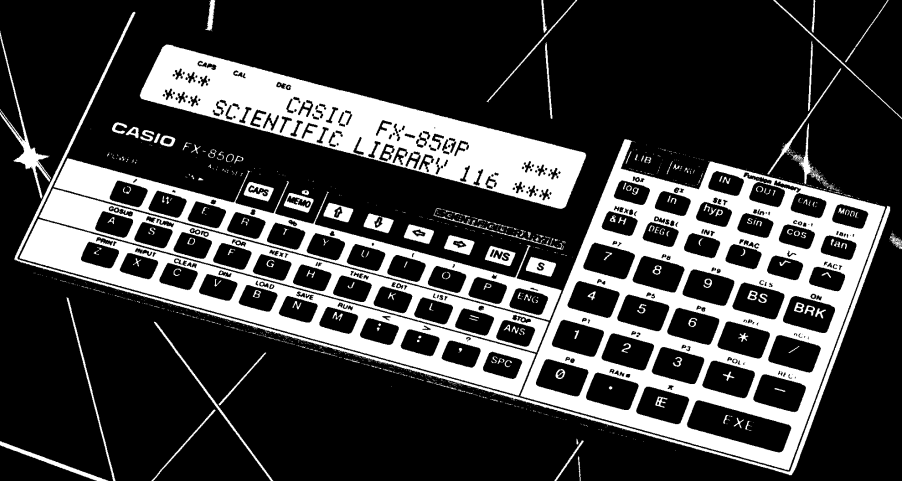
CASIO

MANUEL DE L'UTILISATEUR



SCIENTIFIC LIBRARY 116

ORDINATEUR PERSONNEL

FX-850P

MANUEL DE L'UTILISATEUR

Le contenu de ce manuel est susceptible d'être modifié sans préavis.
La copie illégale de tout ou d'une partie de ce manuel est strictement interdite.
Prière d'être conscient que l'utilisation de ce manuel pour un usage autre que personnel sans la permission de CASIO est interdite conformément aux lois sur les droits d'auteur.
CASIO Computer Co. Ltd. décline toute responsabilité pour tous dommages ou pertes résultant de l'emploi de ce manuel.
En outre, veuillez noter que CASIO décline également toute responsabilité pour des pertes ou réclamations émanant de tiers et qui pourraient résulter de l'emploi de cet appareil.

CASIO®

PREFACE

Félicitations pour votre achat de l'ordinateur personnel FX-850P de CASIO. Ce manuel introduit et explique la fonction de calcul scientifique et le langage de programmation en BASIC utilisés avec cet ordinateur. Il est recommandé que chacun, des novices aux vétérans du BASIC, se familarise avec le nom et la fonction de chaque pièce de l'ordinateur avant de tenter l'opération. Même lorsque la programmation en BASIC n'est pas utilisée, une fonction de mise en mémoire de formules donne des calculs de formules et des calculs de rapports simplifiés, une fonction de la BANQUE DE DONNEES incorporée permet le traitement en mémoire et les recherches. A part cela, cet ordinateur comprend également un utilitaire scientifique incorporé qui assure un total de 116 programmes utilitaires pour des applications statistiques, mathématiques et scientifiques.

PRECAUTIONS

Cet ordinateur est un produit du génie électronique, des essais et du contrôle de qualité de CASIO, tous d'un niveau très élevé. Les points suivants doivent être soigneusement notés afin de permettre à cet appareil de vous assurer les nombreuses années de fonctionnement sans problème pour lesquelles il a été conçu

- Cet appareil est constitué à partir de pièces électroniques de précision et ne devra jamais être démonté. Il ne faudra pas le faire tomber à terre, ni lui faire subir d'autres chocs violents. En effet, des heurts trop forts peuvent entraîner la suppression de l'exécution des programmes ou une altération importante de la mémoire de l'appareil.
- Ne pas utiliser ni ranger cet appareil dans des zones soumises à des températures trop chaudes, trop humides ou exposées à la poussière.
- La réponse de l'affichage peut devenir lente ou même disparaître à des températures trop basses. L'appareil reprendra son fonctionnement normal après un retour à une température normale.
- Les connecteurs de cet appareil sont conçus exclusivement pour un branchement aux seules unités d'extension FA-6 spécifiées.
- L'affichage peut s'assombrir lorsque l'avertisseur sonore retentit, mais cela n'est pas un signal de mauvais fonctionnement et ne cause aucun dérangement.
- Les piles devront être remplacées le plus tôt possible, après que leur usure aura été indiquée par l'assombrissment de l'affichage pendant un fonctionnement normal.
- Remplacer les piles au moins une fois tous les deux ans, même si l'appareil n'est pas utilisé pendant cette période. Des piles mortes laissées dans l'appareil peuvent causer de graves dégâts, à cause du liquide qui peut en sortir. Il faut les enlever le plus tôt possible.
- Maintenir le connecteur de l'appareil recouvert du couvercle de connecteur chaque fois que l'appareil n'est pas connecté à une unité d'extension et éviter de toucher le connecteur.
- De fortes charges d'électricité statique peuvent causer une altération du contenu de la mémoire ou une panne du fonctionnement des touches. Si cette situation venait à se présenter, retirer les piles et les recharger.
- Avant de connecter des dispositifs périphériques, toujours prendre soin de vérifier que l'interrupteur d'alimentation de l'appareil est sur la position "OFF".
- Ne jamais utiliser de diluant, benzine, ni d'autre agent volatil pour nettoyer l'extérieur de l'appareil. Utiliser un chiffon doux humecté dans une solution neutre d'eau et un détergent neutre et bien essorer le chiffon.
- Ne pas mettre l'interrupteur d'alimentation de l'appareil sur la position "OFF" pendant l'exécution d'un programme ou pendant des calculs.
- En cas de mauvais fonctionnement, prendre contact avec le magasin qui a vendu l'ordinateur ou avec un négociant proche.
- Avant de faire appel à leurs services, veuiller relire ce manuel, vérifier l'alimentation, contrôler le programme dans le cas d'erreurs logiques, etc.

TABLE DES MATIERES

PARTIE 1 CONFIGURATION DE L'APPAREIL 1

1-1 GUIDE GENERAL 1
1-2 FONCTIONS OPERATIONNELLES 2
1-3 AFFICHAGE DE SYMBOLES 4
1-4 CLAVIER 5
1-4-1 Fonctions de haut de touche 5
1-4-2 Fonctions notées au-dessus des touches 6
1-5 ECRAN 6
1-5-1 Lignes physiques et lignes logiques 7
1-5-2 Ecran virtuel 7
1-5-3 Editeur d'écran 7
1-5-4 Contraste d'affichage 8
1-6 CARACTERES D'AFFICHAGE 8
1-7 ALIMENTATION 9
1-8 COUPURE AUTOMATIQUE DE L'ALIMENTATION 10
1-9 CONNECTEUR 10

PARTIE 2 OPERATIONS FONDAMENTALES 11

2-1 MODE CAL 11
2-2 MODE BASIC 11
2-3 FONCTION DE MISE EN MEMOIRE DE FORMULES 12
2-4 FONCTION DE LA BANQUE DE DONNEES (MEMO IN) 13
2-5 BIBLIOTHEQUE SCIENTIFIQUE INCORPOREE 14
2-6 RESUME 14

PARTIE 3 FONCTION DE CALCULS 15

3-1 PREPARATIONS AUX CALCULS MANUELS 15
3-2 ENTREE ET CORRECTION DE CALCULS MANUELS 15
3-3 ORDRE DE PRIORITE 19
3-4 CALCULS SCIENTIFIQUES 20
3-5 CALCULS UTILISANT DES VARIABLES 29
3-6 AUTRES CALCULS 30

PARTIE 4 FONCTION DE MISE EN MEMOIRE DE FORMULES 33

4-1 UTILISATION POUR LA PREPARATION DES TABLEAUX 34

PARTIE 5 FONCTION DE LA BANQUE DE DONNEES 37

5-1 ENTREE DES DONNEES 37
5-2 AFFICHAGE DE DONNEES 38
5-3 EDITION DE DONNEES 39
5-4 ADDITION D'ENREGISTREMENTS 40
5-4-1 Addition de données 40
5-4-2 Insertion de données 40
5-5 SUPPRESSION ET EFFACEMENT GENERAL DES DONNEES 41
5-5-1 Suppression de données 41
5-5-2 Effacement général des données 41
5-6 RECHERCHE DE DONNEES 41
5-6-1 Recherche conditionnelle 42
5-7 UTILISATION DES DONNEES DE LA BANQUE DE DONNEES DANS LES PROGRAMMES 42
5-8 APPLICATIONS DES FONCTIONS DE LA BANQUE DE DONNEES 44

PARTIE 6 PROGRAMMATION EN BASIC 46

6-1 CARACTERISTIQUES DU BASIC 46
6-2 CONFIGURATION DE PROGRAMMES EN BASIC 47
6-2-1 Format de programmes en BASIC 47
6-3 ENTREE DE PROGRAMMES EN BASIC 48
6-3-1 Préparation 48
6-3-2 Entrée de programmes 49
6-3-3 Edition de programmes 50
6-4 EXECUTION DE PROGRAMMES EN BASIC 53
6-4-1 Exécution de programmes 53
6-4-2 Erreurs 54
6-5 COMMANDES 55
6-6 OPERATEURS 58
6-7 CONSTANTES ET VARIABLES 60
6-7-1 Constantes 60
6-7-2 Variables 61
6-7-3 Sommaire 65
6-8 SAUVEGARDE ET CHARGEMENT DE PROGRAMMES 67
6-8-1 Sauvegarde de programmes 67
6-8-2 Vérification de programme 67
6-8-3 Chargement de programmes 68

PARTIE 7 DISPOSITIFS PERIPHERIQUES 70

7-1 UNITE D'INTERFACE POUR CASSETTE FA-6 70
7-1-1 Caractéristiques 70
7-1-2 Connexions 71

7-1-3 Interface pour cassette 71
7-1-4 Interface RS-232C 73
7-1-5 Interface Centronics (Interface d'imprimante) 74
7-2 IMPRIMANTE-TABLE TRACANTE (FP-100) 75
7-2-1 Caractéristiques 75
7-2-2 Connexions 75
7-2-3 Impression des données 75
7-3 IMPRIMANTE DE CARACTERES (FP-40) 76
7-3-1 Caractéristiques 76
7-3-2 Connexion 76
7-4 MODULE D'EXTENSION RAM (mémoire vive) (RP-8 (8 KO)), (RP-33 (32 KO)) 77
7-4-1 Carte de mémoire d'extension 77
7-4-2 Manipulaton des modules RAM 78

PARTIE 8 COMPATIBILITE DE LA SERIE PB-100 79

8-1 ENTREE/EDITION DES PROGRAMMES DE LA SERIE PB-100 79
8-2 EXECUTION DES PROGRAMMES DE LA SERIE PB-100 80
8-2-1 Mode DEFM 80
8-2-2 Tableaux en utilisant l'instruction DEFM 81
8-2-3 Affichages du mode DEFM 82
8-2-4 Commande CLEAR, commande DIM, commande DEFM dans mode DIM et mode DEFM 82
8-3 CHARGEMENT DES PROGRAMMES DE LA SERIE PB-100 82
8-4 LECTURE DES DONNEES DE LA SERIE PB-100 83
8-5 FORMAT DE COMMANDES 83

PARTIE 9 ELEMENTS FONDAMENTAUX DE TRAITEMENT DE FICHIERS 85

9-1 CLASSEMENT DES DISPOSITIFS 85
9-2 DESCRIPTEURS DE FICHIER 86
9-2-1 Bandes cassettes 86
9-2-2 Circuit de communications 87

PARTIE 10 REFERENCE DE COMMANDES 89

ELEMENTS DE FORMAT 89
COMMANDES MANUELLES 90
COMMANDES FONDAMENTALES 98
FONCTIONS NUMERIQUES 131
FONCTIONS DE CARACTERES 149
COMMANDES E/S 162
COMMANDES DE LA BANQUE DE DONNEES 174

PARTIE 11 BIBLIOTHEQUE SCIENTIFIQUE 181

11-1 EXECUTION DE BIBLIOTHEQUE 181
11-1-1 Activation de la bibliothèque 181
11-1-2 Fin de bibliothèque 183
11-1-3 Affichage d'activation de bibliothèque 183
11-1-4 Exemples utilisés dans ce manuel 184
11-1-5 Précautions en cas d'utilisation de la bibliothèque 184

1000 CALCULS DE MEMOIRE 186
5010 ANALYSE DE FACTEURS PREMIERS 189
5020 PLUS GRAND COMMUN MULTIPLE/PLUS PETIT COMMUN MULTIPLE 190
5040 EQUATIONS SIMULTANEES (ELIMINATION DE GAUSS-JORDAN) 190
5050 EQUATION DU SECOND DEGRE 193
5060 EQUATIONS DU TROISIEME DEGRE 194
5080 SOLUTION NUMERIQUE D'UNE EQUATION (METHODE DE NEWTON) 196
5090 SOLUTION NUMERIQUE D'UNE EQUATION (METHODE DE BISSECTION) 198
5100 OPERATIONS DE MATRICES 200
5200 INTEGRATION NUMERIQUE (METHODE DE ROMBERG) 210
5220 EQUATION DIFFERENTIELLE ORDINAIRE (METHODE DE RUNGE-KUTTA) 212
5230 INTERPOLATION DE LAGRANGE 213
5250 FONCTION GAMMA Γ(x) 214
5260 FONCTION DE BESSEL Jn(x) 215
5270 FONCTION DE BESSEL Yn(x) 216
5280 FONCTION MODIFIEE DE BESSEL In(x) 217
5290 FONCTION MODIFIEE DE BESSEL Kn(x) 218
5300 NOMBRE COMPLEXE 219
5350 BINAIRE-DECIMAL-HEXADECIMAL 223
5510 DROITE PASSANT PAR DEUX POINTS 228
5520 ANGLE D'INTERSECTION DE DEUX DROITES 229
5530 DISTANCE ENTRE UN POINT ET UNE DROITE 230
5540 MOUVEMENT ROTATIONNEL 231
5550 CERCLE PASSANT PAR TROIS POINTS 232
5560 LONGUEUR DE TANGENTES A PARTIR D'UN POINT VERS UN CERCLE 233
5570 EQUATION DE TANGENTE 234
5600 SUPERFICIE D'UN TRIANGLE 235
5605 SUPERFICIE D'UN TRAPEZE 236
5610 SUPERFICIE D'UN PARALLELOGRAMME 237
5615 SUPERFICIE D'UN CERCLE 238
5620 SUPERFICIE D'UN SECTEUR 239
5625 SUPERFICIE D'UN SEGMENT 240

