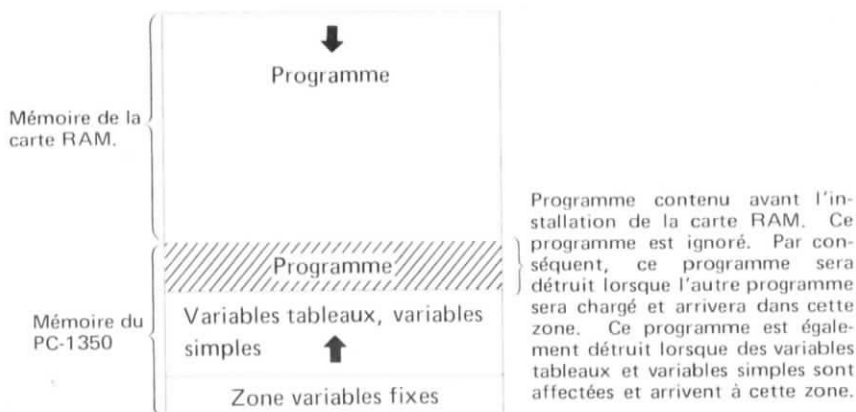
**Remarque:** Le mode MEM "C" est annulé lorsque le programme qui est écrit dépasse la capacité de la carte RAM. On revient au mode MEM "B".

Le mode MEM "C" est enregistrée dans la carte RAM. Le retrait de la carte RAM du PC-1350 rend ce mode invalide. Lorsque la carte RAM est à nouveau installée, ce mode est restauré.

Pour que les données enregistrées sur une carte RAM soient conservées, il faut identifier un programme et l'associer à une touche puis lancer l'exécution du programme à l'aide de la touche définie. Lorsque le programme est exécuté à l'aide de la touche définie. Lorsque le programme est exécuté à l'aide de la commande RUN, les variables tableaux et variables simples sont effacées. Par conséquent, n'utilisez pas la commande RUN. Les variables fixes ne peuvent être sauvegardées sur une carte RAM. Pour sauvegarde des données, utilisez des variables tableaux ou des variables simples.

## 5. Précautions lors de l'utilisation d'une carte RAM

- \* Si la mémoire du PC-1350 comporte un programme lorsqu'une carte RAM est montée, ce programme sera ignoré. Par conséquent, le programme déjà enregistré dans la mémoire du PC-1350 ne pourra être exécuté (ni appelé). Si un programme qui dépasse la capacité de la carte RAM est écrit, le programme enregistré dans la mémoire du PC-1350 sera détruit. Ce programme peut également être détruit si des variables sont affectées dans la mémoire du PC-1350. (Si le programme n'a pas été détruit, il peut être exécuté une fois la carte RAM retirée.)





## Utilisation l'option

- \* Le programme et les données enregistrées sont effacées lorsque la pile de la carte RAM est remplacée (ou retirée). Par conséquent, si le programme enregistré sur une carte RAM, est important, il est recommandé qu'il soit enregistré auparavant sur bande.

Si la pile de la carte RAM est remplacée alors que la carte est installée dans le PC-1350, le contenu de cette carte n'est pas effacé.

Lorsque la nouvelle pile est mise en place, le contenu de la carte RAM peut être retenu pendant environ 34 mois pour la carte CE-201M et 18 mois pour la carte CE-202M (une fois qu'elles sont retirées du PC-1350.)

**Remarque:** N'appuyez pas sur le bouton ALL RESET situé sous le PC-1350 alors qu'une carte RAM est installée.

Lorsque ce bouton est utilisé, le programme et les données enregistrés dans cette carte sont effacés. S'il est nécessaire d'appuyer sur ce bouton, reportez vous à la page 14, et appuyez sur ALL RESET tout en maintenant la touche appropriée enfoncée.

## CHAPITRE 9 REFERENCES BASIC

Le chapitre suivant se divise en trois sections:

**Les commandes:** Ce sont les instructions utilisées en dehors d'un programme et qui permettent de changer le contexte de travail, d'effectuer un certain nombre de tâches et de contrôler les programmes.

**Les verbes:** Ils indiquent une action et sont utilisés dans les programmes pour construire les instructions BASIC.

**Les fonctions:** Ce sont des opérateurs spéciaux utilisés dans les programmes BASIC pour transformer une variable en une autre variable.

Les commandes relatives aux fonctions graphiques, aux entrées-sorties série et aux fonctions texte sont résumées dans les pages suivantes:

Fonction graphique . . . . .	pages 174 ~ 186
Entrée-sortie série . . . . .	pages 187 ~ 203
Fonctions texte . . . . .	pages 204 ~ 206

Les commandes et les verbes sont classés par ordre alphabétique. Chacun d'entre eux se trouve sur une page distincte de façon que vous puissiez vous y référer plus facilement. Les tableaux ci-après indiquent le contenu de chaque section de façon que vous puissiez rapidement identifier la catégorie à laquelle appartient un opérateur. Les fonctions sont regroupées en 4 catégories et classées par ordre alphabétique à l'intérieur de ces catégories.

**Remarque:** Les commandes, les verbes et les fonctions doivent être frappés en majuscules (qui est le mode normal).

## COMMANDES

### Contrôle des programmes

CONT  
GOTO\*  
NEW  
RUN

### Contrôle cassette

CLOAD  
CLOAD?  
CSAVE  
INPUT #\*  
MERGE  
PRINT #\*

### Mise au point

LIST  
LLIST  
TROFF\*  
TRON\*

### Fonctions graphiques

GCURSOR\*  
GPRINT\*  
LINE\*  
POINT  
PRESET\*  
PSET\*

### E-S Série

CLOSE\*  
CONSOLE\*  
INPUT #1\*  
LLIST  
LOAD  
LPRINT\*  
OPEN\*  
OPENS\$  
PRINT #1\*  
SAVE

### Contrôle des variables

CLEAR\*  
MEM\*  
DIM\*

### Contrôle des angles

DEGREE\*  
GRAD\*  
RADIAN\*

### Divers

BEEP\*  
PASS  
RANDOM\*  
USING\*  
WAIT\*

### Fonctions texte

BASIC  
TEXT

\* Ces commandes sont également des verbes BASIC. Leur effet en tant que commandes est identique à leur effet en tant que verbe. C'est pourquoi elles ne sont pas décrites dans la section des commandes. Pour plus d'informations, reportez vous à la section des verbes.

## VERBES

### Contrôle et branchement

CHAIN  
 END  
 FOR...TO...STEP  
 GOSUB  
 GOTO  
 IF... THEN  
 NEXT  
 ON...GOSUB  
 ON...GOTO  
 RETURN  
 STOP

### Affectation et déclaration

CLEAR  
 DIM  
 LET

### Entrée et sortie

AREAD  
 CSAVE  
 CURSOR  
 DATA  
 GCURSOR  
 GPRINT  
 INPUT  
 INPUT #  
 INPUT #1  
 LINE  
 LOAD  
 LPRING  
 PAUSE  
 PRESET  
 PRINT  
 PRINT #  
 PRINT #1  
 PSET  
 READ  
 RESTORE  
 USING  
 WAIT

### Divers

BEEP  
 CLOSE  
 CLS  
 CONSOLE  
 DEGREE  
 GRAD  
 OPEN  
 RADIAN  
 RANDOM  
 REM  
 TROFF  
 TRON

## FONCTIONS

Pseudovariable

INKEY\$  
MEM  
MEM\$  
OPENS  
PI  
POINT

Fonctions relatives aux  
chaînes de caractères

ASC  
CHR\$  
LEFT\$  
LEN  
MID\$  
RIGHT\$  
STR\$  
VAL

Fonctions numériques

ABS  
ACS  
ASN  
ATN  
COS  
DEG  
DMS  
EXP  
INT  
LOG  
LN  
RND  
SGN  
SIN  
SQR  
TAN

## COMMANDES

- 1 CLOAD
- 2 CLOAD "Nom de fichier"

Abréviations: CLO, CLOA

Voir également: CLOAD?, CSAVE, MERGE, PASS

### OBJET

La commande CLOAD permet de charger un programme sauvegardé sur cassette. Elle ne peut être utilisée qu'avec les unités en option CE-126P et CE-152.

### UTILISATION

La première forme de la commande CLOAD efface les programmes existants en mémoire et charge le premier programme enregistré sur bande, à partir de la position à laquelle se trouve la bande.

La seconde forme de cette commande permet d'effacer la mémoire, de rechercher sur la bande le programme dont le nom est indiqué par "nom de fichier" et de charger ce programme.

Si le PC-1350 est en mode PRO ou en mode RUN, le programme est chargé dans la mémoire des programmes. Si le PC-1350 est en mode ReSerVe, le programme est envoyé dans la mémoire de réserve. Il convient donc de faire attention qu'aucun programme ne soit chargé dans la mémoire de réserve ni que des caractères de réserve soient envoyés dans la mémoire des programmes.

### EXEMPLES

CLOAD Charge en mémoire le premier programme trouvé sur la bande.

CLOAD "PRO3" Recherche sur la bande le programme appelé "PRO3" et le charge.

Remarques: 1. L'ordinateur ne peut savoir si les données enregistrées correspondent à un programme ou à des données destinées à la mémoire de réserve. Par conséquent, si vous ne désignez pas le mode correct, le contenu de la mémoire de réserve peut être transféré dans la zone des programmes ou vice-versa, ce qui rendra les données inexploitable. Dans ce cas, restaurez l'ordinateur en appuyant sur le bouton RESET situé au dos de l'ordinateur.

**Commandes**  
**CLOAD**

2. Si l'ordinateur ne trouve pas le nom de fichier indiqué, il continuera à rechercher ce nom même après que la bande soit arrivée en fin de course. Dans ce cas, appuyez sur la touche **ON**  
**BRK** pour arrêter la recherche. Cette remarque est valable également pour les commandes MERGE, CHAIN, CLOAD? et INPUT # décrites ultérieurement.
3. Si une erreur se produit au cours de l'exécution de CLOAD ou CHAIN (décrite ultérieurement), le programme enregistré dans la mémoire ne sera plus valide.

- Au cours du chargement, le signe "\*" est affiché dans le coin inférieur droit de l'écran. Ce signe "\*" disparaît lorsque le chargement est terminé. Aucun signe "\*" ne s'affiche pendant la recherche puisque le chargement n'est pas encore commencé.

**Remarque:** Le signe "\*" disparaît en général à la fin du transfert. Cependant, si une commande telle que CLOAD est exécutée à partir de la 4ème ligne de l'écran, le signe "\*" ne disparaîtra pas mais remontera à la troisième ligne.

Le symbole > s'affiche à nouveau à la fin du chargement.

1 CLOAD?

2 CLOAD? "Nom de fichier"

Abréviations: CLO.?, CLOA?

Voir également: CLOAD, CSAVE, MERGE, PASS

## OBJET

La commande CLOAD? permet de comparer un programme sauvegardé sur bande avec un programme enregistré en mémoire. Cette commande ne peut être utilisée qu'avec les unités en option CE-126P et CE-152.

## UTILISATION

Pour vérifier qu'un programme a été correctement sauvegardé, rembobinez votre bande jusqu'au début du programme et utilisez la commande CLOAD?.

La première forme de la commande CLOAD? permet de comparer le programme enregistré en mémoire avec le premier programme enregistré sur la bande à partir de la position en cours.

La seconde forme de cette commande permet de rechercher sur la bande le programme dont le nom est indiqué par nom de fichier puis de comparer ce programme au programme enregistré en mémoire.

## EXEMPLES

CLOAD? Compare le premier programme de la bande avec le programme en mémoire.

CLOAD? "PRO3" Recherche sur la bande le programme appelé "PRO3" et le compare au programme enregistré en mémoire.

\* Le signe "\*" s'affiche dans le coin inférieur droit de l'écran lorsque le programme est en cours de vérification. En fin de vérification, le signe "\*" disparaît et le symbole > s'affiche.

Cependant, si la commande CLOAD? est exécutée à partir de la quatrième ligne de l'écran, le signe "\*" ne disparaît pas mais remonte à la troisième ligne.



## 1 CONT

Abréviations: C., CO., CON.

Voir également: Verbes RUN, STOP

## OBJET

La commande CONT permet de reprendre un programme qui a été temporairement arrêté.

## UTILISATION

Lorsque le verbe STOP est utilisé pour arrêter temporairement un programme en cours d'exécution, il faut, pour reprendre ce dernier, entrer CONT en réponse au symbole d'attente d'entrée au clavier (>).

Lorsqu'un programme a été temporairement arrêté à l'aide de la touche **BRK**, on peut, pour reprendre son exécution, entrer CONT en réponse au symbole d'attente d'entrée au clavier.

CONT permet également de reprendre un programme arrêté temporairement à cause de commandes telles que PRINT et GPRINT.

## EXEMPLES

CONT      Permet de reprendre un programme dont l'exécution a été interrompue.

- 1 CSAVE
- 2 CSAVE "nom de fichier"
- 3 CSAVE "mot de passe"
- 4 CSAVE "nom de fichier", "mot de passe"

Abréviations: CS., CSA., CSAV

Voir également: CLOAD, CLOAD?, MERGE, PASS

## OBJET

La commande CSAVE permet de sauvegarder un programme sur cassette. Cette commande ne peut être utilisée que l'ordque les unités CE-126P et CE-152 en option sont connectées à l'ordinateur.

## UTILISATION

La première forme de la commande CSAVE permet d'écrire tous les programmes en mémoire sur cassette, sans spécifier de nom de fichiers.

La seconde forme de cette commande permet d'écrire sur la cassette tous les programmes en mémoire et de leur attribuer le nom de fichier indiqué.

La troisième forme de cette commande permet d'écrire sur la cassette tous les programmes en mémoire sans leur donner de nom de fichier mais en leur attribuant un mot de passe. Les programmes sauvegardés avec un mot de passe peuvent être chargés par quiconque mais seules les personnes qui connaissent le mot de passe peuvent en faire liste ou les modifier. (Se reporter à la commande PASS).

La quatrième forme de cette commande permet d'écrire sur cassette tous les programmes en mémoire en leur attribuant le nom de fichier et le mot de passe indiqués.

Si le PC-1350 est en mode PRO ou en mode RUN, ce sont les programmes en mémoire qui sont transférés sur bande. S'il se trouve en mode ReSerVe, c'est le contenu de la mémoire de réserve qui est transféré sur bande.

## EXEMPLES

CSAVE "PRO-3", "SECRET"

Sauvegarde les programmes en mémoire sur la bande et leur attribue le nom de PRO-3. Ces programmes seront protégés par le mot de passe "SECRET".

1 **GOTO** expression

Abréviations: G., GO., GOT.

Voir également: RUN

## OBJET

La commande **GOTO** permet de déclencher l'exécution d'un programme.

## UTILISATION

La commande **GOTO** peut être utilisée à la place de la commande **RUN** pour déclencher l'exécution d'un programme à partir du numéro de ligne spécifié dans l'expression.

Il existe cependant des différences entre les commandes **GOTO** et **RUN**:

- 1) La valeur de l'intervalle pour **WAIT** n'est pas restaurée.
- 2) Le format d'affichage fixé par les instructions **USING** n'est pas effacé.
- 3) Les variables et les tableaux sont préservés.
- 4) **PRINT = LPRINT** n'est pas restauré.
- 5) Le pointer pour l'instruction **READ** n'est pas restauré.
- 6) La spécification de curseur est conservée.
- 7) La direction horizontale du curseur graphique est ramenée à 0. Les indications pour la direction verticale sont conservées.
- 8) Le circuit d'entrée-sortie n'est pas fermé.

Un programme déclenché au moyen de la commande **GOTO** est exécutée de la même manière que s'il avait été déclenché à l'aide de la touche **DEF**.

## EXEMPLES

**GOTO 100** Déclenche l'exécution du programme à partir de la ligne 100.

- 1 LIST
- 2 LIST n° de ligne
- 3 LIST "label"

Abréviations: L., LI., LIS.

Voir également: LLIST

## OBJET

La commande LIST permet d'afficher un programme.

## UTILISATION

La commande LIST ne peut être utilisée que lorsque l'ordinateur est en mode PRO.

- \* La première forme de la commande LIST permet d'afficher un programme à partir de sa première ligne, jusqu'à ce que l'écran soit plein.
- \* La seconde forme de cette commande permet d'afficher un programme à partir de la ligne indiquée par le numéro de ligne et jusqu'à ce que l'écran soit plein.
- \* S'il n'y a pas de ligne correspondant au numéro spécifié, le programme sera affiché à partir de la ligne ayant le premier numéro de ligne supérieur à celui spécifié.
- \* La troisième forme de cette commande permet d'afficher un programme à partir de la ligne écrite avec le label indiqué et jusqu'à ce que l'écran soit plein.
- \* Lorsque des programmes sont fusionnés à l'aide de la commande MERGE, la commande LIST s'applique au dernier programme. Cependant, si le nom spécifié comme label (identification) dans la troisième forme de la commande LIST n'existe pas dans le dernier programme, ce nom est recherché en séquence à partir du premier programme. Si le nom spécifié comme label est trouvé, la ligne sur laquelle il se trouve est affichée.  
Si un mot de passe a été spécifié pour le programme, la commande LIST est ignorée.

## EXEMPLES

LIST 100      Affiche le programme à partir de la ligne 100.

- 1 LLIST
- 2 LLIST expression
- 3 LLIST expression 1, expression 2
- 4 LLIST expression
- 5 LLIST expression

Abréviations: LL., LLI., LLIS.

Voir également: LIST

## OBJET

La commande LLIST permet d'imprimer un programme sur l'imprimante en option.

Lorsque l'interface d'entrée sortie série est ouverte à la suite d'une commande OPEN, la commande LLIST permet de sortir le programme sur le terminal de l'interface d'entrée sortie série. "Voir page 000". Pour que la commande d'impression soit renvoyée sur l'imprimante, il faut exécuter la commande CLOSE.

## UTILISATION

Lorsque l'interface d'entrée-sortie série est ouvert à la suite d'une commande OPEN, la commande LLIST envoie le programme à cette interface (voir page 191).

La commande LLIST peut être utilisée en mode RUN ou en mode PRO.

La première forme de cette commande imprime tous les programmes se trouvant en mémoire.

La seconde forme de LLIST n'imprime que la ligne de programme dont le numéro est donné dans l'expression.

La troisième forme de LLIST imprime les instructions à partir du numéro de ligne égal ou supérieur à la valeur d'expression 1 jusqu'au numéro de ligne égal ou supérieur à la valeur d'expression 2 jusqu'au numéro de ligne égal ou supérieur à la valeur d'expression 2. Il doit y avoir au minimum deux lignes entre les deux nombres.

La quatrième forme de LLIST imprime les lignes de programme commençant à la ligne dont le numéro est donné par l'expression.

La cinquième forme de LLIST imprime toutes les lignes du programme jusques et y compris la ligne dont le numéro est donnée dans l'expression.

\* Lorsque des programmes sont fusionnés à l'aide de la commande MERGE, la commande LLIST n'a d'effet que pour le dernier programme. Pour lister un des programmes précédents, utiliser

LLIST "label"

Si un mot de passe a été défini la commande LLIST est ignorée.

### EXEMPLES

LLIST 100, 200

Liste les instructions entre les numéros de ligne 100 et 200.

- 1 MERGE
- 2 MERGE "nom de fichier"  
(effectif pour l'utilisation manuelle en mode PRO ou en mode RUN".)

Abréviations: MER., MERG.

Voir également: CLOAD

## OBJET

La commande MERGE permet de charger un programme sauvegardé sur bande et de le fusionner avec un programme déjà en mémoire.

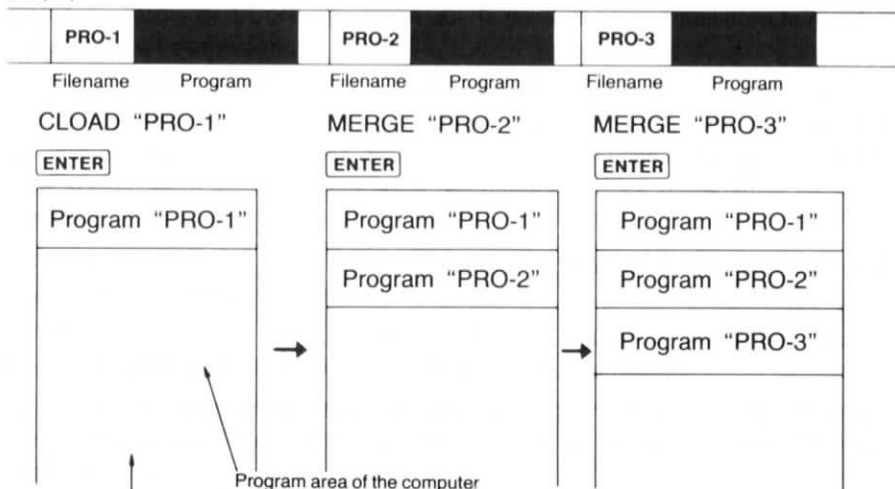
## UTILISATION

La commande MERGE retient le programme déjà enregistré dans le PC-1350 puis chargé un programme enregistré sur bande. Par conséquent, plusieurs programmes différents peuvent être stockés en même temps dans la mémoire du PC-1350.

## EXEMPLES

Lorsque les programmes appelés PRO-1 à PRO-3 doivent être chargés en mémoire et que PRO-1 est chargé au moyen de la commande CLOAD et PRO-2 et PRO-3 sont transférés dans l'ordinateur au moyen de la commande MERGE. La mémoire se présente ainsi:

(Tape)



Transfer the first program to the computer using the CLOAD command.

Les programmes chargés à l'aide de la commande MERGE sont enregistrés comme dans l'exemple ci-après. Les programmes sont traités comme suit par leur numéro de ligne.

● Si le premier numéro de ligne du programme chargé au moyen de la commande MERGE est supérieur au dernier numéro de ligne du programme chargé précédemment, les deux programmes sont considérés comme un seul et même programme.

● Si le premier numéro de ligne du programme chargé au moyen de la commande MERGE est inférieur au dernier numéro de ligne du programme chargé précédemment, les programmes sont considérés comme distincts.

Dans l'exemple ci-dessus, les numéros de ligne des programmes PRO-1, PRO-2 et PRO-3 sont respectivement 10-200, 50-150 et 160-300; PRO-1 et PRO-2 sont considérés comme distincts alors que PRO-2 et PRO-3 sont considérés comme un seul et même programme dont les numéros de ligne vont de 50 à 300.

\* Lorsque des programmes sont chargés avec la commande MERGE, il peut y avoir dans la mémoire du PC-1350 deux ou plusieurs programmes ayant les mêmes numéros de ligne. Dans ce cas, les commandes RUN et GOTO exécutées (expression RUN, expression GOTO) ne concernent que le dernier programme fusionné. Les programmes précédents ne pourront être exécutés.

C'est pourquoi il faut ajouter un label au début du programme à exécuter et l'exécuter à l'aide de la touche **DEF**.

Une fois la commande MERGE exécutée, seul le dernier programme fusionné pourra être édité, les programmes chargés avant lui ne pourront l'être. Par conséquent il faut ajouter un label au programme avant de le fusionner avec le programme suivant.

## FUSION DE PROGRAMMES PROTEGES PAR MOT DE PASSE

Lorsque la commande MERGE est utilisée pour charger des programmes avec mot de passe (protégés par mot de passe), le traitement de ces programmes diffère légèrement suivant que les programmes dans la mémoire de l'ordinateur sont ou non protégés.

Lorsque les programmes en mémoire sont protégés:

les programmes protégés par mot de passe ne peuvent être chargés.

Lorsque les programmes en mémoire ne sont pas protégés:

si les programmes protégés par mot de passe sont chargés à l'aide de la commande MERGE, tous les programmes de la mémoire de l'ordinateur deviennent protégés.

Lorsque les programmes en mémoire sont protégés, même les programmes sans mot de passe deviennent protégés lorsqu'ils sont chargés à l'aide de la commande MERGE.



## EXECUTION DE PROGRAMMES FUSIONNES

"A" PRO-1
"B" PRO-2
"C" PRO-3

La figure ci-contre représente la mémoire lorsque PRO-1 a été chargé et que PRO-2 et PRO-3 ont été chargés à l'aide de la commande MERGE. Si l'exécution d'un programme est lancée au moyen de RUN ou GOTO (expression RUN ou GOTO) c'est PRO-3 qui sera exécuté. Si l'exécution du programme est déclenchée au moyen de la commande RUN "label" ou GOTO "label" ou en appuyant sur la touche **DEF**, le label indiqué est recherché en commençant à partir de PRO-3.

Si ce label n'est pas trouvé dans PRO-3, la recherche se poursuit dans PRO-1. S'il n'est toujours pas trouvé dans PRO-1, il est recherché dans PRO-2. Lorsque le label est trouvé, le programme est exécuté à partir de la ligne comportant le label.

Etant donné la manière dont le label est recherché, si le même label est utilisé dans PRO-1, PRO-2 et PRO-3, PRO-1 et PRO-2 ne peuvent pas être exécutés.

En outre, si la commande LLIST "A", "C" **ENTER** est spécifiée, le label est d'abord recherché dans PRO-3 en premier. Il s'ensuivra une erreur puisque "C" apparaît en premier.

## 1 NEW

Abréviations: Aucune

### OBJET

La commande NEW permet d'effacer les programmes de la mémoire ou le contenu de la mémoire de réserve.

### UTILISATION

Lorsqu'elle est utilisée en mode PRO, la commande NEW efface tous les programmes et toutes les données actuellement en mémoire. (Les programmes protégés par mot de passe ne peuvent être effacés).

Lorsqu'elle est utilisée en mode ReSerVe, la commande NEW efface toutes les positions de la mémoire de réserve.

La commande NEW ne peut être spécifiée en mode RUN; sinon elle provoque une erreur (ERROR 9).

### EXEMPLES

NEW                      Efface les programmes ou la mémoire de réserve.

1 PASS "chaîne de caractères"

Abréviations: PA., PAS.

Voir également: CSAVE, CLOAD, NEW

## OBJET

La commande PASS permet de définir et d'annuler des mots de passe.

## UTILISATION

Les mots de passe servent à protéger les programmes qui ainsi ne peuvent être ni revus ni utilisés par d'autres utilisateurs. Un mot de passe se compose d'un chaîne de sept caractères au maximum qui peuvent être soit des lettres de l'alphabet soit les symboles spéciaux suivants: ! # \$ % & ( ) \* + - / , . : ; < = > ? @  
 $\sqrt{\quad}$   $\pi$   $\wedge$

Lorsqu'une commande PASS a été indiquée, les programmes en mémoire sont protégés. Un programme protégé par mot de passe ne peut être ni examiné ni modifié en mémoire. Il ne peut ni être écrit sur bande ni listé au moyen des commandes LIST ou LLIST. Il n'est pas non plus possible d'ajouter ou de supprimer des lignes à ce programme. Si plusieurs programmes se trouvent en mémoire et si la commande PASS est entrée, tous les programmes de la mémoire deviennent protégés. La seule façon de supprimer cette protection consiste à exécuter une autre commande PASS en spécifiant le même mot de passe.

**Remarque:** Lorsqu'un mot de passe de sept caractères ou plus est indiqué, seuls les sept premiers caractères sont valides et utilisés pour affecter un mot de passe ou le supprimer.

Appuyer sur **ENTER** après le mot de passe.

L'écriture de caractères ou de symboles après le mot de passe provoque une erreur et dans ce cas le mot de passe ne peut être utilisé.

(exemple) PASS "ABCDEFGH": A = 123 **ENTER** → Error

## EXEMPLES

### PASS "SECRET"

Donne le mot de passe "SECRET" à tous les programmes de la mémoire.

#### Mot de passe et carte RAM

Le mot de passe des programmes de la mémoire du PC-1350 et des programmes des cartes RAM doivent être différents.

Lorsqu'une carte RAM est installée et que le mot de passe est défini, si cette carte est retirée, le programme qui y est enregistré devient protégé. Le mot de passe des programmes du PC-1350 ne sera pas fixé à ce stade.

D'autre part, si une carte RAM est installée après que les programmes de la mémoire du PC-1350 ont été protégés (c'est-à-dire qu'un mot de passe a été défini), le programme de la carte RAM ne sera pas protégé.

(Cependant, le mot de passe défini précédemment pour le programme de la carte RAM est toujours valide.)

1 RUN

2 RUN n° de ligne

Abréviations: R., RU.

Voir également: GOTO, MERGE

## OBJET

La commande RUN permet de lancer l'exécution d'un programme en mémoire.

## UTILISATION

La première forme de la commande RUN permet d'exécuter un programme en commençant au numéro de ligne le moins élevé.

La seconde forme de la commande RUN permet d'exécuter le programme en commençant au numéro de ligne spécifié.

Lorsque des programmes sont fusionnés au moyen de la commande MERGE, le dernier programme fusionné sera exécuté avec le format (1) ou "RUN expression" en format (2).

Entre RUN et GOTO, les points suivants diffèrent:

- 1) La valeur de l'intervalle pour WAIT est restauré.
- 2) Le format d'affichage établi par la commande USING est effacé.
- 3) Les variables et tableaux autres que les variables fixes sont effacés.
- 4) L'état PRINT = PRINT est défini.
- 5) Le pointeur pour l'instruction READ est restauré au début de l'instruction DATA.
- 6) La spécification du curseur est effacée.
- 7) La position du curseur graphique est effacée et remise à (0, 7).
- 8) Ferme le circuit d'entrée-sortie série (point de branchement série).

## EXEMPLES

RUN 100      Exécute le programme qui commence à la ligne 100.

# VERBES

## 1 AREAD nom de variable

Abréviations: A., AR., ARE., AREA.

Voir également: Le verbe INPUT et les explications sur l'emploi de la touche DEF au Chapitre 6

### OBJET

Le verbe AREAD permet d'entrer une valeur unique dans un programme lancé au moyen de la touche **DEF**.

### UTILISATION

Lorsqu'un programme est identifié par une lettre de façon à pouvoir être lancé au moyen de la touche **DEF**, le verbe AREAD permet d'entrer une valeur de départ unique sans avoir à utiliser le verbe INPUT. Le verbe AREAD doit figurer sur la première ligne du programme, après l'identification (label). S'il figure ailleurs, il sera ignoré. Il ne doit y avoir qu'une seule valeur de départ par programme, ce peut être une variable numérique ou alphanumérique.

Pour utiliser le verbe AREAD, frappez la valeur voulue en mode RUN, puis appuyez sur la touche **DEF** et entrez la lettre d'identification du programme. Si la variable utilisée est une variable alphanumérique, il n'est pas nécessaire qu'elle soit entre guillemets.

### EXEMPLES

```
10 "X": AREAD N
20 PRINT N ^ 2
30 END
```

Lorsque vous entrez "7 **DEF** X" vous obtiendrez l'affichage "49".

### REMARQUES:

1. Lorsque le symbole ">" est affiché à l'écran au début de l'exécution du programme, la variable désignée est effacée.

2. Lorsque les données sont affichées au moyen du verbe PRINT, au début de l'exécution du programme, voici ce qui est enregistré:

Exemple: { Lorsque le programme ci-dessous est exécuté:  
10 "A": PRINT "ABC", "DEFG"  
20 "S": AREAD A\$: PRINT A\$  
RUN mode  
DEF A → ABC DEFG  
DEF S → DEFG }

- Lorsque PRINT expression numérique, expression numérique, expression numérique... expression numérique ou PRINT "chaîne de caractère", "chaîne de caractère"... "chaîne de caractère" est affiché à l'écran, ce sont les données affichées en dernier qui sont enregistrées.
- Lorsque l'écran indique PRINT expression numérique; expression numérique; expression numérique..., c'est les données affichées en premier (à l'extrême gauche) qui sont enregistrées.
- Lorsque l'écran indique PRINT "chaîne de caractères", "chaîne de caractères", "chaîne de caractères"... , c'est le contenu de "chaîne de caractères" indiqué en dernier (à l'extrême droite) qui est enregistré.

1 BEEP expression

Abréviations: B., BE., BEE.

## OBJET

Le verbe BEEP provoque l'émission d'une tonalité.

## UTILISATION

Le verbe BEEP provoque l'émission par le PC-1350 d'une ou plusieurs tonalités de 4kHz. Le nombre de "bips" dépend de l'expression qui doit être numérique (nombre positif inférieur à 9,999999999E+99).

L'expression est calculée mais seule la partie entière du chiffre obtenu est utilisée pour déterminer le nombre de "bips".

Le verbe BEEP peut également être utilisé comme commande avec des littéraux numériques et des variables pré-définies. Dans ce cas le "bip" se produit immédiatement après que la touche **ENTER** a été enfoncée.

## EXEMPLES

10 A = 5 : B\$ = "9"

20 BEEP 3

Permet d'obtenir 3 "bips"

30 BEEP A

Permet d'obtenir 5 "bips"

40 BEEP (A+4)/2

Permet d'obtenir 4 "bips"

50 BEEP B\$

Cette ligne est erronée et provoquera l'affichage du message ERROR 9.

60 BEEP -4

Ne provoquera aucun bip et aucun message d'erreur.



- 1 **CHAIN**
- 2 **CHAIN** expression
- 3 **CHAIN** "nom de fichier"
- 4 **CHAIN** "nom de fichier" expression

Abréviations: CH., CHA., CHAI.

Voir également: CLOAD, CSAVE, et RUN

## OBJET

Le verbe CHAIN permet de lancer l'exécution d'un programme qui a été enregistré sur cassette. Il ne peut être utilisé que lorsque les unités en option CE-126 et CE-152 sont connectées.

## UTILISATION

Le verbe CHAIN ne peut être utilisé que lorsque un ou plusieurs programmes ont été enregistrés sur cassette. Dans ce cas, lorsque le verbe CHAIN est rencontré dans un programme en cours d'exécution, un programme de la cassette est chargé en mémoire puis exécuté.

La première forme du verbe CHAIN provoque le chargement du premier programme de la bande et son exécution à partir du numéro de ligne le moins élevé. L'effet de ce verbe est identique à celui de CLOAD et RUN lorsque l'ordinateur est en mode RUN.

La seconde forme de CHAIN provoque le chargement du premier programme enregistré sur la bande et son exécution à partir du numéro de ligne indiqué par l'expression.

La troisième forme de CHAIN provoque la recherche, sur la bande, du programme dont le nom est indiqué par "nom de fichier", le chargement de ce programme et son exécution à partir du numéro de ligne le moins élevé.

La quatrième forme de CHAIN provoque la recherche, sur la bande, du programme dont le nom est indiqué "nom de fichier", le chargement de ce programme et son exécution à partir du numéro de ligne indiqué par l'expression.

## EXEMPLES

10 CHAIN

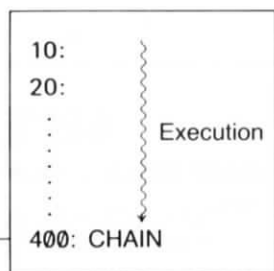
Charge le premier programme de la bande et commence à l'exécuter à partir du numéro de ligne le moins élevé.

20 CHAIN "PRO-2", 480

Recherche sur la bande le programme appelé PRO-2, le charge et commence à l'exécuter à partir du numéro de ligne 480.

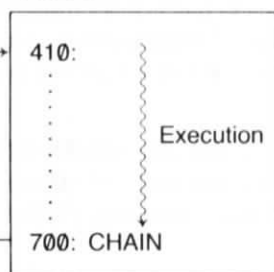
Supposons par exemple que vous ayez trois parties de programme appelés PRO-1, PRO-2, et PRO-3. Chacune de ces parties se termine par le verbe CHAIN.

"PRO-1"



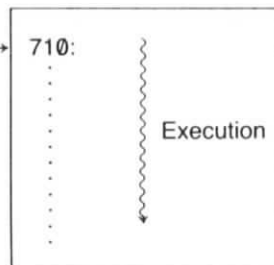
400: CHAIN "PRO-2", 410

"PRO-2"



700: CHAIN "PRO-3", 710

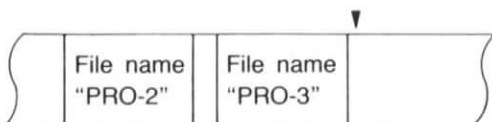
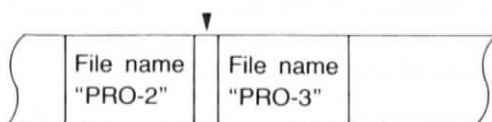
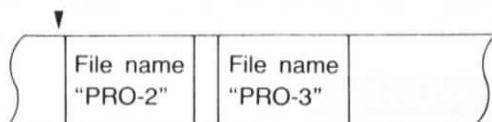
"PRO-3"



990: END

Magnetic tape

("▼" indicates the position of the tape recorder head.)



Au cours de l'exécution, lorsque l'ordinateur rencontre le verbe CHAIN, la partie de programme suivante est appelée en mémoire et exécutée. Ainsi les trois sections sont toutes exécutées.

**REMARQUE:** Lorsque'un programme comportant une commande CHAIN est chargé à partir de la bande au moyen d'une commande MERGE, vérifiez que la commande CHAIN est correcte.

## 1 CLEAR

Abréviations: CL., CLE., CLEA.

Voir également: DIM

### OBJET

Le verbe CLEAR permet d'effacer toutes les variables utilisées dans le programme et de restaurer à 0 ou à NUL toutes les variables pré-affectées.

### UTILISATION

Le verbe CLEAR permet de récupérer la place utilisée pour enregistrer les variables. Ceci peut être utile lorsque les variables utilisées dans la première partie d'un programme ne sont plus nécessaires dans la seconde partie et que la place disponible est limitée. CLEAR peut être également utilisé au début d'un programme lorsque plusieurs programmes résident en mémoire et que vous voulez effacer la place utilisée par l'exécution de programmes antérieurs.

CLEAR ne libère pas la place utilisée par les variables A–Z, A\$–Z\$, ou A(1)–A(26) (sans déclaration DIM) puisqu'un emplacement de mémoire leur est affecté en permanence (voir Chapitre 4). CLEAR restaure les variables numériques à 0 et les variables alphanumériques à NUL.

### EXEMPLES

10 A = 5: DIM C(5)

20 CLEAR

Libère la place attribué à C( ) et restaure A à 0.

**1 CLS**

Abréviations: aucune

Voir également: CURSOR

**OBJET**

La commande CLS permet d'effacer l'écran.

**UTILISATION**

Cette commande efface l'écran et ramène à 0 la première position d'affichage.

**EXEMPLES**

```
10: WAIT 3
20: INPUT A$
30: FOR B = 0 TO 23
40: CLS
50: CURSOR B, 1
60: PRINT A$
70: NEXT B
80: CLS
90: END
```

Ce programme affiche les données d'entrée en les déplaçant de la gauche vers la droite sur l'écran (de la ligne supérieure à la ligne inférieure). Chaque fois que la boucle FOR-NEXT des lignes 30 à 70 est exécutée, l'écran est effacé au moyen de la commande CLS, la position de début d est décalée au moyen de la commande CURSOR et le contenu de A\$ est affiché au moyen de la commande PRINT. En écrivant et en effaçant l'écran de cette manière celui-ci semble bouger. (Supprimez la ligne 40 et exécutez à nouveau le programme. Remarquez la différence.)

- 1 **CURSOR** expression 1, expression 2
- 2 **CURSOR** expression:
- 3 **CURSOR**

Abréviations: CU., CUR., CURS., CURSO.

Voir également: GCURSOR, CLS, INPUT, PRINT, PAUSE

## OBJET

Indique la position de début de l'affichage (n° de colonne).

## UTILISATION

Les formats 1 et 2 indiquent la position de début de l'affichage sous forme de position de caractère pour les données affichées par la commande PRINT, la commande PAUSE, etc.

\* La position d'affichage est spécifiée comme suit:



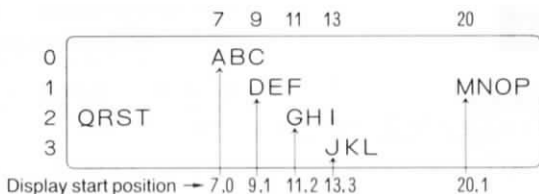
Une position sur l'écran est définie par ses positions verticale et horizontale. Les valeurs de expression 1 et expression 2 indiquent respectivement la position horizontale et la position verticale.

Par conséquent, vérifiez que la valeur d'expression 1 est comprise entre 0 et 23 et celle d'expression 2 entre 0 et 3.

## EXEMPLES

- 5 CLS
- 10 CURSOR 7,0: PRINT "ABC"
- 20 CURSOR 9,1: PRINT "DEF"
- 30 CURSOR 11,2: PRINT "GHI"
- 40 CURSOR 13,3: PRINT "JKL"
- 50 CURSOR 20,1: PRINT "MNOPQRST"

L'exécution du programme provoquera l'affichage suivant:



- \* Le format 2 permet de définir la position de l'affichage comme suit:

1st line	0 1 2 3 . . . . . 2223
2nd line	24 25 26 . . . . . 4647
3rd line	48 49 50 . . . . . 6970
4th line	71 72 73 . . . . . 95

Comme indiqué, chaque position de l'écran correspond à un nombre compris entre 0 et 95, 0 représentant le point supérieur gauche et 95 le point inférieur droit de l'écran. La valeur de l'expression indiquée dans le format 2 indique le numéro de la position de début de l'affichage. Par conséquent, vérifiez que la valeur de l'expression est bien comprise entre 0 et 95. Sinon vous obtiendrez une erreur (ERROR 9).

## EXEMPLES

5 CLS

10 CURSOR 48: PRINT 123

20 CURSOR 26: PRINT "ABCDE"



- \* Le format 3 efface la position de début d'affichage.

Les caractères figurant avant et après ceux affichés par la commande PRINT ou par la commande PAUSE une fois la position de début d'affichage spécifiée par la commande CURSOR sont conservés. Cette possibilité peut être utilisée de nombreuses manières pour ne modifier qu'une partie de l'écran. Pour effacer l'écran, utiliser la commande CLS.

- \* La position de début d'affichage spécifiée dans la commande CURSOR peut également être utilisée par la commande INPUT. Cependant, une fois la commande INPUT exécutée, la position de début de l'affichage ne sera plus modifiée. Par conséquent, si une commande PRINT est exécutée ensuite, les caractères seront affichés à partir de cette position.

### EXEMPLES

```
10 CLS
20 CURSOR 0,2
→30 INPUT "DATA=": A
→40 PRINT A/2
```

La position spécifiée à la ligne 20 s'applique pour les lignes 30 et 40.

```
10 CLS
20 CURSOR 0,2
→30 INPUT "DATA=": A
40 PRINT A/2
```

La position spécifiée à la ligne 20 est effacée.

- \* Si les caractères à afficher ne tiennent pas sur l'écran, l'image écran est remontée vers le haut de façon que tous les caractères soient affichés même si la position de début de l'affichage a été spécifiée à l'aide de la commande CURSOR.

Lorsque la commande INPUT est exécutée après la commande CURSOR, procédez comme suit:

1. Insérez le type de commande GOTO suivant entre la commande CURSOR et la commande INPUT.

Exemple 1)

```
10 CURSOR 5,2
20 GOTO 30
```

Insérez une commande GOTO entre la commande CURSOR et la commande INPUT.

```
30 INPUT A
  :
```

Exemple 2)

```
10 CURSOR 5,2: GOTO 20
20 INPUT A
  :
```

2. Vérifiez que l'écran d'entrée/ou les caractères entrés/de la commande INPUT n'atteigne pas l'extrême droite de la quatrième ligne de l'écran. Le programme peut être détruit ou l'écran risque de changer sans nécessité si une commande GOTO n'a pas été insérée ou si l'écran d'entrée ou les caractères frappés atteint l'extrême droite de la quatrième ligne de l'écran. Dans ces cas, appuyez sur le bouton ALL RESET situé au dos de l'ordinateur afin d'effacer la mémoire et de réentrer le programme.

**1 DATA** liste expressions

Où: liste expression est: expression  
ou: expression, liste expressions

Abréviations: DA., DAT.

Voir également: READ, RESTORE

**OBJET**

Le verbe DATA permet de fournir des valeurs qui seront utilisées par le verbe READ.

**UTILISATION**

Lorsque l'on attribue des valeurs à un tableau pour la première fois, il est commode de lister ces valeurs dans une instruction DATA et d'utiliser ensuite une instruction READ dans une boucle FOR...NEXT pour charger ces valeurs dans le tableau. Lorsque la première instruction READ est exécutée, la première valeur de la première instruction DATA est prise. Les instructions READ suivantes utilisent les valeurs suivantes, dans l'ordre dans lequel elles figurent dans le programme, quelque soit le nombre de valeur listée dans chaque instruction DATA ou quelque soit le nombre d'instructions DATA utilisé.

Les instructions DATA n'ont aucun effet sur l'exécution du programme; par conséquent elles peuvent être insérées aux endroits qui semblent les plus appropriés. La plupart des programmeurs préfèrent les insérer immédiatement après l'instruction READ qui les utilise. Si besoin est, les valeurs d'une instruction DATA peuvent être lues une seconde fois au moyen de l'instruction RESTORE.

**EXEMPLES**

10 DIM B(10)	Définit un tableau
20 WAIT 128	
30 FOR I = 1 TO 10	
40 READ B(I)	Charge les valeurs DATA dans B( ) B(1) sera 10,
50 PRINT B(I)	B(2) sera 20, B(3) sera 30, etc.
60 NEXT I	
70 DATA 10, 20, 30, 40, 50, 60	
80 DATE 70, 80, 90, 100	
90 END	



### 1 DEGREE

Abréviations: DE., DEG., DEGR., DEGREE.

Voir également: GRAD et RADIAN

### OBJET

Le verbe DEGREE permet de convertir en degrés décimaux la valeur d'un angle.

### UTILISATION

Avec le PC-1350, la valeur des angles peut être représentée de trois manières différentes: degrés décimaux, radians et gradients. Ces formes sont utilisés pour spécifier les arguments des fonctions SIN, COS et TAN et renvoyer les résultats des fonctions ASN, ACS et ATN.

Le verbe DEGREE permet de convertir toutes les valeurs d'angles en degrés décimaux jusqu'à ce qu'un verbe GRAD ou RADIAN soit utilisé. Les fonction DMS et DEG peuvent être utilisées pour convertir des degrés décimaux en degré, minutes, secondes et vice-versa.

### EXEMPLES

10 DEGREE

20 X = ASN 1

30 PRINT X

X a maintenant la valeur 90, c'est-à-dire l'arc-sinus de 1.

1 DIM liste dim

où: <u>liste dim</u>	est: <u>spécif. dim</u>
et: <u>spécif. dim</u>	ou: <u>spécif; dim., liste dim</u>
et: <u>spécif. dim numérique</u>	est: <u>spécif. dim numérique</u>
et: <u>spécif. dim alphanumérique</u>	ou: <u>spécif. dim alphanumérique</u>
et: <u>nom numérique</u>	est: <u>nom numérique (taille)</u>
et: <u>nom alpha</u>	est: <u>nom alpha (dims)</u>
et: <u>dims</u>	ou: <u>nom alpha (dims) * longueur</u>
et: <u>taille</u>	est: <u>nom var num valide</u>
et: <u>longueur</u>	est: <u>nom var alpha valide</u>
	est: <u>taille</u>
	ou: <u>taille, taille</u>
	est: <u>nombre d'éléments</u>
	est: <u>longueur de chaque valeur alpha dans un tableau alpha</u>

Abréviations: D., DI.

**OBJET**

Le verbe DIM permet de réserver de la place pour les variables tableaux numériques et alphanumériques.

**UTILISATION**

Le verbe DIM doit être utilisé pour réserver l'emplacement des variables tableaux à l'exception des tableaux de la forme A( ), AS( ), deux caractères( ) et deux caractères \$( ).

Les tableaux ne peuvent avoir plus de deux dimensions et la taille de chacune ne peut être supérieure à 255. La place réservée en mémoire pour les tableaux correspond au nombre d'éléments spécifié dans l'instruction DIM plus un élément supplémentaire (élément 0). Par exemple, DIM B(3) réserve en mémoire B(0), B(1), B(2), et B(3). Dans les tableaux à deux dimensions, il y a une ligne et une colonne 0 en plus.

Dans les tableaux alphanumériques, il faut spécifier, en plus de nombre d'éléments, la taille de chaque élément. Par exemple, DIM BS(3) \* 12 réserve de la place pour 4 éléments alphanumériques ayant chacun un maximum de 12 caractères. Si la longueur n'est pas spécifiée, chaque élément peut comporter 16 caractères au maximum.

Lorsque l'instruction DIM est utilisée pour un tableau numérique, toutes les valeurs sont fixées au départ à 0; lorsqu'elle est utilisée pour un tableau alphanumérique toutes les valeurs sont fixées au départ à NUL.

Pour l'utilisation de DIM avec des tableaux A et AS, reportez-vous aux paragraphes concernant les variables.

### EXEMPLES

10 DIM B(10)

Réserve de la place pour un tableau numérique de 11 éléments.

20 DIM CS(4, 4) \* 10

Réserve de la place pour un tableau alphanumérique à deux dimensions de 5 lignes et de 5 colonnes; chaque élément aura au maximum 10 caractères.

1 END

Abréviations: E., EN.

## OBJET

Le verbe END permet de signaler la fin d'un programme.

## UTILISATION

Lorsque plusieurs programmes sont chargés en mémoire en même temps, il faut inclure une marque pour indiquer la fin de chaque programme de façon que l'exécution ne s'enchaîne pas d'un programme à l'autre. Ceci peut être obtenu au moyen du verbe END inséré comme dernière instruction d'un programme.

## EXEMPLES

10 PRINT "HELLO"

20 END

30 PRINT "GOODBYE"

40 END

Avec ces programmes en mémoire, "RUN 10" imprime "HELLO" mais non "GOODBYE". "RUN 30" imprime "GOODBYE".

- 1 **FOR** variable numérique **TO** expression 1 **TO** expression 2
- 2 **FOR** variable numérique = expression 1 **TO** expression 2  
**STEP** expression 3

Abréviations: F. et FO., STE.

Voir également: NEXT

## OBJET

Le verbe FOR est utilisé avec le verbe NEXT afin de répéter une série d'opérations un nombre de fois déterminé.

## UTILISATION

Les verbes FOR et NEXT sont utilisés ensemble pour encadrer un groupe d'instructions qui doivent être répétées. La première fois que ce groupe d'instructions est exécuté, la variable boucle (la variable appelée immédiatement après FOR) a la valeur d'expression 1.

Lorsque l'exécution arrive au verbe NEXT, la variable boucle est augmentée de la valeur de l'incrément (STEP) puis cette valeur est comparée à expression 2. Si la valeur de la variable boucle est inférieure ou égale à expression 2, le groupe d'instructions entre FOR et NEXT est à nouveau exécuté à partir de la 1ère instruction après FOR. Avec le premier format, la valeur de l'incrément est 1, avec la seconde forme cette valeur est donnée par expression 3. Si la valeur de la variable boucle est supérieure à expression 2, l'exécution se poursuit avec l'instruction suivant immédiatement NEXT. Etant donné que la comparaison se fait à la fin, les instructions comprises entre FOR et NEXT sont toujours exécutées au moins une fois.

Expression 1, expression 2 et expression 3 doivent être comprises entre -9.999999999E99 et 9.999999999E99. Lorsque expression 3 est  $\emptyset$  la boucle FOR/NEXT se répète indéfiniment.

La variable boucle peut être utilisée à l'intérieur d'un groupe d'instructions, par exemple comme index d'un tableau, mais il faut bien prévoir que la valeur de cette variable ne reste pas fixe.

Les programmes doivent être écrits de telle sorte que l'on ne passe pas d'une instruction à l'intérieur d'une boucle FOR/NEXT à une instruction à l'intérieur d'une autre boucle FOR/NEXT. De même, il faut toujours sortir d'une boucle FOR/NEXT en passant par l'instruction NEXT. Pour cela, il faut que la variable boucle ait une valeur supérieur à expression 2.

Le groupe d'instructions compris entre FOR/NEXT peut comporter une autre boucle FOR/NEXT utilisant une variable boucle différente sous réserve que cette boucle soit totalement imbriquée, c'est à dire que si l'instruction FOR se trouve à l'intérieur d'une boucle FOR/NEXT, l'instruction NEXT correspondante soit également à l'intérieur de cette boucle. Il peut y avoir ainsi 5 niveaux d'imbrication.

### EXEMPLES

```
10 FOR I = 1 TO 5
20 PRINT I
30 NEXT I
```

— Ce groupe d'instructions imprime les chiffres  
1, 2, 3, 4, 5.

```
40 FOR N = 10 TO 0 STEP -1
50 PRINT N
60 NEXT N
```

— Ce groupe d'instructions compte à rebour  
de 10 jusqu'à 0.

```
70 FOR N = 1 TO 10
80 X = 1
90 FOR F = 1 TO N
100 X = X * F
110 NEXT F
120 PRINT X
130 NEXT N
```

— Ce groupe d'instructions calcule et imprime  
la factorielle de N pour les nombres 1 à  
10.

**REMARQUE:** L'exécution de la boucle FOR-NEXT doit se terminer même s'il y a saut en dehors de la boucle. Par conséquent on peut obtenir une erreur d'imbrication (ERROR 5) suivant le programme (les programmes qui exécutent la commande FOR un certain nombre de fois).

### 1 GOSUB expression

Abréviations: GOS., GOSU.

Voir également: GOTO, ON...GOSUB, ON...GOTO, RETURN

## OBJET

Le verbe GOSUB permet d'exécuter une routine BASIC.

## UTILISATION

Lorsque vous désirez exécuter le même groupe d'instructions plusieurs fois au cours d'un même programme ou utiliser un jeu d'instructions écrit précédemment dans plusieurs programmes, il est commode d'utiliser les possibilités de BASIC concernant les routines et de spécifier les verbes GOSUB et RETURN.

Le groupe d'instructions est inséré dans le programme à un emplacement où il n'est pas atteint au cours de la séquence normale d'exécution. Il est fréquemment placé après l'instruction END qui marque la fin du programme principal. A l'endroit où, dans le corps du programme principal, la routine doit être exécutée, insérez une instruction GOSUB avec une expression indiquant le ler numéro de ligne de la routine. La dernière ligne de la routine doit être un verbe RETURN. Lorsque GOSUB est exécuté, le PC-1350 transfère le contrôle au numéro de ligne indiqué et traite les instructions jusqu'au verbe RETURN. Le contrôle est alors redonné à l'instruction suivant immédiatement GOSUB.

Une routine peut comporter le verbe GOSUB. Dix niveaux d'imbrications sont autorisés avec les routines.

L'expression d'une instruction GOSUB ne doit pas comporter de virgule, par exemple: 'A(1, 2)' ne peut être utilisé. Puisqu'il existe une structure ON...GOSUB pour choisir différentes routines à des emplacements donnés du programme, l'expression comporte en général uniquement le numéro de ligne voulu. Lorsqu'une expression numérique est utilisée, elle doit être traduite en un numéro de ligne valide (1 à 65279), sinon ERROR 4 s'affichera.

## EXEMPLES

```
10 GOSUB 100
20 END
100 PRINT "HELLO"
110 RETURN
```

Lorsque ce programme est exécuté, il imprime le 'HELLO' une seule fois.

**1 GOTO expression**

Abréviations: C., GO., GOT.

Voir également: ON...GOSUB, ON...GOTO

**OBJET**

Le verbe **GOTO** permet de sauter, au cours de l'exécution d'un programme, à un numéro de ligne spécifié.

**UTILISATION**

Le verbe **GOTO** permet de transférer le contrôle d'un endroit du programme BASIC en un autre endroit. Au contraire du verbe **GOSUB**, **GOTO** ne se "rappelle" pas l'emplacement à partir duquel le transfert a eu lieu.

Une instruction **GOTO** ne doit pas comporter de virgule, par exemple: 'A(1, 2)' ne peut être utilisé. Puisqu'il existe une structure **ON...GOTO** pour choisir différentes routines à des emplacements donnés du programme, l'expression comporte en général uniquement le numéro de ligne voulu. Lorsqu'une expression numérique est utilisée, elle doit être traduite en un numéro de ligne valide (1 à 65279), sinon **ERROR 4** s'affichera.

Les programmes bien conçus se déroulent généralement d'une seule traite de la première à la dernière ligne, à l'exception des routines exécutées au cours du programme. Par conséquent, le verbe **GOTO** est essentiellement utilisé dans l'instruction **IF...THEN**.

**EXEMPLES**

```
10 INPUT A$
20 IF A$ = "Y" THEN GOTO 50
30 PRINT "NON"
40 GOTO 60
50 PRINT "OUI"
60 END
```

Ce programme imprime 'OUI' si un 'Ø' est entré et "NON" si c'est autre chose qui est entré.



## 1 GRAD

Abréviations: GR., GRA.

Voir également: DEGREE et RADIAN

### OBJET

Le verbe GRAD permet de convertir la valeur d'un angle en gradient.

### UTILISATION

Avec le PC-1350, la valeur des angles peut être représentée de trois manières différentes: degrés décimaux, radians et gradients. Ces formes sont utilisées pour spécifier les arguments des fonctions SIN, COS et TAN et renvoyer les résultats des fonctions ASN, ACS et ATN.

Le verbe GRAD permet de convertir toutes les valeurs d'angle en gradients jusqu'à ce qu'un verbe DEGREE ou RADIAN soit utilisé. Les angles sont exprimés en "pourcent" de gradient, par exemple, un angle de  $45^\circ$  donne  $50^\circ$  gradient.

### EXEMPLES

10 GRAD

20 X = ASN 1

30 PRINT X

X a maintenant la valeur 100, c.a.d. un  $100^\circ$  gradient, l'arc-sinus de 1.

- ```
1 IF condition THEN instruction
2 IF condition instruction
```

Abréviations: aucune pour IF, T., TH., THE.

## OBJET

L'ensemble IF...THEN permet d'exécuter ou de ne pas exécuter une instruction suivant que certaines conditions sont ou non remplies au moment de l'exécution du programme.

## UTILISATION

Dans le déroulement normal des programmes BASIC, les instructions sont exécutées dans l'ordre dans lequel elles se présentent. L'ensemble IF...THEN permet de prendre des décisions au cours de l'exécution de façon qu'une instruction donnée ne soit exécutée que lorsqu'il le faut. Lorsque la condition indiquée dans l'instruction IF est vraie, l'instruction est exécutée. Lorsqu'elle est fautive, l'instruction est sautée.

La condition indiquée par IF peut être une expression de relation comme celles décrivent au Chapitre 4. Il est également possible d'utiliser une expression numérique comme condition, bien que le but de l'instruction soit moins clair. Toute expression qui donne 0 ou un nombre négatif est considérée comme fautive; toute expression qui donne un nombre positif est considérée comme vraie.

L'instruction qui suit le mot THEN peut être n'importe quelle instruction BASIC, y compris un autre IF...THEN. Si c'est une instruction LET, le verbe LET lui-même doit figurer. C'est l'instruction située après IF...THEN qui sera exécutée ensuite, que la condition soit vraie ou fautive, à moins que cette instruction ne soit l'instruction END, GOTO ou ON...GOTO.

Les deux formes de l'instruction IF aboutissent au même résultat, mais la première forme est plus claire.

## EXEMPLES

```
10 INPUT "SUITE?"; A$
20 IF A$ = "YES" THEN GOTO 10
30 IF A$ = "NO" THEN GOTO 60
40 PRINT "OUI OU NON"
50 GOTO 10
60 END
```

Ce programme continue à demander 'SUITE' tant que 'OUI' est entré; il s'arrête si 'NON' est entré et sinon pose la question 'OUI OU NON'.

### 1 INPUT liste entrées

|                                   |                                                      |
|-----------------------------------|------------------------------------------------------|
| Où: <u>liste entrées</u>          | est: <u>groupe entrées</u>                           |
|                                   | ou: <u>groupe entrées</u> , <u>liste entrée</u>      |
| Et: <u>groupe entrées</u>         | est: <u>liste var</u>                                |
|                                   | ou: <u>message entrée données</u> , <u>liste var</u> |
|                                   | ou: <u>message entrée données</u> , <u>liste var</u> |
| Et: <u>liste var</u>              | est: <u>variable</u>                                 |
|                                   | ou: <u>variable</u> , <u>liste var</u>               |
| Et: <u>message entrée données</u> | est: <u>une chaîne alpha</u>                         |

Abréviations: I., IN., INP., INPU.

Voir également: INPUT #, READ, CURSOR, PRINT

## OBJET

Le verbe INPUT permet d'entrer une ou plusieurs valeurs à partir du clavier.

## UTILISATION

Lorsque vous désirez entrer des valeurs différentes chaque fois qu'un programme est exécuté, utilisez le verbe INPUT pour entrer ces valeurs à partir du clavier.

Dans sa forme la plus simple, l'instruction INPUT ne comporte pas de paramètre, mais un point d'interrogation est affiché sur le côté gauche de l'écran. Il faut alors entrer une valeur et appuyer sur la touche **ENTER**. Cette valeur est attribuée à la première variable de la liste. Si d'autres variables sont incluses dans la même instruction INPUT, répétez ce processus jusqu'à ce qu'il n'y ait plus de variable dans la liste.

Si l'instruction INPUT est suivie d'un message pour l'entrée de données, le processus est exactement le même à l'exception près suivante: à la place du point d'interrogation, le message est affiché à la gauche de l'écran. Si le message est suivi d'un point virgule, le curseur est placé immédiatement après le message. Si le message est suivi d'une virgule, le message est affiché, puis lorsque vous appuyez sur une touche, l'écran s'efface et le premier caractère entré est affiché à gauche.

Lorsqu'un message pour l'entrée des données est spécifié et qu'il y a plusieurs variables dans la liste qui le suit, la seconde variable et les suivantes sont demandées au moyen du point d'interrogation. Si un second message est inclus dans la liste, il est affiché pour la variable qui le suit immédiatement.

Lorsque la position de début de l'affichage a été spécifiée au moyen de la commande CURSOR avant l'exécution de la commande INPUT, le message d'entrée de données ou '?' sera affiché à partir de cette position. D'autre part, lorsqu'une commande PRINT se terminant par un ";" est en court d'exécution, ces valeurs sont affichées, après quoi c'est le message d'entrée de données qui s'affiche.

Si la touche **ENTER** est enfoncée alors qu'aucune donnée n'a été entrée, la variable conserve la valeur qu'elle avait avant l'instruction INPUT.

## EXEMPLES

10 INPUT A

Efface l'écran et place un point d'interrogation à gauche de l'écran.

20 INPUT "A="; A

Affiche 'A=' et attend les données d'entrée.

30 INPUT "A=", A

Affiche 'A='.



Lorsque des données sont entrées 'A=' disparaît et les données sont affichées à partir du bord gauche de l'écran.

40 INPUT "X=?"; X, "Y=?"; Y

Affiche 'X=?' et attend les premières données d'entrées d'entrée.

L'écran s'efface une fois la touche **ENTER** enfoncée et 'Y=?' s'affiche sur le côté gauche de l'écran.

## REMARQUES:

Si une erreur est effacée à l'aide de la touche **CLS**, la position à laquelle seront affichées les données d'entrée suivantes risque d'être modifiée. Même dans ce cas, les données sont entrées correctement. La commande INPUT est exécutée lorsque vous appuyez sur la touche **ENTER**. Les données affichées juste avant que vous ayez appuyé sur la touche **ENTER** reste à l'écran. Par conséquent, le symbole du curseur  ou  peut rester sur l'écran affiché. En fait, il ne se voit qu'à l'écran et n'affecte pas les données d'entrée.

1 INPUT # liste var  
2 INPUT # "nom fichier"; list var  
Où: liste var est: variable  
ou: variable, list var  
Abréviations: I.#, IN.#, INP.#, INPU.#  
Voir également: INPUT, PRINT #, READ

## OBJET

Le verbe INPUT # permet d'entrer des valeurs à partir de la bande enregistrée.

## UTILISATION ET EXEMPLES

Voici les types de variable qui peuvent être spécifiés dans une instruction INPUT #:

- (1) variables fixes — A, B, C, A(7), D\*, A(20)\*, etc.
- (2) variables simples — AA, B3, CP\$, etc.
- (3) variables tableau — S(\*), HP(\*), K\$(\*), etc.

### 1) Transfert de données dans des variables fixes

Pour transférer des données de la bande dans des variables fixes, spécifiez les noms de variables dans l'instruction INPUT #.

```
INPUT # "DATA 1" ; A, B, X, Y
```

Cette instruction transfère les données du fichier sur bande appelé "DATA 1" dans les variables A, B, X, et Y dans l'ordre indiqué.

Pour transférer dans toutes les variables fixes disponibles et, le cas échéant, dans les variables du tableau étendu (A(27) et au delà) les données de la bande, spécifiez la première variable avec un astérisque (\*) comme indice.

```
INPUT # "D-2"; D*
```

Cette instruction transfère le contenu du fichier sur bande "D-2" dans les variables D à Z et A(27) et au-delà.

```
INPUT # A(10)* (sans déclaration DIM)
```

Cette instruction transfère les données du premier fichier trouvé après la mise en route de la bande dans les variables A(10) et au-delà (de J à Z et A(27) et au-delà).

**Remarque 1.** Si un tableau appelé A est déjà défini par une instruction DIM, il n'est pas possible de définir des variables fixes indice c ayant la forme A( ).

**Remarque 2.** Le transfert des données dans les variables fixes et les variables du tableau étendu (A(27) et au-delà) se poursuivra jusqu'à la fin du fichier de données source, mais si la mémoire de l'ordinateur est pleine avant, (ERROR 6) s'affichera.

## 2) Transfert de données dans des variables simples

Les données d'un fichier sur bande peuvent être transférées dans des variables simples; il suffit pour cela de spécifier dans l'instruction INPUT # les noms des variables voulues.

```
INPUT # "DM-1"; AB, Y1, XY$
```

Cette instruction transfère les données d'un fichier sur bande appelé "DM-1" dans les variables simples AB, Y1, et XY\$.

**Remarque 1.** Les données numériques doivent être transférées dans des variables numériques simples et les données alphanumériques dans des variables alphanumériques. Le contraire n'est pas possible.

**Remarque 2.** Les emplacements des variables simples doivent être réservés dans la zone des programmes et des données avant que l'instruction INPUT # soit exécutée sinon, il se produit une erreur. Utilisez les instructions d'affectations pour réserver la place des variables simples.

```
AA = 0 
```

```
B1$ = "A" 
```

```
INPUT # AA, B1$ 
```

Utilisez les valeurs numériques ou alphanumériques appropriées dans les instructions d'affectation pour réserver la place des variables.

## 3) Transfert de données dans des variables tableau

Pour transférer les données d'un fichier sur bande dans des variables tableau, spécifiez le nom du tableau dans l'instruction INPUT # sous la forme nom de tableau " \*".

```
50 DIM B(5)
```

```
60 INPUT # "DS-4"; B( *)
```

Cette instruction transfère les données du fichier sur bande appelé "DS-4" dans les variables (B(0) à B(5)) du tableau B.

**Remarque 1.** Les données numériques doivent être transférées dans des tableaux numériques de même longueur que les données, et les données alphanumériques doivent être transférées dans les variables tableau alphanumériques de même longueur que ces données. Si cette règle n'est pas observée, il se produit une erreur.

**Remarque 2.** L'emplacement des variables tableau doit être réservé dans la zone des programmes et données avant que l'instruction INPUT # soit exécutée. Sinon, il se produit une erreur. Utilisez l'instruction DIM pour définir le tableau à l'avance.

### — ATTENTION —

Si le nombre de variables spécifié dans l'instruction INPUT # ne correspond pas à la totalité des données enregistrées sur la bande, voici ce qui se produit:

- \* Si le nombre d'éléments de données enregistrées dans le fichier sur bande (à transférer) est supérieur au nombre de variables spécifiées, le transfert sera exécuté jusqu'à la dernière variable et le reste des données sera ignoré.
- \* Si le nombre d'éléments de données enregistrées dans le fichier sur bande (à transférer) est inférieur au nombre de variables spécifiées, toutes les données du fichier seront transférées dans les variables jusqu'à la fin du fichier et les variables restantes conserveront les données qu'elles avaient auparavant. Dans ce cas cependant, l'ordinateur continuera à attendre le transfert des données à partir de la bande. Pour supprimer cet état des choses, il faut appuyer sur la touche **ON** **BRK**.
- \* Si l'instruction INPUT # est exécutée sans qu'aucun nom de variable n'ait été spécifié, il se produit une erreur (ERROR 1).

1 **LET** variable = expression

2 variable = expression

Abréviations: LE.

## OBJET

Le verbe LET permet d'attribuer une valeur à une variable.

## UTILISATION

Le verbe LET attribue la valeur de l'expression à la variable indiquée. Le type de l'expression doit correspondre à celui de la variable; autrement dit les expressions numériques ne peuvent affectées qu'à des variables numériques et les expressions alphanumériques qu'à des variables alphanumériques. Pour passer d'un type à autre, il faut utiliser l'une des fonctions de conversion de type STR\$ ou VAL.

Le verbe LET peut être omis dans toutes les instructions LET à l'exception de celles qui figurent dans une clause THEN d'une instruction IF... THEN. Dans ce cas uniquement le verbe LET est obligatoire.

## EXEMPLES

10 I = 10

Attribue la valeur 10 à I

20 A = 5\*I

Attribue la valeur 50 à A

30 X\$ = STR\$(A)

Attribue la valeur '50' à X\$

40 IF I >= 10 THEN LET Y\$ = X\$ + ".00"

Attribue la valeur '50.00' à X\$.



- 1 LPRINT expr impr
- 2 LPRINT expr impr, expr impr, expr impr, ..., expr impr
- 3 LPRINT liste impr
- 4 LPRINT liste impr;
- 5 LPRINT

Où: print list est: expr impr  
ou: expr impr; liste impr

Et: print expr est: expression  
ou: USING clause; expression

La clause USING est décrite ultérieurement sous USING.

Abréviations: LP., LPR., LPRI., LPRIN.

Voir également: PAUSE, PRINT, USING, et WAIT

## OBJET

Le verbe LPRINT permet d'imprimer des informations sur l'imprimante en option CE-126P.

## UTILISATION

Lorsque l'interface d'entrée-sortie série est en fonction à la suite d'une commande OPEN, la commande LPRINT sort le programme sur le terminal de l'interface d'entrée-sortie série. (Voir page 195.) Pour que la commande de l'impression du programme revienne sur l'imprimante (CE-126P), exécutez la commande CLOSE.

Le verbe LPRINT permet d'imprimer les informations de demande d'entrée de données, les résultats de calcul etc. La première forme de cette instruction imprime une valeur unique. Si l'expression est numérique, la valeur sera imprimée sur le côté droit du papier. S'il s'agit d'une chaîne de caractères, l'impression se fera à partir du côté gauche.

Avec le format 2, les positions d'impression sont divisées en groupe de 12 colonnes. Les valeurs spécifiées sont imprimées à la suite. Autrement dit, la première valeur spécifiée est imprimée à gauche sur la première ligne, la seconde à droite sur la première ligne, la troisième à gauche sur la seconde ligne et la quatrième à droite sur la seconde ligne.

Les valeurs numériques imprimées dans ces colonnes sont imprimées à la droite de l'écran alors que les valeurs alphanumériques (chaîne de caractères) sont imprimées à partir de la gauche. Si la valeur à imprimer dépasse 12 colonnes, les chiffres les moins significatifs de la partie décimale du nombre sont tronqués pour obtenir 12 chiffres s'il s'agit d'une valeur numérique; s'il s'agit d'une valeur alphanumérique seuls les 12 premiers caractères (à partir de la gauche) sont imprimés.

**Remarque 1:** Le nombre de valeurs (éléments) spécifié en format 2 doit être compris entre 2 et 8.

**Remarque 2:** Même si la position de début de l'écran a été spécifiée dans le format 2 avec le format 4 ou avec la commande CURSOR, cette indication sera effacée et l'impression sera exécutée dans le format représenté ci-dessus. Les valeurs sont imprimées à partir du bord gauche du papier.