5630 SUPERFICIE D'UNE ELLIPSE....241
5635 SUPERFICIE D'UN POLYGONE....242
5650 SUPERFICIE DE LA SURFACE D'UNE SPHERE....243
5655 SUPERFICIE DE LA SURFACE D'UNE ZONE D'UNE SPHERE....244
5660 SUPERFICIE DE LA SURFACE D'UN SECTEUR SPHERIQUE....245
5665 SUPERFICIE DE LA SURFACE D'UN CYLINDRE CIRCULAIRE....246
5670 SUPERFICIE DE LA SURFACE D'UN CONE CIRCULAIRE....247
5675 SUPERFICIE DE LA SURFACE D'UN TRONC D'UN CONE CIRCULAIRE....248
5700 VOLUME D'UNE SPHERE....249
5705 VOLUME DE LA ZONE D'UNE SPHERE....250
5710 VOLUME D'UN SECTEUR SPHERIQUE....251
5715 VOLUME D'UN CYLINDRE CIRCULAIRE....252
5720 VOLUME D'UN CONE CIRCULAIRE....253
5725 VOLUME DU TRONC D'UN CONE CIRCULAIRE....254
5730 VOLUME D'UNE CALE....255
5735 VOLUME D'UNE PYRAMIDE....256
5740 VOLUME DU TRONC D'UNE PYRAMIDE....257
5745 VOLUME D'UN ELLIPSOIDE....258
5750 CERCLE INSCRIT ET CERCLES CIRCONSCRITS D'UN POLYGONE....259
5760 POLYEDRE REGULIER....260
5800 FACTORISATION....262
5810 FONCTIONS TRIGONOMETRIQUES....263
5820 FONCTIONS DIFFERENTIELLES....264
5830 INTEGRATION....265
5840 TRANSFORMATION DE LAPLACE....266
5900 TABLEAU PERIODIQUE....268
5910 CONSTANTES SCIENTIFIQUES....273
5920 CONSTANTES DE DISSOCIATION ELECTROLYTIQUE....275
5930 MOUVEMENT ET ENERGIE....276
5932 MOUVEMENT D'ONDES....278
5934 CIRCUITS CA & CC....279
5936 CHAMPS ELECTRIQUE ET MAGNETIQUE....280
5938 THERMODYNAMIQUES ET AUTRES....282
5950 CONVERSIONS METRIQUES DE LONGUEUR....283
5960 CONVERSIONS METRIQUES DE SUPERFICIE....285
5970 CONVERSIONS METRIQUES DE VOLUME....286
5980 CONVERSIONS METRIQUES DE POIDS....288
6210 INTEGRALES DE PROBABILITE SUPERIEURE (REPARTITION NORMALE)....289
6220 INTEGRALES DE PROBABILITE SUPERIEURE (REPARTITION x^2)....290
6230 INTEGRALES DE PROBABILITE SUPERIEURE (REPARTITION t)....291
6240 INTEGRALES DE PROBABILITE SUPERIEURE (REPARTITION F)....292

6310 FREQUENCE CUMULATIVE SUPERIEURE (REPARTITION BINOMIALE)....293
6320 FREQUENCE CUMULATIVE SUPERIEURE (REPARTITION DE POISSON)....294
6330 FREQUENCE CUMULATIVE SUPERIEURE (REPARTITION HYPERGEOMETRIQUE)....295
6410 REPARTITION NORMALE DE POINT DE POURCENTAGE....296
6420 POINT DE POURCENTAGE (REPARTITION x^2)....297
6430 POINT DE POURCENTAGE (REPARTITION t)....298
6440 POINT DE POURCENTAGE (REPARTITION F)....299
6450 NOMBRES ALEATOIRES NORMAUX....300
6460 NOMBRES ALEATOIRES EXPONENTIELS....301
6500 STATISTIQUES A VARIABLE UNIQUE....302
6510 ANALYSE DE REGRESSION LINEAIRE ($y = a + bx$)....305
6520 ANALYSE DE REGRESSION LOGARITHMIQUE ($y = a + b\ln x$)....308
6530 ANALYSE DE REGRESSION EXPONENTIELLE ($y = ab^x$)....311
6540 ANALYSE DE REGRESSION DE PUISSANCE ($y = ax^b$)....314
6610 ESTIMATION D'INTERVALLE DE MOYENNES (POUR VARIANCE CONNUE)....318
6620 ESTIMATION D'INTERVALLE DE MOYENNES (POUR VARIANCE INCONNUE)....322
6630 ESTIMATION D'INTERVALLE DES VARIANCES....326
6640 ESTIMATION D'INTERVALLE D'ECART-TYPE....330
6650 ESTIMATION D'INTERVALLE DE RAPPORT DE VARIANCES....334
6660 ESTIMATION D'INTERVALLE DE DIFFERENCE DE MOYENNES....339
6670 ESTIMATION D'INTERVALLE DES RAPPORTS....344
6680 ESTIMATION D'INTERVALLE DE DIFFERENCE DE RAPPORTS....345
6710 ESSAI DE MOYENNES DE POPULATION (DEUX COTES): POUR VARIANCE CONNUE....347
6711 ESSAI DE MOYENNES DE POPULATION (COTE DROIT): POUR VARIANCE CONNUE....351
6712 ESSAI DE MOYENNES DE POPULATION (COTE GAUCHE): POUR VARIANCE CONNUE....355
6720 ESSAI DE MOYENNES DE POPULATION (DEUX COTES): POUR VARIANCE INCONNUE....359
6721 ESSAI DE MOYENNES DE POPULATION (COTE DROIT): POUR VARIANCE INCONNUE....363
6722 ESSAI DE MOYENNES DE POPULATION (COTE GAUCHE): POUR VARIANCE INCONNUE....367
6730 ESSAI DE VARIANCES DE POPULATION (DEUX COTES)....371
6731 ESSAI DE VARIANCES DE POPULATION (COTE DROIT)....375
6732 ESSAI DE VARIANCES DE POPULATION (COTE GAUCHE)....379
6740 ESSAI DE RAPPORTS DE VARIANCES (DEUX COTES)....383

6741 ESSAI DE RAPPORTS DE VARIANCES (COTE DROIT) 388
6742 ESSAI DE RAPPORTS DE VARIANCES (COTE GAUCHE) 393
6750 ESSAI DE DIFFERENCE DE MOYENNES (DEUX COTES) 398
6751 ESSAI DE DIFFERENCE DE MOYENNES (COTE DROIT) 403
6752 ESSAI DE DIFFERENCE DE MOYENNES (COTE GAUCHE) 408
6760 ESSAI DE RAPPORTS (DEUX COTES) 413
6761 ESSAI DE RAPPORTS (COTE DROIT) 414
6762 ESSAI DE RAPPORTS (COTE GAUCHE) 415
6770 ESSAI DE DIFFERENCE DE RAPPORTS (DEUX COTES) 416
6771 ESSAI DE DIFFERENCE DE RAPPORTS (COTE DROIT) 418
6772 ESSAI DE DIFFERENCE DE RAPPORTS (COTE GAUCHE) 419

PARTIE 12 APPENDICES 421

12-1 TABLEAU DES CODES DE CARACTERES 421
12-2 TABLEAU DES MESSAGES D'ERREURS 423
12-3 TABLEAU DES COMMANDES ET FONCTIONS 425
12-4 LISTE DES MOTS RESERVES 426

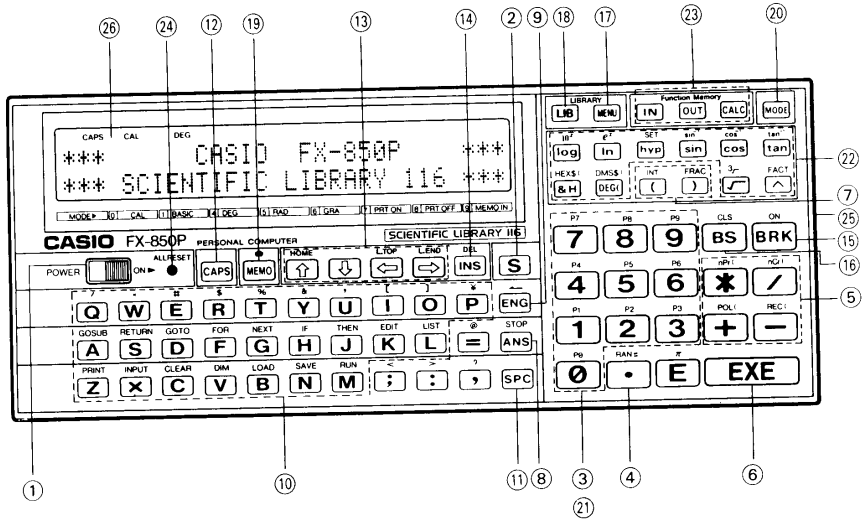
SPECIFICATIONS 427

INDEX 429

PARTIE 1

CONFIGURATION DE L'APPAREIL

1-1 GUIDE GENERAL



① Interrupteur d'alimentation
② Touche de majuscules
③ Touches numériques
④ Touche décimale
⑤ Touches d'opérateur arithmétique
⑥ Touche d'exécution
⑦ Touches de parenthèses
⑧ Touche de réponse
⑨ Touche de données techniques
⑩ Touches alphabétiques
⑪ Touche d'espace
⑫ Touche CAPS (majuscules)
⑬ Touches de curseur
⑭ Touche d'insertion/effacement
⑮ Touche d'interruption
⑯ Touche d'espace arrière/effacement de l'écran
⑰ Touche de menu
⑱ Touche LIB
⑲ Touche de mémoire
⑳ Touche de mode
㉑ Touches de zone de programme
㉒ Touches de fonction
㉓ Touches de mise en mémoire de formules
㉔ Bouton de réinitialisation générale (ALL RESET)
㉕ Bouton P
㉖ Ecran

1-2 FONCTIONS OPERATIONNELLES

① **Interrupteur d'alimentation (POWER ON)**
Glisse vers la droite pour mettre sous tension ("ON") et vers la gauche pour mettre hors tension ("OFF").

② **Touche de majuscules (S)**
Utilisée pour entrer des commandes en BASIC et les symboles notés en rouge sur le panneau de touches. Chaque pression sur cette touche fait apparaître et disparaître le symbole "S" sur l'affichage.
* Tout au long de ce manuel, cette touche est représentée par SHIFT pour la distinguer de la touche alphabétique S .

③ **Touches numériques (0 ~ 9)**
Entrent les valeurs numériques indiquées sur chaque touche.

④ **Touche décimale (•)**
Entre le point des décimales.

⑤ **Touches d'opérateur arithmétique (+ , − , * , /)**
Entrent les opérateurs arithmétiques indiqués sur les touches.
+ : Addition
− : Soustraction
* : Multiplication
/ : Division

⑥ **Touche d'exécution (EXE)**
Termine l'entrée d'un calcul et en donne le résultat. La fonction de cette touche est équivalente à celle de la touche " = " sur une calculatrice ordinaire.
Cette touche est également utilisée pour entrer des lignes d'un programme et pour l'exécution réelle des programmes.

⑦ **Touches de parenthèses (( ))**
Entrent les parenthèses dans le cas de calculs avec parenthèses tels que: $5 \times (10 + 20)$.

⑧ **Touche de réponse (ANS)**
Rappelle le résultat du calcul manuel ou de programme effectué le plus récemment. Une pression sur cette touche pendant l'exécution du programme entraîne la suspension de l'exécution jusqu'à ce que la touche EXE soit enfoncée (STOP affiché).

⑨ **Touche de données techniques (ENG / ⇧)**
Convertit le résultat d'un calcul en un affichage exponentiel.

⑩ **Touches alphabétiques**
Entrent les caractères alphabétiques indiqués sur chaque touche.

⑪ **Touche d'espace (SPC)**
Entre un espace.

⑫ **Touche CAPS (majuscules) (CAPS)**
Change les touches alphabétiques entre les caractères majuscules et minuscules. Le mode majuscules est indiqué par le symbole "CAPS" sur l'affichage.

⑬ **Touches de curseur (← , → , ↑ , ↓)**
Déplacent le curseur sur l'écran. Chaque pression permet de déplacer le curseur dans la direction indiquée sur les touches enfoncées, alors que le maintien enfoncé des touches permet un déplacement continu à grande vitesse. Chaque touche de curseur assume également une fonction différente lorsqu'elle est enfoncée en combinaison avec la touche SHIFT .

TOUCHE	FONCTION	SHIFT +
←	Curseur à gauche	Déplace vers le début de la ligne logique
→	Curseur à droite	Déplace vers la fin de la ligne logique
↑	Curseur en haut	Fait remonter l'écran sans mouvement du curseur
↓	Curseur en bas	Fait descendre l'écran sans mouvement du curseur

(HOME) Voir la page 7 pour les détails sur les lignes physiques et les lignes logiques.

⑭ **Touche d'insertion/effacement (INS / DEL)**
Insère un espace à la position actuelle du curseur en décalant tout à partir de la droite de la position du curseur d'un espace vers la droite. En combinaison avec la touche SHIFT , efface le caractère situé à la position actuelle du curseur et remplit automatiquement l'espace créé en décalant tout vers la droite du curseur d'un espace vers la gauche. Le maintien enfoncée de cette touche pour l'une des fonctions permet l'opération continue à grande vitesse de la fonction respective.

⑮ **Touche d'interruption (BRK)**
Termine les opérations manuelles, l'exécution des programmes, la sortie sur imprimante et la sortie LIST. Réenclenche également l'alimentation lorsque celle-ci a été interrompue par la fonction de coupure automatique (voir page 10).