Avec le format 3, les valeurs sont imprimées à partir du bord gauche du papier. Si la valeur à imprimé est supérieure à 24 colonnes, un retour à la ligne est exécuté automatiquement. Il est possible d'imprimer jusqu'à 96 caractères. Il se produit une erreur lorsque ce 96ème caractère se trouve au milieu d'une valeur numérique.

Avec le format 4, la valeur spécifiée dans la commande LPRINT à exécuter ensuite sera imprimée à la suite de l'emplacement spécifié pour l'impression. Cependant, en raison de la structure de l'imprimante, l'impression se produit dans les cas suivants:

- (1) Lorsque la valeur à imprimer dépasse 24 colonnes.
- (2) Lorsqu'une commande LPRINT ne se terminant pas par un ";" a été exécutée.
- (3) Lorsque l'exécution du programme prend fin.

Avec le format 5, aucune impression n'a lieu mais le papier avance d'une ligne.

## EXEMPLES

```
10 A=10:B=20:X$="ABCDE":Y$="XYZ"
20 LPRINT A
30 LPRINT X$
40 LPRINT A,B,X$,Y$
50 LPRINT X$;A;B
60 LPRINT
70 LPRINT A*B;
80 LPRINT Y$
```

1 **NEXT** variable numérique

Abréviations: N., NE., NEX.

Voir également: FOR

## OBJET

Le verbe **NEXT** est utilisé pour marquer la fin d'un groupe d'instructions qui sont répétées dans une boucle **FOR/NEXT**.

## UTILISATION

L'emploi du verbe **NEXT** est décrit avec l'emploi du verbe **FOR**. La variable numérique d'une instruction **NEXT** doit correspondre à la variable numérique de l'instruction **FOR** correspondante.

## EXEMPLES

```
10 FOR I=1 TO 10  
20 PRINT I  
30 NEXT I
```

Imprime les chiffres de 1 à 10 chaque fois que vous appuyez sur **ENTER**.

1 ON expression GOSUB liste d'expressions

Où: liste d'expression est: expression  
ou: expression, liste d'expressions

Abréviations: O.; GOS., GOSU.

Voir également: GOSUB, GOTO, ON . . . GOTO

## OBJET

Le verbe ON . . . GOSUB permet d'exécuter une routine parmi plusieurs suivant la valeur d'une expression de contrôle.

## UTILISATION

Lorsque le verbe ON . . . GOSUB est exécuté, l'expression entre ON et GOSUB est calculée, puis réduite à un entier. Si la valeur de l'entier est 1, c'est la première routine de la liste qui est exécutée comme avec le verbe GOSUB normal. Si l'expression est 2, c'est la seconde routine de la liste qui est exécutée etc. Après le retour (RETURN) à partir d'une routine, l'exécution se poursuit par l'instruction suivant ON . . . GOSUB.

Si l'expression est nulle, négative ou supérieure au nombre de routines indiquées dans la liste, aucune routine n'est exécutée et l'ordinateur passe à la ligne de programme suivante.

**REMARQUE:** Aucune virgule ne doit être utilisée dans les expressions suivant GOSUB. Le PC-1350 ne peut faire la différence entre les virgules dans les expressions et les virgules entre des expressions.

## EXEMPLES

```
10 INPUT A
20 ON A GOSUB 100, 200, 300
30 END
100 PRINT "PREMIER"
110 RETURN
200 PRINT "SECOND"
210 RETURN
300 PRINT "TROISIEME"
310 RETURN
```

"PREMIER" s'imprime lorsque 1 est entré; "SECOND" lorsque 2 est entré; "TROISIEME" lorsque 3 est entré. Toute autre entrée ne provoque aucune impression.

1 ON expression GOTO liste d'expressions

Où liste d'expressions est: expression  
ou: expression, liste d'expressions

Abréviations: O., G., GO., GOT.

Voir également: GOSUB, GOTO, ON . . . GOSUB

## OBJET

Le verbe ON . . . GOTO permet de transférer le contrôle à différents emplacements suivant la valeur d'une expression de contrôle.

## UTILISATION

Lorsque le verbe ON . . . GOTO est exécuté, l'expression comprise entre ON et GOTO est calculée et réduite à un entier. Si la valeur de l'entier est 1, le contrôle est transféré dans la première position indiquée dans la liste. Si cette valeur est 2, le contrôle est transféré dans la seconde position dans la liste etc.

Si l'expression est nulle, négative ou supérieure au nombre de positions indiqué dans la liste, l'exécution se poursuit et l'ordinateur passe à la ligne suivante du programme.

**REMARQUE:** Il n'est pas possible d'utiliser de virgule dans les expressions GOTO. Le PC-1350 ne peut faire la différence entre les virgules dans les expressions et les virgules entre les expressions.

## EXEMPLES

```
10 INPUT A
20 ON A GOTO 100, 200, 300
30 GOTO 900
100 PRINT "PREMIER"
110 GOTO 900
200 PRINT "SECOND"
210 GOTO 900
300 PRINT "TROISIEME"
310 GOTO 900
900 END
```

"PREMIER" s'imprime lorsque 1 est entré; "SECOND" lorsque 2 est entré; "TROISIEME" lorsque 3 est entré. Toute autre entrée ne provoque aucune impression.

- 1 PAUSE expr impr
- 2 PAUSE expr impr, expr impr, . . . , expr impr
- 3 PAUSE liste impr
- 4 PAUSE liste impr;
- 5 PAUSE

Où: liste impr est: expr impr  
ou: expr impr; liste impr

Et: expr impr est: expression  
ou: USING clause; expression

La clause USING est décrite ultérieurement sous USING.

Abréviations: PAU., PAUS.

Voir également: LPRINT, PRINT, CURSOR, USING, et WAIT

## OBJET

Le verbe PAUSE permet d'imprimer des informations à l'écran pendant un court laps de temps.

## UTILISATION

Le verbe PAUSE permet d'afficher des demandes d'entrée de données, des résultats de calcul, etc. Le verbe PAUSE a les mêmes effets que PRINT mais, après PAUSE, le PC-1350 attend un laps de temps déterminé d'environ 0,85 secondes puis reprend l'exécution du programme sans que vous ayez à appuyer sur la touche **ENTER** ni à attendre le laps de temps spécifié par l'instruction WAIT.

La première forme de l'instruction PAUSE permet d'afficher une valeur unique. Si l'expression est numérique, la valeur est imprimée sur le côté droit de l'écran. S'il s'agit d'une chaîne de caractères, ceux-ci sont affichés à partir de la gauche de l'écran.

Cependant, lorsque la position de début d'affichage est spécifiée au moyen du format 4 ou de la commande CURSOR, l'affichage commence à cette position.

Avec le format 2, l'écran est divisé en groupes de 12 colonnes. Les valeurs sont affichées, en séquence, à partir de la première valeur spécifiée. Dans ce cas également, à l'intérieur des 12 colonnes, les expressions numériques sont affichées à partir de la droite et les expressions alphanumériques à partir de la gauche.

- Le nombre de valeurs (éléments) spécifié dans le format 2 doit être compris entre 2 et 8.

Verbes  
PAUSE

- Si la valeur indiquée dépasse 12 colonnes, voici ce qui se produit:
  - 1) Lorsqu'une valeur numérique comporte plus de 12 chiffres (lorsque la partie située après le E signifiant l'élévation à la puissance comporte 8 chiffres ou plus) le ou les chiffres les moins significatifs sont tronqués.
  - 2) Lorsque l'expression comporte plus de 12 caractères, seuls les 12 premiers caractères (à partir de la gauche) sont affichés.

Avec le format 3, la valeur spécifiée est affichée en continu à partir de la gauche de l'écran. Cependant, si la position de début d'affichage a été spécifiée avec le format 4 ou la commande CURSOR, l'affichage commence à cette position.

**Remarque:** Si la valeur à afficher en format 3 dépasse 96 colonnes, la partie en excès n'est pas affichée. Dans ce cas, l'ordinateur signale une error (ERROR 6) si cette 96ème colonne se trouve à l'intérieur d'une valeur numérique.

Avec le format 4, la valeur spécifiée est affichée sur le côté gauche de l'écran. La colonne suivant immédiatement cette valeur est définie comme position de début d'affichage pour les commandes d'affichage, par exemple pour la commande PRINT suivante.

Avec le format 5, la valeur affichée précédemment est affichée telle quelle.

## EXEMPLES

```
10 A=10:B=20:X$="ABCDEF":  
   Y$="XYZ"
```

|                         | Ecran                                               |
|-------------------------|-----------------------------------------------------|
| 20 PAUSE A              | 10.                                                 |
| 30 PAUSE X\$            | ABCDEF                                              |
| 40 PAUSE X\$, Y\$, A, B | ABCDEF            XYZ<br>10.                    20. |
| 50 PAUSE Y\$:X\$;       | XYZABCDEF                                           |
| 60 PAUSE A*B            | XYZABCDEF200.                                       |

- 1 **PRINT** expr impr
- 2 **PRINT** expr impr, expr impr, expr impr, expr impr
- 3 **PRINT** liste impr
- 4 **PRINT** liste impr;
- 5 **PRINT**
- 6 **PRINT = LPRINT**
- 7 **PRINT = PRINT**

Où: liste impr est: expr impr  
 ou: expr impr; liste impr

Et: expr impr est: expression  
 ou: USING clause; expression

La clause USING est décrite ultérieurement sous USING.

Abréviations: P., PR., PRI., PRIN

Voir également: LPRINT, PAUSE, CURSOR, USING, et WAIT

## OBJET

Le verbe **PRINT** permet d'imprimer des informations à l'écran ou sur l'imprimante de l'unité CE126P.

## UTILISATION

Le verbe **PRINT** permet d'afficher des informations de demande d'entrée de données, des résultats de calcul etc. La première forme de l'instruction **PRINT** permet d'afficher une valeur unique. Si l'expression est numérique, la valeur est imprimée à partir de la droite de l'écran. S'il s'agit d'une chaîne de caractères, elle est imprimée à partir de la gauche de l'écran.

Avec le format 2, l'écran est divisé en groupes de 12 colonnes. Les valeurs sont affichées, en séquence, à partir de la Première valeur indiquée. Dans ce cas également, à l'intérieur des 12 colonnes, les expressions numériques sont affichées à partir de la droite et les chaînes de caractères à partir de la gauche.

- Le nombre de valeurs (éléments) spécifié dans le format 2 doit être compris entre 2 et 8.
- Si la valeur indiquée dépasse 12 colonnes, voici ce qui se produit:
  - 1) Lorsqu'une valeur numérique comporte plus de 12 chiffres (lorsque la partie située après le E signifiant l'élévation à la puissance comporte 8 chiffres ou plus) le ou les chiffres les moins significatifs sont tronqués.
  - 2) Lorsque l'expression comporte plus de 12 caractères, seuls les 12 premiers caractères (à partir de la gauche) sont affichés.



## Verbes PRINT

Avec le format 3, la valeur spécifiée est affichée en continu à partir de la gauche de l'écran. Cependant, si la position de début de l'écran a été spécifiée avec le format 4 ou la commande CURSOR, l'affichage commence à cette position.

**Remarque:** Si la valeur à afficher en format 3 dépasse 96 colonnes, la partie en excès n'est pas affichée. Dans ce cas, l'ordinateur signale une error (ERROR 6) si cette 96ème colonne se trouve à l'intérieur d'une valeur numérique.

Avec le format 4, la valeur spécifiée est affichée sur la côté gauche de l'écran. La colonne suivant immédiatement cette valeur est définie comme position de début d'affichage pour les commandes d'affichage, par exemple pour la commande PRINT suivante.

**Remarque:** Ne combinez pas les commandes d'affichage (PRINT, etc.) avec des commandes d'entrée-sortie série (LPRINT, etc. pour l'interface entrée-sortie série). Vous risqueriez de supprimer la position de début de l'affichage spécifiée avec le format 4.

Avec le format 5, la valeur affichée précédemment est affichée telle quelle. Les formats 6 et 7 de l'instructions PRINT du programme qui suivent sont traitées comme des instructions LPRINT. Le format 7 restaure cette condition de façon que les instructions PRINT provoquent à nouveau l'affichage des données à l'écran.

### EXEMPLES

|                                             | <u>Ecran</u>                              |
|---------------------------------------------|-------------------------------------------|
| 10 A=123:B=5/9:X\$="ABCDEF":<br>Y\$="VWXYZ" | <b>ABCDEF 5.55555E-01</b>                 |
| 20 PRINT X\$,B                              |                                           |
| 30 PRINT A;B                                | <b>123.5.555555556E-01</b>                |
| 40 PRINT X\$;A;                             | <b>ABCDEF123.</b>                         |
| 50 PRINT Y\$;B                              | <b>ABCDEF123.VWXYZ5.555555<br/>56E-01</b> |

- 1 PRINT # "liste var"  
2 PRINT # "nom de fichier"; list var

Où: liste var est: variable  
ou: variable, liste var

Abréviations: P. #, PR. #, PRI. #, PRIN. #

Voir également: INPUT #, PRINT, READ

## OBJET

Le verbe PRINT # permet d'enregistrer des valeurs sur la cassette.

## UTILISATION ET EXEMPLES

Les types de variables suivants peuvent être utilisés pour les noms de variables:

- (1) Variables fixes — A, B, X, A(26), C\*, A(10)\*, etc.
- (2) Variables simples — AA, B2, XY\$, etc.
- (3) Variables tableaux — B(\*), CD(\*), N\$(\*), etc.

### 1) Sauvegarde du contenu des variables fixes sur bande

Pour sauvegarder le contenu des variables fixes sur bande, il faut spécifier les noms de variables voulues (séparées par des virgules) dans l'instruction PRINT #.

```
PRINT # "DATA 1"; A, B, X, Y
```

Cette instruction sauvegarde le contenu des variables A, B, X, et Y dans un fichier sur bande appelé "DATA 1".

Si vous désirez sauvegarder le contenu de la variable fixe indiquée et de toutes les variables fixes suivantes, placer un (\*) après le nom de la variable.

```
PRINT # "D-2"; D*
```

Cette instruction sauvegarde le contenu des variables fixes D à Z (et des variables du tableau étendu A(26) et au-delà, le cas échéant) dans le fichier sur bande appelé "D-2".

```
PRINT # E, X$, A(30)*
```

Cette instruction sauvegarde le contenu des variables fixes E et X\$ et des variables du tableau étendu A(30) et de toutes les autres variables sur bande sans le nom du fichier.

**Remarque:** Les noms des variables fixes indicées A(1) à A(26) peuvent être spécifiés dans l'instruction PRINT # de la même manière pratique-ment que les variables A à Z (ou A\$ à Z\$). Cependant, si le tableau A a déjà été défini par une instruction DIM, A( ) ne peut être utilisé pour définir des variables fixes avec indice.

### 2) Sauvegarde du contenu des variables simples (variables à 2 caractères)

Pour sauvegarder le contenu des variables simples, il faut spécifier les noms de variables voulues.

```
PRINT # "DM-1"; AB, Y1, XY$
```

Cette instruction sauvegarde le contenu des variables simples AB, Y1, et XY\$ dans le fichier sur bande appelé 'DM-1'.

### 3) Sauvegarde du contenu des variables tableaux

Pour sauvegarder sur bande le contenu de toutes les variables d'un tableau déterminé, il faut spécifier le nom du tableau avec un (\*) comme indice.

```
PRINT # "DS-2"; X(*), Y$(*)
```

Cette instruction sauvegarde le contenu de tous les éléments (X(0), X(1), ...) du tableau X et de tous les éléments (Y\$(0), Y\$(1), ...) du tableau Y\$, dans le fichier sur bande appelé "DS-2".

**Remarque:** Il n'est pas possible de sauvegarder le contenu d'un seul élément ou de plusieurs éléments déterminés d'un tableau. Si vous pouvez sauvegarder des parties déterminées des variables fixes ou des variables fixes indicées, vous ne pouvez sauvegarder une partie seulement d'un tableau (par exemple A) défini par une instruction DIM.

\* Si l'instruction PRINT # est exécutée sans qu'aucun nom de variable soit spécifié, il se produit une erreur (ERROR1).

### —ATTENTION—

L'emplacement des variables du tableau étendu tel que A(27) et au-delà, des variables simples et/ou des variables tableaux doit être réservé dans la zone des programmes et des données avant que l'instruction PRINT # soit exécutée. Sinon, l'exécution de cette instruction pour des variables non définies provoquera une erreur.

## 1 RADIAN

Abréviations: RAD., RADI., RADIA.

Voir également: DEGREE et GRAD

### OBJET

Le verbe RADIAN permet de convertir les valeurs des angles en radian.

### UTILISATION

Le **PC-1350** permet de représenter les angles de trois façons différentes — en degrés décimaux, en radians, et gradients. Ces formes permettent de spécifier les arguments des fonctions SIN, COS, et TAN et de renvoyer les résultats des fonctions ASN, ACS, et ATN.

La fonction RADIAN convertit les valeurs des angles en radian jusqu'à ce qu'un verbe DEGREE ou GRAD soit utilisé. RADIAN représente les angles en terme de longueur de l'arc par rapport au rayon, c'est-à-dire que  $360^\circ$  correspond à  $2\text{ PI}$  radians puisque la circonférence d'un cercle est  $2\text{ PI}$  fois le rayon.

### EXEMPLES

```
10 RADIAN
20 X=ASN 1
30 PRINT X
```

X a maintenant le valeur de 1.570796327 ou  $\text{PI}/2$ , l'arc-sinus de 1.

## 1 RANDOM

Abréviations: RA., RAN., RAND., RANDO.

### OBJET

Le verbe **RANDOM** permet de restaurer la valeur de départ pour l'obtention de nombres aléatoires.

### UTILISATION

Lorsque des nombres aléatoires sont obtenus à l'aide de la fonction **RND**, le **PC-1350** commence à partir d'une valeur de départ pré-déterminée. Le verbe **RANDOM** permet de restaurer cette valeur et de prendre une nouvelle valeur de départ au hasard.

La valeur de départ est la même chaque fois que le **PC-1350** est mis sous tension; par conséquent la suite de nombres aléatoires obtenue à l'aide de **RND** est à chaque fois identique, à moins que la valeur de départ ne soit modifiée. Ce verbe est très utile pour la mise au point d'un programme car autrement, le comportement du programme serait identique chaque fois qu'il serait exécuté même s'il comportait la fonction **RND**. Lorsque vous voulez obtenir des nombres vraiment aléatoires, utilisez l'instruction **RANDOM** pour que la valeur de départ soit elle-même un nombre au hasard.

### EXEMPLES

```
10 RANDOM  
20 X=RND 10
```

Lorsque le programme est exécuté à partir de la ligne 20, la valeur de X est basée sur la valeur de départ standard. Lorsqu'il est exécuté à partir de la ligne 10, c'est une nouvelle valeur de départ qui est utilisée.

**1 READ** liste var

Où: variable list est: variable  
ou: variable, liste var

Abréviations: REA.

Voir également: DATA, RESTORE

**OBJET**

Le verbe READ permet de lire les valeurs d'une instruction DATA et de les affecter à des variables.

**UTILISATION**

Lorsque l'on affecte pour la première fois des valeurs à un tableau, il est commode de lister ces valeurs dans une instruction DATA et d'utiliser une instruction READ dans une boucle FOR . . . NEXT pour charger ces valeurs dans le tableau. Lorsque l'instruction READ est exécutée pour la première fois, c'est la première valeur de la première instruction DATA qui est prise. Les instructions READ suivantes prennent les valeurs suivantes dans l'ordre dans lequel elles figurent dans le programme, quelque soit le nombre de valeurs listé dans chaque instruction DATA et quelque soit le nombre de ces instructions.

Si on le désire, on peut spécifier une instruction RESTORE pour que les valeurs d'une instruction DATA soient lues une seconde fois.

**EXEMPLES**

|                                |                                                |
|--------------------------------|------------------------------------------------|
| 10 DIM B (10)                  | Définit un tableau                             |
| 20 WAIT 32                     |                                                |
| 30 FOR I=1 TO 10               |                                                |
| 40 READ B(I)                   | Charge la valeur 10 de l'instruction DATA dans |
| 50 PRINT B(I)*2;               | B( )—B(1), la valeur 20 dans B(2), la valeur   |
| 60 NEXT I                      | 30 dans B(3), etc.                             |
| 70 DATA 10, 20, 30, 40, 50, 60 |                                                |
| 80 DATA 70, 80, 90, 100        |                                                |
| 90 END                         |                                                |

1 REM remarque

Abréviations: aucune

## OBJET

Le verbe REM permet d'insérer des commentaires dans un programme.

## UTILISATION

Il est souvent utile d'insérer des commentaires explicatifs dans un programme. Ces commentaires peuvent indiquer un titre, le nom des auteurs, la date des dernières modifications, des remarques sur l'utilisation, des rappels au sujet des algorithmes utilisés, etc. Pour insérer ces commentaires, il faut utiliser l'instruction REM.

L'instruction REM n'a aucun effet sur l'exécution du programme et peut être insérée en importe quel endroit de ce programme. Tout ce qui suit le verbe REM sur la même ligne que celui-ci est traité comme un commentaire.

## EXEMPLES

10 REM CETTE LIGNE EST UNE EXPLICATION

- 1 RESTORE  
2 RESTORE expression

Abréviations: RES., REST., RESTO., RESTOR.

Voir également: DATA, READ

## OBJET

Le verbe RESTORE permet de relire les valeurs d'une instruction DATA ou de modifier l'ordre dans lequel elles sont lues.

## UTILISATION

Lorsque le verbe READ est utilisé normalement, le PC-1350 commence par lire la première valeur de l'instruction DATA et poursuit en séquence jusqu'à la dernière valeur. La première forme de l'instruction RESTORE ramène le pointeur à la première valeur de la première instruction DATA de façon qu'elle puisse être relue. La seconde forme de l'instruction RESTORE ramène le pointeur à la première valeur de la première instruction DATA dont le numéro de ligne est supérieur à la valeur de l'expression.

## EXEMPLES

|                  |                                                     |
|------------------|-----------------------------------------------------|
| 10 DIM B(10)     | Définit un tableau                                  |
| 20 WAIT 32       |                                                     |
| 30 FOR I=1 TO 10 |                                                     |
| 40 RESTORE       |                                                     |
| 50 READ B(I)     | Affecte la valeur 20 à chacun des éléments de B( ). |
| 60 PRINT B(I)*I; |                                                     |
| 70 NEXT I        |                                                     |
| 80 DATA 20       |                                                     |
| 90 END           |                                                     |



## 1 RETURN

Abréviations: RE., RET., RETU., RETUR.

Voir également: GOSUB, ON . . . GOSUB

### OBJET

Le verbe RETURN est utilisé à la fin d'une routine pour redonner le contrôle à l'instruction suivant l'instruction GOSUB qui a provoqué le débranchement à cette routine.

### UTILISATION

Une routine peut comporter plusieurs instructions RETURN, mais la première rencontrée met fin à son exécution. L'instruction exécutée ensuite sera celle qui suit l'instruction GOSUB ou ON . . . GOSUB qui a appelé la routine. Si une instruction RETURN est exécutée sans qu'il y ait d'instruction GOSUB, il se produit une Erreur 5.

### EXEMPLES

```
10 GOSUB 100
20 END
100 PRINT "MERCI"
110 RETURN
```

Lorsque ce programme est exécuté, il imprime le mot "MERCI" une fois.

## 1 STOP

Abréviations: S., ST., STO.

Voir également: END; CONT commande

### OBJET

Le verbe **STOP** est utilisé pour arrêter temporairement l'exécution d'un programme afin de permettre les diagnostics.

### UTILISATION

Lorsque le verbe **STOP** est rencontré au cours de l'exécution d'un programme, le **PC-1350** s'arrête temporairement et un message s'affiche, par exemple 'BREAK IN 200', 200 étant le numéro de ligne comportant le verbe **STOP**. Cette instruction est utilisée au cours de la mise au point d'un programme pour vérifier le déroulement du programme et examiner l'état des variables. Pour relancer l'exécution du programme, utilisez la commande **CONT**.

### EXEMPLES

10 STOP

"BREAK IN 10" apparaîtra à l'écran.

## 1 TROFF

Abréviations: TROF.

Voir également: TRON

### OBJET


Le verbe TROFF permet d'annuler le mode traçage.

### UTILISATION

L'exécution du verbe TROFF permet de reprendre l'exécution normale du programme.

### EXEMPLES

```
1Ø TRON  
2Ø FOR I=1 TO 3  
3Ø NEXT I  
4Ø TROFF
```

numéros de ligne 1Ø, 2Ø, 3Ø, 3Ø, 3Ø et 4Ø appuyant sur la touche  .

## 1 TRON

Abréviations: TR., TRO.

Voir également: TROFF

### OBJET

Le verbe TRON permet de déclencher le mode traçage.

### UTILISATION


Le mode traçage facilite la mise au point des programmes. Lorsque l'ordinateur est dans ce mode, le numéro de ligne de chaque instruction est affiché une fois l'instruction exécutée. Le **PC-1350** s'arrête alors et attend que la touche flèche vers le bas soit enfoncée pour passer à l'instruction suivante. Il est possible d'appuyer sur la touche flèche vers le haut pour voir l'instruction qui vient d'être exécutée. Le mode traçage se poursuit jusqu'à ce que le verbe TROFF soit exécuté ou jusqu'à ce que les touches **SHIFT** et **CA**/**CLS** soient utilisées.

Lorsqu'un résultat est affiché à la position spécifiée par la commande CURSOR en mode traçage, le numéro de ligne suivant est affiché à la ligne suivante de l'écran. (Reportez-vous à la commande CURSOR page 125).

Lorsque, en mode traçage, la position de début d'affichage est spécifiée après l'exécution de la commande CURSOR, celui-ci est effacé si des variables sont appelées ou si des calculs sont effectués en mode manuel.

### EXEMPLES

```
10 TRON
20 FOR I=1 TO 3
30 NEXT I
40 TROFF
```

Lorsqu'il est exécuté, ce programme affiche les numéros de ligne 10, 20, 30, 30, 30, et 40 appuyant sur la touche  .

- 1 USING
- 2 USING "spécification d'édition"
- 3 USING variable alphanumérique

Abréviations: U., US., USI., USIN.

Voir également: LPRINT, PAUSE, PRINT

D'autres explications sur l'emploi de USING sont données à l'Annexe C

## OBJET

Le verbe USING permet de contrôler le format des données affichées ou imprimées en sortie.

## UTILISATION

Le verbe USING peut être utilisé seul ou comme une clause avec l'instruction LPRINT, PAUSE ou PRINT. Ce verbe définit un format de sortie qui est utilisé pour toutes les données en sortie qui le suivent jusqu'à ce qu'il soit modifié par un autre verbe USING.

Le spécification d'édition du verbe USING consiste en une chaîne de caractères entre guillemets pouvant comporter les caractères d'édition suivants:

- # Caractère de zone numérique cadré à droite
- Point décimal.
- ^ Indiqué que les nombres doivent être affichés en notation scientifique.
- & Zone alphanumérique cadrée à gauche

Par exemple, "#####" est une spécification d'édition pour une zone numérique cadrée à droite comportant 3 chiffres et le signe. Dans les zones numériques, il faut inclure une position pour le signe même s'il est toujours positif.

Les spécifications d'édition peuvent comporter plusieurs zones. Par exemple "####&&&&" peut être utilisé pour imprimer une zone numérique et une zone alphanumérique juxtaposées.

Si, comme dans le format 1, la spécification d'édition n'est pas indiquée, ce sont les règles d'affichage normales qui sont utilisées.

## EXEMPLES

Ecran

10 A=125: X\$="ABCDEF"

20 PRINT USING "##.##^";A

1.25E 02

30 PRINT USING "&amp;&amp;&amp;&amp;&amp;&amp;&amp;";X\$

ABCDEF

40 PRINT USING "####&amp;&amp;";A;X\$

125ABC

- 1 WAIT expression
- 2 WAIT

Abréviations: W., WA., WAI.

Voir également: PAUSE, PRINT

## OBJET

Le verbe WAIT permet de déterminer le laps de temps pendant lequel les informations affichées resteront à l'écran avant que l'exécution du programme se poursuive.

## UTILISATION

Au cours de l'exécution normale d'un programme, le **PC-1350** s'arrête après la commande PRINT, GPRINT, PSET, PRESET ou LINE et attend que vous appuyez sur la touche **ENTER**. La commande WAIT permet au **PC-1350** d'afficher les données pendant un laps de temps spécifié puis de poursuivre automatiquement (semblable au verbe PAUSE). L'expression qui suit le verbe WAIT définit le laps de temps voulu et doit être comprise entre 0 et 65535. Chaque incrément est d'environ 1/59<sup>ème</sup> de seconde. WAIT 0 est trop rapide pour permettre la lecture; WAIT 65535 correspond à une attente de 19 minutes. Lorsque WAIT n'est suivi d'aucune expression, le **PC-1350** revient à sa condition initiale et attend que vous appuyez sur la touche **ENTER** pour poursuivre.

## EXEMPLES

10 WAIT 59

PRINT restera affiché environ 1 seconde.

# FONCTIONS

## LES PSEUDOVARIABLES

Les pseudovariables constituent un groupe de fonctions qui ne comportent pas d'argument et sont utilisées comme des variables simples chaque fois que l'on en a besoin.

### 1 INKEY\$

INKEY\$ est une pseudovariable alphanumérique qui donne la valeur de la touche appuyée à la variable spécifiée. **ENTER**, **CLS**, **SHIFT**, **DEF**, **MODE**, **INS**, **DEL**, **SML**, **↑**, **↓**, **▶**, et **◀** ont tous une valeur NULLE. INKEY\$ permet de répondre à l'utilisation de touches individuelles sans attendre que la touche **ENTER** soit enfoncée pour mettre fin aux données d'entrée.

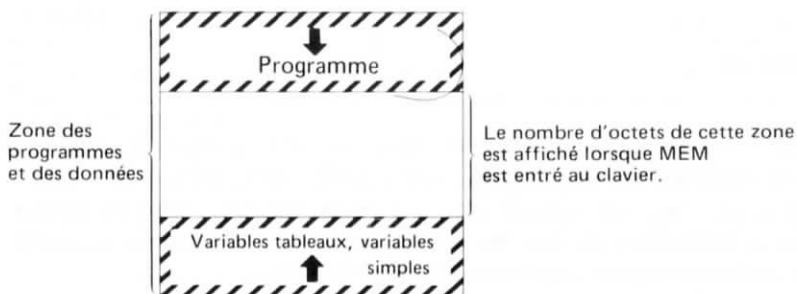
```
10 A$=INKEY$
20 B=ASC A$
30 IF B=0 THEN GOTO 10
40 IF B...
```

Les lignes 40 et suivantes comportent des tests pour cette touche et indiquent les mesures à prendre (par exemple: 40 PRINT A\$).

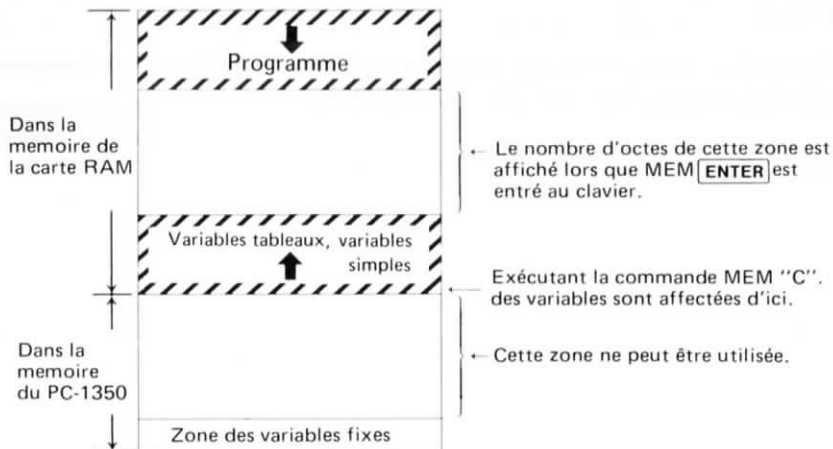


- |   |         |
|---|---------|
| 1 | MEM     |
| 2 | MEM "C" |
| 3 | MEM "B" |

Avec le format 1, le nombre d'octets libres (zone non utilisée par un programme, par des variables tableaux ou des variables simples) de la zone des programmes et des données est obtenu.

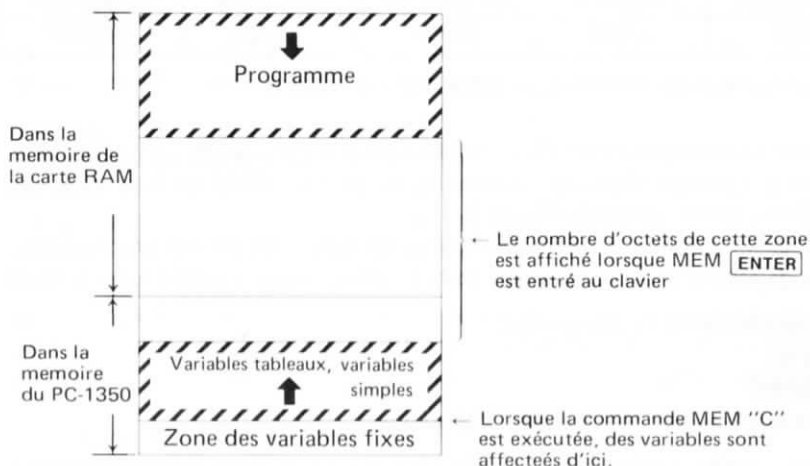


Lorsque la carte RAM en option est installée, il est possible de spécifier le format 2 pour que les variables soient affectées à la carte RAM.



Le format 3 permet de restaurer dans la mémoire du PC-1350 les variables qui avaient été envoyées au moyen du format 2 dans la mémoire de la carte RAM.

La modification de la position de début d'affectation des variables en format (2) ou en format (3) rend les autres variables précédemment affectées invalides.



- S'il n'y a pas de carte RAM installée, l'exécution de MEM "C" provoque une erreur (ERROR 1).
- Le mode défini pour la commande MEM "C" est enregistrée dans la carte RAM. Le retrait de cette carte du PC-1350 rend ce mode invalide. (L'ordinateur passe automatiquement à MEM "B".) Si la carte est à nouveau installée, l'ordinateur repasse en mode MEM "C".

## REFERENCE

1. La taille du programme (en octets) peut être obtenue de la manière suivante:

Exemple: PC-1350 uniquement

Mode RUN

CLEAR [ENTER] (efface les variables simples, les variables tableaux, etc.)

3070-MEM [ENTER] ← Affiche le nombre d'octets du programme

↑  
Affiche le nombre d'octets du programme  
Cette valeur varie suivant que la carte RAM est ou non utilisée.  
Utilisez les valeurs (nombre d'octets) à la page suivante.

Capacité de la zone des programmes et des données (en octet).

| PC-1350<br>uniquement | PC-1350 +<br>CE-201M | CE-201M<br>uniquement | PC-1350 +<br>CE-202M | CE-202M<br>uniquement |
|-----------------------|----------------------|-----------------------|----------------------|-----------------------|
| 3070                  | 11262                | 8142*                 | 19454                | 16334*                |

\* Capacité de la carte RAM lorsque MEM "C" est spécifié.

## 2. Contrôle de l'enregistrement du programme dans la carte RAM

Lorsqu'un programme dépassant la capacité de la carte RAM est écrit, celui-ci est détruit si la carte est retirée du PC-1350.

Par conséquent, utilisez la méthode suivante lorsque vous écrivez un programme long afin de vérifier que celui-ci est complètement enregistré dans la carte RAM.

(a) Lorsque MEM "B" a été spécifié.

Appuyez sur

MEM

et vérifiez que la valeur obtenue est 3120 ou plus.

**Remarque:** 3120 correspond au nombre d'octets de la zone des programmes et des données du PC-1350 (3070) + la zone de gestion du système (50)

(b) Lorsque MEM "C" a été spécifié.

Appuyez sur

MEM\$

et vérifiez que "C" est affiché.

\* Si un programme qui dépasse la capacité de la carte RAM est écrit lorsque MEM "C" a été spécifié, MEM "C" est effacé et MEM "B" est restauré.

Si c'est "B" qui s'affiche après MEM\$  , appuyez sur 3120-MEM  et abrégez le programme de la valeur (en octets) qui s'est affichée.

1 MEM\$

Cette commande permet de définir si la position actuelle de début de la zone affectée aux variables se trouve dans la mémoire de la carte RAM ou dans la mémoire du PC-1350.

(Exemple)

MEM\$

Si la position de début de la zone affectée aux variables se trouve dans le PC-1350 (MEM "B" spécifié), "B" s'affiche. Si elle se trouve dans la carte RAM (MEM "C" spécifié), "C" s'affiche.

- "B" s'affiche lorsque la carte RAM n'est pas installée.

## 1 PI

PI est une pseudovariable numérique qui a la valeur de PI. Elle a les mêmes effets que l'emploi du caractère spécial PI ( $\pi$ ) au clavier. Comme tous les autres nombres, la valeur de PI est indiquée avec une précision de 10 chiffres (3.141592654).

## FNCTIONS NUMERIQUES

On appelle fonctions numériques les opérations mathématiques qui prennent une valeur numérique unique et renvoient une valeur numérique. Ces fonctions comportent les fonctions trigonométriques, les fonctions logarithmiques, et les fonctions qui agissent sur l'entier ou la partie signe d'un nombre. Dans de nombreuses versions BASIC, l'argument d'une fonction doit être écrit entre parenthèses. Le PC-1350 n'a pas besoin de ces parenthèses sauf s'il est nécessaire d'indiquer quelle est la partie d'une expression plus complexe qui doit être incluse dans l'argument.

LOG 100 + 100 sera interprété comme:

(LOG 100) + 100 et non comme LOG (100+ 100).

1 ABS expression numérique

ABS est une fonction numérique qui renvoie la valeur absolue de l'argument numérique. La valeur absolue est la valeur d'un nombre sans tenir compte de son signe. ABS - 10 est 10.

1 ACS expression numérique

ACS est une fonction numérique qui renvoie l'arc-cosinus de l'argument numérique. L'arc cosinus est l'angle dont le cosinus est égal à l'expression. La valeur renvoyée dépend du mode du PC-1350 pour les angles (degrés, radians ou grades). ACS. 5 donne 60° décimaux.

1 ASN expression numérique

ASN est une fonction numérique qui renvoie l'arc-sinus de l'argument numérique. L'arc-sinus est l'angle dont le sinus est égal à l'expression. La valeur renvoyée dépend du mode du PC-1350 pour les angles (degrés, radians, ou grades). ASN. 5 donne 30° en mode degrés décimaux.

1 **ATN** expression numérique

ATN est une fonction numérique qui renvoie l'arc-tangente de l'argument numérique. L'arc-tangente est l'angle dont la tangente est égale à l'expression. La valeur renvoyée dépend du mode dans lequel se trouve le PC-1350 pour les angles (degrés, radians, ou grades). ATN 1. donne  $45^\circ$  en mode degrés décimaux.

1 **COS** expression numérique

COS est une fonction numérique qui renvoie le cosinus de l'argument angle. La valeur renvoyée dépend du mode dans lequel se trouve le PC-1350 pour les angles (degré, radian, ou gradient). COS 60 donne 0.5 en mode degrés décimaux.

1 **DEG** expression numérique

La fonction DEG convertit un argument angle exprimé DMS (Degré, Minute, Seconde) en DEG (degrés décimaux). En format DMS, la partie entière du nombre représente les degrés, les premiers et les seconds chiffres décimaux représentent les minutes et les troisièmes et quatrièmes les secondes, les chiffres suivants représentant les centièmes de secondes. Par exemple,  $55^\circ 10' 44.5''$  est représenté sous la forme 55.10445. En format DEG, la partie entière correspond aux degrés et la partie décimale aux degrés décimaux. DEG 55.10445 correspond à 55.17902778.

1 **DMS** expression numérique

DMS est une fonction numérique qui convertit un angle exprimé en format DEG en format DMS (voir DEG). DMS 55.17902778 devient 55.10445.

1 **EXP** expression numérique

EXP est une expression numérique qui renvoie la valeur de  $e$  (2.718281828—la base des logarithmes naturels) élevée à la valeur de l'argument numérique. EXP 1 est 2.718281828.

1 INT expression numérique

INT est une fonction numérique qui renvoie la partie entière de son argument numérique. INT PI est 3.

1 LN expression numérique

LN est une fonction numérique qui renvoie le logarithme à la base e (2.718281828) de son argument numérique. LN 100 est 4.6051170186.

1 LOG expression numérique

LOG est une fonction numérique qui renvoie le logarithme, en base 10 de son argument numérique. LOG 100 est 2.

1 RND expression numérique

RND est une fonction numérique qui génère des nombres aléatoires. Si la valeur de l'argument est inférieure à 1 mais supérieure ou égale à 0, le nombre aléatoire est inférieur à 1 et supérieur ou égal à 0. Si l'argument est un entier supérieur ou égal à 1, le résultat est un nombre aléatoire supérieur ou égal à 1 et inférieur ou égal à l'argument. Si l'argument est supérieur ou égal à 1 et n'est pas un entier, le résultat est un nombre aléatoire supérieur ou, égal à 1 et inférieur ou égal à l'entier le plus petit supérieur à l'argument. (Dans ce cas, la génération du nombre aléatoire change, suivant la valeur de la partie décimale de l'argument.):

| Argument | ----- Résultat ----- |                   |
|----------|----------------------|-------------------|
|          | Limite inférieure    | Limite supérieure |
| .5       | 0<                   | <1                |
| 2        | 1                    | 2                 |
| 2.5      | 1                    | 3                 |

Chaque fois que le PC-1350 est mis sous tension, c'est la même suite de nombres aléatoires qui est créée puisque c'est la même "valeur de départ" qui est utilisée à chaque fois. Pour rendre cette valeur de départ aléatoire, reportez-vous au verbe RANDOM.

1 **SGN** expression numérique

SGN est une fonction numérique qui renvoie une valeur en fonction du signe de l'argument. Si l'argument est positif, le résultat est 1; s'il est nul le résultat est 0 et s'il est négatif le résultat est -1.  $\text{SGN } -5$  est -1.

1 **SIN** expression numérique

SIN est une fonction numérique qui renvoie le sinus de l'argument angle. La valeur renvoyée dépend du mode dans lequel se trouve le PC-1350 pour les angles (degrés décimaux, radians ou grades).  $\text{SIN } 30^\circ$  est 0.5.

1 **SQR** expression numérique

SQR est une fonction numérique qui renvoie la racine carrée de son argument. Cette fonction a les mêmes effets que l'emploi du symbole spécial racine carrée ( $\sqrt{\quad}$ ) au clavier.  $\text{SQR } 4$  est 2.

1 **TAN** expression numérique

TAN est une fonction numérique qui renvoie la tangente de son argument angle. La valeur renvoyée dépend du mode dans lequel se trouve le PC-1350 pour les angles (degrés décimaux, radians ou grades).  $\text{TAN } 45^\circ$  est 1.



## FONCTIONS ALPHANUMÉRIQUES

On appelle fonctions alphanumériques les opérations utilisées pour manipuler les chaînes de caractères. Certaines prennent un argument alphanumérique et renvoient une valeur numérique. D'autres prennent un argument alphanumérique et renvoient des valeurs alphanumériques. Certaines prennent une valeur numérique et renvoient une valeur alphanumérique. Certaines encore prennent un argument alphanumérique et un ou deux arguments numérique et renvoient une valeur alphanumérique.

Dans de nombreuses versions BASIC, l'argument d'une fonction doit être écrit entre parenthèses. Le PC-1350 n'a pas besoin de ces parenthèses sauf s'il est nécessaire d'indiquer quelle est la partie d'une expression plus complexe qui doit être incluse dans l'argument. Les fonctions alphanumériques comportant deux ou trois arguments requièrent toutes des parenthèses.

### 1 ASC expression alphanumérique

ASC est une fonction alphanumérique qui renvoie la valeur numérique en code caractère du premier caractère de son argument. Le tableau des codes caractères et leur relation avec les caractères sont donnés à l'Annexe B. ASC "A" est 65. Le PC-1350 utilise les codes ASCII et leurs caractères.

### 1 CHR\$ expression numérique

CHR\$ est une fonction alphanumérique qui renvoie le caractère correspondant au code caractère numérique de son argument. Le tableau des codes caractères et leur relation avec les caractères est donné à l'Annexe B. CHR\$ 65 est "A".

**Remarque:** Si le code caractère 13 est spécifié lorsque vous exécutez la commande CHR\$ manuellement, le contenu spécifié ci-dessous ne sera pas affiché.

Exemple

CHR\$ 70 + CHR 71 + CHR\$ 13 + CHR\$ 75 + CHR\$ 76

**ENTER** → FG

Les caractères K et L pour les codes 75 et 76 ne sont pas affichés.

### 1 LEFT\$ (expression alphanumérique, expression numérique)



**Fonctions**  
**Fonctions Alphanumeriques**

LEFT\$ est une fonction alphanumérique qui renvoie la partie d'extrême gauche du premier argument alphanumérique. Le nombre de caractères renvoyés est déterminé par l'expression numérique. LEFT\$ ("ABCDEF", 2) est "AB".

1 **LEN** expression alphanumérique

LEN est une fonction alphanumérique qui renvoie la longueur de l'argument alphanumérique. LEN "ABCDEF" est 6.

1 **MIDS** (expression alphanumérique, exp. num. 1, exp. num. 2)

MIDS est une fonction alphanumérique qui renvoie la partie du milieu du premier argument alphanumérique. Le premier argument numérique indique la position du premier caractère à inclure dans le résultat. Le second argument numérique indique le nombre de caractères à inclure. MIDS ("ABCDEF", 2,3) est "BCD".

1 **RIGHT\$** (expression alphanumérique, expression numérique)

RIGHT\$ est une fonction alphanumérique qui renvoie la partie d'extrême droite du premier argument alphanumérique. Le nombre de caractères renvoyés dépend de l'argument numérique. RIGHT\$ ("ABCDEF", 3) est 'DEF'.

1 **STR\$** expression numérique

STR\$ est une fonction alphanumérique renvoyant une valeur alphanumérique qui est la représentation alphanumérique de son argument numérique. C'est l'inverse de VAL. STR\$ 1.59 est '1.59'.

1 **VAL** expression alphanumérique

VAL est une fonction alphanumérique qui renvoie la valeur numérique de son argument alphanumérique. C'est l'inverse de STR\$. VAL pour les caractères autres que les nombres est nul. VAL "1.59" est 1.59.

**Remarque:** Les valeurs alphanumériques que la fonction VAL peut convertir en valeur numérique sont les chiffres (0 à 9), les symboles (+ et -) et le symbole (E) indiquant une partie exponentielle. Rien d'autre ne peut être converti. Si une valeur alphanumérique comporte d'autres caractères et symboles, les valeurs alphanumériques à la droite de celle-ci seront ignorées. Si un espace est inclus dans une valeur alphanumérique, l'espace est en général ignoré.

1 **GCURSOR** (expression 1, expression 2)

Abréviation: GC., GCU., GCUR., GCURS., GCURSO.

Voir également: CURSOR, GPRINT

## OBJET

Indique la position de début de l'affichage en prenant le point comme unité.

## UTILISATION

La commande **GCURSOR** indique la position de début de l'affichage pour la commande **GPRINT**. (Amène le curseur graphique à la position indiquée.)

- \* L'écran se compose de 150 points horizontaux et 32 points verticaux. Chaque point est défini par un nombre compris entre 0 et 149 dans le sens horizontal (X) et entre 0 et 31 dans le sens vertical (Y). La position de début de l'affichage est définie en spécifiant le nombre correspondant à sa position horizontale dans l'expression 1 et le nombre correspondant à sa position verticale dans l'expression 2.

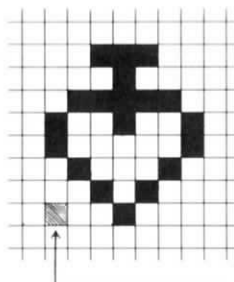


## EXEMPLES

10 **GCURSOR** (71, 20)

20 **GPRINT** "1824458F452418"

L'exécution de ce programme affiche le dessin suivant près du centre de l'écran. (La partie sombre n'est pas affichée.)



Dans le programme ci-dessus cette position en noir est la position de début du l'affichage (71, 20).

**Remarque:** La valeur des expressions 1 et 2 peut aller de  $-32768$  à  $+32767$ . Cependant si la valeur de ces expressions dépasse  $0$  à  $149$  et  $0$  à  $31$  respectivement, la position spécifiée se trouvera en dehors des limites de l'écran et une position verticale (position qui n'existe pas réellement) sera spécifiée comme position de début de l'affichage.

- \* La position de début de l'affichage reviendra à  $(0, 7)$  lorsque la commande RUN ou la commande CLS sera exécutée ou lorsque les touches **SHIFT** et **CA** seront enfoncées.

Lorsque le programme est déclenché à l'aide de la commande GOTO ou en utilisant la touche **DEF**, la valeur pour la direction Y est conservée. La valeur pour la direction X revient à  $0$ .

- 1 **GPRINT** chaîne de caractères
- 2 **GPRINT** expression; expression; expression; . . .
- 3 **GPRINT**

Abréviations: **GP.**, **GPR.**, **GPRI.**, **GPRIN**

Voir également: **GCURSOR**, **PRINT**

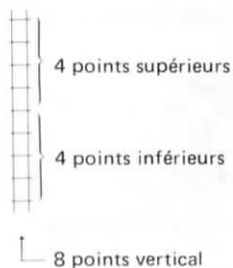
## OBJET

Permet d'afficher la configuration de points spécifiée.

## UTILISATION

La commande **GPRINT** permet d'afficher la configuration de points spécifiée. Une ligne verticale de 8 points est spécifiée comme une seule configuration.

- \* Avec le format 1, la configuration de 8 points, divisés en 4 points inférieurs et 4 points supérieurs, est spécifiée par une chaîne de caractères comprise entre guillemets, les configurations de 4 points étant représentées par des chiffres hexadécimaux.



| Nombre hexadécimal | Configuration | Nombre hexadécimal | Configuration | Nombre hexadécimal | Configuration | Nombre hexadécimal | Configuration |
|--------------------|---------------|--------------------|---------------|--------------------|---------------|--------------------|---------------|
| 0                  |               | 4                  |               | 8                  |               | C                  |               |
| 1                  |               | 5                  |               | 9                  |               | D                  |               |
| 2                  |               | 6                  |               | A                  |               | E                  |               |
| 3                  |               | 7                  |               | B                  |               | F                  |               |

## Commandes Graphiques

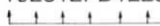
GPRINT "○○ ○○ ○○ ..."



Un seul groupe (pair) de 2 chiffres indique une configuration verticale (8 points). S'il reste un chiffre seul à la fin de la chaîne de caractères, celui-ci est ignoré.

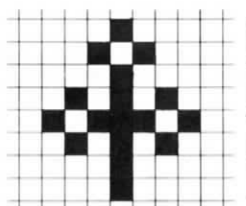
### EXEMPLES

GPRINT "102812FD122810"



Représente la configuration supérieure (4 points).

Représente la configuration inférieure (4 points).



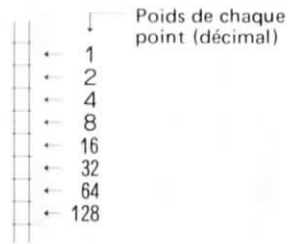
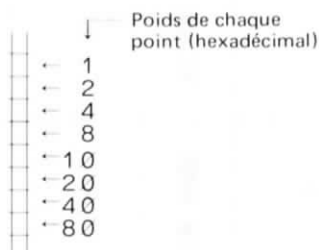
4 points supérieure

4 points inférieure

0 8 2 D 2 8 0 ← Configuration supérieure (4 points) représentée en hexadécimal.

1 2 1 F 1 2 1 ← Configuration inférieure (4 points) représentée en hexadécimal.

- \* Avec le format 2, une configuration de 8 points verticaux est spécifiée sous la forme d'une valeur hexadécimale ou la forme d'une valeur décimale. Un "poids" est attribué à chaque point de la configuration des 8 points verticaux comme indiqué ci-dessous.



Spécifiez la configuration de points à l'aide d'une valeur numérique qui correspond à la somme des poids des points à afficher sur l'écran.

### EXEMPLES

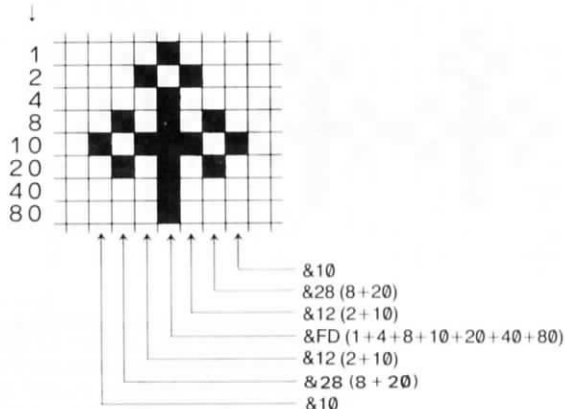
- \* Spécification de configurations de points en hexadécimal.

GPRINT &10, &28; &12; &FD; &12; &28; &10

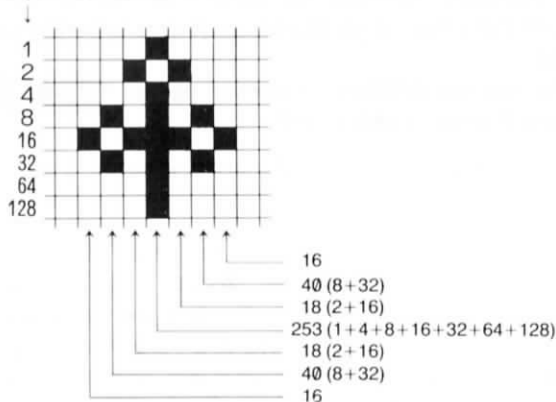
- \* Spécification de configurations de points en décimal.

GPRINT 16; 40; 18; 253; 18; 40; 16

## Poids en hexadécimal



## Poids en décimal



Avec le format 3, l'écran graphique précédent continue à être affiché.

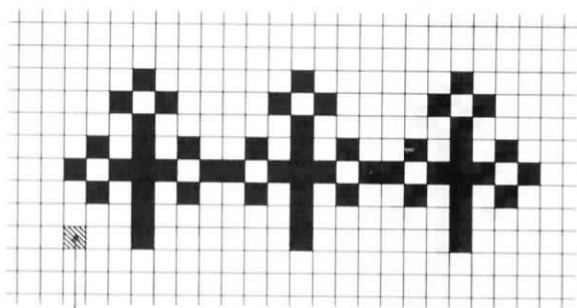
- \* Lorsque la position de début de l'affichage a été spécifiée au moyen de la commande GCURSOR, la première configuration de points de l'ensemble de la configuration affichée à l'aide de la commande GPRINT est affichée à partir de la position spécifiée pour le début de l'affichage et en utilisant les 8 points qui se trouvent au-dessus de celui-ci.

**EXEMPLES**

```

10 AAS= "102812FD12810"
20 GCURSOR (60, 20)
30 GPRINT AAS;AAS;AAS
    
```





Position spécifiée dans la commande GCURSOR (60, 20). L'affichage de la configuration commence à partir de cette position et utilise les 8 points supérieures partir de cette position.

**Remarque:** Lorsque l'instruction GPRINT se termine par un ";" , la colonne à la droite de l'affichage après exécution devient la position de début de l'affichage.  
Lorsque l'instruction GPRINT se termine par un "." ou  , la valeur pour l'axe des x redevient 0.

1 **LINE** (expression 1, expression 2) – (expression 3, expression 4),  $\left. \begin{matrix} S \\ R \\ X \end{matrix} \right\}$ ,  
 expression 5,  $\left. \begin{matrix} B \\ BF \end{matrix} \right\}$  (C)  
 (A) (B)  
 (D) (E)  
 Abréviation: LIN.  
 Voir également: GCURSOR, PSET

**OBJET**

Trace une ligne entre 2 points spécifiés.

**UTILISATION**

Une ligne est tracée entre les 2 points spécifiés par (expression 1, expression 2) et (expression 3, expression 4).

**EXEMPLES**

LINE (0, 0) – (149, 31)

Une ligne sera tracée à partir du coin supérieur gauche jusqu'au coin inférieur droit de l'écran.

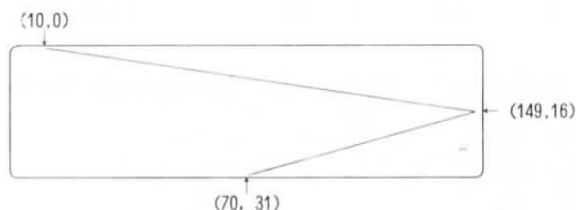
- \* Les valeurs de expression 1 à expression 4 dans les termes (A) et (B) peuvent aller de  $-32768$  à  $+32767$  mais pour que les points soient visibles à l'écran, expression 1 et expression 3 doivent être comprises entre 0 et 149, et expression 2 et expression 4 entre 0 et 31.

Lorsque les valeurs spécifiées pour un point se trouvent en dehors des limites de l'écran mais sont comprises entre  $-32768$  et  $+32767$ , il ne se produit aucune erreur. Seules les parties correspondantes de la zone située dans les limites de l'écran seront affichées. Il se produit une erreur (ERROR 3) lorsque les valeurs spécifiées pour un point se trouvent au-delà de ces limites.

- \* Le terme (A) (expression 1, expression 2) peut être omis. Dans ce cas, la ligne est tracée soit à partir de la position (0, 0) soit à partir de la position spécifiée par le terme (B) (expression 3, expression 4) de la commande LINE exécutée immédiatement avant.

**EXEMPLES**

```
5 CLS : WAIT 0
10 LINE (10, 0) – (149, 16)
20 WAIT : LINE – (70, 31)
```



**Remarque:** Etant donné que l'écran est constitué par des points les lignes diagonales peuvent ne pas sembler droites. D'autre part, les courbes peuvent ne pas être affichées avec exactitude.

\* S, R, ou X dans le terme (C) spécifient les points de la ligne tracée qui s'afficheront, s'effaceront ou apparaîtront en image inversée.

S: Trace la ligne en faisant apparaître les points (définit les points).

R: Trace la ligne en effaçant les points. Est utilisé pour tracer une ligne dans une zone où les points environnant sont affichés ou pour effacer une ligne existante. (Restaure les points.)

X: Trace la ligne et efface les points s'ils étaient déjà affichés ou les affiche s'ils n'y étaient pas. (Inverse les points.) Si ni S, ni R ni X n'est spécifié, S est pris par défaut.

\* La valeur de l'expression 5 dans le terme (D) spécifie le type de ligne.

Par exemple, lorsque la valeur de expression 5 est 26214 (&6666), voici le type de ligne qui est dessiné.



16 points

Une ligne est dessinée en répétant la configuration représentée à gauche.

Le nombre 26214 (&6666) peut être exprimé en binaire comme suit:

0110011001100110

Si les 16 points de la ligne représentés dans la figure ci-dessus et le nombre binaire sont comparés, on peut voir que les points correspondants aux 1 sont affichés et les points correspondants aux 0 effacés. Ainsi, le type de ligne est spécifié par les 0 et les 1 après que la valeur de expression 5 ait été convertie en un nombre binaire de 16 chiffres. Par conséquent, la ligne n'apparaît pas sur l'écran lorsque la valeur de expression 5 est 0 et une ligne en trait plein apparaît lorsque la valeur est 65535 (&FFFF). Une ligne en trait plein est également affichée si expression 5 est omise. Cependant, si R est spécifié dans le terme (C), l'opposé se produit et si X est spécifié, les points correspondants au chiffre 1 sont inversés.

- \* La valeur spécifiée pour expression 5 doit être comprise entre 0 et 63535 (&FFFF).
- \* Le terme (E) dessine un carré dont la diagonale est une ligne reliant les 2 points spécifiés par le terme (A) et (B).  
 B: Dessine un carré  
 BF: Dessine un carré rempli de lignes.

## EXEMPLES

```

10 CLS : WAIT 0
20 AA$= "102812FD122810"
30 GCURSOR (64, 20)
40 GPRINT AA$; AA$; AA$
50 LINE (24, 0) - (124, 31), &F18F, B
60 LINE (34, 3) - (114, 28), X, BF
70 GOTO 60
    
```

1 POINT (expression 1, expression 2)

Abréviation: POI., POIN.

Voir également: GCURSOR, PSET, PRESET

## OBJET

Lit l'état d'un point spécifié.

## UTILISATION

Si le point spécifié par (expression 1, expression 2) est affiché, un "1" est renvoyé, et s'il est effacé c'est un "0" qui est renvoyé.

Si le point indiqué se trouve au-delà des limites de l'écran, un "-1" est renvoyé.

\* Les valeurs spécifiées pour expression 1 et expression 2 peuvent être comprises entre -32768 et +32767. Cependant, les points affichés à l'écran sont compris entre 0 et 149 pour expression 1 et 0 et 31 pour expression 2.

## EXEMPLES

|                            |                                                                                       |
|----------------------------|---------------------------------------------------------------------------------------|
| 10 CLS : WAIT 0:A=75       |                                                                                       |
| 20 LINE (50, 0)-(50, 31)   | ← Trace 2 lignes verticales                                                           |
| 30 LINE (100, 0)-(100, 31) |                                                                                       |
| 40 PSET (A, 16)            | ← Affiche un point entre les 2 lignes                                                 |
| 50 B= POINT (A+1, 16)      | ← Effectue une vérification pour voir si le point suivant à droite est affiché.       |
| 60 IF B THEN 150           | ← Si oui, sauter à la ligne 150.                                                      |
| 70 PSET (A+1, 16)          | ← Si non, afficher ce point.                                                          |
| 80 PRESET (A, 16)          | ← Effacer ensuite le point affiché précédemment.                                      |
| 90 A=A+1                   | ← Déplacer un point vers la droite.                                                   |
| 100 GOTO 50                | ← Revenir à ligne 50.                                                                 |
| 150 B= POINT (A-1, 16)     | ← Vérification pour voir si le point à gauche du point affiché est également affiché. |
| 160 IF B THEN 50           | ← Si oui, aller à la ligne 50.                                                        |
| 170 PSET (A-1, 16)         | ← Si non, l'afficher.                                                                 |
| 180 PRESET (A, 16)         | ← Effacer ensuite le point affiché précédemment.                                      |
| 190 A=A-1                  | ← Se déplacer d'un point vers la gauche.                                              |
| 200 GOTO 150               | ← Revenir à la ligne 150.                                                             |

L'exécution de ce programme provoque le recul et l'avancée d'un point compris entre 2 lignes verticales dessinées sur l'écran.

1 **PRESET** (expression 1, expression 2)

Abréviation: PRE., PRES., PRESE.

Voir également: PSET, GCURSOR, POINT

## OBJET

Efface (restaure) le point indiqué de l'écran.

## UTILISATION

Efface le point défini par (expression 1, expression 2).

\* Les valeurs spécifiées pour expression 1 et expression 2 peuvent être comprises entre -32768 et +32767. Cependant, les points de l'écran sont compris entre 0 et 149 pour expression 1 et 0 et 31 pour expression 2.

## EXEMPLES

```
10 CLS : WAIT 0
20 LINE (20, 0)-(130, 31), BF
30 FOR X=-25 TO 25 STEP 0.5
40 Y=-1* SQR ABS (25*25-X*X)
50 PRESET (X+75, Y+31)
60 NEXT X
70 WAIT : GPRINT
```

L'exécution de ce programme trace un demi cercle à l'intérieur d'un carré rempli.

1 PSET (expression 1, expression 2)

2 PSET (expression 1, expression 2), X

Abréviation: PS., PSE.

Voir également: PRESET, GCURSOR, POINT

## OBJET

Affiche ou efface le point spécifié de l'écran.

## UTILISATION

- \* Le format (1) affiche le point spécifié par (expression 1, expression 2).
- \* Le format (2) efface le point spécifié par (expression 1, expression 2) s'il est affiché et l'affiche s'il est effacé.
- \* Les valeurs spécifiées pour expression 1 et expression 2 peuvent être comprises entre -32768 et +32767. Cependant les points de l'écran sont compris entre 0 et 149 pour expression 1 et 0 et 31 pour expression 2.

## EXEMPLES

```
10 CLS : WAIT 0 : DEGREE
```

```
20 FOR A=0 TO 600
```

```
30 B=-1* SIN A
```

```
40 Y= INT (B*16) +16
```

```
50 X= INT (A/4)
```

```
60 PSET (X, Y)
```

```
70 NEXT A
```

```
80 WAIT : GPRINT
```

L'exécution de ce programme trace une sinusoïde sur l'écran.

## 1 CLOSE

Abréviation: CLOS.

Voir également: OPEN

### OBJET

Ferme le circuit de l'interface d'entrée-sortie série.

### UTILISATION

Cette commande permet de fermer le circuit (en terme de logiciel) de l'interface d'entrée-sortie série qui avait été ouvert par la commande OPEN.

Par conséquent, une fois cette commande exécutée, toute sortie vers le terminal d'entrée-sortie série ou toute entrée à partir de ce terminal ne peut plus être exécutée.



1 **CONSOLE** expression

Abréviations: CPNS., CONSO., CONSOL.

Voir également: OPEN, LPRINT, LLIST

## OBJET

Définit le nombre de colonnes par ligne pour l'envoi des données.

## UTILISATION

Cette commande définit le nombre de colonnes par ligne pour l'envoi de données à partir de l'interface d'entrée-sortie série (terminal) lorsque la commande LPRINT ou LLIST a été utilisée.

Le PC-1350 envoie un code de fin (CR, LF, ou CR + LF) après l'envoi la ligne de données pré-définie.

- \* Il n'est possible de spécifier pour l'expression que des entiers compris entre 1 et 160. Si la valeur de l'expression est supérieure à 160, il y aura 160 colonnes par ligne. Si cette valeur est 0 ou négative, il se produira une erreur (ERROR 3).  
Si aucune expression n'est spécifiée, la commande est ignorée et le nombre de colonnes précédemment défini est conservé.
- \* Le nombre de colonnes passe à 40 lorsque les piles sont remplacées ou lorsque vous appuyez sur le bouton RESET.

1 **INPUT#1** variable, variable, variable...

Abréviations: I.#1, IN.#1, INP.#1, INPU.#1

Voir également: open, PRINT#1

## OBJET

Affecte les données entrées par l'intermédiaire de l'interface (terminal) d'entrée-sortie série aux variables spécifiées.

## UTILISATION

- \* Cette commande n'est valide que lorsque le circuit de l'interface d'entrée-sortie série est ouvert (à la suite d'une commande OPEN); sinon elle est ignorée.
- \* La commande INPUT#1 affecte les données (envoyées sous la forme décrite dans les paragraphes de la commande PRINT#1) aux variables spécifiées. Par conséquent, les variables sont spécifiées comme dans la commande PRINT#1

## EXEMPLES

INPUT#1A, AB, C\$, E(\*)

Les données entrées par l'intermédiaire de l'interface d'entrée-sortie sont attribuées aux variables A, AB, et C\$ et à la variable tableau E ( ).

- \* Vérifiez que le type des variables spécifié et le type des données d'entrée correspondent (c.a.d alphanumérique ou numérique). Avec les codes ASCII, si un caractère est attribué à une variable numérique, sa valeur devient 0. Si un chiffre est affecté à une variable alphanumérique, son contenu devient une chaîne de caractères. Par conséquent, si la type de variable spécifié et les données d'entrée ne correspondent pas, vous risquez d'obtenir des valeurs inattendues.

Même si des données telle que la fonction "SIN30" sont affectées à une variable numérique, ces données sont présumées être alphanumériques et par conséquent le contenu de la variable numérique sera 0.

Lorsque les données se présentent sous la forme "10 +40", les caractères (chiffres) après l'opérateur sont ignorés.

- \* Par conséquent dans ce cas les données seront "10".

**Remarque 1:** Si CR (code de contrôle: 0DH) ou NUL (00H) est inséré dans les données d'entrée, toutes les données suivant ces caractères risquent d'être ignorées.

**Remarque 2:** Les variables simples et les variables tableaux doivent être affectées dans la zone des programmes et des données avant l'exécution d'une commande INPUT#1, sinon il se produira une erreur.

- 1 **LLIST** ENTER
- 2 **LLIST** { expression  
"label" } ENTER
- 3 **LLIST** expression 1, expression 2 ENTER

Abréviations: LL., LLI., LLIS

Voir également: OPEN, CONSOLE

## OBJET

Envoie le contenu d'un programme à l'interface d'entrée-sortie série (terminal).

## UTILISATION

La commande **LLIST** peut être utilisée en opération manuelle en mode **PRO** ou **RUN**.

Lorsque le circuit de l'interface d'entrée-sortie série est ouvert à la suite d'une commande **OPEN**, le programme est envoyé en sortie sous forme de codes ASCII. Lorsque le circuit est fermé, le programme est imprimé sur l'imprimante. (Voir page 104.)

\* Avec le format (1), tous les programmes du PC-1350 sont envoyés en sortie.

### EXEMPLE

Lorsque le programme ci-dessous se trouve dans le PC-1350, le fait d'appuyer sur

**LLIST** **ENTER**

envoie le programme en sortie sous la forme représentée ci-dessous.

```
10: OPEN
100: REM **ABC-12**
65279: END
```

|       |   |   |   |   |   |   |       |       |       |
|-------|---|---|---|---|---|---|-------|-------|-------|
| Space | 1 | 0 | : | O | P | E | N     | Space | CR    |
| .1    | 0 | 0 | : | R | E | M | Space | Space | *     |
| *     | A | B | C | - | 1 | 2 | *     | *     | CR    |
| 6     | 5 | 2 | 7 | 9 | : | E | N     | D     | Space |
| CR    |   |   |   |   |   |   |       |       |       |

**Remarque:** CR est un code de fin. C'est soit LF soit CR + LF suivant la définition de la commande **OPEN**.

- \* Avec le format (2), la ligne indiquée par la valeur de l'expression ou la ligne avec le label spécifié est envoyée en sortie.
- \* Avec le format (3), le programme, à partir de la ligne indiquée par la valeur de expression 1 jusqu'à la ligne indiquée par expression 2 est envoyé en sortie. (les labels peuvent également être utilisés pour expression 1 et expression 2.) Expression 1 ou expression 2 peuvent être omises avec le format (3).
- Si expression 1 est omise, le programme est envoyé en sortie à partir de la première ligne jusqu'à la ligne indiquée par la valeur de expression 2.
- Si expression 2 est omise, le programme est envoyé en sortie à partir de la ligne indiquée par la valeur de expression 1 et jusqu'à la dernière ligne.
- \* S'il n'y a pas de ligne correspondant à la valeur d'expression 1 ou expression 2, la ligne avec le numéro suivant le plus élevé sera envoyé à l'interface. Il se produit une erreur (ERROR 1) si les lignes spécifiées dans expression 1 et expression 2 sont identiques.
- \* La commande LLIST est ignorée si un mot de passe a été défini.
- \* Si les programmes ont été fusionnés à l'aide de la commande MERGE, la commande LLIST n'est exécutée que pour le dernier programme fusionné. Pour lister les programmes enregistrés précédemment, il faut exécuter

LLIST "label",

- \* Le nombre de colonnes imprimé par ligne est défini par la commande CONSOLE. Si ce nombre est fixé à 23 ou moins, l'exécution de la commande LLIST provoque une erreur (ERROR 3).

**1 LOAD ENTER**

Abréviation: LOA.

Voir également: OPEN, CLOAD

**OBJET**

Charge les données en provenance de l'interface, d'entrée-sortie série (terminal) dans la zone des programmes et des données.

**UTILISATION**

La commande **LOAD** est valide lorsque le circuit de l'interface d'entrée-sortie série a été ouvert par une commande **OPEN**. Lorsque le circuit est fermé, elle est ignorée.