⑯ **Touche d'espace arrière/effacement de l'écran (BS / CLS)**
Efface le caractère placé immédiatement à la gauche du curseur et remplit automatiquement l'espace créé en décalant tout à partir de la droite de la position du curseur d'un espace vers la gauche. En combinaison avec la touche SHIFT , efface le contenu de l'écran et place le curseur au coin supérieur gauche de l'écran.

⑰ **Touche de menu (MENU)**
Uniquement applicable dans le mode CAL, cette touche est utilisée pour afficher un menu de la bibliothèque scientifique incorporée. Voir la PARTIE 11 "BIBLIOTHEQUE SCIENTIFIQUE" pour les détails.

⑱ **Touche LIB (LIB)**
Uniquement applicable dans le mode CAL, cette touche exécute l'opération correspondant à un numéro de bibliothèque entré. Voir la PARTIE 11 "BIBLIOTHEQUE SCIENTIFIQUE" pour les détails.

⑲ **Touche de mémoire (MEMO)**
Utilisée pour afficher et rechercher les données de la BANQUE DE DONNEES. Voir la PARTIE 5 "FONCTION DE LA BANQUE DE DONNEES" pour les détails.

⑳ **Touche de mode (MODE)**
Utilisée en combinaison avec les touches numériques pour spécifier les modes d'opération.
MODE 0 ..... Mode CAL (sélectionné lors de la mise sous tension)
MODE 1 ..... Mode BASIC (écriture/édition de programmes)
MODE 4 ..... Mode DEG (unité d'angle = degrés)
MODE 5 ..... Mode RAD (unité d'angle = radians)
MODE 6 ..... Mode GRA (unité d'angle = grades)
MODE 7 ..... Activation d'impression
MODE 8 ..... Désactivation d'impression
MODE 9 ..... Mode MEMO IN (fonction de la BANQUE DE DONNEES)

㉑ **Touches de zone de programme (SHIFT PB ~ PB)**
Exécute le programme dans la zone de programme correspondante dans le mode CAL. Spécifie une zone de programme pour l'écriture ou l'édition dans le mode BASIC.

㉒ **Touches de fonction ([log] , [ln] , [sin] , etc.)**
Permettent une entrée à une touche de fonctions souvent utilisées.
- Fonctions à entrée directe
  [log] , [ln] , [hyp] , [sin] , [cos] , [tan] , [&H], [DEG(] , [√‾] , [^]
- Fonctions [SHIFT]
  [10^x] , [e^x] , [SET] , [sin⁻¹] , [cos⁻¹] , [tan⁻¹] , [HEX$(] , [DMS$(] , [INT] , [FRAC] , [ˣ/ᵧ] , [FACT] , [nPr(] , [nCr(] , [POL(] , [REC(] , [RAN#] , [π]

㉓ **Touches de mise en mémoire de formules ([IN] , [OUT] , [CALC])**
Utilisées lorque l'on travaille avec la fonction de mise en mémoire de formules. Voir la PARTIE 4 "FONCTION DE MISE EN MEMOIRE DE FORMULES" pour les détails.

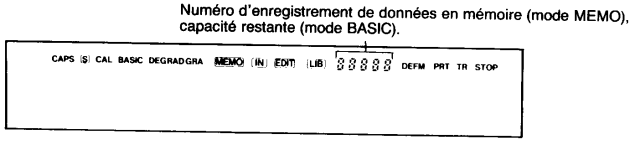
㉔ **Bouton de réinitialisation générale (ALL RESET •)**
Efface tout le contenu de la mémoire et passe dans le mode CAL. Toutes les données importantes doivent être sauvegardées ailleurs avant d'appuyer sur ce bouton. Si une pression sur ce bouton ne réussit pas à effacer le contenu de la mémoire, appuyer d'abord sur le bouton P et ensuite à nouveau sur le bouton ALL RESET.

㉕ **Bouton P (• P)** (panneau arrière)
Bouton de réinitialisation du matériel destiné à arrêter un mauvais fonctionnement dû à l'électricité statique. Bien que l'exécution soit interrompue, le contenu de la mémoire est retenu. Le bouton ALL RESET doit être utilisé lorsque le mauvais fonctionnement altère le contenu de la mémoire. Noter que l'alimentation est coupée, puis remise lorsque le bouton P est enfoncé.

㉖ **Ecran**
Un affichage à cristaux liquides à 32 colonnes et 2 lignes sur lequel apparaissent des caractères de 5 × 7 points.

1-3 AFFICHAGE DE SYMBOLES

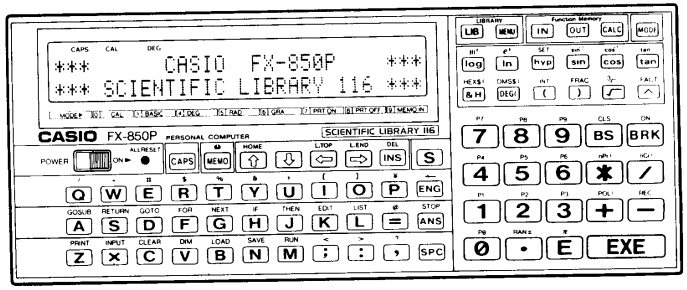
Les symboles notés sur l'affichage illustré ci-dessous apparaissent pour indiquer l'état actuel d'un calcul.



CAPS	:	Caractères alphabétiques majuscules (minuscules lorsque non affiché)
[S]	:	Mode de majuscules (les commandes/fonctions marquées au-dessus des touches peuvent être entrées)
CAL	:	Mode CAL (calculs arithmétiques élémentaires ou calculs de fonctions)
BASIC	:	Mode BASIC (entrée, édition, exécution de programmes en BASIC)
DEG	:	Unité d'angle = degrés
RAD	:	Unité d'angle = radians
GRA	:	Unité d'angle = grades
[MEMO]	:	Recherche, affichage, entrée, édition de données de la BANQUE DE DONNEES
[IN]	:	Entrée, édition de données de la BANQUE DE DONNEES
[EDIT]	:	Edition de données de la BANQUE DE DONNEES, édition de programmes en BASIC (en utilisant la commande EDIT)
DEFM	:	Mode DEFM (pour l'exécution des programmes de la série PB-100 de CASIO)
[LIB]	:	Mode de fonction de la bibliothèque scientifique

PRT	:	Mode d'impression (sortie du contenu de l'affichage sur l'imprimante)
TR	:	Mode d'analyse (analyse l'exécution des programmes en BASIC)
STOP	:	Affichage suspendu (par instruction PRINT ou commande STOP en BASIC)

1-4 CLAVIER



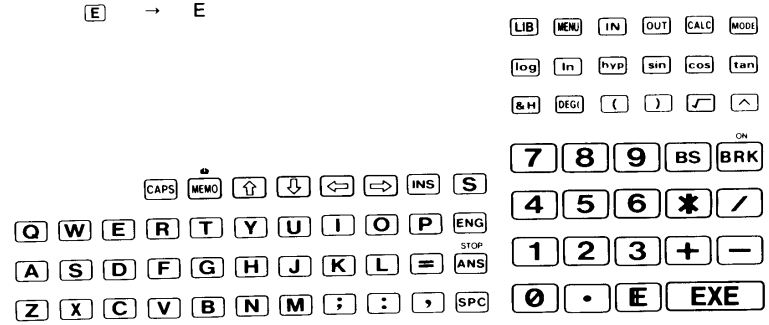
Un regard sur le clavier de l'appareil révèle les caractères et les symboles situés au-dessus des touches. Ceux-ci sont accédés en utilisant les touches [CAPS] et [SHIFT] .

1-4-1 Fonctions de haut de touche

Mode normal
Dans ce mode, chaque touche entre les caractères, symboles ou commandes notés sur les touches elles-mêmes. (Cet état est automatiquement réglé lors de la mise sous tension et immédiatement après la procédure ALL RESET.)

EXEMPLE:

Opération		Affichage
[A]	→	A
[E]	→	E



Mode de minuscules

Une pression sur la touche [CAPS] permet de changer les touches alphabétiques (uniquement) en caractères minuscules, indiqué par le symbole CAPS disparaissant de l'affichage. Une seule pression sur la touche [CAPS] permet de verrouiller le clavier dans le mode de minuscules alors qu'une nouvelle pression ramène aux majuscules.

EXEMPLE:

Opération		Affichage
[CAPS] [A]	→	a
[B]	→	b
[D]	→	d

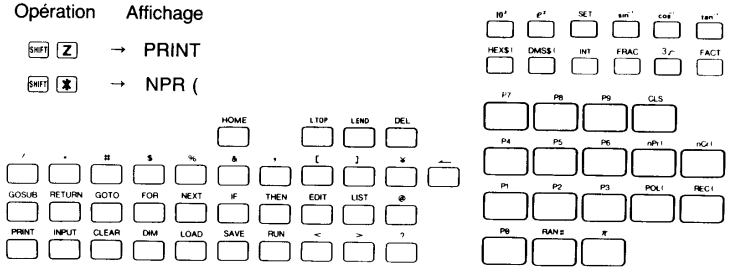
[q] [w] [e] [r] [t] [y] [u] [i] [o] [p]
[a] [s] [d] [f] [g] [h] [j] [k] [l]
[z] [x] [c] [v] [b] [n] [m]

1-4-2 Fonctions notées au-dessus des touches

Les commandes à une touche en BASIC, les symboles et les commandes notés au-dessus des touches sont entrés lorsque les touches correspondantes sont enfoncées après la touche [SHIFT]. Noter cependant qu'une pression sur les touches numériques ([0] ~ [9]) après [SHIFT] dans le mode CAL permet d'exécuter le programme dans la zone de programme correspondante, alors que, dans le mode BASIC, elle permet de passer à la zone de programme correspondante.

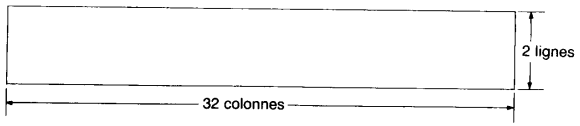
EXEMPLE:

Opération		Affichage
[SHIFT] [Z]	→	PRINT
[SHIFT] [*]	→	NPR (



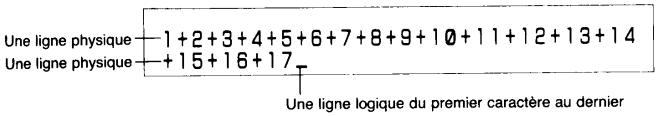
1-5 ECRAN

L'écran est un affichage à cristaux liquides de 32 colonnes × 2 lignes. Les caractères sont formés par une matrice de 5 × 7 points.



1-5-1 Lignes physiques et lignes logiques

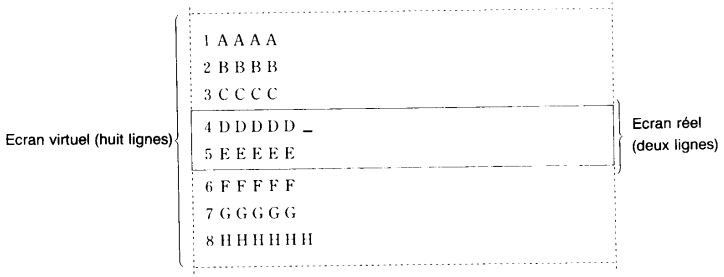
La capacité maximale d'affichage d'une ligne est de 32 colonnes, mais intérieurement, l'appareil est capable de traiter des lignes d'une longueur allant jusqu'à 255 caractères. La ligne de capacité d'affichage (32 caractères) est ce qu'on appelle la ligne physique, alors que la ligne de capacité interne est appelée la ligne logique. Une ligne logique est une ligne continue de caractères dans laquelle chaque colonne située à l'extrême droite de l'écran n'est pas un caractère nul.



Une pression sur [SHIFT] [⇦] permet de déplacer le curseur au début de la ligne logique, alors qu'une pression sur [SHIFT] [⇨] permet de déplacer le curseur à la fin de la ligne logique. Ces opérations sont utiles pour déterminer l'étendue des lignes logiques.

1-5-2 Ecran virtuel

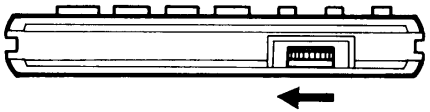
L'écran peut afficher deux lignes à la fois et lorsque la troisième ligne est entrée, la première ligne disparaît par le haut de l'écran. Les lignes qui disparaissent de l'écran peuvent cependant être ramenées en vue en utilisant les touches de curseur ([⇧] / [⇩]), car l'appareil est capable de mettre en mémoire interne jusqu'à huit lignes. Ces huit lignes constituent l'écran virtuel, alors que les deux lignes actuellement affichées sont appelées l'écran réel.



1-5-3 Editeur d'écran

Toutes les données ou lignes de programme comprises sur l'écran virtuel peuvent être éditées. Tout d'abord, la portion du programme ou des données est portée sur l'écran réel et ensuite le curseur est placé à la position à éditer.

1-5-4 Contraste d'affichage



L'affichage peut apparaître sombre ou pâle suivant la puissance des piles ou l'angle de vision. Le contraste de l'affichage peut être ajusté au niveau désiré en faisant tourner le cadran de contrôle. Une rotation du cadran vers le bas (sens de la flèche) assombrit l'affichage, alors qu'une rotation vers le haut éclaire l'affichage.
Un affichage faible alors que le contraste est réglé à un niveau élevé est la preuve d'un affaiblissement des piles, qui devront alors être remplacées le plus rapidement possible (voir page 9).

1-6 CARACTERES D'AFFICHAGE

Le tableau suivant illustre la relation entre les caractères et les codes de caractères.