- \* Les données en provenance de l'interface d'entrée-sortie série sont lues jusqu'à ce que l'ordinateur rencontre un code de fin. Ces données sont considérées comme la première ligne de programme. Le PC-1350 convertit les données en une forme sous laquelle elles pourront être enregistrées comme programme puis les écrit dans la zone des programmes et des données. L'ordinateur lit à nouveau des données à partir de l'interface d'entrée-sortie série, les convertit de la même manière et les écrit dans la mémoire.  
Cette opération se poursuit jusqu'à ce que le code de fin de texte (voir la commande **OPEN**) soit lu.
- \* Le nombre maximal d'octets pouvant être lu en une seule fois est de 256. Par conséquent, si l'ordinateur lit plus de 256 octets avant de trouver le code de fin, il se produit une erreur.
- \* Les données lues sont converties puis écrites dans la zone des programmes et des données. Si une ligne, y compris le numéro de ligne, dépasse 80 octets, il se produit une erreur. Il se produit également une erreur si le début de la ligne n'est pas une valeur numérique (numéro de ligne).
- \* Au cours de l'exécution de la commande **LOAD**, les lignes ne sont pas réorganisées en fonction de leur numéro de ligne (par exemple placées par ordre croissant).

**Remarques:**

- L'exécution de la commande **LOAD** prend fin lorsque le code de fin de texte est lu (en provenance de l'interface).  
Même si l'unité émettrice a envoyé la totalité du programme, le PC-1350 ne met pas fin à l'exécution de la commande tant que le code de fin de texte n'a pas été lu. Dans ce cas, mettez fin à l'exécution de la manière suivante:

## Commandes Relatives Aux E/S Serie

- (1) Après l'envoi du programme, demandez également l'envoi du code de fin de texte.
  - (2) Ou appuyez sur la touche **BRK** pour mettre fin à l'exécution de la commande.
- Le contenu de la mémoire réservée ne peut être lu à partir de l'interface d'entrée-sortie série.

- 1 **LPRINT** { expression  
chaîne de caractère }
- 2 **LPRINT** { expression  
chaîne de caractère } , { expression  
chaîne de caractère. }
- ... , { expression  
chaîne de caractère }
- 3 **LPRINT** { expression  
chaîne de caractère } ; { expression  
chaîne de caractère }
- ... ; { expression  
chaîne de caractère }
- 4 **LPRINT** ... { expression  
chaîne de caractère } ;
- (Formater là où un “;” est ajouté à la fin de 1 et 3)
- 5 **LPRINT**

Abréviations: LP., LPR., LPRI., LPRIN.

Voir également: OPEN, CONSOLE, USING

## OBJET

Cette commande provoque l'envoi des informations spécifiées par l'intermédiaire de l'interface d'entrée-sortie série (terminal).

## UTILISATION

Lorsque le circuit de l'interface d'entrée-sortie série a été ouvert au moyen de la commande OPEN, les informations indiquées sont envoyées en code ASCII au terminal par l'intermédiaire de l'interface d'entrée-sortie série.

Lorsque le circuit est fermé, cette commande provoque l'impression des informations sur l'imprimante (Voir page 140.)

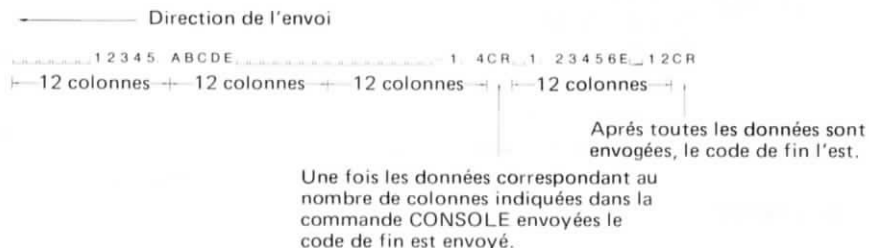
- \* Avec le format 1, la valeur de l'expression ou de la chaîne de caractères est envoyée à partir de son début.  
Si la valeur de l'expression est négative, un signe “-” est envoyé avant la valeur.  
Si elle est positive, c'est un espace qui est envoyé.
- \* Avec le format 2, des séries de 12 colonnes sont automatiquement définies.  
La valeur d'une expression ou d'une chaîne de caractères sera envoyée dans l'une de ces 12 colonnes (chiffre).



(Exemple)

```
10 OPEN "1200, N, 8, 1, A, C"
20 CONSOLE 36
30 LPRINT 12345, "ABCDE",
    -7/5 1. 23456789E12
```

L'exécution de ce programme envoie les informations sous la forme suivante.



Si la chaîne de caractères spécifiée dépasse 12 colonnes dans ce format, seuls les 12 premiers caractères sont envoyés. De même si la valeur de l'expression dépasse 12 chiffres (en notation exponentielle), l'expression sera envoyée après que les chiffres décimaux en trop auront été tronqués.

Si la valeur de l'expression est négative, un signe "-" est envoyé avant la valeur. Si elle est positive, c'est un espace qui est envoyé.

- \* Avec le format 3, les valeurs numériques ou alphanumériques spécifiées sont envoyées dans l'ordre indiqué. Avec ce format, aucun espace n'est envoyé avant les nombres positifs.

(Exemple)

```
⋮
50 LPRINT -123; "ABC"; 567. 89
```

← Direction de l'envoi  
-123. ABC567. 89CR

↑  
Code de fin (LF ou CR + LF est envoyé suivant la façon dont a été spécifiée la commande OPEN.)

- \* Avec le format 4, le code de fin indiquant la fin des données n'est pas envoyé. Mais une fois les données correspondant au nombre de colonnes spécifiées dans la commande CONSOLE envoyées, le code de fin de texte est envoyé.

- \* Avec le format 5, seul le code de fin de texte est envoyé.
- \* Lorsque le format a été spécifié dans la commande USING, les formats 1 à 4 envoient les données en fonction de ce qui a été spécifié.
- \* Lorsque PRINT=LPRINT est spécifié, les commandes PRINT ont les mêmes effets que les commandes LPRINT.  
PRINT=LPRINT n'est valide que si la commande est exécutée lorsque l'imprimante est connectée au terminal de l'imprimante ou lorsque le circuit de l'interface d'entrée-sortie serie a été ouvert par une commande OPEN.

**Remarque:** Pour envoyer des caractères ou des codes de contrôle qui ne peuvent être entrés directement au clavier, il faut les spécifier à l'aide de la commande CHR\$ comme indiquée ci-dessous.

(Exemple) Pour envoyer [ ]

① :

```
50 LPRINT CHR$&5B, CHR$&5D
```

② :

```
50 A$=CHR$&5B, B$=CHR$&5D
```

```
60 LPRINT A$, B$
```

NUL 900H) n'est valide qu'avec le format ① et sera ignoré avec le format ② .

1 OPEN "vitesse en baud, parité, longueur de mot, bit d'arrêt, type de code, code de fin, code de fin de texte"

2 OPEN

Abréviations: OP., OPE.

Voir également: CLOSE

## OBJET

Cette commande permet de transférer des données par l'intermédiaire de l'interface d'entrée-sortie. Elle définit également les conditions d'entrée-sortie.

## UTILISATION

Le format 1 permet de transférer des données par l'intermédiaire de l'interface d'entrée-sortie (terminal d'entrée-sortie série). Elle définit également les conditions du transfert des données avec le matériel connecté. Ces conditions sont spécifiées de la manière suivante:

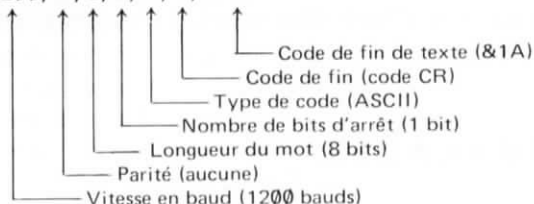
"vitesse en baud, parité, longueur de mot, bit d'arrêt, type de code, code de fin, code de fin de texte"

|                         |                                                                                                                                                                    |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Vitesse en baud:        | 300, 600, 1200<br>Indique la vitesse de modulation (vitesse de transfert).<br>Pour le PC-1350, on peut sélectionner 300, 600 ou 1200 bauds<br>(1 baud = 1 bit/sec) |
| Parité:                 | N, E, O<br>Indique le type de parité pour les caractères<br>N: Aucun bit de parité n'est transmis ni reçu.<br>E: parité paire.<br>O: Parité impaire.               |
| Longueur de mot:        | 7, 8<br>Indique le nombre de bits à transmettre ou à recevoir par caractère. On peut spécifier soit 7 soit 8 bits.                                                 |
| Nombre de bits d'arrêt: | 1, 2                                                                                                                                                               |
| Type de code:           | A<br>Seuls les codes ASCII peuvent être transmis ou reçus.<br>Par conséquent A doit toujours être spécifié.                                                        |
| Code de fin de ligne:   | C, F, L<br>Précise le type de code de fin à indiquer à la fin des données (délimitation) à la fin d'une ligne de programme, etc.                                   |

Code de fin de texte: C: Code CR (retour marge)  
 F: Code LF (ligne suivante)  
 L: Code CR + Code LF.  
 &00-&FF  
 Précise le code de fin de texte qui indiquera la fin du programme, etc.  
 (Peut être obligatoire lorsque les commandes SAVE ou LOAD sont utilisés.)

## EXEMPLES

OPEN "1200, N, 8, 1, A, C, & 1 A"



Les conditions ci-dessus sont celles en vigueur lorsque les piles ont été remplacées ou lorsque le bouton RESET a été enfoncé.

- \* Il n'est pas obligatoire de spécifier toutes ces indications après la commande OPEN. Dans ce cas, les conditions déjà spécifiées restent valides.

## EXEMPLES

OPEN ",, ,2"

Seul le nombre de bits d'arrêt est modifié.

- \* Avec le format 2, toutes les conditions définies précédemment sont conservées. Ce format permet le transfert des données par l'intermédiaire de l'interface d'entrée-sortie.
- \* L'exécution de la commande OPEN alors que le circuit de l'interface d'entrée-sortie est déjà ouvert et prêt pour le transfert des données (en raison d'une commande OPEN précédente) provoque une erreur (ERROR 8). Pour fermer le circuit, exécutez la commande CLOSE.  
 (Le circuit se ferme également lorsque la commande RUN est exécutée, lorsque le programme se termine ou lorsque l'interrupteur de mise sous tension est placé dans la position OFF (hors tension). Les conditions définies sont conservés même après que la commande CLOSE a été exécutée.)

## 1 OPEN\$

Abréviations: OP.\$, OPE.\$

Voir également: OPEN

### OBJET

Permet d'obtenir les conditions d'entrée-sortie définies à ce moment.

### UTILISATION

Les conditions d'entrée-sortie en vigueur sont obtenues sous forme d'une chaîne de caractères.

### EXEMPLES

OPEN\$ **ENTER** 1200, N, 8, 1, A, C, &1A

1 **PRINT#1** variable, variable, variable ...

Abréviations: P.#1, PR.#1, PRI.#1, PRIN.#1

Voir également: OPEN, INPUT#1

## OBJET

Cette commande permet l'envoi du contenu des variables spécifiées par l'intermédiaire de l'interface d'entrée-sortie série (terminal).

## UTILISATION

Cette commande n'est valide que si la circuit de l'interface d'entrée-sortie série a été ouvert (par une commande OPEN). Sinon elle est ignorée.

\* Les variables sont spécifiées de la manière suivante.

Variables fixes: Spécifiez chaque nom de variable.

[Exemple] A, B, C\$

Remarque: Les variables fixes ne peuvent être spécifiées sous la forme A\*.

Variables simples: Spécifiez chaque nom de variable.

[Exemple] AA, B1\$, C2

Variables tableaux: Spécifiez-les sous la forme: nom de tableau (\*).

[Exemple] B(\*), C\$(\*)

Lorsque les tableaux sont ainsi spécifiés, le contenu de tous les éléments est envoyé. (Les éléments d'un tableau ne peuvent être spécifiés individuellement.)

[Exemple] 50 PRINT #1A, AB, C\$, E(\*)

\* Lorsque les données sont envoyées, le code de fin est ajouté à la fin du contenu de chaque variable.

Pour les variables tableaux, le code de fin est ajouté à la fin du contenu de chaque élément.

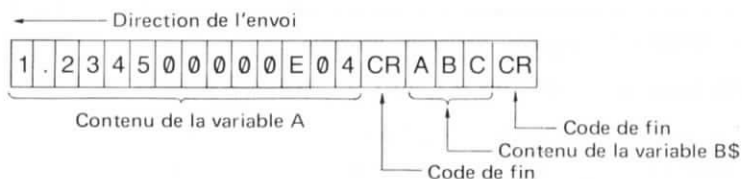
[Exemple] Lorsque:

A = 12345 et B\$ = "ABC"

l'exécution de l'instruction

PRINT#1A, B\$

Envoie A et B\$ sous la forme suivante. (sous réserve que le code de fin soit CR.)



\* Si la valeur de la variable numérique est négative, un signe "-" est envoyé devant la valeur.

- \* Les éléments d'un tableau sont envoyés dans l'ordre suivant:

Tableau à 1 dimension

[Exemple] Pour B (3)

B (0) → B(1) → B(2) → B(3)

Tableau à 2 dimensions

[Exemple] Pour C(2, 3)

C (0, 0) → C(0, 1) → C(0, 2) .....

**Remarques:**

- \* Les emplacements des variables du tableau étendu A(27) et au-delà, des variables simples et/ou des variables tableaux doivent être affectées dans la zone des programmes et des données avant que la commande PRINT #1 soit exécutée. Il se produit une erreur lorsqu'il y a tentative d'envoi du contenu d'une variable qui n'a pas été affectée.  
Il se produit également une erreur si le type de variable envoyé (numérique ou alphanumérique) ne correspond pas au type de variable à l'intérieur du PC-1350.
- \* Si des symboles tels que "π" et "√" se trouvent dans les données à envoyer, ils sont convertis en "PI" et "SQR" respectivement, puis envoyés.

## 1 SAVE

Abréviations: SA., SAV.

Voir également: OPEN, LLIST

### OBJET

Cette commande provoque l'envoi du programme qui se trouve dans la mémoire du PC-1350 à l'interface d'entrée-sortie série (terminal).

### UTILISATION

Lorsque le circuit de l'interface d'entrée-sortie série est ouvert à la suite d'une commande OPEN, le programme est envoyé en code ASCII.

Lorsque le circuit est fermé, cette commande est ignorée.

- \* Une fois la totalité du programme envoyée, le code de fin de texte est envoyé à son tour.
- \* Lorsque le programme est protégé par un mot de passe, la commande SAVE est ignorée.



## 1 BASIC

Abréviations: BA., BAS., BASI.

Voir également: TEXT

### OBJET

Efface le mode texte.

(Valide uniquement en opération manuelle dans le mode programme)

### UTILISATION

L'exécution de cette commande efface le mode texte et permet de revenir au mode BASIC.

Lorsque l'ordinateur revient en mode BASIC, le symbole d'entrée des données passe de "<" à ">".

- Le fait de passer du mode texte en mode BASIC transforme en général le texte qui se trouve dans la mémoire du PC-1350 en un programme (code interne).
- Cependant les abréviations telles que "P." et "I." ne sont pas converties dans les commandes correspondantes. (Dans ce cas, appelez le curseur sur la ligne et appuyez sur la touche **ENTER** pour les convertir en une commande.)  
En raison des caractéristiques de la fonction texte, les commandes et les formats qui n'existent dans le PC-1350 risquent de ne pouvoir être exécutés.
- Au cours de la conversion d'un programme, le signe "\*" est affiché à droite sur la quatrième ligne de l'écran.
- Si un mot de passe a été défini, l'exécution de la commande BASIC provoque une erreur (ERROR 1).

**1 TEXT**

Abréviations: TE., TEX.

Voir également: BASIC

**OBJET**

Place l'ordinateur en mode texte.

(Valide uniquement en opération manuelle dans le mode programme)

**UTILISATION**

La fonction texte permet d'entrer un programme écrit pour un ordinateur personnel de niveau plus élevé. Le programme entré sur le PC-1350 est envoyé à l'ordinateur personnel par l'intermédiaire de l'interface d'entrée-sortie série.

- L'exécution de la commande TEXT place l'ordinateur en mode texte. Dans ce mode, entrez un nombre correspondant au numéro de ligne, puis les informations correspondant aux commandes ou aux données du programme. Puis appuyez sur la touche **ENTER** pour écrire les entrées dans la zone des programmes et des données.  
Au contraire du mode BASIC, le contenu ainsi écrit n'est pas converti en commandes (codes internes). Le texte est enregistré tel quel (sous forme de caractères et/ou chiffres) en code ASCII. Le texte est classé dans l'ordre des numéros correspondant aux numéros de ligne et qui figurent au début de chaque ligne. (Fonction d'édition de numéro de ligne.)
- Le texte écrit en mode texte est enregistré en mémoire tel quel. Par conséquent les abréviations de commande BASIC (tel que I. pour INPUT) sont affichées et enregistrées en mémoire telles quelles.
- Si un programme est enregistré en code interne alors que l'ordinateur est en mode text, ce programme est converti en code ASCII.
- Au cours de la conversion d'un programme, le signe "\*" est affiché à droite sur la quatrième ligne de l'écran.
- En mode text, le symbole d'entrée des données est "<". (d'ordinaire ce symbole est ">".)
- En mode text, une ligne (y compris le numéro de ligne et **ENTER**) ne doit pas dépasser 80 caractères (80 octets). Si une ligne dépasse 80 caractères en raison de la conversion d'un programme, la partie en trop sera effacée.

Exemple:           80 octets (le fin est **ENTER**)

10:PRINT"ABC..... 1234567890"  
                                  converti en code ASCII.

10:PRINT "ABC..... 123456"  
                                  80 octets (le fin est **ENTER**)

Dans cette exemple, la commande PRINT occupe un octet en code interne mais elle en occupe 6 en code ASCII. C'est pourquoi, les derniers caractères sont supprimés (7890).

- Le nombre d'octets augmente lorsque l'on convertit un programme du code interne en code ASCII comme l'exemple le montre. S'il en résulte un dépassement de capacité de la zone des programmes, le programme converti jusqu'à ce point est reconverti en code interne et ERROR 6 s'affiche.
- Si un mot de passe a été défini, il se produit une erreur (ERROR 1) lorsque la commande TEXT est exécutée.

# CHAPITRE 10 DETECTION ET CORRECTION DES INCIDENTS

Ce chapitre vous donne quelques conseils sur ce que vous devez faire lorsque le PC-1350 ne répond pas comme vous l'attendiez. Ce chapitre est divisé en 2 parties: La première partie traite du fonctionnement général de l'ordinateur et la seconde de la programmation en BASIC. Pour chaque problème, une série de mesures vous est proposée. Essayez les une par une jusqu'à ce que vous ayez réglé votre problème.

## FONCTIONNEMENT DE L'ORDINATEUR

### Si:

Vous mettez l'ordinateur sous tension mais rien ne s'affiche à l'écran

Il y a quelque chose d'affiché à l'écran, mais rien ne se passe lorsque vous appuyez sur les touches

Vous avez entré une opération ou une réponse et il ne se passe rien

Vous êtes en train d'exécuter un programme BASIC; quelque chose s'affiche puis l'ordinateur s'arrête

Vous avez entré une opération et elle est affichée dans le format des instructions BASIC (deux points après le 1er nombre)

### Vous devriez:

1. Vérifiez que l'interrupteur de mise sous tension se trouve bien sur ON.
2. Appuyez sur la touche **ON BRK** pour voir si l'ordinateur n'a pas été mis automatique hors tension.
3. Remplacez les piles.
4. Réglez le contraste de l'écran.

1. Appuyez sur la touche **CLS** .
2. Appuyez sur la touche **CA** ( **SHIFT** **CLS** )
3. Mettez l'ordinateur hors tension puis à nouveau sous tension.
4. Maintenez une touche enfoncée et appuyez sur le bouton RESET.
5. Appuyez sur le bouton RESET sans appuyer sur aucune touche.

1. Appuyez sur **ENTER** .

1. Appuyez sur **ENTER** .

1. Passez du mode PROgramme en mode RUN.




L'ordinateur ne répond pas quelle que soit la touche sur laquelle vous appuyez

1. Appuyez simultanément sur une touche quelconque et sur le bouton RESET.
2. Si cette première tentative n'a rien donné, appuyez sur RESET sans appuyer sur aucune touche. Cette opération effacera la zone des programmes et des données ainsi que les positions de mémoire réservées.



### MISE AU POINT DES PROGRAMMES BASIC

Lorsque l'on entre un nouveau programme BASIC, en général il ne marche pas du premier coup. Même si vous frappez au clavier un programme que vous savez correct tel que ceux qui vous sont fournis dans cette brochure, il est courant que l'on fasse au moins une erreur de frappe. Si c'est un nouveau programme, quelque soit sa longueur, il comportera probablement également une erreur de logique. Voici quelques conseils pour vous aider à repérer et corriger vos erreurs.

Vous exécutez votre programme et obtenez un message d'erreur :

1. Revenez en mode PROgramme et utilisez les touches  et  pour revenir à la ligne erronée. Le curseur sera placé à l'endroit de la ligne où le PC-1350 n'a pas su que faire.
2. Si la ligne écrite vous semble correcte, le problème peut provenir des valeurs utilisées. Par exemple, CHR\$(A) provoquera une erreur si A a une valeur de 1 car CHR\$(1) n'est pas valide. Vérifiez les valeurs des variables soit en mode RUN soit en mode PRO en frappant le nom de la variable et en appuyant sur  .

Vous exécutez le programme et n'obtenez pas de message d'erreur mais il ne fait pas ce que vous voulez.

3. Vérifiez chaque ligne de programme à l'aide de LIST et les touches  et  pour vérifier s'il a été correctement entré. Le nombre d'erreurs que l'on peut corriger en relisant simplement un programme est surprenant.
4. Au fur et à mesure que vous lisez chaque ligne mettez-vous à la place de l'ordinateur. Prenez des exemples de valeurs et tentez d'effectuer les opérations de chaque ligne pour voir si vous obtenez le résultat escompté.
5. Rajoutez une ou plusieurs instructions PRINT dans votre programme afin d'afficher les valeurs clé et les emplacements clé. Utilisez ces instructions pour

isoler les parties de programme qui marchent correctement et repérez les erreurs. Cette méthode est également utile pour trouver quelles sont les parties d'un programme qui ont été exécutées. Vous pouvez également utiliser STOP pour arrêter temporairement l'exécution à des endroits particulièrement importants de façon à pouvoir examiner plusieurs variables.

- Utilisez TRON et TROFF soit comme commande soit directement à l'intérieur d'un programme pour suivre le déroulement du programme ligne par ligne. Arrêtez-vous pour examiner le contenu des variables importantes aux points cruciaux. Cette méthode de détection des incidents est très lente mais c'est parfois la seule.

Pour poursuivre le programme, appuyez sur la touche  une fois. La ligne à exécuté ensuite est affichée ainsi que le numéro de ligne. Là encore vous pouvez revoir la ligne en utilisant la touche . Vous pouvez également vérifier le contenu d'une variable en frappant son nom et en appuyant sur .

(lorsque A=4 est entré avant  )

| 4|

Il est obligatoire d'appuyer sur la touche  pour chacune des lignes à exécuter jusqu'à la fin du programme. Si vous ne désirez pas poursuivre l'exécution ligne par ligne, appuyez sur la touche  pour arrêter l'exécution du programme. Si vous changez à nouveau d'avis, utilisez la commande CONT pour reprendre les programmes.

Voici un exemple de correction d'un programme calculant l'hypothénuse:

| Entrée  | Ecran                           |
|---------|---------------------------------|
|         | >                               |
| T R O N | TRON_                           |
| ENTER   | >                               |
| R U N   | RUN_                            |
| ENTER   | ?                               |
| 3       | 3_                              |
| ENTER   | ?                               |
| 4       | 4_                              |
| ENTER   | 10:                             |
| ↑       | 10: INPUT A, B                  |
| ↓       | 20:                             |
| ↑       | 20: A = A * A : B = B * B       |
| A       | A_                              |
| ENTER   | 9.                              |
| B       | B_                              |
| ENTER   | 16.                             |
| ↓       | 30:                             |
| H       | H_                              |
| ENTER   | 5.                              |
| ↓       | HYPOTENUSE = 5.                 |
| ↑       | 40: PRINT "HYPOTENUSE = ";<br>H |
| ↓       | 40:                             |
| ↓       | >                               |

Même si vous faites très attention, vous allez créer en fin de compte un programme qui ne fait pas tout à fait ce que vous voulez. Pour repérer le problème, les concepteurs de SHARP ont prévu une méthode spéciale d'exécution des programmes appelée le mode "Traçage". En mode Traçage, le PC-1350 affiche le numéro de chaque ligne de programme et s'arrête après l'exécution de cette ligne. Cela vous permet de suivre la séquence dans laquelle les instructions sont véritablement exécutées. Lorsque le programme s'arrête après l'exécution d'une ligne, vous pouvez

examiner ou modifier les valeurs des variables.

L'instruction qui permet de passer en mode Traçage est tout simplement TRON. Cette instruction peut être émise comme une commande (en mode RUN) ou bien elle peut être insérée comme instruction dans un programme. En temps que TRON est utilisée comme commande elle informe le PC-1350 que le traçage est requis au cours de l'exécution de tous les programmes suivants. Les programmes à tracer sont alors lancés de manière normale au moyen d'une commande GOTO ou RUN.

Si TRON est utilisée comme instruction, l'ordinateur ne passera en mode Traçage que lorsque la ligne contenant TRON sera exécutée. Si pour une raison ou pour une autre, cette ligne n'est jamais atteinte, l'ordinateur ne passera pas en mode Traçage.

Une fois déclenché, le mode Traçage reste en vigueur jusqu'à ce qu'il soit annulé par une instruction TROFF. Cette instruction peut également être émise soit comme une commande soit comme une instruction. Le mode Traçage peut également être annulé par l'utilisation des touches suivantes:

**SHIFT** **CA**  
**CLS**

Pour vous montrer comment utiliser le mode Traçage, entrez le programme ci-dessous pour calculer la longueur de l'hypoténuse d'un triangle dont la longueur des côtés est donnée:

Liste du programme:

```
10 INPUT A, B
20 A = A * A : B = B * B
30 H =  $\sqrt{A + B}$ 
40 PRINT "HYPOTENUSE = ";H
```

En mode RUN, frappez la commande TRON suivie de la commande RUN. Remarquez que la commande INPUT fonctionne de la manière habituelle et que le point d'interrogation est affiché à chaque fois qu'une valeur d'entrée est requise. Dès que vous avez entré deux valeurs, le numéro de ligne de l'instruction INPUT apparaît:

| 10: |

Maintenez la touche **↑** enfoncée pour revoir toute la ligne:

| 10: INPUT A, B |

\* En mode Traçage, une fois que le résultat calculé affiche à l'emplacement spécifié par la commande CURSOR, le numéro de ligne suivant s'affiche à la ligne suivante. (Pour la description CURSOR, reportez-vous à la page 120.)



- \* En mode Traçage, si des variables sont appelées ou si un calcul est exécuté manuellement lorsque la position de début d'affichage a été spécifiée avec la commande CURSOR, cette position est annulée.

## CHAPITRE 11

# MAINTENANCE DU PC-1350

Voici quelques recommandations pour vous éviter d'avoir des problèmes avec votre PC-1350:

- \* Manipulez toujours votre ordinateur avec précaution car l'écran à cristaux liquides est fragile.
- \* Ne placez pas votre ordinateur dans des endroits soumis à des changements de température extrêmes; évitez également les endroits humides et poussiéreux. Lorsqu'il fait chaud, les véhicules stationnés au soleil sont soumis à des températures très élevées. Une station prolongée tant de telles conditions peut abîmer votre ordinateur.
- \* Utilisez uniquement des chiffons secs et doux pour nettoyer votre ordinateur. N'utilisez ni solvant, ni eau, ni chiffon humide.
- \* Pour éviter des fuites provenant des piles, retirez celles-ci lorsque l'ordinateur n'est pas utilisé pendant une période de temps prolongée.
- \* Pour toute réparation, renvoyez votre ordinateur à des centres agréés par SHARP.
- \* Si votre ordinateur est soumis à une forte électricité statique ou à des bruits externes importants, il peut "rester en suspend" (toutes les touches deviennent inopérantes). Dans ce cas, appuyez sur le bouton ALL RESET tout en maintenant une autre touche enfoncée. (Reportez-vous au chapitre Détection et correction des incidents).
- \* Gardez ce manual pour pouvoir vous y référer ultérieurement.

## ANNEXE A MESSAGES D'ERREUR

Le PC-1350 comporte 9 codes d'erreur différents. Le tableau suivant explique ces codes:

| Code d'erreur | Signification                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1             | <b>Erreur de syntaxe</b> <ul style="list-style-type: none"><li>Le PC-1350 ne peut pas comprendre ce que vous avez entré. Vérifiez différents éléments: les points virgules à la fin des instructions PRINT, les mots mal orthographiés, les emplois incorrects...<br/><code>3 */2</code></li></ul>                                                                                                                                                                                                                                                      |
| 2             | <b>Erreur de calcul</b> <p>Voici sans doute ce qui s'est passé:</p> <ol style="list-style-type: none"><li>Vous avez tenté d'utiliser un nombre trop grand.<br/>Les résultats des calculs sont supérieurs à 9.999999999E 99.</li><li>Vous avez tenté de diviser par 0<br/><code>5/0</code></li><li>Vous avez spécifié un calcul illogique<br/><code>LN -30</code> ou <code>ASN 1.5</code></li></ol>                                                                                                                                                      |
| 3             | <b>Fonction illégale (erreur dans l'instruction DIM ou erreur d'argument)</b> <ul style="list-style-type: none"><li>La variable tableau existe déjà.<br/>Le tableau a été spécifié sans que sa dimension ait été définie.<br/>L'indice du tableau dépasse la taille de ce dernier spécifiée dans une instruction DIM.<br/><code>DIM B(256)</code></li><li>Argument illégal. Cela veut dire que vous avez demandé à l'ordinateur de faire quelque chose qu'il ne peut pas faire.<br/>Intervalle supérieur à 65535.<br/><code>WAIT 66000</code></li></ul> |
| 4             | <b>Numéro de ligne trop élevé</b> <p>Là vous avez:</p> <ol style="list-style-type: none"><li>Soit tenté d'utiliser un numéro de ligne inexistant avec GOTO, GOSUB, RUN, LIST ou THEN etc.</li><li>Soit tenté d'utiliser un numéro de ligne trop élevé, le numéro de ligne le plus élevé étant 65279.</li></ol>                                                                                                                                                                                                                                          |

**5 NEXT sans FOR...**

Il y a plus de 10 routines imbriquées.

Il y a plus de 5 boucles FOR imbriquées.

Vous avez spécifié RETURN sans GOSUB, NEXT sans FOR ou READ alors qu'il n'y a pas d'instruction DATA.

Il n'y a plus de place dans la mémoire tampon.

**6 Dépassement de capacité de la mémoire.**

Généralement cette erreur survient lorsque vous avez tenté de donner à un tableau des dimensions trop importantes pour la mémoire. Ceci peut également survenir lorsqu'un programme devient trop long.

- Le contenu de la mémoire de réserve dépasse 144 octets.

**7 Erreur PRINT USING.**

Vous avez placé une indication de format illégal dans une instruction USING.

**8 Erreur d'unité d'entrée-sortie.**

Cette erreur peut se produire lorsque l'imprimante en option ou l'enregistreur de cassettes est connecté au PC-1350.

Cette erreur peut également se produire lorsque vous utilisez l'entrée-sortie série. Ce code signifie qu'il y a un problème de communication entre l'unité d'entrée-sortie et le PC-1350.

**9 Autres erreurs.**

Ce code d'affiche chaque fois que l'ordinateur a un problème qui ne correspond pas à l'un des codes précédents. Voici l'une des causes les plus courantes de cette erreur: vous tentez d'avoir accès aux données d'une variable d'une manière (par exemple A\$), alors que les données ont été enregistrées au départ dans la variable d'une autre manière (par exemple A).

**Les erreurs d'entrées**

Lors de l'exécution d'un programme, une erreur peut survenir en raison des erreurs commises lors de l'entrée du programme. Dans ce cas, notez les points suivants:

Exemple: Lorsque KPRINT est entré à la place de LPRINT

```
10: K PRINT AS
  ▶ 10 K PRINT AS
L ENTER 10: L PRINT AS
      ↑
      Espace
```

**ANNEXE A**  
**Messages d'Erreur**

Lorsque l'erreur est corrigée de cette manière, l'ordinateur ne reconnaît pas LPRINT comme une commande. Dans cet exemple, effacez KPRINT et reffapez LPRINT.

```
10: K PRINT AS
  ► 10 K PRINT AS
DEL DEL 10 AS
INS ... INS 10 ████████ AS$
LPRINT ENTER 10: LPRINT AS
```

↑  
Espace non repus lorsque LPRINT est reconnu  
comme une commande.

Vous pouvez ensuite vérifier si la commande a été entrée correctement à l'aide des touches de déplacement du curseur.

(Entrée correcte)

```
10: RADIAN
  ► 10 RADIAN
  ► 10 RADIAN_
```

(Entrée erronée)

```
10: RADAN
  ► 10 RADAN
  ► 10 RADAN
  ► 10 RADAN
```

## ANNEXE B TABLEAU DES CODES CARACTÈRES

Le tableau suivant montre les valeurs de conversion à utiliser avec CHR\$ et ASC. L'en-tête horizontal indique le premier caractère hexadécimal ou les 4 premiers bits binaires; l'en-tête vertical indique le second caractère hexadécimal ou les seconds bits binaires. Dans le coin supérieur gauche de chaque case se trouve indiqué le chiffre décimal correspondant au caractère. Le caractère lui-même se trouve dans le coin inférieur droit. Si aucun caractère n'est indiqué, ce caractère n'est pas valide sur le PC-1350.

Par exemple le caractère "A" correspond au nombre 65 ou en hexadécimal à 41 et en binaire à 01000001. Le caractère " $\sqrt{\quad}$ " correspond au nombre 252, à FC en hexadécimal et à 11111100 en binaire.

- Remarque:**
- Lorsque vous utilisez l'imprimante en option CE-126P, n'oubliez pas que les codes caractères 92(&5C), 249(&F9) et 250(&FA) pour le PC-1350 (caractères affichés) et pour l'imprimante (caractères imprimés) sont des caractères différents.
  - Lorsque vous utilisez l'imprimante en option CE-126P, n'utilisez pas le code 0 (&00).

ANNEXE B  
Tableau des Codes Caracteres

4 Premiers bits

4  
D  
e  
r  
n  
i  
e  
r  
s  
b  
i  
t  
s

| Hex<br>Binaire | 0    | 1    | 2      | 3    | 4    | 5    | 6    | 7    | 8    | E    | F    |
|----------------|------|------|--------|------|------|------|------|------|------|------|------|
|                | 0000 | 0001 | 0010   | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1110 | 1111 |
| 0              | 0    | 16   | 32     | 48   | 64   | 80   | 96   | 112  | 128  | 224  | 240  |
| 0000           | NUL  |      | ESPACE | 0    | @    | P    | €    | p    |      |      |      |
| 1              | 1    | 17   | 33     | 49   | 65   | 81   | 97   | 113  | 129  | 225  | 241  |
| 0001           |      |      | !      | 1    | A    | Q    | a    | q    |      |      |      |
| 2              | 2    | 18   | 34     | 50   | 66   | 82   | 98   | 114  | 130  | 226  | 242  |
| 0010           |      |      | "      | 2    | B    | R    | b    | r    |      |      |      |
| 3              | 3    | 19   | 35     | 51   | 67   | 83   | 99   | 115  | 131  | 227  | 243  |
| 0011           |      |      | #      | 3    | C    | S    | c    | s    |      |      |      |
| 4              | 4    | 20   | 36     | 52   | 68   | 84   | 100  | 116  | 132  | 228  | 244  |
| 0100           |      |      | \$     | 4    | D    | T    | d    | t    |      |      |      |
| 5              | 5    | 21   | 37     | 53   | 69   | 85   | 101  | 117  | 133  | 229  | 245  |
| 0101           |      |      | %      | 5    | E    | U    | e    | u    |      |      | ♣    |
| 6              | 6    | 22   | 38     | 54   | 70   | 86   | 102  | 118  | 134  | 230  | 246  |
| 0110           |      |      | &      | 6    | F    | V    | f    | v    |      |      | ♥    |
| 7              | 7    | 23   | 39     | 55   | 71   | 87   | 103  | 119  | 135  | 231  | 247  |
| 0111           |      |      | .      | 7    | G    | W    | g    | w    |      |      | ♦    |
| 8              | 8    | 24   | 40     | 56   | 72   | 88   | 104  | 120  | 136  | 232  | 248  |
| 1000           |      |      | (      | 8    | H    | X    | h    | x    |      |      | ♣    |
| 9              | 9    | 25   | 41     | 57   | 73   | 89   | 105  | 121  | 137  | 233  | 249  |
| 1001           |      |      | )      | 9    | I    | Y    | i    | y    |      |      | ■    |
| A              | 10   | 26   | 42     | 58   | 74   | 90   | 106  | 122  | 138  | 234  | 250  |
| 1010           |      |      | *      | :    | J    | Z    | j    | z    |      |      | □    |
| B              | 11   | 27   | 43     | 59   | 75   | 91   | 107  | 123  | 139  | 235  | 251  |
| 1011           |      |      | +      | ;    | K    | [    | k    | {    |      |      | π    |
| C              | 12   | 28   | 44     | 60   | 76   | 92   | 108  | 124  | 140  | 236  | 252  |
| 1100           |      |      | ,      | <    | L    | \    | l    |      |      |      | √    |
| D              | 13   | 29   | 45     | 61   | 77   | 93   | 109  | 125  | 141  | 237  | 253  |
| 1101           |      |      | -      | =    | M    | ]    | m    | }    |      |      |      |
| E              | 14   | 30   | 46     | 62   | 78   | 94   | 110  | 126  | 142  | 238  | 254  |
| 1110           |      |      | .      | >    | N    | ^    | n    | ~    |      |      |      |
| F              | 15   | 31   | 47     | 63   | 79   | 95   | 111  | 127  | 143  | 239  | 255  |
| 1111           |      |      | /      | ?    | O    | -    | o    |      |      |      |      |

Les codes caractères sont représentés comme suit:

Exemple

- Code pour "\*" 

|                |              |
|----------------|--------------|
| En hexadécimal | &2A          |
| En décimal     | 42 (32 + 10) |
- Code pour "√" 

|                |                |
|----------------|----------------|
| En hexadécimal | &FC            |
| En décimal     | 252 (240 + 12) |

Remarques:

- Affichage d'un caractère à l'aide de la commande CHR\$:  
Le caractère pour le code 0 (&00) dans la table est nul. Rien n'est affiché. Les carrés du tableau dans lesquels aucun caractère n'est indiqué correspondent à des espaces.
- Impression d'un caractère sur l'unité CE-126P à l'aide de la commande CHR\$:
  - N'utilisez pas le code 0 (&00).
  - Les carrés du tableau dans lesquels aucun caractère n'est indiqué correspondent à des espaces.
  - Les codes 249 (&F9) et 250 (&FA) sont des espaces.
  - Le caractère pour le code 92 (&5C) est "↵".



## ANNEXE C MISE EN FORME DES DONNEES EN SORTIE

Il est parfois important ou utile de contrôler le format ainsi que le contenu des données en sortie. Le PC-1350 contrôle les formats d'affichage au moyen du verbe USING. Ce verbe vous permet de spécifier:

- \* Le nombre de chiffres
- \* L'emplacement du point décimal
- \* Le format en notation scientifique
- \* Le nombre de caractères dans une chaîne

Ces différents formats sont spécifiés à l'aide d'un "masque de sortie". Ce masque peut être une constante ou une variable alphanumérique:

10: USING "####"

20: M\$="&&&&&&"

30: USING M\$

Lorsque le verbe USING est utilisé sans masque, il n'y a pas de mise en forme spéciale.

40: USING

Un verbe USING peut également être utilisé avec une instruction PRINT:

50: PRINT USING M\$; N

Lorsqu'un verbe USING est utilisé, il contrôle le format de tous les données de sortie jusqu'à ce qu'un autre verbe USING soit rencontré.

### LES MASQUES NUMERIQUES

Un masque USING numérique ne peut être utilisé que pour afficher des valeurs numériques, c'est-à-dire des constantes ou des variables numériques. Si une constante ou une variable alphanumérique est affichée alors que le masque USING en vigueur est numérique ce masque sera ignoré. La place prévue par le masque doit toujours être suffisante pour contenir la valeur à afficher. Le masque doit réserver la place du signe, même si le nombre est toujours positif. Ainsi un masque qui indique quatre positions ne peut être utilisé que pour afficher des nombres de trois chiffres.

## INDICATION DU NOMBRE DE CHIFFRES

Le nombre de chiffres voulus est spécifié à l'aide du caractère "#". Chaque "#" du masque réserve la place d'un chiffre. Les données affichées ou imprimées contiennent autant de caractères qu'il en a été spécifié dans le masque. Le chiffre figure à l'extrême droite de cette zone; les positions restant à gauche sont complétées par des espaces.

Par conséquent, les nombres positifs ont toujours au moins un espace à gauche. Etant donné que le PC-1350 garde dix chiffres significatifs au maximum, il ne faut pas utiliser plus de onze caractères "#" dans un masque numérique. Lorsque le nombre total de colonnes de la partie entière spécifiée dépasse onze, cette partie entière est considérée comme onze chiffres dans le PC-1350.

**Remarque:** Dans tous les exemples de cet annexe, le début et la fin de la zone affichée seront indiqués par un "I" pour montrer la dimension de la zone.

### Instruction

### Ecran

10: USING "####"

(Placez le PC-1350 en mode RUN, frappez RUN et appuyez sur **ENTER** .)

20: PRINT 25

| 25 |

30: PRINT -350

| -350 |

40: PRINT 1000

ERROR 7 IN 40

Remarquez que la dernière instruction a provoqué une erreur car cinq positions (4 chiffres et un espace pour le signe) était requis alors que le masque n'avait prévu que 4 positions.

## INDICATIONS DU POINT DECIMAL

Le caractère point décimal "." peut être inclus dans un masque numérique pour indiquer l'emplacement voulu du point décimal. Si le masque comporte plus de positions décimales qu'il n'en faut pour la valeur à afficher, les positions à la droite du dernier chiffre seront complétées par des 0. S'il y a plus de chiffres décimaux dans la valeur que dans le masque, les chiffres en plus seront tronqués (pas d'arrondi):

| <u>Instruction</u>  | <u>Ecran</u> |
|---------------------|--------------|
| 10: USING "####.##" |              |
| 20: PRINT 25        | 25.00        |
| 30: PRINT -350.5    | -350.50      |
| 40: PRINT 2.547     | 2.54         |

### INDICATION DE LA NOTATION SCIENTIFIQUE

Un caractère "^" peut être inséré dans le masque pour indiquer que le nombre est à afficher en notation scientifique. Les caractères "#" et "." seront utilisés dans le masque pour spécifier le format de la partie "caractéristique" du nombre, c'est-à-dire la partie affichée à la gauche de la lettre E. Deux caractères "#" doivent toujours être indiqués à la gauche du point décimal pour permettre d'indiquer le signe et un entier. Le point décimal doit être inclus mais il n'est pas obligatoire. Neuf caractères "#" au maximum peuvent figurer à la droite du point décimal. Après la partie caractéristique, le caractère d'élevation à la puissance, E, sera affiché suivi d'une position pour le signe et de 2 positions pour l'exposant. Ainsi, la zone la plus petite pour la notation scientifique correspondrait au masque "##^" et permettrait d'imprimer des nombres de format "2E 99". La zone la plus grande pour la notation scientifique serait "##.#####^" qui permettrait d'imprimer des nombres tels que "-1.234567890 E-12":

| <u>Instruction</u> | <u>Ecran</u> |
|--------------------|--------------|
| 10: USING "##.#^"  |              |
| 20: PRINT 2        | 2.00E 00     |
| 30: PRINT -365.278 | -3.65E 02    |

### INDICATION DES MASQUES ALPHANUMERIQUES

Les constantes et les variables alphanumériques sont affichées au moyen du caractère "&". Chaque "&" indique un caractère dans la zone à afficher. Les caractères seront affichés à partir de la gauche de cette zone. Si le nombre de caractères est inférieur à la longueur de la zone, les espaces en trop à droite seront complétés par des blancs. Si le nombre de caractères est supérieur à la longueur de la zone, ils seront tronqués de façon à correspondre à la longueur de la zone:

| <u>Instruction</u>    | <u>Ecran</u> |
|-----------------------|--------------|
| 10: USING "&&&&&"     |              |
| 20: PRINT "ABC"       | ABC          |
| 30: PRINT "ABCDEFGHI" | ABCDEF       |

## MASQUES MIXTES

Dans la plupart des applications, un masque USING comportera soit uniquement des caractères de mise en forme numérique soit uniquement des caractères de mise en forme alphanumérique. Cependant, dans certains cas, les 2 types de caractères peuvent être inclus dans un même masque USING. Dans ces cas, chaque passage des caractères de mise en forme numérique aux caractères de mise en forme alphanumérique ou vice-versa marque la limite pour une valeur différente. Ainsi, le masque "#####&&&" permet d'afficher deux valeurs distinctes – une valeur numérique pour laquelle sont réservées 5 positions et une valeur alphanumérique pour laquelle sont réservées 4 positions:

| <u>Instruction</u>                   | <u>Ecran</u> |
|--------------------------------------|--------------|
| 10: PRINT USING "###.##&&"; 25; "CR" | 25.00CR      |
| 20: PRINT -5.789; "DB"               | -5.78DB      |

**Nota:** Une fois spécifié, le format USING est utilisé pour toutes les données en sortie suivantes jusqu'à ce que ce format soit annulé ou modifié par un autre verbe USING.

## ANNEXE D CALCUL DES EXPRESSIONS ET PRIORITE DES OPERATEURS

Lorsqu'une expression complexe est envoyée à l'ordinateur, il calcule les différentes parties de l'expression dans un ordre qui dépend de la priorité de ces différentes parties. Si vous entrez l'expression:

$$100/5+45$$

soit comme un calcul soit comme partie d'un programme, le PC-1350 ne sait pas si vous voulez dire:

$$\frac{100}{5+45} = 2 \quad \text{ou} \quad \frac{100}{5} + 45 = 65$$

Le PC-1350 doit avoir le moyen de se décider entre ces deux possibilités; il utilise donc les règles de priorité des opérateurs. Etant donné que la division a une "priorité" supérieure à celle de l'addition (voir ci-dessous) il choisira d'effectuer la division en premier puis l'addition, c'est-à-dire qu'il choisira la seconde possibilité et renverra la valeur 65.

### PRIORITE DES OPERATEURS

Avec le BASIC utilisé sur le PC-1350, voici la priorité des opérateurs, en allant de la priorité la plus élevée à la priorité la plus faible:

Niveau

de priorité    Opérations

- 1    Parenthèses
- 2    Variables et pseudovariables
- 3    Fonctions
- 4    Élévation à la puissance ( ^ )
- 5    Moins unaire, signe négatif ( - )
- 6    Multiplication et division ( \* , / )
- 7    Addition et soustraction ( + , - )
- 8    Opérateurs de relation ( < , <= , = , <> , >= , > )
- 9    Opérateurs logiques ( AND , OR , NOT )

Lorsque, dans une expression, 2 opérateurs ou plus ont la même priorité, l'expression sera calculée de gauche à droite. (L'élévation à la puissance est calculée de droite à gauche). Notez que si  $A + B - C$  est spécifié par exemple, la réponse est la même que l'addition ou la soustraction soit exécutée en premier.

Lorsqu'une expression comporte plusieurs parenthèses imbriquées, les parenthèses les plus à l'intérieur sont calculées d'abord et le calcul se poursuit par les parenthèses suivantes la plus à l'intérieur et ainsi de suite.

Pour le niveau 3 et 4, la dernière opération entrée a la plus haute priorité.

Par exemple:  $-2^4 \rightarrow -(2^4)$

$$3^{-2} \rightarrow 3^{-2}$$

### EXEMPLE DE CALCULS D'EXPRESSIONN

Prenons l'expression:

$$((3+5-2) * 6+2)/10^{\wedge} \text{LOG } 100$$

Le PC-1350 calculerait d'abord les parenthèses les plus à l'intérieur. Etant donné que "+" et "-" sont du même niveau, il commencera par la gauche et effectuera l'addition en premier:

$$((8-2) * 6+2)/10^{\wedge} \text{LOG } 100$$

Puis il effectuera la soustraction:

$$((6) * 6+2)/10^{\wedge} \text{LOG } 100$$

Ou:

$$(6 * 6+2)/10^{\wedge} \text{LOG } 100$$

Dans le jeu suivant de parenthèses il effectuera la multiplication en premier:

$$(36+2)/10^{\wedge} \text{LOG } 100$$

Puis l'addition:

$$(38)/10^{\wedge} \text{LOG } 100$$

Ou:

$$38/10^{\wedge} \text{LOG } 100$$

Maintenant que les parenthèses ont été supprimées, la fonction LOG a la priorité la plus élevée; c'est donc elle qui sera calculée ensuite:

$$38/10^{\wedge} 2$$

L'élévation à la puissance sera calculée ensuite:

$$38/100$$

Et enfin la division sera exécutée:

$$0.38$$

Voici la valeur de l'expression

## ANNEXE E FONCTION DES TOUCHES EN BASIC

**ON**  
**BRK**

(ON)

Permet de remettre le PC-1350 sous tension après une mise hors tension automatique.

(BREAK)

- L'utilisation de cette touche au cours de l'exécution d'un programme a les mêmes effets que la touche BREAK et provoque l'interruption de l'exécution du programme.
- Lorsqu'elle est utilisée au cours de l'exécution en mode manuel, ou avec des commandes d'entrée-sortie telles que BEEP, CLOAD, etc., c'est l'exécution de la commande qui est interrompue.

**SHIFT**

- La touche jaune marquée "SHIFT" doit être utilisée pour appeler une seconde fonction (fonction figurant en brun au dessus de chaque touche).

Ex. **SHIFT** **?** → ? est entré.

**CLS**

- Permet d'effacer l'entrée affichée à l'écran.
- Utilisée pour supprimer une erreur.

**SHIFT** **CA**

- Efface non seulement le contenu de l'écran mais replace exactement l'ordinateur dans son état initial.

– Etat initial –

- Restaure l'horloge WAIT.
- Restaure le format d'affichage (format USING).
- Restaure le mode TRON (TROFF).
- Restaure PRINT=LPRINT.
- Restaure les erreurs.

**MODE**

- Permet de changer le mode de fonctionnement de l'ordinateur et de passer du mode RUN en mode PRO et vice-versa.

**SHIFT** **MODE**

- Permet de passer en mode réserve.

**0** à **9**

- Touches numériques.

**.**

- Point décimal.
- Permet d'entrer l'abréviation d'une commande, d'un verbe ou d'une fonction.
- Permet de désigner la partie décimale dans le format USING.

**E**


















- Permet d'indiquer un exposant en notation scientifique. (Cette touche est la lettre E: majuscule.)

- /
  - Touche de division.
  
- \*
  - Touche de multiplication.
  - Permet de désigner une variable tableau dans les instructions INPUT#, PRINT#, etc.
  
- +
  - Touche d'addition.
  
- - Touche de soustraction.
  
- SHIFT ^
  - Utilisée pour les instructions demandant l'élévation à la puissance.
  - Utilisé pour spécifier le système d'affichage exponentiel pour les données numériques dans les instructions USING.
  
- SHIFT < }  
SHIFT > }
  - Utilisée pour l'entrée d'opérateurs logiques dans les phrases IF.
  
- DEF
  - Lorsque l'une des 18 touches (A, S, D, F, G, H, J, K, L, =, Z, X, C, V, B, N, M, eSPaCe) est utilisée après que la touche **DEF** a été enfoncée, lance l'exécution du programme dont la ligne de programme comporte la même identification (label) que la touche utilisée.
  
- A à Z
  - Touches alphabétiques. Vous les connaissez certainement puisqu'elles sont identiques à celles d'un clavier de machine à écrire standard. Si une lettre de l'alphabet est utilisée telle quelle, c'est une lettre en majuscule qui est affiché. Si une lettre de l'alphabet est utilisée après la touche **SML** c'est une lettre en minuscule qui est affichée.
  
- SPC
  - Permet d'obtenir un espace lors de l'entrée des programmes ou des caractères.
  
- =
  - Dans les instructions d'affectation, permet d'affecter les chiffres ou caractères figurant à la droite de ce signe à la variable spécifiée à sa gauche.
  - Utilisée lors de l'entrée d'opérateurs logiques dans une phrase IF.
  
- SHIFT !
  - Utilisée pour désigner ces symboles.
- "
  - " : ● Permet de désigner et d'annuler des caractères.
- #
  - # : ● Spécifie l'identification d'un programme.
- \$
  - \$ : ● Dans l'instruction USING, définit le format d'affichage de données numériques.
- %
  - % : ● Utilisée lors de l'affectation de variables alphanumériques.
- &
  - & : ● Utilisée dans les instructions USING pour définir le format d'affichage des valeurs alphanumériques.
- @
  - @ : ● Utilisée pour désigner un nombre en hexadécimal.
- SHIFT @
  - @ : ● Utilisée pour le contenu de la mémoire réservée lorsque la touche de réserve est utilisée comme touche de programme.



Exemple: GOTO 100 @

!%: • Utilisée comme chaîne de caractères à l'intérieur de  
" "

-  • Permet d'entrer CLOAD?
-  • Permet de séparer 2 ou plusieurs instructions figurant sur une même ligne.
-  • Permet d'obtenir une pause entre 2 équations et entre des variables ou des commentaires.
-  • Permet d'afficher plusieurs éléments à la fois sur l'écran.  
• Permet d'obtenir une pause entre l'instruction et la variable.
-   • Utilisé pour entrer des parenthèses.
-  • Déplace le curseur vers la droite (appuyez une seule fois pour avancer d'une position, maintenez la touche enfoncée pour avancer automatiquement).  
• Exécute les instructions de retour arrière.  
• En utilisation manuelle efface une condition d'erreur.
-  • Déplace le curseur vers la gauche (appuyez une seule fois pour avancer d'une position, maintenez la touche enfoncée pour avancer automatiquement).  
• Sinon même chose que la touche .
-  • Insère un espace (  apparaît) correspondant à un caractère entre l'adresse (N) indiquée par le curseur et l'adresse précédente (N-1).
-  • Supprime le contenu de l'adresse indiquée par le curseur.
-  • Utilisé pour désigner PI ( $\pi$ ).
-  • Utilisé pour désigner la racine carrée.
-  • Entre une ligne de programme dans la mémoire de l'ordinateur.  
• Utilisée lors de l'écriture des programmes.  
• Demande le calcul manuel ou l'exécution directe d'une commande par l'ordinateur.  
• Relance un programme arrêté temporairement par une commande INPUT ou PRINT.
-  • Permet de passer en mode impression ou de supprimer ce mode lorsque l'imprimante en option est connectée au PC-1350.
-  • Place l'ordinateur en mode minuscule et le restaure en mode majuscule. (Mets en fonction et hors fonction le symbole SML.)

- Le symbole SML est affiché lorsque **SML** est enfoncé. Maintenant si **A**, **B** et **C** sont enfoncés, c'est a, b et c qui s'affichent. Si **SML** est à nouveau enfoncé, le symbole SML disparaît et les caractères sont à nouveau entrés en majuscule.

Les touches **↑** et **↓** ont les fonctions suivantes suivant le mode spécifié et l'état de l'ordinateur.

| Mode                                                                                             | Etat                                                               | <b>↓</b>                                      | <b>↑</b>                                                                                                   |
|--------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|-----------------------------------------------|------------------------------------------------------------------------------------------------------------|
| RUN                                                                                              | Programme en cours d'exécution                                     |                                               |                                                                                                            |
|                                                                                                  | Le programme est temporairement interrompu                         | Pour exécuter la ligne suivante               | Pour afficher la ligne de programme en cours d'exécution ou déjà exécuté, maintenez cette touche enfoncée. |
|                                                                                                  | Instruction PRINT, LINE, PSET, PRESET, GPRINT en cours d'exécution |                                               |                                                                                                            |
|                                                                                                  | Lors d'une interruption                                            |                                               |                                                                                                            |
|                                                                                                  | Instruction INPUT en cours d'exécution                             |                                               |                                                                                                            |
|                                                                                                  | Condition d'erreur au cours d'exécution d'un programme             |                                               | Maintenez cette touche enfoncée pour afficher la ligne qui est cause d'erreur.                             |
|                                                                                                  | Condition TRON                                                     | Pour exécution une opération de mise au point | Pour afficher la ligne de programme cours d'exécution ou déjà exécuté, maintenez cette touche enfoncée     |
| (Lorsque le mode passe de RUN à PRO et que la ligne de programme n'est pas en cours d'affichage) |                                                                    |                                               |                                                                                                            |
| PRO                                                                                              | Le programme est temporairement interrompu                         | Pour afficher la ligne interrompue            | Même chose que colonne précédente                                                                          |
|                                                                                                  | Condition d'erreur                                                 | Pour afficher la ligne erronée                | Même chose que colonne précédente                                                                          |
|                                                                                                  | Autre condition                                                    | Pour afficher la première ligne               | Pour afficher la dernière ligne                                                                            |
| (Lorsque la ligne de programme est en cours d'affichage)                                         |                                                                    |                                               |                                                                                                            |
|                                                                                                  |                                                                    | Pour afficher la ligne de programme suivante  | Pour afficher la ligne de programme précédente                                                             |
| RESERVE                                                                                          |                                                                    |                                               |                                                                                                            |

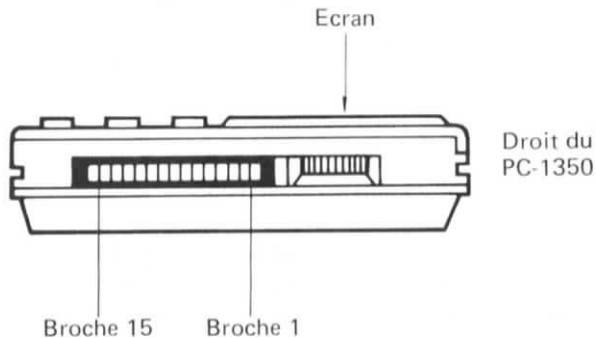
## ANNEXE E

### Fonction des Touches en Basic

- Sur l'écran, la touche **ENTER** a les mêmes effets qu'un espace.
- Si l'ordinateur attend l'entrée de données et si aucune touche n'est utilisée pendant environ 11 minutes, il est automatiquement mis hors tension (fonction de mise hors tension automatique).

## ANNEXE F SIGNAUX UTILISES AVEC LE TERMINAL D'ENTREE-SORTIE SERIE

Le PC-1350 est équipé d'un connecteur de 15 broches pour le terminal d'entrée-sortie série. Les broches utilisées et leurs signaux sont décrits ci-dessous.



### Affectation des broches

| Broches | Nom                        | Symbole | E/S | Fonctions                                 |
|---------|----------------------------|---------|-----|-------------------------------------------|
| 1       | Mise à la terre du chassis | FG      |     | Mise à la terre pour la maintenance       |
| 2       | Transmission données       | SD      | s   | Signal de sortie courant continu          |
| 3       | Reception données          | RD      | e   | Signal entré courant continu              |
| 4       | Demande de données         | RS      | s   | ON: Envoie la porteuse                    |
| 5       | Prêt à émettre             | CS      | e   | ON: Transmission possible                 |
| 7       | Signal terre               | SG      | —   | Référence tension 0 pour tous les signaux |
| 8       | Détection porteuse données | CD      | e   | ON: Signal porteuse reçu                  |
| 10      |                            | VC      |     | Alimentation en courant                   |
| 11      | Prêt à recevoir            | RR      | s   | ON: Réception validée                     |
| 13      |                            | VC      |     | Alimentation en courant                   |
| 14      | Terminal prêt              | ER      | s   | ON: Terminal local prêt                   |

**ANNEXE F**  
**Signaux Utilisés Avec le Terminal**

- Remarque 1.** Le niveau de tension de VC est le niveau haut. Le niveau de tension de SG est le niveau bas.
- Remarque 2.** Si la tension appliquée au PC-1350 est supérieure aux limites autorisées (comprises entre SG et VC) les composants internes risquent d'être abîmés puisque le PC-1350 utilise des composants CMOS.

# ANNEXE G

## SPECIFICATIONS

|                                                   |                                                                                                                                                                                                                                                                                   |          |                           |
|---------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------------------------|
| <b>Modèle:</b>                                    | Ordinateur de poche PC-1350                                                                                                                                                                                                                                                       |          |                           |
| <b>Processeur:</b>                                | CPU CMOS 8 bits                                                                                                                                                                                                                                                                   |          |                           |
| <b>Langage de programmation:</b>                  | BASIC                                                                                                                                                                                                                                                                             |          |                           |
| <b>ROM système:</b>                               |                                                                                                                                                                                                                                                                                   |          | 40 K octets               |
| <b>Capacité mémoire:</b>                          | <b>ROM système:</b>                                                                                                                                                                                                                                                               |          |                           |
|                                                   | <b>RAM:</b>                                                                                                                                                                                                                                                                       |          | 1.6 K octets              |
|                                                   | <b>Système</b>                                                                                                                                                                                                                                                                    |          |                           |
|                                                   | <b>Utilisateur:</b>                                                                                                                                                                                                                                                               |          | 208 octets                |
|                                                   | <b>Zone mémoire fixe</b>                                                                                                                                                                                                                                                          |          |                           |
|                                                   | <b>(A – Z, A\$ – Z\$)</b>                                                                                                                                                                                                                                                         |          | 3070 octets               |
|                                                   | <b>Zone des programmes/données</b>                                                                                                                                                                                                                                                |          |                           |
|                                                   | <b>Zone réservée</b>                                                                                                                                                                                                                                                              |          | 144 octets                |
| <b>Piles:</b>                                     | <b>Routines:</b>                                                                                                                                                                                                                                                                  | 10 piles | <b>Fonction:</b> 16 piles |
|                                                   | <b>FOR-NEXT:</b>                                                                                                                                                                                                                                                                  | 5 piles  | <b>Données:</b> 8 piles   |
| <b>Opérateurs:</b>                                | Addition, soustraction, multiplication, division, fonctions trigonométriques et trigonométriques inverses, fonctions logarithmiques et exponentielles, conversion des angles, carré et racine carrée, signe, valeur absolue, entier, opérateurs de relation, opérateurs logiques. |          |                           |
| <b>Précision numérique:</b>                       | 10 chiffres (mantisse) + 2 chiffres (exposant).                                                                                                                                                                                                                                   |          |                           |
| <b>Possibilité d'édition:</b>                     | Déplacement du curseur vers la droite et vers la gauche, affichage de la ligne supérieure ou inférieure, insertion et suppression de caractères.                                                                                                                                  |          |                           |
| <b>Protection mémoire:</b>                        | Sauvegarde grâce aux piles CMOS.                                                                                                                                                                                                                                                  |          |                           |
| <b>Caractéristiques de l'entrée-sortie série:</b> | Norme: système de transmission départ-arrêt (asynchrone) semi duplex uniquement.                                                                                                                                                                                                  |          |                           |
|                                                   | Vitesses en bauds: 300, 600, 1200 bauds                                                                                                                                                                                                                                           |          |                           |
|                                                   | Bits de données: 7 ou 8 bits                                                                                                                                                                                                                                                      |          |                           |
|                                                   | Bits de parité: Parité paire, impaire, ou pas de parité                                                                                                                                                                                                                           |          |                           |
|                                                   | Bits d'arrêt: 1 ou 2 bits                                                                                                                                                                                                                                                         |          |                           |
|                                                   | Connecteur utilisé: 15 broches (pour l'équipement externe)                                                                                                                                                                                                                        |          |                           |
|                                                   | Niveau du signal de sortie: Niveau C-MOS (4 à 6 volts)                                                                                                                                                                                                                            |          |                           |
|                                                   | Signaux d'interface: Entrées: RS, CS, CD                                                                                                                                                                                                                                          |          |                           |
|                                                   | Sorties: SD, RS, RR, ER                                                                                                                                                                                                                                                           |          |                           |
|                                                   | Autres: SG, FG, VC                                                                                                                                                                                                                                                                |          |                           |



## ANNEXE H UTILISATION DE PROGRAMMES MIS AU POINT POUR LE PC-1245, LE PC-1250/1251, LE PC-1260/1261 OU LE PC-1401

**Remarques:** La série PC-1210 comporte le PC-1210 et le PC-1211.  
La série PC-1245 comporte le PC-1245, le PC-1250, et le PC-1251.  
La série PC-1260 comporte le PC-1260 et le PC-1261.

Les écrans des ordinateurs de poche de la série PC-1245, PC-1210, PC-1260 et PC-1401 comportent soit une soit deux lignes. Par conséquent, pour modifier les données affichées à l'écran il faut tout d'abord effacer les données précédentes. Avec le PC-1350, les données affichées (4 lignes) remontent vers le haut chaque fois qu'une nouvelle ligne est affichée.

Par conséquent, lors de l'exécution de programmes qui déplacent des caractères ou des symboles ou de programmes utilisant la commande CURSOR avec les PC de la série 1260, on peut obtenir des mouvements ou des affichages inattendus. Il est donc nécessaire de modifier le programme, par exemple en effaçant l'écran à l'aide de la commande CLS.

D'autre part, étant donné que les fonctions risquent d'être légèrement différentes suivant la série de l'ordinateur de poche, il faut modifier le programme de la façon décrite ci-après.

### MODIFICATIONS REQUISES POUR LES PROGRAMMES DES ORDINATEURS DE LA SERIE PC-1245 (PC-1245, PC-1250, PC-1251)

Lorsque vous utilisez des programmes mis au point pour les PC de la série 1245, il est nécessaire de modifier ce qui suit:

1. Multiplication sans utiliser l'opérateur "\*" .  
Sur les PC de la série 1245, l'opérateur (\*) de multiplication peut être omis comme AB à la place de A\*B ou CD à la place de C\*D. Sur le PC-1350, l'opérateur de multiplication (\*) ne peut pas être omis puisque l'ordinateur traite deux caractères consécutifs comme AB ou CD comme des variables simples. Utilisez la spécification de droite dans l'exemple suivant:

(exemple) A=SIN BC → A=SIN (B\*C)



2. Définition des variables indicées (tel que A ( )) à l'aide de l'instruction DIM. Sur les PC de la série 1245, si, par exemple, DIM A(30) est exécuté, les emplacements de mémoire pour A(27) à A(30) sont réservés comme une extension de la zone de définition de variable fixe. Cependant sur le PC-1350, l'exécution de DIM A(30) réserve une zone de mémoire distincte de variable fixe pour les variables tableaux A(0) à A(30) pour le tableau appelé A. Pour définir des variables indicées (tel que A ( )) comme extension de variables fixes, utilisez la spécification de droite dans l'exemple ci-dessous:

DIM A(30) → A(30)=0

3. Instruction d'entrée-sortie de données pour les fichiers sur bandes. Sur les PC de la série 1245, l'exécution de l'instruction PRINT#C, par exemple, sauvegarde le contenu de la variable C et de toutes les variables suivantes dans un fichier sur bande. Cependant, sur le PC-1350, l'exécution de cette même instruction sauvegarde le contenu de la variable C uniquement. Pour sauvegarder le contenu d'une variable déterminée et de toutes les variables suivantes, utilisez la spécification de droite dans les exemples ci-dessous:

(exemple) PRINT#A → PRINT#A\*  
PRINT#C → PRINT#C\*

4. Valeur d'une variable boucle à la fin d'une boucle FOR-NEXT: La valeur d'une variable boucle obtenue après exécution d'une boucle FOR-NEXT sur le PC-1350 est différente de celle obtenue sur les PC de la série 1245. Si la valeur d'une variable boucle est utilisée comme expression conditionnelle dans un programme écrit pour les PC de la série 1245, ajoutez-lui 1 lorsqu'elle est utilisée sur le PC-1350.

(exemple) 10 FOR I=0 TO 10

~

50 NEXT I  
60 IF I=10 THEN 100

Modifie la valeur de I à la ligne 60 comme suit:

60 I=11 THEN 100

(sur le PC-1350, la valeur d'une variable boucle doit être augmentée de la valeur de l'incrément. Le nombre de cycles d'exécution de la boucle reste cependant le même.)

5. Symbole d'élévation à la puissance "IE":

Le PC-1350 utilise la lettre majuscule "E" comme symbole d'élévation à la puissance. La modification suivante doit être apportée:

A=1.234 IE 5 → A=1.234E5  
B=IE 6 → B=1E6

Si un programme pour le PC-1245 est lu à partir d'un fichier sur bande dans le PC-1350, la modification pour le symbole d'élévation à la puissance qui vient d'être décrite ci-dessus sera effectuée automatiquement par le PC-1350.

6. Le code caractère des PC de la série 1245 est légèrement différent de celui du PC-1350.

Lorsque les codes suivants sont désignés par la fonction CHR\$, modifier les codes.

| Code de caractères | PC-1245  | PC-1350 |
|--------------------|----------|---------|
| 39 (&27)           | ⏏        | ,       |
| 91 (&5B)           | √        | [       |
| 92 (&5C)           | ¥        | \       |
| 93 (&5D)           | π        | ]       |
| 96 (&60)           | IE       | '       |
| 250 (&EA)          | -(Error) | ⏏       |
| 251 (&FB)          | -(Error) | π       |
| 252 (&FC)          | -(Error) | √       |

**Remarque:** Comme indiqué ci-dessus, le PC-1350 n'a pas le caractère IE.

## MODIFICATIONS SUPPLEMENTAIRES

1. Les PC-1245, PC-1250 et PC-1251 utilisent des numéros de ligne allant de 1 à 999, alors que, avec ce modèle, vous pouvez utiliser des numéros de ligne allant de 1 à 65279. Par conséquent le numéro de ligne utilise 3 octets dans la RAM (les PC de la série 1245 n'en utilisent que 2). Cette modification est automatiquement effectuée lorsque le programme est chargé à partir d'une bande. Il peut se produire un dépassement de capacité de la mémoire (ERROR 6) lors du changement ou de l'exécution d'un programme long.


De plus lorsqu'une ligne de programme comporte presque 80 octets, cette modification peut provoquer l'effacement de la fin de la ligne.

2. Lorsqu'un programme écrit pour les PC de la série 1245 est chargé à partir d'une bande, l'ordinateur reste occupé (BUSY) pendant une ou deux secondes après que la bande se soit arrêtée, en raison des changements des numéros de ligne (passage de deux à trois octets) comme indiqué précédemment. Au cours de ce laps de temps, le symbole "\*" sera affiché dans le coin inférieur de l'écran comme pour le chargement d'un programme.

**Remarque:** Les PC de la série 1245 ne peuvent lire une bande qui contient des programmes mis au point pour le PC-1350.

## MODIFICATIONS REQUISES POUR LES PROGRAMMES DU PC-1401

(1) Modifications concernant la touche  :

La touche  ne fait pas partie dans cet ordinateur des touches permettant d'identifier un programme. Par conséquent, il faut choisir une autre touche pour les programmes définis par cette touche.

Exemple:

100 " , " : → 100 " = " :

(2) Autres remarques:

Le PC-1401 comporte plus de commandes de fonction que le PC-1350. Par conséquent, une erreur se produira au cours de l'exécution si un programme qui utilise des commandes n'existant pas sur le PC-1350 est écrit ou lu à partir d'une bande. Dans ce cas, ces commandes seront remplacées par "~" puis affichées.

## MODIFICATIONS DES PROGRAMMES DE LA SERIE PC-1210

Pour utiliser les programmes des PC de la série 1210 sur le PC-1350, ceux-ci doivent être modifiés de la même façon que les programmes de la série 1245 (sauf les points 2 et 6). De plus, les modifications suivantes sont nécessaires:

(1) Instruction IF

Si, par exemple,

```
50 IF A > L PRINT "A" (affichage "A" si A > L)
```

est rencontré dans le programme des ordinateurs de poche de la série 1210, cette instruction est interprétée de la manière suivante:

```
50 IF A > LPRINT "A" (imprime "A" si A >)
```

et il s'ensuit une erreur.