Tableau de codes de caractères

Chiffre de poids fort →

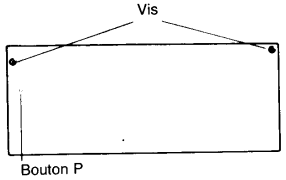
↓ Chiffre de poids faible		0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
	HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0			(SPC)	0	@	P	`	p	Å	0	(SPC)	ー	タ	ミ	⋰	×
1	1		(DEL)	!	1	A	Q	a	q	∫	1	。	ア	チ	ム	⋱	円
2	2	(LINE TOP)	(INS)	〃	2	B	R	b	r	√	2	「	イ	ツ	メ	∓	年
3	3			#	3	C	S	c	s	∴	3	」	ウ	テ	モ	↑	月
4	4	(SHIFT RELEASE)		$	4	D	T	d	t	Σ	4	、	エ	ト	ヤ	←	日
5	5	(LINE CANCEL)		%	5	E	U	e	u	Ω	5	・	オ	ナ	ユ	↓	千
6	6	(LINE END)		&	6	F	V	f	v	▮	6	ヲ	カ	ニ	ヨ	→	万
7	7	(BEL)		'	7	G	W	g	w	▮	ァ	ァ	キ	ヌ	ラ	π	£
8	8	(BS)		(	8	H	X	h	x	α	8	ィ	ク	ネ	リ	♠	ø
9	9	(CAPS L→U)		)	9	I	Y	i	y	β	9	ゥ	ケ	ノ	ル	♥	±
10	A	(LF)		*	:	J	Z	j	z	γ	+	ェ	コ	ハ	レ	◆	∓
11	B	(HOME)		+	;	K	[	k	{	ε	−	ォ	サ	ヒ	ロ	♣	○
12	C	(CLS)	(⇨)	,	<	L	¥	l	¦	θ	n	ャ	シ	フ	ワ	□	
13	D	(CR)	(⇦)	−	=	M	]	m	}	μ	x	ュ	ス	ヘ	ン	○	
14	E	(SHIFT SET)	(⇧)	.	>	N	^	n	~	σ	-1	ョ	セ	ホ	″	△	
15	F	(CAPS U→L)	(⇩)	/	?	O	_	o		ϕ	÷	ッ	ソ	マ	。	＼	

* Les segments vides ne sont pas sortis.
* Les notations entre parenthèses sont des codes de commande et ne sont pas affichés.
* Les caractères ne pouvant pas être affichés en utilisant l'entrée du clavier peuvent être affichés en utilisant la fonction CHR$.

1-7 ALIMENTATION

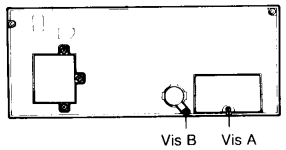
Cet appareil est équipé d'une alimentation principale (deux piles au lithium CR2032) et d'une alimentation de protection (une pile au lithium CR1220). Les piles doivent être remplacées chaque fois que l'affichage reste pâle, même après le réglage du contraste. Les piles doivent aussi être remplacées une fois tous les deux ans sans tenir compte de l'utilisation de l'appareil.

■ Remplacement des piles



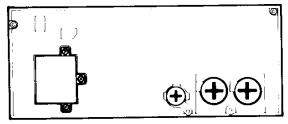
1) Mettre l'interrupteur d'alimentation de l'appareil sur la position "OFF" et retirer le panneau arrière de l'appareil après avoir enlevé les deux vis le maintenant en place.

2) Retirer le support des piles principales en enlevant la vis A et/ou le support de la pile de protection en enlevant la vis B.



3) Enlever les vieilles piles en tournant l'appareil pour que le compartiment des piles soit dirigé vers le bas et en tapotant légèrement.

4) Essuyer les surfaces des nouvelles piles avec un chiffon sec et les charger dans le compartiment des piles en s'assurant que le pôle positif (+) est dirigé vers le haut.



5) Remettre en place la vis du support des piles tout en enfonçant les piles.

6) Remettre en place le panneau arrière de l'appareil et les deux vis pour le fixer.

* Le retrait simultané de l'alimentation principale et de l'alimentation de protection entraîne l'effacement des programmes et des données sauvegardés dans la mémoire.
* Toujours enlever les piles mortes de l'appareil pour éviter les dégâts dus à une fuite d'électrolyte.
* Toujours remplacer les deux piles principales.
* Toujours s'assurer que la polarité des piles (+/−) est correcte.
* Ne jamais se débarrasser des piles en les brûlant. Car le fait de les exposer à des températures très élevées peut les faire EXPLOSER.
* Bouton P
  Bouton de réinitialisation du matériel destiné à arrêter un mauvais fonctionnement dû à l'électricité statique. Toujours couper l'alimentation, puis la remettre lorsque le bouton P est enfoncé.

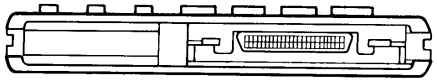
* ***Garder les piles hors d'atteinte d'enfants. Faire immédiatement appel à un docteur si une pile est avalée par inadvertance.***

1-8 COUPURE AUTOMATIQUE DE L'ALIMENTATION

L'alimentation de l'appareil est automatiquement coupée environ six minutes après la dernière opération de touche (sauf pendant l'exécution de programmes) ou la dernière entrée d'une instruction INPUT ou d'une instruction PRINT. L'alimentation peut être rétablie en mettant l'interrupteur d'alimentation sur la position "OFF" et ensuite sur la position "ON" ou en appuyant sur la touche [BRK].

* Le contenu des programmes et des données est retenu, même lorsque l'interrupteur d'alimentation est mis sur la position "OFF", mais les réglages tels que le nombre de chiffres ou le mode (p.ex., mode BASIC, mode MEMO IN) sont annulés.

1-9 CONNECTEUR



Le connecteur noté dans l'illustration est équipé pour la connexion de dispositifs périphériques (p.ex., FA-6). Toujours recouvrir le connecteur avec le couvercle de connecteur lorsque les dispositifs périphériques ne sont pas connectés.

PARTIE 2

OPERATIONS FONDAMENTALES

Ce chapitre couvre les divers modes disponibles avec le FX-850P en utilisant une série d'exemples simples. Ces procédures doivent être maîtrisées avant d'essayer des opérations plus complexes.

2-1 MODE CAL

Le mode CAL est actif chaque fois que l'appareil est mis sous tension. Les calculs arithmétiques, les calculs de fonction, l'exécution de la bibliothèque scientifique, les calculs de mise en mémoire de formules, l'exécution de programmes et le rappel de données peuvent être effectués dans ce mode.

EXEMPLE:

2,5 + 3,5 − 2 =

OPERATION:

2 [.] 5 [+] 3 [.] 5 [−] 2 [EXE]

```
2.5+3.5-2
  4
```

Bien que la touche [EXE] soit utilisée à la place de la touche [=], l'opération est identique à celle utilisée dans une calculatrice normale.
Le mode CAL peut être entré à partir d'un autre mode en appuyant sur [MODE] [0]. Voir la PARTIE 3 "FONCTION DE CALCULS" (page 15) pour les détails.

2-2 MODE BASIC

Le mode BASIC est utilisé pour la création, l'exécution et l'édition de programmes en BASIC. Le mode BASIC peut être entré à partir d'un autre mode en appuyant sur [MODE] [1].

EXEMPLE:

Créer et exécuter un programme qui calcule la somme de deux valeurs A et B.

ENTREE DE PROGRAMME

[MODE] 1

[SHIFT] [P0]

10 [A] [=] 5 [EXE]

20 [B] [=] 6 [EXE]

30 [P] [R] [I] [N] [T] [A] [+] [B] [EXE]

40 [E] [N] [D] [EXE]

EXECUTION DE PROGRAMME

[R] [U] [N] [EXE]

```
RUN
 11
```

Voir la PARTIE 6 "PROGRAMMATION EN BASIC" (page 46) pour les détails sur l'utilisation du langage BASIC.

2-3 FONCTION DE MISE EN MEMOIRE DE FORMULES

La fonction de mise en mémoire de formules permet de sauvegarder dans la mémoire des formules souvent utilisées pour des calculs lorsque des valeurs sont affectées aux variables. Cette fonction est appliquée dans le mode CAL en utilisant les touches [IN], [OUT] et [CALC].

EXEMPLE:

Déterminer le prix de vente d'un produit en appliquant un taux de profit basé sur le prix d'achat et le prix de vente.

SELLING PRICE (PRIX DE VENTE) = PURCHASE PRICE (PRIX D'ACHAT) ÷ (1 − PROFIT% (% DE PROFIT))

ENTREE DE TOUCHE

[MODE] [0]

[S] [E] [L] [L] [=] [P] [U] [R] [C] [H] [A] [S] [E] [/] [(] 1 [−] [P] [R] [O] [F] [I] [T] [)] [IN]

Nécessaire pour sauvegarder la formule dans la mémoire.

S'assurer que l'entrée de la formule est correcte en appuyant sur la touche [OUT].

OPERATION:

[OUT]

```
SELL=PURCHASE/(1-PROFIT)
SELL=PURCHASE/(1-PROFIT)_
```


Calculer maintenant les prix de vente suivants:

PRIX D'ACHAT	PROFIT
$1.000	30%
$960	25%

[CALC]

```
PURCHASE?_
```

1000 [EXE]

```
PURCHASE?1000
PROFIT?_
```

0 [.] 3 [EXE]

```
PROFIT?0.3
SELL= 1428.571429
```

[CALC]

```
PURCHASE?_
```

960 [EXE]

```
PURCHASE?960
PROFIT?_
```

[.] 25 [EXE]

```
PROFIT?.25
SELL= 1280
```

Comme on peut le voir dans cette exemple, une fois que la formule est entrée, elle peut être utilisée de manière répétitive en affectant simplement des valeurs aux variables. Voir la PARTIE 4 "FONCTION DE MISE EN MEMOIRE DE FORMULES "(page 33) pour les détails.

* La touche [BRK] peut être utilisée pour terminer cette fonction.

2-4 FONCTION DE LA BANQUE DE DONNEES (MEMO IN)

La fonction de la BANQUE DE DONNEES permet la mise en mémoire de larges volumes de données pour les rappeler si nécessaire. Une pression sur [MODE] [9] permet de passer dans le mode MEMO IN et de faire apparaître le curseur dans le coin supérieur gauche de l'affichage, prêt pour l'entrée de données.

EXEMPLE:

J. SMITH 347-237-4811

[MODE] [9]

[S] [M] [I] [T] [H] [SPC] [J] [.] [SPC] 347 [−] 237 [−] 4811 [EXE]

```
CAPS      DEG      MEMO IN
SMITH J.  347-237-4811
```

Une fois les données sauvegardées, les articles de données spécifiques peuvent être rappelés en utilisant la touche [MEMO].

[MODE] 0

```
-
```

(Mode CAL)

[S] [MEMO]

```
SMITH J.  347-237-4811
```

(Recherche de l'article de données commençant par "S".)

La touche [BRK] peut être utilisée pour terminer la fonction de la BANQUE DE DONNEES. Pour les détails sur la mise en mémoire et le rappel de données, voir la PARTIE 5 "FONCTION DE LA BANQUE DE DONNEES" (page 37).

2-5 BIBLIOTHEQUE SCIENTIFIQUE INCORPOREE

Cette fonction assure une large variété de bibliothèque scientifique pratique pouvant être rappelée et utilisée dans des calculs dans le mode CAL.
Opérations mathématiques/statistiques — 116 types
Les opérations sont sélectionnées en utilisant les touches [MENU], [⇧], [⇩], [EXE] et [LIB].
* Voir la PARTIE 11 "BIBLIOTHEQUE SCIENTIFIQUE" (page 181) pour les détails.

2-6 RESUME

Tableau de fonctions

Nom de la fonction	Opération de touche
Mode CAL	[MODE] [0]
Mode BASIC	[MODE] [1]
BANQUE DE DONNEES	[MODE] [9]
Mise en mémoire de formules	[MODE] [0] + [IN], [OUT], [CALC]
Bibliothèque scientifique incorporée	[MODE] [0] + N° de bibliothèque [LIB]

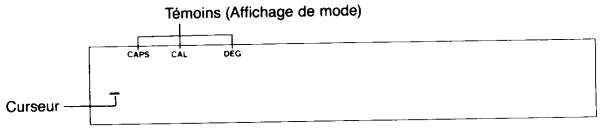
PARTIE 3

FONCTION DE CALCULS

Ce chapitre couvre les calculs arithmétiques et les calculs de fonctions fondamentaux qui sont effectués manuellement.

3-1 PREPARATIONS AUX CALCULS MANUELS

Mettre l'appareil sous tension (ON)



L'affichage illustré ci-dessus apparaît à chaque mise sous tension. Il indique le mode CAL dans lequel les calculs manuels peuvent être effectués. Toutefois, l'unité d'angle actuellement spécifiée est retenue, même lorsque l'interrupteur d'alimentation est mis sur la position "OFF".

3-2 ENTREE ET CORRECTION DE CALCULS MANUELS

Effectuer les calculs fondamentaux suivants pour se familariser avec ce mode.

EXEMPLE:

123 + 456 = 579

123 [+] 456	123+456_	(Entrée de formule)
[EXE]	123+456 579	(Donne le résultat.)

Comme on peut le voir ici, la touche [EXE] est enfoncée à la place de [=]. La touche [*] est utilisée pour la multiplication et la touche [/] pour la division.

La procédure suivante peut être utilisée pour corriger des données entrées.

EXEMPLE:

33 × 5 + 16 = 181

Pour servir d'exemple, la valeur 33 est ici entrée par erreur avec 34.

34 [*] 5 [+] 16 | 34*5+16_

Appuyer six fois sur [⇦] pour ramener le curseur sur le "4". Ceci peut également être accompli avec [SHIFT] [LTOP] [⇨] .

[⇦] [⇦] [⇦] [⇦] [⇦] [⇦] | 34*5+16

3 | 33*5+16 (Remplace 4 par 3.)

[EXE] | 33*5+16
181

EXEMPLE:

33 × 5 + 16 = 181

Pour servir d'exemple, le calcul ci-dessus est effectué avec la valeur 33 entrée par erreur à la place de 34.

34 [*] 5 [+] 16 [EXE] | 34*5+16
186

[⇧] [⇦] [⇦] [⇦] | 34*5+16
186 (Déplacer le curseur à la position pour la correction.)

3 | 33*5+16
186

[EXE] | 33*5+16
181 (Réexécuter le calcul.)

EXEMPLE:

33 × 5 + 16 = 181

Pour servir d'exemple, le signe de la multiplication (*) est omis par erreur et le calcul est effectué.

335 [+] 16 [EXE] | 335+16
351

[⇧] [⇦] [⇦] | 335+16
351 (Déplacer le curseur à la position pour insérer le *.)

[INS] | 33_5+16
351 (Ouvrir un espace d'un caractère.)

[*] | 33*5+16
351 (Entrer le symbole de la multiplication.)

[EXE] | 33*5+16
181 (Obtenir une nouvelle fois le résultat.)