L'erreur a lieu car une commande qui n'existe pas dans les PC de la série 1210 existe en fait sur le PC-1350.

Pour résoudre ce problème, insérez une commande THEN dans l'instruction IF comme suit:

```
50 IF A > L THEN PRINT "A"
```

(2) Format spécifié dans USING

La fonction de la commande USING dans le PC-1350 et le PC-1210 diffère comme indiqué ci-dessous:

[Exemple]

10 A = -123,456

20 PAUSE USING "####.##" ; A

30 PAUSE A, USING "####" ; A

L'exécution de ce programme provoque l'affichage de ce qui suit:

|                   |         |         |
|-------------------|---------|---------|
| * PC-1210/PC-1211 |         | -123.45 |
|                   | -123    | -123    |
| * PC-1350         |         | -123.45 |
|                   | -123.45 | -123    |

Pour l'exécution de la ligne 30 dans les PC de la série 1210, l'affichage sur le côté gauche suit également le format affiché sur le côté droit. Avec le PC-1350, l'affichage suit le format spécifié précédemment. Ceci ne s'applique pas uniquement à la commande PAUSE mais également aux commandes PRINT et LPRINT.

(3) Omission de ""

Avec les PC de la série 1210, la "" qui vient immédiatement avant **ENTER** ou : (2 points) peut être omise. Elle ne peut pas être omise avec le PC-1350. Par conséquent, n'oubliez d'ajouter "" dans le programme si elle est omise.

(4) Commande d'impression

Le PC-1350 a une commande PRINT pour l'affichage à l'écran et une commande LPRINT pour l'impression. Cependant, toutes les commandes PRINT peuvent être utilisées pour l'impression si PRINT = LPRINT a été spécifiée. (Voir page 147.)

Les PC de la série 1210 n'ont pas de commande LPRINT. Pour imprimer lorsque vous utilisez un programme écrit pour le PC de la série 1210, ajoutez PRINT = LPRINT au programme, ou exécutez l'impression en mode manuel.

(5) Variables

Une fois la commande RUN exécutée sur les PC de la série 1210, toutes les variables sont conservées. Avec le PC-1350, toutes les variables à partir de A(27) sont effacées. (Voir page 48.)

Par conséquent, si vous devez conserver les variables au début de l'exécution d'un programme, commencez l'exécution de votre programme par la commande GOTO ou par les touches définissant une fonction.

## MODIFICATION DES PROGRAMMES DES PC DE LA SERIE 1260

### Modification du code caractère

Dans les PC de la série 1260, le caractère correspondant au code caractère 96 (&60) est un espace, alors que dans le PC-1350 c'est une apostrophe (').

Par conséquent, si un espace a été spécifié à l'aide du code 96 dans une commande CHR\$, changez ce code à 32 (\$20)

## EXEMPLES DE PROGRAMME

La lecture de cette brochure vous a permis d'acquérir un certain nombre de connaissances sur les commandes des programmes. Il est cependant nécessaire que vous écriviez vous même de vrais programmes de façon à maîtriser le langage BASIC et à pouvoir créer librement vos programmes. De même que la conduite d'une voiture et que votre technique en tennis s'améliorent par la pratique, de même vous ne pourrez vous améliorer en programmation qu'en écrivant le plus de programmes possible, quelque soit votre niveau. Il est également important pour vous de vous référer aux programmes créés par d'autres. Dans ce but, les pages ci-dessous contiennent un certain nombre de programmes utilisant des commandes BASIC.

(Sharp Corporation et/ou ses filiales ne peuvent être tenues pour responsables des pertes ou dommages pouvant survenir de l'emploi des logiciels indiqué dans cette brochure).

## TABLE DES MATIERES

| Titre du programme                                                                                                                                                                        | Page |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| ● JEU D'ATTERRISSAGE EN DOUCEUR . . . . .                                                                                                                                                 | 243  |
| ● CALCUL DU NOMBRE DE JOURS ENTRE DEUX DATES.<br>CALCUL DU JOUR A PARTIR D'UNE DATE. . . . .                                                                                              | 247  |
| ● ANALYSE ABC. . . . .                                                                                                                                                                    | 252  |
| ● MATRICE . . . . .                                                                                                                                                                       | 258  |
| ● MOYENNE MODIFIEE DE MANIERE DYNAMIQUE . . . . .                                                                                                                                         | 263  |
| ● METHODE D'ARRANGEMENT UNIQUE. . . . .                                                                                                                                                   | 268  |
| ● TEST F . . . . .                                                                                                                                                                        | 274  |
| * L'écran du PC-1350 affiche 4 lignes mais avec la séquence d'utilisation des touches dans un programme partiel, seules les parties véritablement affichées sont enregistrées en mémoire. |      |
| * Le nombre d'octets utilisé par chaque programme est indiqué à la fin de chaque programme.                                                                                               |      |

## ■ PRESENTATION

Bientôt plus de combustible! Atterissage d'urgence!

Comment faire atterir une fusée avec le peu de combustible restant? La fusée est en chute libre. Le moteur de la fusée a été mis en marche pour ralentir sa chute libre. Si l'allumage a lieu trop tôt ou si trop de combustible est utilisé, la fusée se désagrège en poussière. Si vous allumez le moteur trop tard, la fusée s'écrase à la surface de la planète.

Et maintenant, servez-vous de votre intelligence et de votre courage pour faire atterir la fusée sans dégât. Le secret pour réussir est de contrôler le moment de l'allumage et la consommation de combustible.

## ■ INSTRUCTIONS

1. Pour lancer le programme, appuyez sur les touches **DEF** **A** .
2. Entrez le montant de combustible utilisé au moyen des touches **0** à **9** . Pour faire atterir la fusée, il faut régler avec soin la consommation de combustible. Lorsque vous réussissez, le combustible restant s'affiche; lorsque vous échouez "Good bye!!" s'affiche. Le programme s'arrête à moins que vous n'entriez une instruction pour demander si vous voulez jouer à nouveau ou non.

## ■ CONTENU

La gravité est fixée à  $5m/(\text{unité de temps})^2$ .

Si 5 unités de combustible pour une unité de temps sont brûlées, la gravité est alors.

$$H = H_0 + V_0 t + \frac{1}{2} a t^2$$

$$V = V_0 + a t$$

$$V^2 = V_0^2 + 2aH$$

$$H_0 = 500, V_0 = -50, F_0 = 200$$

H: Hauteur             $H_0$ : Hauteur initiale

V: Vitesse             $V_0$ : Vitesse initiale

a: Accélération de la gravitation     $F_0$ : Combustible au départ

t: Temps            F: Combustible utilisé

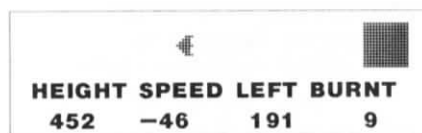
**Remarque:** Vous pouvez rendre le jeu encore plus intéressant en changeant les valeurs numériques de temps d'attente à la ligne 40, le combustible de départ à la ligne 50 et la vitesse initiale à la ligne 60.

## ■ SEQUENCE ES TOUCHES

1)   (Début du programme)



2)  (Combustible brûlé)



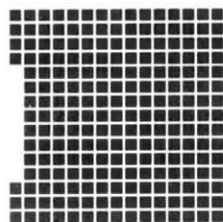
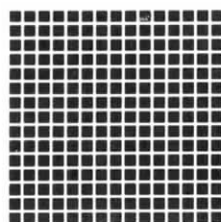
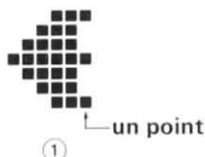
3)



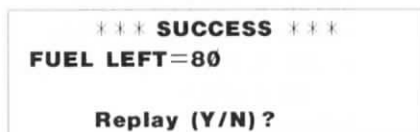
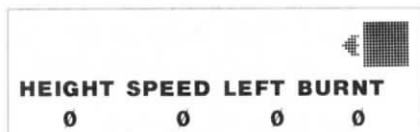
Entrez le combustible brûlé de la même manière et pour suivez le jeu.

### Référence

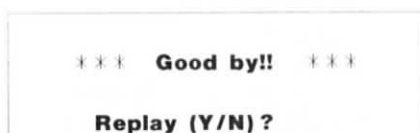
Configuration de points pour le jeu d'atterrissage en douceur



“Exemples d'écrans affichés lorsque vous avez réussi”



“Exemples d'écrans lorsque vous avez raté”



\* Entrer  si vous voulez jouer à nouveau.

Reportez-vous aux configurations: de la page suivante.



## ■ IMPRESSION EN SORTIE

```

10:"A": WAIT 0: CLEAR :
   USING : CLS
20:S=-50:A=0:D$="":V=30
30:CURSOR 0,1: PRINT "*"
   ** SOFTLANDING GAME
   ***": BEEP 3: CLS
40:DATA "TIME=",5
50:DATA "FUEL=",200
60:DATA "HEIGHT=",500
70:RESTORE : READ B$,W,
   B$,F,B$,H
80:WAIT W
90:FOR P=1 TO 2:
   GCURSOR (135,P*7):
   GPRINT "FFFFFFFFFFFFFF
   FFFFFFFFFFFFFFFFFF":
   NEXT P
100:CURSOR 0,2: PRINT "H
   EIGHT SPEED LEFT BUR
   NT"
110:GOSUB 300
120:L=V
130:IF F<=0 LET C=0:
   BEEP 1: GOTO 180
140:BEEP 1:D$= INKEY$
150:IF D$="" LET C=A:
   GOTO 180
160:C= VAL D$
170:A=C
180:IF C>F LET C=F
190:F=F-C:X=C-5:H=H+S+X/
   2:S=S+X
200:IF H>0 LET V=(500-H)
   /5+30: GOTO 110
210:IF ABS H<5 OR ABS S<
   5 LET H=0:S=0:L=V:V=
   129: GOSUB 300:
   BEEP 5: CLS : PRINT
   " *** SUCCESS **
   *": GOTO 250
220:GCURSOR (L,11):
   GPRINT "000000000000
   "
230:GCURSOR (129,11):
   GPRINT "15A006A00A21

```

```

08"
240:BEEP 3: CLS :
   CURSOR 0,1: PRINT "
   *** Good by!! ***
   ": GOTO 260
250:PRINT USING "####":
   FUEL LEFT =":F
260:CURSOR 0,3: PRINT "
   Replay (Y/N) ?"
270:Z$= INKEY$: IF Z$=
   Y" THEN 10
280:IF Z$="N" END
290:GOTO 270
300:CURSOR 0,3: PRINT
   USING "####":H;
   USING "####":S;F;
   USING "####":C
310:GCURSOR (L,11):
   GPRINT "000000000000
   "
320:GCURSOR (V,11):
   GPRINT "081C3E7F7F49
   "
330:RETURN

```

779 octets

## ■ CONTENU DE LA MEMOIRE

|     |                                              |
|-----|----------------------------------------------|
| A   | ✓                                            |
| B\$ | ✓                                            |
| C   | Combustible brûlé                            |
| D\$ | Combustible brûlé                            |
| E   |                                              |
| F   | Combustible au départ<br>Combustible restant |
| G   |                                              |
| H   | Hauteur initiale<br>Hauteur                  |
| I   |                                              |
| J   |                                              |
| K   |                                              |
| L   | ✓                                            |
| M   |                                              |
| N   |                                              |
| O   |                                              |
| P   | Compteur de beucle                           |
| Q   |                                              |
| R   |                                              |
| S   | Vitesse                                      |
| T   |                                              |
| U   |                                              |
| V   | ✓                                            |
| W   | Temps d'attente                              |
| X   | ✓                                            |
| Y   |                                              |
| Z\$ | ✓                                            |

Tire du programme: **CALCUL DU NOMBRE DE JOURS ENTRE DEUX DATES. CALCUL DU JOUR A PARTIR D'UNE DATE**

## ■ PRESENTATION

Deux types de calcul sont possibles avec ce programme. Vous pouvez calculer le nombre de jours entre une date et une autre ou quel sera le jour, le mois et l'année un certain nombre de jours après une date donnée. Ce programme vous permet de trouver rapidement ces résultats; il suffit d'indiquer la date de départ et la date finale ou la date de départ et le nombre de jours. Par exemple tentez de calculer le nombre de jours écoulés depuis votre naissance jusqu'à ce jour ou quel sera le jour, le mois et l'année dans 10.000 ans.

## ■ INSTRUCTIONS

**DEF** **A** . . . . . Calcul du nombre de jours

1. Entrez l'année, le mois et le jour de départ en fonction de ce qui est affiché à l'écran, utilisez **N** **ENTER** lorsque vous voulez modifier un détail et **Y** **ENTER** lorsque vous ne voulez rien modifier car l'ordinateur demandera s'il y a une erreur ou pas dans le contenu affiché à l'écran.
2. Ensuite si vous entrez la date de départ et la date finale, le nombre de jours s'affichera. De plus, une fois le nombre de jours affiché, il est possible d'entrer à nouveau une date de départ en utilisant la touche **ENTER**.
3. Pour mettre fin au programme, appuyez sur la touche **ENTER** alors que l'ordinateur attend une nouvelle fois l'entrée de la date de départ.

**DEF** **B** . . . . . Calcul du jour à partir d'une date

Suivez la même procédure que ci-dessus.

## ■ EXEMPLES

1. Calcul du nombre de jours  
Calculez le nombre de jours entre le 7 février 1961 et le 9 mai 1984.
2. Calcul du jour à partir d'une date  
Calculez à partir du 7 février 1961 quel sera le jour, le mois et l'année après que 8492 jours se soient écoulés.

## ■ SEQUENCE D'UTILISATION DES TOUCHES

“Dans le cas de l'exemple 1”

1) **DEF** **A** [Lancement du programme]

```
** NO. OF DAYS CALC. **
START Y=? M= D=
TARGET Y= M= D=
```

2) 1961 **ENTER** 2 **ENTER** 7 **ENTER**

```
** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
TARGET Y= M= D=
OK(Y/N)?_
```

3) **Y** **ENTER**

```
** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
TARGET Y=? M= D=
```

4) 1984 **ENTER** 5 **ENTER** 9 **ENTER**

```
** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
TARGET Y=1984 M=5 D=9
OK(Y/N)?_
```

5) **Y** **ENTER** [Affichage de la date]

```
** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
TARGET Y=1984 M=5 D=9
PLEASE WAIT !!
```

```
** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
TARGET Y=1984 M=5 D=9
NO. OF DAYS =>8492_
```

6) **ENTER** [Le système attend l'entrée de l'année de départ]

```
** NO. OF DAYS CALC. **
START Y=? M= D=
TARGET Y= M= D=
```

7) **ENTER** [Fin du programme]

>

## ■ SEQUENCE D'UTILISATION DES TOUCHES

"Dans le cas de l'exemple 2"

- 1) **DEF** **B** [Lancement du programme]

```

** NO. OF DAYS CALC. **
START Y=? M= D=
DAYS =
    
```

- 2) 1960 **ENTER** 2 **ENTER** 7 **ENTER**

```

** NO. OF DAYS CALC. **
START Y=1960 M=2 D=7
DAYS=
OK(Y/N)?_
    
```

- 3) **N** **ENTER** [Une erreur a été détectée dans les données entrées]

```

** NO. OF DAYS CALC. **
START Y=? M= D=
DAYS =
    
```

- 4) 1961 **ENTER** [Corrigez les données entrées]

```

** NO. OF DAYS CALC. **
START Y=1961 M=? D=
DAYS =
    
```

- 5) **Y** **ENTER**

```

** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
DAYS =?
    
```

- 6) 8492 **ENTER** [Entrée du nombre de jours]

```

** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
DAYS =8492
OK(Y/N)?_
    
```

- 7) **Y** **ENTER** [Affichage de la date finale]

```

** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
DAYS =8492
PLEASE WAIT!!
    
```

```

** NO. OF DAYS CALC. **
START Y=1961 M=2 D=7
DAYS =8492
TARGET Y=1984. M=5. D=9.
    
```

- 8) **ENTER** [Le système attend l'entrée de la date de début]

```

** NO. OF DAYS CALC. **
START Y=? M= D=
DAYS =
    
```

- 9) **ENTER** [Fin du programme]

>

## ■ LISTE DU PROGRAMME

```

10:REM *****
20:REM NO. OF DAYS CALC
30:REM *****
40:"A": CLEAR : CLS :
   WAIT 0: USING :A=0:
   GOTO 50
45:"B": CLEAR : CLS :
   WAIT 0: USING :A=1
50:PRINT "** NO. OF DAY
   S CALC. **"
60:BEEP 1
80:CURSOR 0,1: PRINT "S
   TART Y=":" M=":
   " D= "
85:IF A=1 CURSOR 0,2:
   PRINT "DAYS =
   " : GOTO 95
90:CURSOR 0,2: PRINT "T
   ARGET Y=":" M=":
   " D= "
95:CURSOR 0,3: PRINT "
   "
97:IF C=1 THEN 180
100:CURSOR 9,1: INPUT R:
   GOTO 120
110:CLS : END
120:CURSOR 16,1: INPUT S
130:CURSOR 21,1: INPUT T
140:CURSOR 0,3: PRINT "
   "
150:CURSOR 13,3: INPUT "
   OK(Y/N)?" : F$
160:IF F$="N" THEN 80
165:IF F$(">")Y" GOTO 150
170:CURSOR 0,3: PRINT "
   "
180:CURSOR 9,2: INPUT U
189:IF A=1 LET X=U:
   GOTO 210
190:CURSOR 16,2: INPUT V
200:CURSOR 21,2: INPUT W
210:CURSOR 13,3: INPUT "
   OK(Y/N)?" : F$
220:IF F$="N" LET C=1:
   GOTO 85
225:IF F$(">")Y" THEN 210
230:CURSOR 0,3: PRINT "
   PLEASE WAIT !!
   "
235:IF A=1 THEN 400
240:H=R:G=S:I=T: GOSUB 3
   00:J=I:H=U:G=V:I=W:
   GOSUB 300:X=I-J
250:BEEP 1: CURSOR 0,3:
   PRINT "
   "
260:CURSOR 0,3: PRINT "N
   O. OF DAYS =>":
   USING "#####":X
270:CURSOR 20,3: INPUT "
   " : K
280:USING :C=0: GOTO 80
300:IF G-3>=0 LET G=G+1:
   GOTO 320
310:G=G+13:H=H-1
320:I= INT (365.25*H)+
   INT (30.6*G)+I:I=I-
   INT (H/100)+ INT (H/
   400)-428: RETURN
400:H=R:G=S:I=T: GOSUB 6
   00:J=Q+X:G=1:I=0
410:H=H+1: GOSUB 600
420:IF J>Q THEN 410
430:H=H-1
440:G=G+1: GOSUB 600
450:IF J>Q THEN 440
460:G=G-1: GOSUB 600:I=J
   -Q:H=H-B: BEEP 1:
   CURSOR 0,3: PRINT "
   "
470:CURSOR 0,3: PRINT "T
   ARGET Y=":" M=":
   " D= "
475:CURSOR 9,3: PRINT H:
   CURSOR 16,3: PRINT G
   : CURSOR 21,3:
   PRINT I
480:CURSOR 21,2: INPUT "
   " : K
490:GOTO 80
600:IF G-3>=0 LET O=G+1:
   P=H: GOTO 620
610:O=G+13:P=H-1
620:O= INT (365.25*P)+
   INT (30.6*O)+I:O=O-
   INT (P/100)+ INT (P/
   400)-428: RETURN

```

1244 octets

■ CONTENU DE LA MEMOIRE

|     |                 |
|-----|-----------------|
| A   | ✓               |
| B   | ✓               |
| C   | ✓               |
| D   |                 |
| E   |                 |
| F\$ | ✓               |
| G   | ✓               |
| H   | ✓               |
| I   | ✓               |
| J   | ✓               |
| K   | ✓               |
| L   |                 |
| M   |                 |
| N   |                 |
| O   | ✓               |
| P   | ✓               |
| Q   | ✓               |
| R   | Année de départ |
| S   | Mois de départ  |
| T   | Jour de départ  |
| U   | Année finale    |
| V   | Mois final      |
| W   | Jour final      |
| X   | Nombre de jours |
| Y   |                 |
| Z   |                 |

## ■ PRESENTATION

Ce programme permet de savoir quelle est la place d'un produit. Il est important pour les responsables des ventes de savoir quelle position occupe chaque produit dans les statistiques générales des ventes. Lorsque vous utilisez l'analyse ABC qui est avant tout un moyen de gérer les stocks et lorsque vous entrez le montant vendu et le numéro du produit, ce programme calcul le pourcentage des ventes, le pourcentage dans le total des ventes et l'imprime ainsi que sa place parmi les autres produits.

## ■ INSTRUCTIONS

1. Lancez le programme en frappant **R** **U** **N** **ENTER**.  
Entrez le numéro du produit et les statistiques de vente en fonction de ce qui s'affiche à l'écran.  
Lorsque l'ordinateur attend l'entrée du numéro du produit, vous pouvez passer à l'étape 2 si vous appuyez sur la touche **ENTER** uniquement.
2. Choisissez si vous voulez modifier les données que vous allez entrer ensuite. Appuyez sur **Y** **ENTER** pour modifier ces données. Dans ce cas si les données sont erronées entrez les données correctes lorsque les statistiques de vente et les numéros des produits sont affichés à l'écran en utilisant uniquement la touche **ENTER**. Lorsqu'il n'est pas nécessaire de modifier les données appuyez sur **N** **ENTER**.
3. Sortez le taux de vente, le taux par rapport au total et la place du produit. Pour imprimer ces résultats sur l'imprimante, appuyez sur les touches **P** **ENTER**. Pour afficher ces résultats à l'écran, appuyez sur **L** **ENTER**. De plus lorsque tout ne peut être affiché, appuyez sur la touche **U** (défilement vers le haut) **D** (défilement vers le bas). Pour mettre fin au programme appuyez **E** **ENTER**.
4. Lorsque vous imprimez les résultats ou lorsque vous avez entré **Q** après qu'ils aient été affichés, vous revenez à l'étape 3.

**Remarque:** Les numéros de produit peuvent comporter jusqu'à 7 lettres et les statistiques de vente jusqu'à 6 chiffres au maximum. De plus le nombre maximum de cases dans lequel il est possible d'entrer des données est de 26.



## ■ EXEMPLE

Le taux des ventes et les taux globaux sont calculés à partir des données ci-dessous et la place du produit est recherchée.

| No. Produit | Ventes  |
|-------------|---------|
| TR-300      | 89.000  |
| BT-650      | 17.000  |
| GZ-90       | 23.000  |
| RS-15       | 65.000  |
| JO-230      | 56.000  |
| PQ-180      | 108.000 |
| YC-30       | 21.000  |

## ■ IMPRESSION EN SORTIE

\*\* ABC ANALYSIS \*\*

```

*PROD. NO.*      (RANK)
SALES %         SUM %
-----
* PQ-180 *      ( A )
108000 28.4    108000 28.4
-----
* TR-300 *      ( A )
89000  23.4    197000 51.9
-----
* RS-15  *      ( B )
65000  17.1    262000 69.1
-----
* JO-230 *      ( B )
56000  14.7    318000 83.9
-----
* GZ-90  *      ( B )
23000   6.0    341000 89.9
-----
* YC-30  *      ( B )
21000   5.5    362000 95.5
-----
* BT-650 *      ( C )
17000   4.4    379000 100
-----

```

## ■ SEQUENCE D'UTILISATION DES TOUCHES

1) **R** **U** **N** **ENTER**

|                                   |
|-----------------------------------|
| <b>(1) PRODUCT NO. SALES</b><br>? |
|-----------------------------------|

2) TR-300 **ENTER**

[Entrée n° 1er produit]

|                                          |
|------------------------------------------|
| <b>(1) PRODUCT NO. SALES</b><br>TR-300 ? |
|------------------------------------------|

3) 88000 **ENTER** [Entrée produit des ventes 1er produit]

|                                              |                                   |
|----------------------------------------------|-----------------------------------|
| <b>(1) PRODUCT NO. SALES</b><br>TR-300 88000 | <b>(2) PRODUCT NO. SALES</b><br>? |
|----------------------------------------------|-----------------------------------|

Entrée comme ci-dessus

4) YC-30 **ENTER** [Entrée n° 7ème produit]

|                                               |                                         |
|-----------------------------------------------|-----------------------------------------|
| <b>(6) PRODUCT NO. SALES</b><br>PQ-180 108000 | <b>(7) PRODUCT NO. SALES</b><br>YC-30 ? |
|-----------------------------------------------|-----------------------------------------|

5) 21000 **ENTER** [Entrée produit des ventes 7ème produit]

|                                             |                                   |
|---------------------------------------------|-----------------------------------|
| <b>(7) PRODUCT NO. SALES</b><br>YC-30 21000 | <b>(8) PRODUCT NO. SALES</b><br>? |
|---------------------------------------------|-----------------------------------|

6) **ENTER** [Confirmation données choisies?]

|                                |
|--------------------------------|
| <b>CONFIRM OR AMEND (Y/N)_</b> |
|--------------------------------|

7) **Y** **ENTER**

[Confirmation données]

|                          |
|--------------------------|
| <b>(1) TR-300-&gt; ?</b> |
|--------------------------|

8) **ENTER**

|                                        |
|----------------------------------------|
| <b>(1) TR-300-&gt; ?</b><br>88000 -> ? |
|----------------------------------------|

9) 89000 **ENTER** [Correction données entrées]

|                                            |                          |
|--------------------------------------------|--------------------------|
| <b>(1) TR-300-&gt; ?</b><br>88000 -> 89000 | <b>(2) BT-650-&gt; ?</b> |
|--------------------------------------------|--------------------------|

Recommencez comme ci-dessus

10) **ENTER**

|                                         |                                        |
|-----------------------------------------|----------------------------------------|
| <b>(6) PQ-180-&gt; ?</b><br>108000 -> ? | <b>(7) YC-30 -&gt; ?</b><br>21000 -> ? |
|-----------------------------------------|----------------------------------------|

## ■ SEQUENCE D'UTILISATION DES TOUCHES

11)

[Reconfirmation des données?]

**CONFIRM OR AMEND (Y/N)\_**

16)

[Choisir à nouveau la méthode de sortie]

**LCD or PRINTER or END  
(L/P/E)?\_**

12)

[Pas de confirmation des données]

**LCD or PRINTER or END  
(L/P/E)?\_**

17)

[Impression de la sélection de la méthode de sortie]

**LCD or PRINTER or END  
(L/P/E)?\_**

13)

[Ecran de sélection de la méthode de sortie]

| PR.NO. | RATE(%) | SUM(%) | R |
|--------|---------|--------|---|
| PQ-180 | 28 ■■■■ | 28     | A |
| TR-300 | 23 ■■■■ | 51     | A |
| RS-15  | 17 ■■■■ | 69     | B |

18)

[Fin du programme]

>

14)

[Confirmation de l'écran par défilement vers le bas]

| PR. NO. | RATE(%) | SUM(%) | R |
|---------|---------|--------|---|
| TR-300  | 23 ■■■■ | 51     | A |
| RS-15   | 17 ■■■■ | 69     | B |
| JO-230  | 14 ■■■■ | 83     | B |

Recommencez comme ci-dessus

15)

| PR. NO. | RATE(%) | SUM(%) | R |
|---------|---------|--------|---|
| GZ-90   | 6 ■■■■  | 89     | B |
| YC-30   | 5 ■■■■  | 95     | B |
| BT-650  | 4 ■■■■  | 100    | C |

# ■ IMPRESSION EN SORTIE

```

40: CLEAR : CLS : WAIT 0
   :A=26:B=14: DIM B$(A
   )*B:C(A),D(A),K$(A)*
   1:Z$(0)*6
50: FOR I=1 TO A:
   CURSOR : IF I>2
   GOSUB 700: PRINT " "
60: CURSOR 0,(I)>1)*2:
   PRINT "<" STR$ I;" "
   PRODUCT NO. SALES":
   PRINT " "
70: BEEP 1: CURSOR 2+
   LEN STR$ I:(I=1)+(I
   )>1)*3: INPUT B$(I)
80: IF B$(I)="" LET C=I-
   1:I=A: GOTO 130
90: B$(I)= LEFT$( B$(I),
   7)
95: CURSOR 2+ LEN STR$ I
   ,(I=1)+(I)>1)*3:
   PRINT B$(I);" "
100: CURSOR 14+ LEN STR$
   I,(I=1)+(I)>1)*3:
   BEEP 1: INPUT Z$(0)
107: CURSOR 14+ LEN STR$
   I,(I=1)+(I)>1)*3:
   PRINT Z$(0);" "
110: C(I)= VAL Z$(0): IF
   C(I)<=0 BEEP 1:
   GOTO 100
120: E=E+C(I):C=I:Z$(0)="
   "
130: NEXT I: IF C<2 BEEP
   2: GOTO 40
140: BEEP 1: CLS : INPUT
   "CONFIRM OR AMEND(Y/
   N)":Z$
150: IF Z$="N" GOSUB 520:
   GOTO 240
160: IF Z$(">")"Y" THEN 140
170: CLS : FOR I=1 TO C
180: CURSOR : IF I>2
   GOSUB 700: PRINT " "
190: CURSOR 0,(I)>1)*2:
   PRINT "<" STR$ I;" "
   "B$(I);" -> " "
   CURSOR 6+ LEN STR$ I
   + LEN B$(I),(I)>1)*2
200: INPUT B$(I):B$(I)=
   LEFT$( B$(I),7)
210: CURSOR 2+ LEN STR$ I
   ,(I=1)+(I)>1)*3:
   PRINT STR$ C(I);" ->
   " : CURSOR LEN STR$
   C(I)+7,(I=1)+(I)>1)*3
   :Z$(0)= STR$ C(I)
220: INPUT Z$(0): IF VAL
   Z$(0)<=0 BEEP 1:
   GOTO 210
230: E=E-C(I):C(I)= VAL Z
   $(0):E=E+C(I): NEXT
   I: CLS : GOTO 140
240: CLS :H=0:V=0: PRINT
   "LCD or PRINTER or E
   ND"
250: BEEP 1: CURSOR 7,1:
   INPUT "(L/P/E)?":Z$
260: IF Z$="E" BEEP 3:
   CLS : END
270: IF Z$="P" THEN 430
280: IF Z$(">")"L" THEN 240
290: CLS : CURSOR 0,0:
   PRINT "PR. NO. RATE<
   %>": CURSOR 16,0:
   PRINT " SUM<%>R":Y=1
300: FOR I=1 TO 3-(C=2):
   CURSOR 0,I: GOSUB 70
   0:V=D(Y-1)
310: CURSOR 0,I: PRINT B$
   (Y): CURSOR 8,I:0$=
   STR$ ( INT (D(Y)-V))
315: IF LEN 0$=1 LET 0$="
   "0$
317: PRINT 0$
320: CURSOR (64+D(Y)/2)/6
   -.9,I: PRINT USING "
   #####" D(Y): CURSOR 2
   3,I: PRINT K$(Y)
330: LINE (66,I*8+1)-(66+
   (D(Y)-V)/2,I*8+5),BF
340: LINE (66+(D(Y)-V)/2,
   I*8+1)-(66+D(Y)/2,I*
   8+5),B
350: Y=Y+1: NEXT I
360: LINE (2,7)-(149,7):
   LINE (2,31)-(149,31)
   : LINE (2,0)-(2,31)
370: LINE (48,0)-(48,31):
   LINE (143,0)-(143,31
   ): LINE (149,0)-(149
   ,31)
380: IF INKEY$="" GOTO 3
   80
390: IF INKEY$="Q" GOTO
   240
400: IF INKEY$="D" AND Y
   <C+1 BEEP 1:Y=Y-2:
   GOTO 300
410: IF INKEY$(">")"U" OR Y
   <5 BEEP 2: GOTO 300
420: Y=Y-4: BEEP 1: GOTO
   300
430: LPRINT " ** ABC AN
   ALYSIS **": LPRINT
440: LPRINT " *PROD. NO.*
   (RANK)"
450: LPRINT " SALES %
   SUM %"
460: FOR I=1 TO C: GOSUB
   690:H=H+C(I)
470: LPRINT " * " USING "
   #####" B$(I):
   USING ";" * (
   " ;K$(I);" )
480: LPRINT USING "#####
   " ; STR$ C(I): USING
   "###.##" D(I)-V:
   USING "#####" H:
490: IF D(I)<>100 USING "
   ###.##" : GOTO 510
500: USING "#####"
510: LPRINT D(I):V=D(I):
   NEXT I: GOSUB 690:
   LPRINT : LPRINT :
   GOTO 240
520: L= INT (C/2+1):R=C
530: IF L>1 LET L=L-1:X=C
   (L):B$(0)=B$(L):
   GOTO 560
540: X=C(R):B$(0)=B$(R):C
   (R)=C(1):B$(R)=B$(1
   ):R=R-1
550: IF R<=1 LET C(1)=X:B
   $(1)=B$(0): GOTO 630
560: J=L
570: I=J:J=2*J
580: IF J>R THEN 620
590: IF J<R THEN IF C(J)>>
   C(J+1) LET J=J+1
600: IF X<C(J) THEN 620
610: C(I)=C(J):B$(I)=B$(J
   ): GOTO 570
620: C(I)=X:B$(I)=B$(0):
   GOTO 530
630: FOR I=1 TO C
640: H=H+C(I):D(I)=H/E*10
   0
650: IF D(I)<=66 OR I=1
   LET K$(I)="A": GOTO
   680
660: IF D(I)<=99 LET K$(I
   )="B": GOTO 680
670: K$(I)="C"
680: NEXT I: RETURN
690: USING : LPRINT "-----
   "
   : RETURN
700: PRINT " "
   :
   RETURN

```

2096 octets

## ■ CONTENU DE LA MEMOIRE

|        |                                    |
|--------|------------------------------------|
| A      | Nombre maximum de cases de données |
| B      | Longueur numéro du produit         |
| C      | Nombre de cases de données         |
| D      |                                    |
| E      | Produit global des ventes          |
| F      |                                    |
| G      |                                    |
| H      | ✓                                  |
| I      | Compteur de boucles                |
| J      | Compteur de boucles                |
| K      |                                    |
| L      | ✓                                  |
| M      |                                    |
| N      |                                    |
| O\$    | ✓                                  |
| P      |                                    |
| Q      |                                    |
| R      | Pointer de fin                     |
| S      |                                    |
| T      |                                    |
| U      |                                    |
| V      | ✓                                  |
| W      |                                    |
| X      | ✓                                  |
| Y      | ✓                                  |
| Z\$    | ✓                                  |
| C(A)   | Produit des ventes                 |
| D(A)   | ✓                                  |
| B\$(A) | Nom du client                      |
| K\$(A) | ✓                                  |
| Z\$(Ø) | ✓                                  |

**■ PRESENTATION**

Ce programme vous aide à ne pas faire de fautes. Bien que la méthode de calcul par elle même soit facile à comprendre, elle prend du temps en raison des erreurs qui surviennent dans les calculs. Ce programme a été conçu dans ce but. Il permet de rechercher les valeurs d'une matrice. La correction et l'impression des valeurs en entrée sont également possible.

**■ INSTRUCTIONS**

- Lancez le programme à l'aide des touches **DEF** **A**
  - Tout d'abord entrez les degrés de la matrices et entrez chaque donnée dans celle-ci.
  - Quand toutes les données ont été entrées, choisissez si vous voulez imprimer ou non les données d'entrée.
- Ensuite, choisissez si vous voulez corriger chacun des éléments d'entrée.
  - Pour pouvoir corriger chaque donnée d'entrée, appuyez sur **Y** **ENTER** , entrez le nombre de lignes et de colonnes des données à corriger et ensuite, entrez les données correctes.
  - Quand il n'y a pas de correction à faire appuyez sur **N** **ENTER** .
- Demandez la sortie des résultats des calculs et mettez fin au programme.
  - Lorsque vous avez terminé la correction des données, choisissez si vous désirez imprimer à nouveau les données d'entrée et au niveau **N** **ENTER** choisissez si vous voulez corriger chaque donnée d'entrée ou non.