Comme on peut le voir dans l'exemple mentionné ci-dessus, la touche [INS] est utilisée pour insérer des espaces à la position actuelle du curseur pour l'entrée de caractères ou de symboles.

EXEMPLE:

33 × 5 + 16 = 181

Pour servir d'exemple, le calcul ci-dessus est effectué avec la valeur 16 entrée par erreur à la place de 216.

33 [*] 5 [+] 216 [EXE]

33*5+216
381

[⇧] [⇨] | 33*5+216
381 (Ramener le curseur à la position de 2.)

[SHIFT] [DEL] | 33*5+16
381 (Effacer 2.)

[EXE] | 33*5+16
181 (Réexécuter le calcul.)

Comme on peut le voir dans l'exemple mentionné ci-dessus, la touche [DEL] est utilisée pour effacer des caractères à la position actuelle du curseur.

- La touche [BS] peut également être utilisée pour effacer des caractères, mais son fonctionnement est légèrement différent de [SHIFT] [DEL] .

[SHIFT] [DEL] ..... Efface le caractère à la position actuelle du curseur.

ABCDEFGH → [SHIFT] [DEL] → ABCEFGH

[BS] ..... Efface le caractère à la position située à la gauche de la position actuelle du curseur.

ABCDEFGH → [BS] → ABDEFGH

Pratiquer les exemples suivants pour se familariser avec la procédure de calculs fondamentaux.

EXEMPLE 1:

9 + 7,8 ÷ 6 − 3,5 × 2 = 3,3

OPERATION:

9 [+] 7.8 [/] 6 [−] 3.5 [*] 2 [EXE]

9+7.8/6-3.5*2
3.3

EXEMPLE 2:

56 × (−12) ÷ (−2,5) = 268,8

OPERATION:

56 [*] [−] 12 [/] [−] 2.5 [EXE]

```
56*-12/-2.5
 268.8
```

Des valeurs négatives sont entrées en appuyant sur la touche [−] avant d'entrer la valeur.

EXEMPLE 3:

$(4{,}5 \times 10^{75}) \times (-2{,}3 \times 10^{-78}) = -0{,}01035$

OPERATION:

4.5 [E] 75 [*] [−] 2.3 [E] [−] 78 [EXE]

```
4.5E75*-2.3E-78
-0.01035
```

Les exposant sont entrés en appuyant sur la touche [E] (ou la touche alphabétique [E]) avant d'entrer la valeur.

L'exemple suivant indique comment le résultat d'un calcul peut être immédiatement intégré dans un calcul suivant.

EXEMPLE 4:

$(23 + 456) \times 567 = 271593$

OPERATION:

23 [+] 456 [EXE]

```
23+456
 479
```

en continuant par

[*] 567 [EXE]

```
479*567
271593
```

Le dernier résultat obtenu peut être entré à tout moment dans un calcul suivant en appuyant sur la touche [ANS].

EXEMPLE 5:

$81{,}2 \div (5{,}6 + 8{,}9) = 5{,}6$

Ce calcul est effectué en premier.

OPERATION:

5.6 [+] 8.9 [EXE]

```
5.6+8.9
 14.5
```

en continuant par

81.2 [/] [ANS]

```
 14.5
81.2/ 14.5_
```

[EXE]

```
81.2/ 14.5
 5.6
```

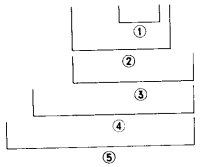

3-3 ORDRE DE PRIORITE

Les opérations arithmétiques, de relation et logiques sont effectuées dans l'ordre de priorité suivant:

1. (,)
2. Fonctions
3. Puissance
4. Signes (+, −)
5. ∗, /, ¥, MOD
6. +, −
7. Opérateurs de relation
8. NOT
9. AND
10. OR, XOR

EXEMPLE:

2 + 4 ∗ COS (14 + 16)^2 = 5



NOTE:

a. Les calculs sont effectués de gauche à droite lorsque l'ordre de priorité est identique.
b. Les fonctions complexes (sin, cos 60) sont effectuées de droite à gauche.
c. Les puissances consécutives (5^4^3) sont effectuées de gauche à droite.

Les résultats des calculs sont affichés de la manière suivante:

1. Nombre entier inférieur à 1×10^{10}:	Nombre entier
2. 10 chiffres ou moins dans la partie fractionnaire:	Décimale
3. Autres:	Exponentielle

Nombre de chiffres

- Les calculs internes sont effectués avec une mantisse de 12 chiffres + un exposant de 2 chiffres. Cependant le nombre PI est exprimé avec 11 chiffres (3,1415926536): arrondi et affiché en 10 chiffres (3,141592654).
- Les résultats des calculs sont affichés arrondis à une mantisse de 10 chiffres + un exposant de 2 chiffres.
- La capacité d'entrée maximum pour une seule ligne est de 255 caractères.

Spécification du nombre de chiffres significatifs et du nombre de décimales

"SET" est utilisé pour ces spécifications.

Spécification du nombre de décimales ..... [SHIFT] [SET] [F] (0 ~ 9) [EXE]
Spécification du nombre de chiffres significatifs ..... [SHIFT] [SET] [E] (0 ~ 9) [EXE]

* "SET E0" utilisé pour spécifier le nombre de chiffres significatifs spécifié à 10 chiffres.

Libération de spécification ..... [SHIFT] [SET] [N] [EXE]

* Lorsqu'une spécification est faite, le résultat est affiché par le nombre de chiffres spécifiés. (Le chiffre situé après le dernier chiffre spécifié est arrondi.) La valeur initiale reste dans l'ordinateur.

EXEMPLE:

Nombre spécifié de décimales: 2

OPERATION:

[SHIFT] [SET] [F] **2** [EXE]

```
SET F2
_
```

10 [/] **3** [EXE]

```
10/3
 3.33
```

EXEMPLE:

Nombre spécifié de chiffres significatifs: 3

OPERATION:

[SHIFT] [SET] [E] **3** [EXE]

```
SET E3
_
```

12 [*] **34** [EXE]

```
12*34
 4.08E+02
```

3-4 CALCULS SCIENTIFIQUES

Les fonctions scientifiques (voir le tableau des fonctions scientifiques de la page 28) peuvent être utilisées aussi bien à l'intérieur des programmes que pour des calculs manuels. Pour notre explication, tous les exemples donnés ici ne couvrent que des calculs manuels.

Fonctions trigonométriques et trigonométriques inverses

sin:	sinus	$\sin^{-1}$:	arc sinus
cos:	cosinus	$\cos^{-1}$:	arc cosinus
tan:	tangente	$\tan^{-1}$:	arc tangente

Ces fonctions donnent la valeur d'une fonction trigonométrique pour un angle donné ou bien la valeur d'un angle pour la valeur d'une fonction trigonométrique donnée. La commande ANGLE doit être utilisée pour spécifier l'unité de valeur d'angle lorsque ces fonctions sont utilisées. La spécification de l'unité d'angle n'est nécessaire qu'une fois pour toutes les fonctions trigonométriques et trigonométriques inverses qui viennent ensuite. Les unités d'angle peuvent être spécifiées en utilisant la touche [MODE] ou la commande ANGLE.

ANGLE 0..... DEG (degrés) → ([MODE] [4])
ANGLE 1..... RAD (radians) → ([MODE] [5])
ANGLE 2..... GRAD (grades) → ([MODE] [6])

La relation entre ces trois spécifications est donnée ci-après:

$$90 \text{ degrés} = \frac{\pi}{2} \text{ radians} = 100 \text{ grades}$$

L'unité d'angle actuelle est retenue lorsque l'interrupteur d'alimentation est mis sur la position "OFF" et l'unité d'angle devient ANGLE 0 (DEG) lorsque le bouton ALL RESET est enfoncé.

La valeur de π peut être directement entrée dans une formule qui utilise le nombre "PI" (3,141592654).

EXEMPLE 1:

$\sin 30° = 0{,}5$

OPERATION:

[SHIFT] [SET] [N] [EXE]

[MODE] **4**

```
CAPS CAL DEG
_
```

[sin] (ou [S] [I] [N]) **30** [EXE]

```
CAPS CAL DEG
SIN30
 0.5
```

EXEMPLE 2:

$\cos \frac{\pi}{3} = 0{,}5$

OPERATION:

[MODE] **5**

```
CAPS CAL RAD
_
```

[cos] (ou [C] [O] [S])
[(] [SHIFT] [π] [/] **3** [)] [EXE]

```
CAPS CAL RAD
COS(PI/3)
 0.5
```

[SHIFT] [π] peuvent également être entrés par [P] [I] .

EXEMPLE 3:

$2\sin \frac{\pi}{3} + \cos \frac{\pi}{3} = 2{,}232050808$

OPERATION:

[MODE] **5**

```
CAPS  CAL   RAD
-
```

2 [*] [sin] [(] [SHIFT] [π] [/] **3** [)] [+] [cos] [(] [SHIFT] [π] [/] **3** [)] [EXE]

```
CAPS  CAL   RAD
2*SIN(PI/3)+COS(PI/3)
 2.232050808
```

EXEMPLE 4:

tan 60° = 1,732050808

OPERATION:

[MODE] **4**

```
CAPS  CAL   DEG
-
```

[tan] (ou [T] [A] [N]) **60** [EXE]

```
CAPS  CAL   DEG
TAN60
 1.732050808
```

EXEMPLE 5:

$\sin^{-1} 0{,}5 = 30°$

OPERATION:

[SHIFT] [sin⁻¹] **0.5** [EXE]

```
CAPS  CAL   DEG
ASN0.5
 30
```

EXEMPLE 6:

$\cos^{-1} \frac{2^{0,5}}{2} = 45°$

OPERATION:

[SHIFT] [cos⁻¹] [(] **2** [^] **0.5** [/] **2** [)] [EXE]

```
CAPS  CAL   DEG
ACS(2^0.5/2)
 45
```


EXEMPLE 7:

$\tan^{-1}\sqrt{3} = 60° = 1{,}047197551$

OPERATION:

[SHIFT] [tan⁻¹] [√] **3** [EXE]

```
CAPS  CAL   DEG
ATNSQR3
 60
```

[MODE] **5**

```
CAPS  CAL   RAD
ATNSQR3
 60
```

[SHIFT] [tan⁻¹] [√] **3** [EXE]

```
CAPS  CAL   RAD
ATNSQR3
 1.047197551
```

Fonctions hyperboliques et hyperboliques inverses

sinh:	sinus hyperbolique	$\sinh^{-1}$:	arc sinus hyperbolique
cosh:	cosinus hyperbolique	$\cosh^{-1}$:	arc cosinus hyperbolique
tanh:	tangente hyperbolique	$\tanh^{-1}$:	arc tangente hyperbolique

EXEMPLE 1:

sinh 5 = 74,20321058

OPERATION:

[hyp] [sin] **5** [EXE]

```
HYPSIN5
 74.20321058
```

(La fonction HYPSIN est utilisée pour sinh.)

EXEMPLE 2:

$\cosh^{-1} 1{,}5 = 0{,}9624236501$

OPERATION:

[hyp] [SHIFT] [cos⁻¹] **1.5** [EXE]

```
HYPACS1.5
 0.9624236501
```

(La fonction HYPACS est utilisée pour $\cosh^{-1}$.)

Fonctions logarithmiques, fonctions exponentielles

$\log_{10}$:	logarithme décimal	e^x:	exponentielle
$\log_e$:	logarithme népérien		

EXEMPLE 1:

$\log_e 123 = 4{,}812184355$

OPERATION:

[ln] **123** [EXE]

```
LN123
 4.812184355
```

(La fonction LN est utilisée pour $\log_e$.)

EXEMPLE 2:

$\log_{10} 100 = 2$

OPERATION:

[log] **100** [EXE]

```
LOG100
 2
```

(La fonction LOG est utilisée pour $\log_{10}$.)

EXEMPLE 3:

$e^5 = 148{,}4131591$

OPERATION:

[SHIFT] [e^x] **5** [EXE]

```
EXP5
 148.4131591
```

(La fonction EXP est utilisée pour e^x.)

Autres fonctions

SGN:	Signe	RAN#:	Nombre aléatoire	ABS:	Valeur absolue
INT:	Valeur entière	FIX:	Partie entière	FRAC:	Fraction
ROUND:	Arrondissage				

• SGN

Pour SGN (x), donne un 1 lorsque x>0, un −1 lorsque x<0 et un 0 lorsque x = 0.

OPERATION:

SGN6 [EXE]

```
SGN6
 1
```

SGN − 2 [EXE]

```
SGN-2
-1
```

• RAN#

Produit un nombre aléatoire entre 0 et 1 avec un maximum de 10 décimales. Pour les détails, voir la PARTIE 10 "REFERENCE DE COMMANDES".

OPERATION:

[SHIFT] [RAN#] [(] [−] **1** [)] [EXE]

```
RAN#(-1)
 0.2466393388
```

* La valeur mentionnée ci-dessus n'est qu'une valeur d'exemple.

• ABS

Donne la valeur absolue de x pour ABS (x).

$|78{,}9 \div -5{,}6| = 14{,}08928571$

OPERATION:

ABS [(] **78.9** [/] [−] **5.6** [)] [EXE]

```
ABS(78.9/-5.6)
 14.08928571
```


• INT

Pour INT (x), donne le nombre entier le plus grand ne dépassant pas la valeur de x.

OPERATION:

[SHIFT] [INT] [−] **64.5** [EXE]

```
INT-64.5
-65
```

• FIX

Donne la partie entière de x pour FIX (x).

Partie entière de 8000 ÷ 96.

OPERATION:

FIX [(] **8000** [/] **96** [)] [EXE]

```
FIX(8000/96)
 83
```

• FRAC

Donne la partie fractionnaire de x pour FRAC (x).

Partie fractionnaire de 8000 ÷ 96.

OPERATION:

FRAC [(] **8000** [/] **96** [)] [EXE]

```
FRAC(8000/96)
 0.3333333333
```

• ROUND

La fonction ROUND (X, −Y) donne le résultat de X à la Y^ème^ décimale (résultant en Y − 1 décimales).
Arrondir le résultat de 8000/96 à trois décimales.