**■ CONTENU**

Ce programme convertit la matrice en une matrice triangulaire à l'aide de la méthode d'élimination puis donne la réponse.

Soit une matrice:  $[a_{ij}]$  ( $i, j = 1 \sim n$ )

$$P = a_{mm} \quad (m = 2 \sim n)$$

$$q = a_{im}/P \quad (i = 1 \sim m - 1)$$

$$a_{ij} = a_{ij} - q \cdot a_{mj} \quad (j = 1 \sim m)$$

Après calcul, voici ce qui est obtenu:

$$a_{ij} = \emptyset \text{ for } i < j$$

Ceci abouti à:  $\det = a_{11} \cdot a_{22} \cdot a_{33} \cdot \dots \cdot a_{nn}$

Cependant si  $P = \emptyset$ , vous obtiendrez une erreur puisqu'il est impossible d'effectuer de calcul.

## ■ EXEMPLE

Recherche du point de détermination suivant

$$\begin{bmatrix} 4 & 7 & 1 & 8 \\ 5 & 3 & 9 & 3 \\ 2 & 5 & 3 & 5 \\ 1 & 4 & 7 & 8 \end{bmatrix}$$

## ■ RESULTAT IMPRIME

$\hat{A}(1,1)=4.$

$\hat{A}(1,2)=7.$

$\hat{A}(1,3)=1.$

$\hat{A}(1,4)=8.$

$\hat{A}(2,1)=5.$

$\hat{A}(2,2)=3.$

$\hat{A}(2,3)=9.$

$\hat{A}(2,4)=3.$

$\hat{A}(3,1)=2.$

$\hat{A}(3,2)=5.$

$\hat{A}(3,3)=3.$

$\hat{A}(3,4)=5.$

$\hat{A}(4,1)=1.$

$\hat{A}(4,2)=4.$

$\hat{A}(4,3)=7.$

$\hat{A}(4,4)=8.$

$\det = -525.$

## ■ SEQUENCE D'UTILISATION DES TOUCHES

1) **DEF** **A** [Début du programme]

```
DEGREES = _
```

2) 4 **ENTER** [Entrée des degrés]

```
DEGREES = 4  
A(1,1) = ?
```

3) 4 **ENTER** [Entrée de l'élément]

```
DEGREES = 4  
A(1,1) = 4  
A(1,2) = ?
```

4) 6 **ENTER**

```
DEGREES = 4  
A(1,1) = 4  
A(1,2) = 6  
A(1,3) = ?
```

·  
·  
·

Recommencer comme ci-dessus

·  
·  
·

5) 8 **ENTER**

```
A(4,2) = 4  
A(4,3) = 7  
A(4,4) = 8  
PRINT OUT DATA (Y/N) _
```

6) **N** **ENTER**

```
A(4,3) = 7  
A(4,4) = 8  
PRINT OUT DATA (Y/N) N  
AMEND DATA (Y/N) _
```

7) **Y** **ENTER**

[Correction des données]

```
X = ?                    Y =
```

8) 1 **ENTER**

[Entrée du nombre de colonnes]

```
X = 1                    Y = ?
```

9) 2 **ENTER**

[Entrée du nombre de lignes]

```
X = 1                    Y = 2  
A(1,2) = 6.  
A(1,2) = ?
```

10) 7 **ENTER**

[Correction des données entrées]

```
X = 1                    Y = 2  
A(1,2) = 6.  
A(1,2) = 7  
AMEND DATA (Y/N) _
```

11) **N** **ENTER**

```
A(1,2) = 6.  
A(1,2) = 7  
AMEND DATA (Y/N) N  
PRINT OUT DATA (Y/N) _
```

12) **Y** **ENTER** [Impression de la  
liste des données après correction]

```
A(1,2) = 7  
AMEND DATA (Y/N) N  
PRINT OUT DATA (Y/N) Y  
AMEND DATA (Y/N) _
```

13) **N** **ENTER** [Sortie des résultats  
et fin du programme]

```
>
```



## ■ LISTE DU PROGRAMME

```

10:REM *****
20:REM * MATRIX CALC. *
30:REM *****
40:"A": CLEAR : CLS :
   WAIT 0
50:INPUT "DEGREES=":IN
60:N=N-1
70:DIM A(N,N)
80:GOSUB 230
90:F=0
100:INPUT "PRINT OUT DAT
   A (Y/N)":0$
110:IF 0$="N" THEN 140
120:IF 0$(">")"Y" THEN 90
130:GOSUB 350
140:INPUT "AMEND DATA (Y
   /N)":0$
150:IF (0$="N")*(F=0)
   THEN 200
160:IF (0$="N")*(F=1)
   THEN 90
170:IF 0$(">")"Y" THEN
   CURSOR 0,3: GOTO 140
180:GOSUB 430
190:GOTO 140
200:GOSUB 580
210:CLS
220:END
230:L=1
240:FOR I=0 TO N
250:FOR J=0 TO N
260:AA$="A(" + STR$ (I+1)
   +"," + STR$ (J+1)+")="
   "
270:PRINT AA$:
280:CURSOR LEN (AA$),L
290:INPUT A(I,J)
300:L=L+1: IF L=4 THEN
   LET L=3
310:CURSOR
320:NEXT J
330:NEXT I
340:RETURN
350:FOR I=0 TO N
360:FOR J=0 TO N
370:AA$="A(" + STR$ (I+1)
   +"," + STR$ (J+1)+")="
   "
380:LPRINT AA$:A(I,J)
390:NEXT J
400:NEXT I
410:LPRINT : LPRINT
420:RETURN
430:CLS
440:PRINT "X=      Y="
450:CURSOR 2,0: INPUT X
460:IF (X<1)+(X>N+1)
   THEN 450
470:CURSOR 10,0: INPUT Y
480:IF (Y<1)+(Y>N+1)
   THEN 470
490:CURSOR
500:AA$="A(" + STR$ (X)+
   "," + STR$ (Y)+")="
510:PRINT AA$:A(X-1,Y-1)
520:PRINT AA$
530:CURSOR LEN (AA$),2
540:INPUT A(X-1,Y-1)
550:CURSOR
560:F=1
570:RETURN
580:FOR M=N TO 1 STEP -1
590:P=A(M,M)
600:IF P=0 THEN 760
610:FOR I=0 TO M-1
620:Q=A(I,M)/P
630:FOR J=0 TO M
640:A(I,J)=A(I,J)-Q*A(M,
   J)
650:NEXT J
660:NEXT I
670:NEXT M
680:D=A(0,0)
690:FOR I=1 TO N
700:D=D*A(I,I)
710:NEXT I
720:BEEP 1
730:LPRINT "det=" ;D
740:LPRINT : LPRINT
750:RETURN
760:BEEP 2: CLS
770:PRINT "CALCULATE IMP
   POSSIBLE !!"
780:END

```

985 octets

## ■ CONTENU DE LA MEMOIRE

|         |                                              |
|---------|----------------------------------------------|
| A       |                                              |
| B       |                                              |
| C       |                                              |
| D       | Matrice                                      |
| E       |                                              |
| F       | Indicateur                                   |
| G       |                                              |
| H       |                                              |
| I       | Compte                                       |
| J       | Compte                                       |
| K       |                                              |
| L       | ✓                                            |
| M       | Compte                                       |
| N       | Degrés                                       |
| O\$     | Indicateur                                   |
| P       | ✓                                            |
| Q       | ✓                                            |
| R       |                                              |
| S       |                                              |
| T       |                                              |
| U       |                                              |
| V       |                                              |
| W       |                                              |
| X       | Modification dans les éléments de la matrice |
| Y       | Modification dans les éléments de la matrice |
| Z       |                                              |
| AAS     | Ecran                                        |
| A(N, N) | éléments de la matrice                       |

## ■ PRESENTATION

Même les modifications difficiles à lire peuvent être perçues d'un seul coup d'oeil! Ce programme permet de rechercher la moyenne entre des éléments fixes des données et de plus de poursuivre pour connaître les tendances d'ensemble de ces données.

Autrement dit, il est possible de saisir avec exactitude les tendances à long terme d'éléments tels que les statistiques de vente, les prix des stocks et le taux d'échange qui sont soumis à des modifications brusques de jour en jour.

## ■ INSTRUCTIONS

1. Lancez le programme en appuyant sur **DEF** **A** . Entrez le nombre d'éléments et poursuivez en entrant chaque donnée. Lorsque vous entrez ces données, elles seront imprimées et, également, les données d'entrée et les valeurs moyennes seront imprimées à partir de  $n+1$ .
2. Une fois le programme terminé, appuyez sur **ENTER** pour redonner le contrôle à l'ordinateur qui attend des données d'entrée.

## ■ CONTENU

Le traitement varie suivant que le nombre d'éléments en moyenne ( $n$ ) est pair ou impair.

1.  $n$  est un nombre impair

$$\bar{X}_1 = \sum_{i=1}^n X_i/n$$

$$\bar{X}_2 = \sum_{i=2}^{n+1} X_i/n$$

⋮

2.  $n$  est un nombre pair

$$\bar{X}_1 = \left( \frac{X_1}{2} + \frac{X_{n+1}}{2} + \sum_{i=2}^n X_i \right) / n$$

$$\bar{X}_2 = \left( \frac{X_2}{2} + \frac{X_{n+2}}{2} + \sum_{i=3}^{n+1} X_i \right) / n$$

⋮

## ■ EXEMPLE

Entrez les données suivantes et recherchez la moyenne mouvante modifiée. Fixez le nombre d'éléments à 4.

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_9$ | $X_{10}$ | $X_{11}$ | $X_{12}$ | $X_{13}$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|----------|----------|----------|
| 82    | 15    | 8     | 53    | 0     | 41    | 22    | 3     | 18    | 75       | 64       | 39       | 46       |

## ■ RESULTATS IMPRIMES

```
* X(1)=82.  
* X(2)=15.  
* X(3)=8.  
* X(4)=53.  
* X(5)=0.  
Average=29.25  
* X(6)=41.  
Average=22.25  
* X(7)=22.  
Average=27.25  
* X(8)=3.  
Average=22.75  
* X(9)=18.  
Average=18.75  
* X(10)=75.  
Average=25.25  
* X(11)=64.  
Average=34.75  
* X(12)=39.  
Average=44.5  
* X(13)=46.  
Average=52.5
```

## ■ SEQUENCE D'UTILISATION DES TOUCHES

1) **DEF** **A** [Début du programme]

**NUMBER OF ITEMS=** \_

2) 4 **ENTER**

[Nombre d'éléments entrés]

**NUMBER OF ITEMS=4**  
**X(1)=?**

3) 82 **ENTER** [Entrée de X(1)]

**NUMBER OF ITEMS=4**  
**X(1)=82**  
**X(2)=?**

.  
. .  
. . .  
. . . .  
. . . . .

Répéter l'opération ci-dessus

4) 46 **ENTER** [Entrée de X(13)]

**X(11)=64**  
**X(12)=39**  
**X(13)=46**  
**X(14)=?**

5) **ENTER** [Fin du programme]

>

## ■ LISTE DU PROGRAMME

```

10:REM *****
20:REM MODIFIED MOVING
25:REM AVERAGE
30:REM *****
40:"A": CLEAR : CLS :
    WAIT 0
50:N=1:E=0:C=1
60:INPUT "NUMBER OF ITE
    MS=";A
70:DIM X(A-1)
80:IF A<> INT (A/2)*2
    THEN 220
90:FOR I=0 TO A-1
100:GOSUB 430
110:NEXT I
120:FOR I=0 TO A-1
130:PRINT "X(" + STR$ (N)
    +")=";
140:CURSOR 5+ LEN (
    STR$ (N)),C: INPUT D
    : GOTO 160
150:LPRINT : LPRINT :
    CLS : END
160:E=E+D
170:LPRINT "* X(" + STR$
    (N)+")=";D
180:LPRINT " Average=";(
    E-.5*(D+X(I)))/A
190:E=E-X(I):X(I)=D:N=N+
    1:C=C+1: IF C=4 LET
    C=3
200:NEXT I
210:GOTO 120
220:FOR I=0 TO A-2
230:GOSUB 430
240:NEXT I
250:B=A-1
260:PRINT "X(" + STR$ (N)
    +")=";
270:CURSOR 5+ LEN (
    STR$ (N)),C
280:INPUT X(B)
290:LPRINT "* X(" + STR$
    (N)+")=";X(B)
300:LPRINT " Average=";
    E/A
310:E=E+X(B):N=N+1:C=C+1
    : IF C=4 LET C=3
320:FOR I=0 TO B
330:PRINT "X(" + STR$ (N)
    +")="
340:CURSOR 5+ LEN (
    STR$ (N)),C
350:INPUT D: GOTO 370
360:LPRINT : LPRINT :
    CLS : END
370:E=E-X(I)+D:X(I)=D
380:LPRINT "* X(" + STR$
    (N)+")=";D
390:LPRINT " Average=";
    E/A
400:N=N+1:C=C+1: IF C=4
    LET C=3
410:NEXT I
420:GOTO 320
430:PRINT "X(" + STR$ (N)
    +")=";
440:CURSOR 5+ LEN (
    STR$ (N)),C
450:INPUT D
460:LPRINT "* X(" + STR$
    (N)+")=";D
470:E=E+D:X(I)=D:N=N+1:C
    =C+1: IF C=4 THEN
    LET C=3
480:RETURN

813 octets

```

## ■ CONTENU DE LA MEMOIRE

|        |                     |
|--------|---------------------|
| A      | Nombre d'éléments   |
| B      | ✓                   |
| C      | ✓                   |
| D      | Données             |
| E      | Total des données   |
| F      |                     |
| G      |                     |
| H      |                     |
| I      | Compteur de boucles |
| J      |                     |
| K      |                     |
| L      |                     |
| M      |                     |
| N      | ✓                   |
| O      |                     |
| P      |                     |
| Q      |                     |
| R      |                     |
| S      |                     |
| T      |                     |
| U      |                     |
| V      |                     |
| W      |                     |
| X      |                     |
| Y      |                     |
| Z      |                     |
| X(A-1) | Données             |

## ■ PRESENTATION

Ce programme apporte une aide pour l'analyse des fractions. Ce programme permettra de supprimer les ennuis qui accompagnent le nombre de niveau, le nombre de reproduction et pour chaque donnée il calculera la somme carré (S), le degré de liberté (DF) une estimation impartiale de la variance de la population (V) et une estimation impartiale du taux de variance de la population (F).

## ■ INSTRUCTION

- Lancez le programme en appuyant sur **DEF** **A** .
  - Choisissez la méthode de sortie des résultats calculés (imprimante: 1 **ENTER** , écran: 2 **ENTER** ).
  - Entrez le nombre de niveaux et le nombre de reproductions. Pour poursuivre, entrez les données en fonction de ce qui est indiqué à l'écran.
  - Une fois toutes les données entrées, choisissez d'imprimer ou non ces données.
- Choisissez ensuite si vous voulez corriger ou non vos données d'entrée.
  - Pour les corriger appuyez sur **Y** **ENTER** et indiquez le nombre de lignes et de colonnes des données à corriger et ensuite entrez les données correctes.
  - Lorsqu'il n'y a pas besoin de corriger les données, appuyez sur **N** **ENTER** .
- Demandez les résultats des calculs et mettez fin au programme.
  - Lorsque la correction des données a été effectuée, choisissez si vous voulez ou non imprimer les données en entrée à nouveau et au niveau de **N** **ENTER** choisissez si vous voulez à nouveau corriger ces données d'entrée.

## ■ CONTENU

Nbre de niveaux:  $a$

Nbre de reproductions:  $n$

Données:  $X_{ij}$  ( $i = 1 \sim a, j = 1 \sim n$ )      Nbre de données:  $an$

- $[X] = x^2/an$       2.  $S_A = [A] - [X]$       3.  $\phi_A = a - 1$   
 $[A] = x^2/n$        $S_T = [AS] - [X]$        $\phi_E = a_n - a$   
 $[AS] = \sum x^2_{ij}$        $S_E = [AS] - [A]$        $\phi_T = a_n - 1$
  - $\{V\} = \{S\} / \{\phi\}$
  - $F = V_A / V_E$
- $V_T$  n'est pas calculé.



## ■ EXEMPLE

Dans le tableau ci-dessous, de l'engrais avait été donné pour planter des boutures à trois temps différents et leur croissance avait été mesurée après une semaine. A partir de ces résultats, il faut chercher les sommes carrées, le degré de liberté, la variance et le taux de variance sans partie pris.

|                 |    | Growth after 1 week [cm] |      |      |      |      |
|-----------------|----|--------------------------|------|------|------|------|
|                 |    | 10                       | 20   | 30   | 40   | 50   |
| Manure [g/sold] | 10 | 7,80                     | 7,64 | 7,49 | 7,72 | 7,78 |
|                 | 20 | 8,02                     | 8,28 | 8,16 | 8,36 | 8,36 |
|                 | 30 | 8,52                     | 8,41 | 8,49 | 8,37 | 8,37 |

## ■ RESULTATS IMPRIMES

Factor value=

3.

\*Square sums

a=1.596

Replication value=

5.

e=0.15956

t=1.75556

X(1,1)= 7.8

X(1,2)= 7.64

X(1,3)= 7.49

X(1,4)= 7.72

X(1,5)= 7.78

\*Degree of freedom

a=2.

e=12.

t=14.

X(2,1)= 8.02

X(2,2)= 8.28

X(2,3)= 8.16

X(2,4)= 8.31

X(2,5)= 8.36

\*Unbiased variance

a=0.798

e=1.329666667E-02

X(3,1)= 8.52

X(3,2)= 8.41

X(3,3)= 8.54

X(3,4)= 8.49

X(3,5)= 8.37

\*Unbiased variance ratio

a=60.01504135

## ■ SEQUENCE D'UTILISATION DES TOUCHES

1) **DEF** **A** [Début du programme]

**\*SINGLE ARRANGE. METHOD \***

**PRINTER ON=1 OFF=2\_**

2) 1 **ENTER**

[Selection de la sortie sur imprimante]

**\*SINGLE ARRANGE. METHOD \***

**NO. OF FACTORS= \_**

3) 3 **ENTER** 5 **ENTER**

[Entrer ?? Degrés]

**\*SINGLE ARRANGE. METHOD \***

**NO. OF FACTORS=3**

**NO. OF REPLICATIONS=5**

**X(1,1)=?**

4) 7.8 **ENTER** 7.64 **ENTER**

7.58 **ENTER**

[Entrée de chacune des données]

**X(1,1)=7.8**

**X(1,2)=7.64**

**X(1,3)=7.58**

**X(1,4)=?**

recommencer comme ci-dessus

5) 8.54 **ENTER** 8.49 **ENTER**

8.37 **ENTER**

[Entrée de chacune des données]

**X(3,3)=8.54**

**X(3,4)=8.49**

**X(3,5)=8.37**

**PRINT OUT DATA(Y/N)\_**

6) **N** **ENTER**

**X(3,4)=8.49**

**X(3,5)=8.37**

**PRINT OUT DATA(Y/N)N**

**AMEND DATA(Y/N)\_**

7) **Y** **ENTER**

[Correction des données]

**X=?**

**Y=?**

8) 1 **ENTER**

[Entrée du nombre de colonne]

**X=1**

**Y=?**

9) 3 **ENTER**

[Entrée du nombre de lignes]

**X=1**

**Y=3**

**X(1,3)=7.58**

**X(1,3)=?**

10) 7.49 **ENTER**

[Correction des données entrées]

**X=1**

**Y=3**

**X(1,3)=7.58**

**X(1,3)=7.49**

**AMEND DATA(Y/N)\_**

11) **N** **ENTER**

**X(1,3)=7.58**

**X(1,3)=7.49**

**AMEND DATA(Y/N)N**

**PRINT OUT DATA(Y/N)\_**

12)  Y  ENTER

[Impression de la liste des données  
après correction]

```
X(1,3)=7.49  
AMEND DATA(Y/N)N  
PRINT OUT DATA(Y/N)Y  
AMEND DATA(Y/N)_
```

13)  N  ENTER

[Résultats et fin du programme]

```
>
```

## ■ LISTE DU PROGRAMME

```

5:REM *****
10:REM SINGLE ARRANGE.
20:REM METHOD
30:REM *****
40:"A": CLEAR : WAIT 0:
CLS
50:BEEP 1
60:CURSOR 0,0: PRINT "*
SINGLE ARRANGE. METH
OD*"
70:CURSOR 2,2: INPUT "P
RINTER ON=1 OFF=2 ";
P
80:IF (P<>1)*(P<>2)
THEN 70
90:CURSOR 0,2: PRINT "
"
100:CURSOR 0,1: INPUT "N
O. OF FACTORS=";A
110:CURSOR 0,2: INPUT "N
O. OF REPLICATIONS="
IN
120:CURSOR
130:DIM D(A,N)
140:FOR I=1 TO A
150:FOR J=1 TO N
160:BB$="X(" + STR$(I)+
"," + STR$(J)+")="
170:CURSOR : PRINT BB$
180:CURSOR LEN (BB$),3
190:INPUT D(I,J)
200:NEXT J
210:NEXT I
220:IF (P=2)*(F=0) THEN
370
230:IF (P=2)*(F=1) THEN
520
240:CURSOR :F=0: INPUT "
PRINT OUT DATA(Y/N)"
;0$
250:IF 0$="N" THEN 370
260:IF 0$<>"Y" THEN 240
270:LPRINT : LPRINT "Fac
tor value=": LPRINT
A
280:LPRINT "Replication
value=": LPRINT N:
LPRINT
290:FOR I=1 TO A
300:FOR J=1 TO N
310:BB$="X(" + STR$(I)+
"," + STR$(J)+")="
320:LPRINT BB$;D(I,J)
330:NEXT J
340:LPRINT
350:NEXT I
360:LPRINT
370:CURSOR : INPUT "AMEN
D DATA(Y/N)";0$
380:IF (0$="N")*(F=0)
THEN 520
390:IF (0$="N")*(F=1)
THEN 220
400:IF 0$<>"Y" THEN 370
410:CLS :F=1
420:PRINT "X= Y="
430:CURSOR 3,0: INPUT X
440:IF (X<1)+(X>A) THEN
430
450:CURSOR 13,0: INPUT Y
: CURSOR
460:IF (Y<1)+(Y>N) THEN
450
470:BB$="X(" + STR$(X)+
"," + STR$(Y)+")="
480:PRINT BB$;D(X,Y)
490:PRINT BB$
500:CURSOR LEN (BB$),2:
INPUT D(X,Y):
CURSOR
510:GOTO 370
520:FOR I=1 TO A
530:E=0
540:FOR J=1 TO N
550:E=E+D(I,J)
560:Z=Z+D(I,J)*D(I,J)
570:NEXT J
580:S=S+E:T=T+E*E
590:NEXT I
600:S=S*S/(A*N):T=T/N:T=
T-S:Z=Z-S:L=Z-T:F=A-
1
610:U=T/F:O=A*(N-1):M=L/
O:Q=A*N-1:G=U/M
620:ON P GOSUB 640,720
630:CLS : END
640:LPRINT "*Square sums
"
650:LPRINT "a=";T:
LPRINT "e=";L:
LPRINT "t=";Z:
LPRINT
660:LPRINT "*Degree of f
reedom"
670:LPRINT "a=";F:
LPRINT "e=";O:
LPRINT "t=";Q:
LPRINT
680:LPRINT "*Unbiased va
riance"
690:LPRINT "a=";U:
LPRINT "e=";M:
LPRINT
700:LPRINT "*Unbiased va
riance ratio"
710:LPRINT "a=";G:
LPRINT : LPRINT :
RETURN
720:CLS : PRINT "*Square
sums"
730:PRINT "a=";T: PRINT
"e=";L: WAIT :
PRINT "t=";Z: WAIT
740:WAIT 0: CLS : PRINT
"*Degree of freedom"
750:PRINT "a=";F: PRINT
"e=";O: WAIT :
PRINT "t=";Q
760:WAIT 0: CLS : PRINT
"*Unbiased variance"
770:PRINT "a=";U: WAIT :
PRINT "e=";M
780:WAIT 0: CLS : PRINT
"*Unbiased variance
ratio"
790:WAIT : PRINT "a=";G
800:CLS : RETURN
1533 octets

```

## ■ CONTENU DE LA MEMOIRE

|         |                                 |
|---------|---------------------------------|
| A       | Nombre de facteurs              |
| B       |                                 |
| C       |                                 |
| D       |                                 |
| E       | Somme des données de la colonne |
| F       | Indicateur degré de liberté     |
| G       | ✓                               |
| H       |                                 |
| I       | Compteur de boucles             |
| J       | Compteur de boucles             |
| K       |                                 |
| L       | ✓                               |
| M       | Variance                        |
| N       | Nombre de reproductions         |
| O, O\$  | ✓, ✓                            |
| P       | ✓                               |
| Q       | ✓                               |
| R       |                                 |
| S       | Somme des données               |
| T       | ✓                               |
| U       | Variance                        |
| V       |                                 |
| W       |                                 |
| X       | Correction                      |
| Y       | Correction                      |
| Z       | ✓                               |
| BB\$    | Ecran                           |
| D(A, N) | Données                         |

## ■ PRESENTATION

Ce programme vous aidera à exécuter le test F.

Ce test est très utile lorsque vous voulez savoir s'il y a ou non une relation entre les conditions et les résultats d'expériences. Ce programme vous permet d'exécuter les calculs ennuyeux mais indispensables pour ce test. Lorsque vous entrez vos chiffres et les déviations standard ou les chiffres et les données, la valeur du test et le degré de liberté s'impriment.

## ■ INSTRUCTIONS

Lancez le programme avec **DEF** **A** lorsque vous connaissez les variables standard.

1. Entrez les chiffres du groupe 1, les variables standard, les chiffres du groupe 2 et les variables standard au fur et à mesure qu'ils vous sont demandés à l'écran.
2. Une fois les données entrées, celles-ci seront imprimées en sortie avec la valeur du test F et le programme prendra fin.

Lancez le programme avec **DEF** **B** lorsque vous ne connaissez pas les variables standards.

1. Entrez les chiffres du groupe 1 et chaque donnée, puis les chiffres du groupe 2 et chaque donnée au fur et à mesure qu'ils vous sont demandés à l'écran.
2. Une fois les données entrées, celles-ci seront imprimées en sortie avec la moyenne, la variable standard la valeur du test F et le degré de liberté, puis le programme prendra fin.

## ■ CONTENU

- Calcul des sommes carrées

$$\text{Somme du carré entre des classes} = \sum n_i (\bar{X}_i - \bar{X})^2$$

$$\text{Somme du carré à l'intérieur d'une classe} = \sum \sum (X_{ij} - \bar{X}_i)^2$$

$$\text{Total} = \sum \sum (X_{ij} - \bar{X})^2$$

$\bar{X}_i$  : Moyenne à l'intérieur d'un groupe

$\bar{X}$  : Moyenne totale des données

- Calcul du degré de liberté

$$\text{Degré de liberté entre des classes} = \text{Nbre de groupes} - 1$$

$$\text{Degré de liberté à l'intérieur d'une classe} = \text{Total des données} - \text{Degré de liberté entre les classes} - 1$$

Total = Total des données – 1

- Calcul du carré de la moyenne

Carré de la moyenne entre des classes = Somme du carré entre les classes  
Degrés de liberté entre les classes

Carré de la moyenne à l'intérieur d'une classe = Somme des carrés à l'intérieur d'une classe  
Degrés de liberté à l'intérieur d'une classe

- Calcul de F (taux de variance)

F = Carré de la moyenne entre les classes / Carré de la moyenne à l'intérieur d'une classe.

## ■ EXEMPLES

1. Entrez le tableau suivant et recherchez la valeur du test F et le degré de liberté.

|             |                                |
|-------------|--------------------------------|
| groupe<br>1 | nombre de données: 10          |
|             | variable standard: 0,326114176 |
| groupe<br>2 | nombre de données: 8           |
|             | variable standard: 0,356452736 |

2. Entrez les données suivantes et recherchez la moyenne, la variable standard, la valeur du test F et le degré de liberté dans chaque groupe.

|             |     |     |     |    |    |    |     |
|-------------|-----|-----|-----|----|----|----|-----|
|             | 1   | 2   | 3   | 4  | 5  | 6  | 7   |
| groupe<br>1 | 70  | 67  | 82  | 89 | 93 | 76 | 81  |
| groupe<br>2 | 110 | 108 | 101 | 98 | 99 | 96 | 103 |

## ■ RESULTATS IMPRIMES

### [Exemple 1]

\*\*\*\*\* F TEST \*\*\*\*\*

Data numbers 1= 10.

Std. deviation 1= 0.326114176

Data numbers 2= 8.

Std. deviation 2= 0.356452736

-----  
F Test value= 1.194715641

Degree of freedom 1= 7.

Degree of freedom 2= 9.

### [Exemple 2]

\*\*\*\*\* F TEST \*\*\*\*\*

Data numbers 1= 7.

Data 1(1)=70.

Data 1(2)=67.

Data 1(3)=82.

Data 1(4)=89.

Data 1(5)=93.

Data 1(6)=76.

Data 1(7)=81.

Average 1= 79.71428571

Std. deviation 1= 9.481812206

Data numbers 2= 7.

Data 2(1)=110.

Data 2(2)=103.

Data 2(3)=101.

Data 2(4)=98.

Data 2(5)=99.

Data 2(6)=96.

Data 2(7)=103.

Average 2= 102.1428571

Std. deviation 2= 5.209881704

-----  
F Test value= 3.312279483

Degree of freedom 1= 6.

Degree of freedom 2= 6.



## ■ SEQUENCE D'UTILISATION DES TOUCHES

### DANS LE CAS DE L'EXEMPLE 1

1) **DEF** **A** [Début du programme]

**DATA NUMBERS1 = \_**

2) 10 **ENTER**

[Entrée des chiffres du groupe 1]

**DATA NUMBERS1 = 10**  
**STD. DEVIATION1 = \_**

3) .326114176 **ENTER**

[Entrée de la variable standard du groupe 1]

**DATA NUMBERS1 = 10**  
**STD. DEVIATION1 = .32114176**  
**DATA NUMBERS2 = \_**

4) 8 **ENTER**

[Entrée des chiffres du groupe 2]

**DATA NUMBERS1 = 10**  
**STD. DEVIATION1 = .326114176**  
**DATA NUMBERS2 = 8**  
**STD. DEVIATION2 = \_**

5) .356452736 **ENTER**

[Entrée de la variable standard du groupe 2]

**STD.DEVIATION1 = .326114176**  
**DATA NUMBERS2 = 8**  
**STD.DEVIATION2 = .356452736**  
>

(Contrôle des résultats en sortie terminé)

### DANS LE CAS DE L'EXEMPLE 2

1) **DEF** **B** [Début du programme]

**DATA NUMBERS1 = \_**

2) 7 **ENTER**

[Entrée des chiffres du groupe 1]

**DATA NUMBERS1 = 7**  
**DATA 1(1) = ?**

3) 70 **ENTER**

[Donnée 1 du groupe 1]

**DATA NUMBERS1 = 7**  
**DATA 1(1) = 70**  
**DATA 1(2) = ?**

⋮  
Continuer comme ci-dessus  
⋮

4) 81 **ENTER**

[Donnée 2 du groupe 1]

**DATA NUMBERS2 = \_**

5) 7 **ENTER**

[Entrée des chiffres du groupe 2]

**DATA NUMBERS2 = 7**  
**DATA 2(1) = ?**

⋮  
Continuer comme ci-dessus  
⋮

6) 96 **ENTER**

[Donnée 6 du groupe 2]

**DATA 2(4) = 98**  
**DATA 2(5) = 99**  
**DATA 2(6) = 96**  
**DATA 2(7) = ?**

7) 103 **ENTER**

[Donnée 7 du groupe 2]

**DATA 2(5) = 99**  
**DATA 2(6) = 96**  
**DATA 2(7) = 103**  
>

(Contrôle de résultats de sorti terminé)

## ■ LISTE DU PROGRAMME

```

10:REM *****
20:REM *   F TEST   *
30:REM *****
40:"A": CLEAR : CLS :
   WAIT 0
50:INPUT "DATA NUMBERS
   1=";M
60:INPUT "STD. DEVIATIO
   N 1=";R
70:INPUT "DATA NUMBERS
   2=";N
80:INPUT "STD. DEVIATIO
   N 2=";S
90:GOTO 330
100:"B": CLEAR : CLS :
   WAIT 0:F=1
110:INPUT "DATA NUMBERS
   1=";M
120:DIM X(M)
130:PRINT ""
140:L=2
150:FOR I=1 TO M
160:CURSOR : PRINT "DATA
   1(" + STR$(I) + ")=";
170:CURSOR 9+ LEN (
   STR$(I));L: INPUT X
   (I)
180:X=X+X(I):R=R+X(I)*X(
   I)
190:L=L+1: IF L=4 LET L=
   3
200:NEXT I
210:X=X/M:R=R-M*X*X:R=
   SQR (R/(M-1))
220:CLS
230:INPUT "DATA NUMBERS
   2=";N
240:DIM Y(N)
250:PRINT "":L=2
260:FOR I=1 TO N
270:CURSOR : PRINT "DATA
   2(" + STR$(I) + ")=";
280:CURSOR 9+ LEN (
   STR$(I));L: INPUT Y
   (I)
290:Y=Y+Y(I):S=S+Y(I)*Y(
   I)
300:L=L+1: IF L=4 LET L=
   3
310:NEXT I
320:Y=Y/N:S=S-N*Y*Y:S=
   SQR (S/(N-1))
330:T=R*R:U=S+S:O=M:Q=N
340:IF U>T LET P=M:M=N:N=
   =P:P=U:U=T:T=P
350:LPRINT "***** F T
   EST *****"
360:LPRINT
370:LPRINT "Data numbers
   1=": LPRINT 0
380:IF F<>1 THEN GOTO 44
   0
390:FOR I=1 TO 0
400:LPRINT "Data 1(" +
   STR$(I) + ")=";X(I)
410:NEXT I
420:LPRINT
430:LPRINT "Average 1=
   ": LPRINT X
440:LPRINT "Std. deviati
   on 1=": LPRINT R
450:LPRINT
460:LPRINT "Data numbers
   2=": LPRINT 0
470:IF F<>1 THEN GOTO 53
   0
480:FOR I=1 TO 0
490:LPRINT "Data 2(" +
   STR$(I) + ")=";Y(I)
500:NEXT I
510:LPRINT
520:LPRINT "Average 2=
   ": LPRINT Y
530:LPRINT "Std. deviati
   on 2=": LPRINT S
540:LPRINT : LPRINT "----
   -----
   -: LPRINT
550:LPRINT "F Test value
   =" : LPRINT T/U
560:LPRINT
570:LPRINT "Degree of fr
   eedom 1=": LPRINT M-
   1
580:LPRINT "Degree of fr
   eedom 2=": LPRINT N-
   1
590:LPRINT : LPRINT
600:END

```

1094 octets

## ■ CONTENU DE LA MEMOIRE

|      |                       |
|------|-----------------------|
| A    |                       |
| B    |                       |
| C    |                       |
| D    |                       |
| E    |                       |
| F    | ✓                     |
| G    |                       |
| H    |                       |
| I    | Compteur de boucles   |
| J    |                       |
| K    |                       |
| L    | ✓                     |
| M    | Chiffres données 1    |
| N    | Chiffres de données 2 |
| O    | ✓                     |
| P    | ✓                     |
| Q    | ✓                     |
| R    | Variable standard 1   |
| S    | Variable standard 2   |
| T    | Variance 1            |
| U    | Variance 2            |
| V    |                       |
| W    |                       |
| X    | Somme donnée 1, carré |
| Y    | Somme donnée 2, carré |
| Z    |                       |
| X(M) | Données 1             |
| Y(N) | Données 2             |



**SHARP CORPORATION**

**OSAKA, JAPAN**

© 1984 SHARP CORPORATION  
Printed in Japan/Imprimé au Japon

5B1T(TINSF4220CCZZ)③