OPERATION:

ROUND [(] **8000** [/] **96** [,] [−] **4** [)] [EXE]

```
ROUND(8000/96,-4)
 83.333
```

Conversions des systèmes décimal et sexagésimal

DEG:	Sexagésimal en décimal
DMS$:	Décimal en sexagésimal

EXEMPLE 1:

12°34′56″ = 12,58222222°

OPERATION:

[DEG] **12** [,] **34** [,] **56** [)] [EXE]

```
DEG(12.34.56)
 12.58222222
```

EXEMPLE 2:

12,3456° = 12°20′44,16″

OPERATION:

[SHIFT] [DMS(] **12.3456** [)] [EXE]

```
DMS$(12.3456)
 12°20'44.16
```

Conversions des systèmes décimal en hexadecimal

&H: Hexadécimal en décimal
HEX$: Décimal en hexadecimal

EXEMPLE 1:

$10_{(16)} = 16_{(10)}$

OPERATION:

[&H] **10** [EXE]

```
&H10
 16
```

EXEMPLE 2:

$1000_{(10)} = 3E8_{(16)}$

OPERATION:

[SHIFT] [HEX$(] **1000** [)] [EXE]

```
HEX$(1000)
03E8
```

* Les valeurs hexadécimales A, B, C, D, E, F corespondent aux valeurs décimales 10, 11, 12 13, 14, 15.

FACT, NPR, NCR

Ces fonctions donnent la factorielle, permutation et combinaison des valeurs entrées.

EXEMPLE 1:

10! = 3628800

OPERATION:

[SHIFT] [FACT] **10** [EXE]

```
FACT10
 3628800
```


EXEMPLE 2:

$_{10}P_4 = 5040$

OPERATION:

[SHIFT] [nPr(] 10 [,] **4** [)] [EXE]

```
NPR(10.4)
 5040
```

EXEMPLE 3:

$_{10}C_4 = 210$

OPERATION:

[SHIFT] [nCr(] **10** [,] **4** [)] [EXE]

```
NCR(10.4)
 210
```

REC ↔ POL

Convertir des coordonnées rectangulaires en coordonnées polaires et inversement.

EXEMPLE 1:

Convertir les coordonnées polaires $(5, \frac{\pi}{6})$ en coordonnées rectangulaires (X, Y).

OPERATION:

[MODE] **5** (Unité d'angle = RAD)

[SHIFT] [REC(] **5** [,] [SHIFT] [π] [/] **6** [)] [EXE]

```
REC(5.PI/6)
 4.330127019
```

(Coordonnée X)

[Y] [EXE]

```
Y
 2.5
```

(Coordonnée Y)

EXEMPLE 2:

Convertir les coordonnées rectangulaires (1, 1) en coordonnées polaires (r, θ).

OPERATION:

[SHIFT] [POL(] **1** [,] **1** [)] [EXE]

```
POL(1.1)
 1.414213562
```

(Coordonnée r)

[Y] [EXE]

```
Y
 0.7853981634
```

(Coordonnée θ)

Tableau des fonctions scientifiques

Nom de la fonction	Formule	Format	Détails
Trigonométrique	sin	SIN (expression numérique)	$-1440° <$ expression numérique $< 1440°$ (8πrad, 1600grad)
	cos	COS (expression numérique)	$-1440° <$ expression numérique $< 1440°$ (8πrad, 1600grad)
	tan	TAN (expression numérique)	$-1440° <$ expression numérique $< 1440°$ (8πrad, 1600grad) * A l'exclusion des cas où l'expression numérique est $(2n\text{-}1) \times 90°$ (π/2rad, 100grad) * n : nombre entier
Trigonométrique inverse	$\sin^{-1}$	ASN (expression numérique)	|expression numérique| $\leqq 1$, $-90° \leqq$ ASN (expression numérique) $\leqq 90°$
	$\cos^{-1}$	ACS (expression numérique)	|expression numérique| $\leqq 1$, $0° \leqq$ ACS (expression numérique) $\leqq 180°$
	$\tan^{-1}$	ATN (expression numérique)	|expression numérique| $< 10^{100}$, $-90° \leqq$ ATN (expression numérique) $\leqq 90°$
Hyperbolique	sinh	HYPSIN (expression numérique)	|expression numérique| $\leqq 230{,}2585092$
	cosh	HYPCOS (expression numérique)	|expression numérique| $\leqq 230{,}2585092$
	tanh	HYPTAN (expression numérique)	|expression numérique| $\leqq 10^{100}$, $-1 \leqq$ HYPTAN (expression numérique) $\leqq 1$
Hyperbolique inverse	$\sinh^{-1}$	HYPASN (expression numérique)	|expression numérique| $< 5 \times 10^{99}$
	$\cosh^{-1}$	HYPACS (expression numérique)	$1 \leqq$ expression numérique $< 5 \times 10^{99}$
	$\tanh^{-1}$	HYPATN (expression numérique)	|expression numérique| < 1
Exponentielle	e^x	EXP (expression numérique)	$-227 \leqq$ expression numérique $\leqq 230{,}2585092$
Logarithme népérien	$\log_e x$	LN (expression numérique)	expression numérique > 0
Logarithme décimal	$\log_{10} x$	LOG (expression numérique)	expression numérique > 0
Racine carrée	$\sqrt{x}$	SQR (expression numérique)	expression numérique $\geqq 0$
Racine cubique	$\sqrt[3]{x}$	CUR (expression numérique)	expression numérique $\geqq 0$
Valeur absolue	|x|	ABS (expression numérique)	Donne la valeur absolue de l'expression numérique
Signe		SGN (expression numérique)	expression numérique < 0 : -1 expression numérique $= 0$: 0 expression numérique > 0 : 1
Nombre entier		INT (expression numérique)	Fonction de Gauss: Donne la valeur entière maximum ne dépassant pas la valeur de l'expression numérique.
Fraction		FRAC (expression numérique)	Donne la partie fractionnaire de l'expression numérique.

Nom de la fonction	Formule	Format	Détails
Arrondissage		ROUND (x, y) x, y : expression numérique	Arrondit x à la position spécifiée par y.
Fix		FIX (expression numérique)	Donne la partie entière de x.
Degré	Sexagésimal → Décimal	DEG (d [,m[,s]]) d, m, s : expression numérique	Convertit une valeur sexagésimale en valeur décimale.
PI	π	PI	3,141592654
Nombre aléatoire		RAN # (expression numérique)	Donne un nombre aléatoire avec 10 décimales. $0 <$ RAN # < 1
Factorielle	x !	FACT (expression numérique)	nombre entier : $0 \leqq$ expression numérique $\leqq 69$
Permutation	nPr	NPR (n, r)	nombre entier : $0 \leqq r \leqq n \leqq 10^{10}$
Combinaison	nCr	NCR (n, r)	nombre entier : $0 \leqq r \leqq n \leqq 10^{10}$
Conversions de coordonnées		POL (x, y)	Convertit les coordonnées rectangulaires spécifiées par |x| + |y| > 0 en coordonnées polaires.
		REC (r, θ)	Convertit les coordonnées polaires spécifiées par $0 \leqq r < 10^{100}$ en coordonnées rectangulaires.

* A l'exception de ROUND, DEG, NPR, NCR, POL et REC, toutes les valeurs utilisées avec ces fonctions n'ont pas besoin d'être comprises entre parenthèses.

3-5 CALCULS UTILISANT DES VARIABLES

Les calculs algébriques peuvent également être effectués en utilisant des variables. La liste de calculs suivante, par exemple, devient beaucoup plus facile à effectuer si une variable est assignée pour le terme commun.

$2 \times 3{,}1415 + 5 =$
$3 \times 3{,}1415 + 6 =$
$4 \times 3{,}1415 + 7 =$
$5 \times 3{,}1415 + 8 =$

1. Tout d'abord, assigner la valeur 3,1415 à la variable X.

X [=] 3.1415 [EXE]

```
X=3.1415
_
```

2. Puis, utiliser la variable à la place de la valeur pour chacun des calculs.

2 [×] X [+] 5 [EXE]

```
2*X+5
 11.283
```

3 [×] X [+] 6 [EXE]

```
3*X+6
 15.4245
```

Variables

Les règles suivantes s'appliquent aux noms de variables de tous les types de variables utilisées avec l'appareil.

Noms de variables:

1. Ce sont des chaînes de caractères ayant un caractère alphabétique majuscule (A ~ Z, code décimal interne 65 ~ 90) ou un caractère alphabétique minuscule (a ~ z, code décimal interne 97 ~ 122) dans la position de tête (la première). (Voir le tableau des codes de caractères à la page 421 pour les codes internes.)
2. Ils sont composés de caractères alphabétiques majuscules ou minuscules ou de chiffres (0 ~ 9, code décimal interne 48 ~ 57) après le caractère alphabétique de tête.
3. Ne peuvent pas utiliser de mots réservés (voir page 426) comme caractères de tête.
4. Peuvent avoir une longueur maximum de 15 caractères.

3-6 AUTRES CALCULS

En plus des calculs arithmétiques fondamentaux de l'addition, la soustraction, la multiplication et la division et des calculs exponentiels, le FX-850P est également capable d'utiliser une foule d'autres opérateurs arithmétiques et de relation.

Opérateurs arithmétiques

Les opérateurs arithmétiques suivants sont utilisés dans les formules:

Signes	(+, −)
Addition	(+)
Soustraction	(−)
Multiplication	(∗)
Division	(/)
Puissance	(∧)
Division de nombre entier	(¥)
Reste de division de nombre entier	(MOD)

Les valeurs utilisées avec les opérateurs ¥ et MOD sont limitées à la gamme de −32768 à 32767 et la partie fractionnaire des valeurs non entières est tronquée.

EXEMPLE:

5¥2.9 = 2 (5÷2.9 = 2 .5)
7.8 MOD2.7 = 1* (7÷2 = 3····· 1)

(Les parties fractionnaires barrées par un "x" sont tronquées avant que le calcul ne soit effectué.)

* Lorsqu'une variable est utilisée à la place de 7,3, un espace est nécessaire entre la variable et l'opérateur MOD.

Avec les deux opérateurs ¥ et MOD, les valeurs sont converties en leurs valeurs absolues avant d'effectuer la division. Le signe attribué au résultat de l'opération ¥ suit les règles de la division normale, alors que le signe attribué au résultat de l'opération MOD est le signe du dividende.

EXEMPLE:

−15¥7 = −2 −15÷7 = −2 ····· −1
−15MOD7 = −1 −15¥7 −15MOD7

Opérateurs logiques

L'applicaton des opérateurs logiques est similaire à celle des opérateurs arithmétiques. Les parties fractionnaires des données sont tronquées et l'opération logique spécifiée est effectuée bit par bit (chaque bit du résultat est obtenu en examinant le bit dans la même position pour chaque argument). Il y a quatre différents opérateurs logiques disponibles avec l'appareil:

NOT Fait qu'une expression n'est pas vraie.
AND L'expression est vraie si les deux parties le sont, autrement l'expression est fausse.
OR L'expression est vraie si l'une des deux parties est vraie, autrement l'expression est fausse.
XOR L'expression est fausse si l'une des deux parties est vraie ou fausse, l'expression est vraie si une partie est vraie et une partie est fausse.

Négation

X	NOT X
0	1
1	0

Produit logique

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

Somme logique

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

OR exclusif

X	Y	X XOR Y
0	0	0
0	1	1
1	0	1
1	1	0

EXEMPLE:

Déterminer la somme logique de 10 et 3.

10 [O] [R] 3 [EXE]

```
10OR3
 11
```

10 = 1010(2)
3 = 0011(2)
11 = 1011(2)

* 1010(2) représente la valeur binaire de 10.

Opérateur de caractères

Le seul opérateur de chaîne de caractères disponible est l'opérateur "plus" (+). La longueur du résultat est limitée à 255 caractères.

EXEMPLE:

"A" + "B" → "AB"

Opérateurs de relation

Les opérateurs suivants peuvent être utilisés à l'intérieur de programmes (uniquement) pour comparer deux valeurs ou deux chaînes. Un résultat vrai donne une valeur de – 1, alors qu'un résultat faux donne 0.

Egal à	=
Différent de	< >, > <
Plus petit que	<
Plus grand que	>
Plus petit ou égal à	= <, < =
Plus grand ou égal à	= >, > =

Avec les comparaisons des chaînes de caractères, chaque caractère situé dans la chaîne à la gauche de l'opérateur est comparé à chaque caractère situé à la position correspondante dans la chaîne à la droite de l'opérateur. Les comparaisons sont faites en utilisant le code de caractères pour chaque caractère. Si deux chaînes sont de longueur différente et que la chaîne plus courte est identique aux caractères de tête de la chaîne plus longue, la chaîne plus courte est considérée comme étant la moindre des deux.

EXEMPLE:

```
10 PRINT 125>12
20 PRINT "DEF"<"ABCD"
30 PRINT "ABCD"="ABC"
```

[MODE] 1 [R] [U] [N] [EXE]

```
RUN
-1
```

Etant donné que 125 est en fait supérieur à 12, une valeur de – 1 (VRAI) est renvoyée.

[EXE]

```
-1
0
```

Le code de caractères de "DEF" est supérieur à celui de "ABCD", donc 0 (FAUX) est renvoyé.

[EXE]

```
0
0
```

La chaîne "ABCD" n'est pas égale à la chaîne "ABC", donc 0 (FAUX) est renvoyé.

* Les chaînes de caractères sont comparées jusqu'à ce qu'une différence soit trouvée et un jugement est rendu à la première différence rencontrée. Dans l'exemple mentionné ci-dessus, le "A" de la première position d'une chaîne diffère du "D" de la première position de l'autre chaîne, donc la comparaison est basée sur "A" et "D". Bien que la chaîne "ABCD" soit plus longue, la chaîne "DEF" est considérée comme étant la plus grande des deux, parce que le code de caractères de "D" est supérieur au code de caractères de "A".

PARTIE 4

FONCTION DE MISE EN MEMOIRE DE FORMULES

La fonction de mise en mémoire de formules est très utile lorsque l'on effectue des calculs à répétition. Trois touches différentes sont utilisées en cas de travail avec la fonction de mise en mémoire de formules.

Touche [IN] .....Met en mémoire la formule actuellement affichée.
Touche [OUT] .....Affiche la formule mise en mémoire.
Touche [CALC] .....Attribue des valeurs aux variables comprises dans la formule et affiche le résultat du calcul de la formule.

Application échantillon

EXEMPLE:

Obtenir une valeur de y pour chacune des valeurs attribuées à x lorsque y = 3,43 cos x. (Calculer avec trois décimales.)

x	8°	15°	22°	27°	31°
y					

OPERATION:

Spécifier tout d'abord l'unité d'angle et le nombre de décimales.

[MODE] 0 [MODE] 4 (Unité d'angle: "DEG")
[SHIFT] [SET] [F] 3 [EXE] (Obtenir avec trois décimales en arrondissant la 4ème décimale.)

Entrer ensuite une formule et appuyer sur la touche [IN] pour la sauvegarder.

[Y] [=] 3 [.] 43 [*] [cos] [X] [IN]

Appuyer sur la touche [OUT] pour confirmer que la formule a été sauvegardée.

[SHIFT] [CLS]

```
-
```

[OUT]

```
Y=3.43*COSX_
```

Ensuite, commencer à calculer en appuyant sur la touche [CALC].

CALC	X?_
8 EXE	X?8 Y= 3.397
EXE	Y= 3.397 X?_
15 EXE	X?15 Y= 3.313
EXE	Y= 3.313 X?_
22 EXE	X?22 Y= 3.180
EXE	Y= 3.180 X?_
27 EXE	X?27 Y= 3.056
EXE	Y= 3.056 X?_
31 EXE	X?31 Y= 2.940
BRK	Y= 2.940 _

La touche CALC peut être utilisée à la place de la touche EXE pour effectuer des calculs à répétition.
* La touche BRK peut être utilisée pour terminer cette fonction afin de retourner automatiquement au mode CAL.

4-1 UTILISATION POUR LA PREPARATION DES TABLEAUX

Des formules multiples peuvent être utilisées en séparant par des deux-points (:). Des tableaux tels que ceux indiqués ci-dessous peuvent facilement être préparés en utilisant cette méthode.

EXEMPLE:

Compléter le tableau suivant. (Calculer avec trois décimales en arrondissant.)

X	Y	P=X∗Y	Q=X/Y
4,27	1,17		
8,17	6,48		
6,07	9,47		
2,71	4,36		
1,98	3,62		

OPERATION:

SHIFT SET F 3 EXE ..... Spécification du nombre de décimales
P = X ∗ Y : Q = X / Y IN ..... Mise en mémoire de la formule

CALC	X?_	(Le calcul commence.)
4 • 27 EXE	X?4.27 Y?_	(Valeur X)
1 • 17 EXE	Y?1.17 P= 4.996	(Valeur Y)
EXE	P= 4.996 Q= 3.650	
EXE	Q= 3.650 X?_	

⋮

Continuer à entrer les valeurs de X et Y de cette manière et les valeurs de P et de Q seront calculées dans l'ordre successif et le tableau sera achevé de la manière indiquée ci-dessous.

X	Y	P=X∗Y	Q=X/Y
4,27	1,17	4,996	3,650
8,17	6,48	52,942	1,261
6,07	9,47	57,483	0,641
2,71	4,36	11,816	0,622
1,98	3,62	7,168	0,547

Les noms de variables peuvent comprendre jusqu'à 15 caractères alphabétiques majuscules ou minuscules. Ceci signifie que des noms de variables peuvent être créés qui décrivent en fait leur contenu. Des remarques peuvent également être portées en affixe après les noms de variables en entourant les remarques de parenthèses carrées [ ]. Tout caractère à l'exception des virgules peut être utilisé à l'intérieur des parenthèses carrées.

EXEMPLE:

Compléter le tableau suivant. (Calculer avec deux décimales en arrondissant.)

Rayon r (m)	Hauteur h (m)	Volume d'un cylindre ($V_0 = \pi r^2 h$) (m^3)	Volume d'un cône ($V_1 = {}^1/_3\ V_0$) (m^3)
1,205	2,227		
2,174	3,451		
3,357	7,463		

OPERATION:

[SHIFT] [SET] [F] 2 [EXE]
[C] [Y] [L] [I] [N] [D] [E] [R] [SHIFT] [←] [M] 3 [SHIFT] [↓] [=] [SHIFT] [π] [✱] [R] [A] [D] [I] [U] [S] [SHIFT] [←] [M]
[SHIFT] [↓] [^] 2 [✱] [H] [E] [I] [G] [H] [T] [SHIFT] [←] [M] [SHIFT] [↓] [:] [C] [O] [N] [E] [SHIFT] [←] [M] 3 [SHIFT] [↓]
[=] [C] [Y] [L] [I] [N] [D] [E] [R] [/] 3 [IN]

[CALC]	RADIUS[M]?_	(Le calcul commence.)
1 [•] 205 [EXE]	RADIUS[M]?1.205 HEIGHT[M]?	(Rayon)
2 [•] 227 [EXE]	HEIGHT[M]?2.227 CYLINDER[M3]= 10.16	(Hauteur)
[EXE]	CYLINDER[M3]= 10.16 CONE[M3]= 3.39	
[EXE]	CONE[M3]= 3.39 RADIUS[M3]?_	
2 [•] 174 [EXE]	RADIUS[M2]?2.174 HEIGHT[M]?_	(Rayon)

⋮

Si les valeurs des rayon (r) et hauteur (h) sont entrées de cette manière, le volume (V_0) du cylindre et le volume (V_1) du cône seront calculées successivement et le tableau sera achevé de la manière indiquée ci-dessous.

Rayon r (m)	Hauteur h (m)	Volume d'un cylindre ($V_0 = \pi r^2 h$) (m^3)	Volume d'un cône ($V_1 = {}^1/_3\ V_0$) (m^3)
1,205	2,227	10,16	3,39
2,174	3,451	51,24	17,08
3,357	7,463	264,22	88,07

IMPORTANT

1. Jusqu'à 255 caractères peuvent être mis en mémoire en utilisant la touche [IN]. La mise en mémoire de la nouvelle formule permet d'effacer la formule actuellement mise en mémoire.
2. Le contenu de la mémoire est retenu, même lorsque l'interrupteur d'alimentation de l'appareil est mis sur la position "OFF", manuellement ou grâce à la coupure automatique de l'alimentation.
3. La touche [CALC] ne peut être utilisée que pour exécuter des expressions numériques sauvegardées à l'aide de la touche [IN].
4. Une erreur est générée lorsqu'une entrée mise en mémoire par la touche [IN] n'est pas une expression numérique.
5. Des chaînes et des tableaux sont simplement affichés tels qu'ils ont été mis en mémoire si on les rappelle.
6. Les mêmes limites qui s'appliquent aux variables BASIC s'appliquent aux variables de la fonction de mise en mémoire de formules (voir page 30).
7. Les calculs sont terminés dans les conditions suivantes:
   - En appuyant sur la touche [BRK].
   - Lorsqu'une erreur est générée.

PARTIE 5

FONCTION DE LA BANQUE DE DONNEES

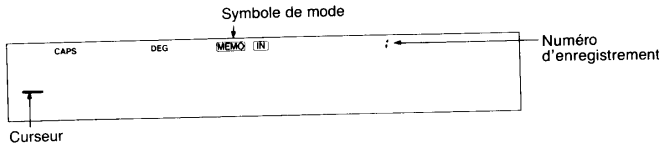
La fonction de la BANQUE DE DONNEES incorporée dans cet appareil lui donne la possibilité de complètement remplacer un agenda. Pour servir ici d'exemple, le tableau suivant de constantes scientifiques sera entré dans la BANQUE DE DONNEES de l'appareil.

TABLEAU DE CONSTANTES SCIENTIFIQUES

Nom	Symbole	Valeur numérique	Unité	Remarques
Accélération de chute libre	g	9,80665	ms^{-2}	FREE FALL
Vitesse de la lumière (dans l'espace)	c	$2{,}99792458 \times 10^{8}$	ms^{-1}	SPEED LIGHT
Constante de Planck	h	$6{,}626176 \times 10^{-34}$	Js	PLANCK'S
Constante de gravitation	G	$6{,}672 \times 10^{-11}$	Nm^2kg^{-2}	GRAVITATION
Charge élémentaire	e	$1{,}6021892 \times 10^{-19}$	C	ELEMENTARY
Masse de l'électron	me	$9{,}109534 \times 10^{-31}$	kg	ELECTRON
Masse atomique	u	$1{,}6605655 \times 10^{-27}$	kg	ATOMIC
Constante d'Avogadro	N_A	$6{,}022045 \times 10^{23}$	mol^{-1}	AVOGADRO
Constante de Boltzmann	k	$1{,}380662 \times 10^{-23}$	JK^{-1}	BOLTZMANN'S
Volume molaire de gaz idéal à p.t.s.	Vm	$2{,}241383 \times 10^{-2}$	m^3mol^{-1}	IDEAL GAS

5-1 ENTREE DES DONNEES

Le mode MEMO doit être entré en utilisant l'opération [MODE] [9] pour permettre l'entrée de données dans la BANQUE DE DONNEES. A ce moment, l'affichage doit apparaître de la manière illustrée ci-dessous:



Les symboles apparaissant au centre de la ligne supérieure de l'affichage indiquent que le mode actuel est le mode MEMO IN. La valeur située sur le coin supérieur droit indique le numéro d'enregistrement, qui est actuellement le numéro de la ligne des données de la BANQUE DE DONNEES. Le numéro d'enregistrement 1 indique qu'il n'y a pas encore de données sauvegardées. La procédure suivante est la procédure pour entrer la constante d'accélération de chute libre:

OPERATION:

[CAPS] [G] [,] → Minuscules pour l'entrée de g

9 [.] **80665** [,] → Entrée de valeur numérique

[M] [S] [−] **2** → Minuscules pour l'entrée de ms^{-2}

[SPC] [CAPS] [F] [R] [E] [E] [SPC] [F] [A] [L] [L] [EXE] → Majuscules pour l'entrée de remarques

Les articles multiples (p.ex., symboles et valeurs) peuvent être compris dans une ligne en les séparant par des virgules. L'étape finale de l'opération est la touche [EXE] qui écrit les données dans la mémoire. Cette opération fait aussi disparaître le curseur de l'affichage. Appuyer à nouveau sur [EXE] pour faire apparaître le curseur au coin supérieur gauche ou entrer simplement le premier caractère du prochain enregistrement. L'une des deux procédures permet de passer au numéro d'enregistrement suivant pour l'entrée de l'article suivant.

OPERATION:

[CAPS] [C] [,] → Minuscules pour l'entrée de c

2.99792458 [E] **8** [,] → Entrée de valeur numérique (exposant entré en utilisant [E])

[M] [S] [−] **1** → Minuscules pour l'entrée de ms^{-1}

[SPC] [CAPS] [S] [P] [E] [E] [D] [SPC] [L] [I] [G] [H] [T] [EXE] → Majuscules pour l'entrée de remarques

Dans cet exemple, la valeur utilisée comme exposant est entrée en utilisant la touche [E]. Noter que des lettres en majuscules et en minuscules ont été utilisées dans les deux premières lignes. Toujours vérifier le mode actuel sur l'affichage. Le témoin **CAPS** indique le mode de majuscules, alors qu'un affichage net à la position de **CAPS** indique les minuscules.
Répéter les procédures décrites ci-dessus jusqu'à ce que les dix constantes soient sauvegardées, puis passer au mode CAL en appuyant sur [MODE] [0].

5-2 AFFICHAGE DE DONNEES

Toutes les données sauvegardées peuvent maintenant être affichées pour vérifier si l'entrée a été correcte. Tout en étant dans le mode CAL, appuyer sur la touche [MEMO] pour afficher les enregistrements 1 et 2. Noter qu'ici, seul l'enregistrement 1 est affiché si sa longueur dépasse 32 caractères.

```
g.9.80665.ms-2 FREE FALL
c.2.99792458E8.ms-1 SPEED LIGHT
```

A ce moment, une pression sur [⇩] ou sur [EXE] permet d'afficher les enregistrements 2 et 3. A tout moment, une pression sur [⇧] ou sur [SHIFT] [EXE] permet d'afficher l'enregistrement précédent.

[⇩]

```
c.2.99792458E8.ms-1 SPEED LIGHT
h.6.626176E-34.Js PLANCK'S
```


* Noter que les opérations des touches de défilement notées ci-dessus diffèrent de la manière suivante:

OPERATION	RESULTAT
[EXE]	Défile d'un enregistrement vers le bas.
[SHIFT] [EXE]	Défile d'un enregistrement vers le haut.
[⇧]	Défile d'une ligne vers le haut.
[⇩]	Défile d'une ligne vers le bas.

5-3 EDITION DE DONNEES

L'édition des données sauvegardées est effectuée dans le mode MEMO IN ([MODE] [9]). Bien sûr, les données peuvent aussi être changées pendant la procédure d'entrée (avant que [EXE] ne soit enfoncée) en déplaçant le curseur à l'emplacement désiré à l'aide des touches de curseur (touches [⇦] et [⇨]) et en entrant ensuite les données correctes.

La procédure suivante est utilisée pour éditer des données qui ont déjà été sauvegardées.

1. Appuyer sur [MODE] [9] (curseur n'est pas affiché).
2. Appuyer sur [MEMO] .
3. Placer l'enregistrement à changer dans la première ligne de l'affichage.
4. Appuyer sur [⇦] ou sur [⇨] pour afficher le curseur ([EDIT] apparaît sur l'affichage pour indiquer le mode EDIT).
5. Déplacer le curseur à l'emplacement désiré et entrer les données correctes.
6. Appuyer sur [EXE] ([EDIT] disparaît de l'affichage).

L'exemple suivant suppose qu'une erreur est découverte dans l'enregistrement 5 (charge élémentaire) pendant l'affichage dans le mode CAL.

```
CAPS CAL DEG MEMO S
e.1.6.21892E-19.C ELEMENTARY
me.9.109534E-31.kg ELECTRON
```

Appuyer sur [MODE] [9] pour passer dans le mode MEMO IN.

[MODE] **9**

```
CAPS DEG MEMO IN S
e.1.6.21892E-19.C ELEMENTARY
me.9.109534E-31.kg ELECTRON
```

Le curseur peut ici être affiché en appuyant sur [⇦] ou sur [⇨]. A ce moment, le symbole [EDIT] apparaît également pour indiquer le mode EDIT.

[⇨]

```
CAPS DEG MEMO IN EDIT S
e.1.6.21892E-19.C ELEMENTARY
```

Curseur

Déplacer le curseur à l'emplacement désiré et entrer les données correctes. Appuyer enfin sur [EXE] pour terminer la procédure ([EDIT] disparaît de l'affichage).

[⇨] [⇨] [⇨] [⇨] [⇨] 0

```
CAPS        DEG        (MEMO) IN  (EDIT)        S
e.1.6021892E-19.C ELEMENTARY
```

[EXE]

```
CAPS        DEG        (MEMO) IN                S
e.1.6021892E-19.C ELEMENTARY
me.9.109534E-31.kg ELECTRON
```

5-4 ADDITION D'ENREGISTREMENTS

De nouveaux enregistrements peuvent être ajoutés aux enregistrements déjà entrés. Les enregistrements peuvent être ajoutés à la fin d'enregistrements existants ou insérés entre deux enregistrements existants.

5-4-1 Addition de données

1. Appuyer sur [MODE] [9] . L'appareil est en attente pour l'entrée du prochain enregistrement successif situé après les enregistrements déjà sauvegardés.
2. Entrer les données pour ajouter le nouvel enregistrement.
3. Appuyer sur [EXE] pour terminer la procédure.

5-4-2 Insertion de données

1. Appuyer sur [MODE] [9] .
2. Appuyer sur la touche [MEMO] .
3. Utiliser [EXE] ou [SHIFT] [EXE] pour afficher l'enregistrement existant afin de suivre l'enregistrement nouvellement inséré.
4. Entrer les données.
5. Appuyer sur [EXE] pour terminer la procédure.

L'exemple suivant décrit comment entrer un enregistrement contenant la constante de température absolue de l'eau à 0°C entre l'enregistrement 8 (Constante d'Avogadro) et l'enregistrement 9 (Constante de Boltzmann).

Nom	Symbole	Valeurs numériques	Unité	Remarques
Température absolue de l'eau à 0°C	T0	273,15	K	ABS TEMP

Entrer les données mentionnées ci-dessus après avoir affiché l'enregistrement 8 sur la première ligne de l'affichage.

OPERATION:

[T] **0** [,] **273** [.] **15** [,] [K] [SPC] [A] [B] [S] [SPC] [T] [E] [M] [P] [EXE]

Le résultat de l'opération est le suivant:

Enregistrement 8	Constante d'Avogadro
Enregistrement 9	Température absolue de l'eau à 0°C
Enregistrement 10	Constante de Boltzmann
Enregistrement 11	Volume molaire de gaz idéal à p.t.s.

* Pour insérer de nouvelles données dans l'enregistrement 1, appuyer sur [SHIFT] [EXE] après avoir affiché l'enregistrement 1, puis entrer les données de l'enregistrement 1. A ce moment, tous les enregistrements suivants sont décalés vers le bas.

5-5 SUPPRESSION ET EFFACEMENT GENERAL DES DONNEES

5-5-1 Suppression de données

La procédure suivante est utilisée pour effacer des enregistrements spécifiques de données déjà sauvegardées.

1. Appuyer sur [MODE] [9] .
2. Appuyer sur la touche [MEMO] .
3. Appuyer sur [EXE] et rappeler le numéro de l'enregistrement à supprimer.
4. Appuyer sur la touche de curseur [⇦] ou [⇨] pour afficher le symbole [EDIT].
5. Appuyer sur [SHIFT] [CLS] [EXE] pour supprimer l'enregistrement actuellement affiché. Tous les enregistrements suivants sont décalés vers le haut.

5-5-2 Effacement général des données

Le contenu de la BANQUE DE DONNEES est retenu lorsque l'interrupteur d'alimentation de l'appareil est mis sur la position "OFF" et lorsque les commandes NEW, NEW ALL et CLEAR sont exécutées. La procédure suivante est utilisée pour effacer tout le contenu actuel de la BANQUE DE DONNEES.

1. Appuyer sur [MODE] [1] pour passer dans le mode BASIC.
2. Entrer [N] [E] [W] [SHIFT] [#] [EXE] pour exécuter une commande NEW# et effacer toutes les données sauvegardées dans la BANQUE DE DONNEES.

IMPORTANT
Les données effacées en utilisant les procédures décrites ci-dessus ne peuvent pas être récupérées. Ne supprimer ou effacer les données que lorsqu'elles ne sont plus nécessaires.

5-6 RECHERCHE DE DONNEES

Une pression sur la touche [MEMO] tout en étant dans le mode CAL ou dans le mode MEMO IN permet d'afficher l'enregistrement 1. Maintenant, chaque pression sur la touche [MEMO] permet de décaler le curseur vers les données situées à la droite de la prochaine virgule située après la position actuelle du curseur.
La touche [EXE] peut également être utilisée pour décaler le curseur vers l'article de données suivant et [SHIFT] [EXE] peuvent être utilisés pour décaler vers l'article de données précédent.

```
CAPS  CAL        DEG        (MEMO) IN
g.9.80665.ms-2 FREE FALL
c.2.99792458E8.ms-1 SPEED LIGHT
```

↓ [MEMO]

```
CAPS CAL      DEG  MEMO IN          :
g.9.80665.ms-2 FREE FALL
c.2.99792458E8.ms-1 SPEED LIGHT
```

```
CAPS CAL      DEG  MEMO IN          :
g.9.80665.ms-2 FREE FALL
c.2.99792458E8.ms-1 SPEED LIGHT
```

5-6-1 Recherche conditionnelle

La recherche conditionnelle permet de désigner une lettre, valeur ou mot spécifique (jusqu'à huit caractères de long) afin de rapidement repérer un enregistrement désiré à l'intérieur d'un large fichier. L'entrée de <données objet> [MEMO] permet d'afficher le premier article de données dans lequel les <données objet> apparaissent immédiatement après une virgule. Chaque pression consécutive de [MEMO] affiche l'article de données suivant contenant les <données objet> après une virgule.
Dans l'exemple suivant, entrer [N] [MEMO] pour localiser la constante d'Avogadro.

```
CAPS CAL      DEG  MEMO          Y
G.6.672E-11.Nm2kg-2 GRAVITATION
e.1.6021892E-19.C ELEMENTARY
```

Le premier enregistrement à apparaître est l'enregistrement 4 (constante de gravitation), parce qu'il contient la lettre N suivie d'une virgule. Appuyer à nouveau sur [MEMO] pour afficher l'article de données suivant qui satisfait à la condition posée.

```
CAPS CAL      DEG  MEMO          8
Na.6.022045E23.mol-1 AVOGADRO
k.1.380662E-23.JK-1 BOLTZMANN'S
```

L'article de données désiré est ici localisé. Bien sûr, la touche [MEMO] peut être enfoncée autant de fois que désiré jusqu'à ce que les <données objet> soient localisées. Si aucun des enregistrements ne contient les <données objet> spécifiées, le curseur est affiché et l'appareil attend une autre entrée.

5-7 UTILISATION DES DONNEES DE LA BANQUE DE DONNEES DANS LES PROGRAMMES

Les données sauvegardées dans la BANQUE DE DONNEES peuvent aussi être accédées à partir d'un programme en BASIC en utilisant les commandes de programme suivantes.

READ #

La commande standard READ est généralement utilisée pour lire les instructions DATA contenues dans un programme. La commande READ #, d'un autre côté, lit les données de la BANQUE DE DONNEES. Les données sont lues en unités à partir du début d'un groupe de données jusqu'à la virgule suivante.

FORMAT: READ # nom de variable [, nom de variable]

Comme indiqué ci-dessus, des noms de variables multiples peuvent être spécifiés, avec les noms de variables étant séparés par des virgules. Comme avec la commande standard READ, les données numériques ne peuvent être affectées qu'aux variables numériques et les données de chaînes de caractères qu'aux variables de chaînes de caractères. Une différence de types de variable entraîne une erreur TM et l'exécution de la commande READ # lorsqu'aucune donnée n'existe génère une erreur DA. Tous les espaces de tête d'un groupe de données sont sautés, à moins que le groupe ne soit compris entre guillemets.

RESTORE #

Comme avec la commande standard RESTORE, RESTORE # peut être utilisée pour désigner une position spécifique à partir de laquelle l'opération READ # est à effectuer.

FORMAT: RESTORE #

La simple exécution de RESTORE # permet de spécifier que la prochaine opération READ # ou WRITE # est à effectuer au début des données actuellement sauvegardées dans la BANQUE DE DONNEES.

FORMAT: RESTORE # "chaîne de caractères objet"

L'inclusion d'une chaîne de caractères objet avec la commande RESTORE # permet de spécifier que la prochaine opération READ # ou WRITE # est à effectuer à partir de l'article de données commençant par la chaîne de caractères spécifiée contenue dans la BANQUE DE DONNEES. Une erreur DA est générée lorsque la chaîne de caractères objet spécifiée n'existe pas. La capacité maximum de l'opération WRITE # est de 255 caractères et le dépassement de cette valeur entraîne une erreur.

FORMAT: RESTORE # "chaîne de caractères objet", 0

Le format mentionné ci-dessus est identique à RESTORE # "chaîne de caractères objet".

FORMAT: RESTORE # "chaîne de caractères objet", 1

Le format mentionné ci-dessus spécifie que la prochaine opération READ # ou WRITE # est à effectuer à partir de l'enregistrement qui commence par la chaîne de caractères objet spécifiée.

FORMAT: RESTORE # "chaîne de caractères objet", {0 ou 1},
{numéro de ligne ou # numéro de zone de programme}

Le format mentionné ci-dessus désigne un saut au numéro de ligne ou au numéro de zone de programme spécifié pour la prochaine opération READ # ou WRITE # lorsque la chaîne de caractères objet spécifiée n'existe pas.

WRITE #

La commande WRITE # est utilisée dans un programme pour réécrire ou effacer des données de la BANQUE DE DONNEES.

FORMAT: WRITE # données de la BANQUE DE DONNEES

Le format mentionné ci-dessus remplace des articles de données existants par les données spécifiées de la BANQUE DE DONNEES, en commençant par la position actuelle de READ #/WRITE # Au cas où les données A, B, C existent dans la BANQUE DE DONNEES, avec les données B spécifiées pour la prochaine opération READ #/WRITE #, l'exécution de WRITE # Y, Z entraîne le changement du fichier des données de la BANQUE DE DONNEES en A, Y, Z. L'exécution de WRITE # "Y, Z" entraîne le changement en A, Y, Z, C. La ligne de données spécifiée pour la prochaine opération READ #/WRITE # est effacée lorsque la commande WRITE # est exécutée sans spécification des données de la BANQUE DE DONNEES.

5-8 APPLICATIONS DES FONCTIONS DE LA BANQUE DE DONNEES

La fonction de la BANQUE DE DONNEES peut être utilisée pour effectuer une foule de taches en plus des applications décrites dans ce chapitre du manuel. Toutes les données imaginables peuvent virtuellement être sauvegardées.

EXEMPLE:

La fonction de mise en mémoire de formules peut être utilisée en combinaison avec la BANQUE DE DONNEES pour sauvegarder, rappeler et exécuter des formules chaque fois qu'elles sont nécessaires.

1	V=4*PI*R^3/3
2	S=PI*R^2
3	Y=3*X^2+4
4	Z=SINX+COSY
5	A=Z*1.13+X*1.24

Les cinq formules énumérées ci-dessus sont sauvegardées dans la BANQUE DE DONNEES. Rappeler la troisième formule, la transférer à la fonction de mise en mémoire de formules et ensuite l'exécuter (dans le mode CAL).

MEMO	V=4*PI*R^3/3 S=PI*R^2	(Affiche la 1ère formule.)
EXE	S=PI*R^2 Y=3*X^2+4	(Affiche la 2ème formule.)
EXE	Y=3*X^2+4 Z=SINX+COSY	(Affiche la 3ème formule.)
IN	Z=SINX+COSY A=Z*1.13+X*1.24	(Met la 3ème formule en mémoire.)
CALC	X?_	(Exécute la formule mise en mémoire.)

IMPORTANT

Noter que les lignes d'enregistrement de la BANQUE DE DONNEES sont limitées à 255 caractères. Il faut faire attention en procédant à des changements à l'aide de la commande WRITE # à ne pas dépasser cette limite pour ne pas produire d'erreur.

PARTIE 6

PROGRAMMATION EN BASIC

Le BASIC standard est utilisé comme langage de programmation pour cet appareil et ce chapitre couvre l'application du langage BASIC.

6-1 CARACTERISTIQUES DU BASIC

1. Le BASIC est plus facile à utiliser que d'autres langages de programmation tels que le FORTRAN, le rendant approprié même pour des novices.
2. L'écriture de programmes est également plus facile, parce que la création, l'édition et l'exécution des programmes sont toutes effectuées par interaction avec l'ordinateur lui-même.

Les fonctions suivantes sont aussi disponibles:

1. Des calculs de haute précision sont permis par l'affichage des valeurs numériques avec des mantisses de 10 chiffres et des exposants de 2 chiffres (mantisses de 13 chiffres et exposant de 2 chiffres pour opérations internes).

2. Une large sélection de fonctions intégrées facilitent le fonctionnement.

① Fonctions mathématiques standard

SIN COS TAN ASN ACS ATN LOG LN EXP SQR ABS SGN
INT FIX FRAC PI ROUND RAN# DEG

② Fonctions puissantes de traitement de chaîne de caractères

CHR$ STR$ MID$ LEFT$ RIGHT$ HEX$ DMS$ ASC VAL LEN

③ Fonctions mathématiques de haut niveau

POL REC NCR NPR HYPSIN HYPCOS HYPTAN HYPASN HYPACS
HYPATN CUR

3. 10 zones de programme indépendantes

Un maximum de 10 programmes peuvent être sauvegardés indépendamment en même temps (P0 ~ 9).

4. Noms de variables étendus

Des noms de variables jusqu'à 15 caractères de long peuvent être utilisés, permettant d'utiliser des noms qui rendent le contenu facile à comprendre.

5. Puissante fonction de mise au point

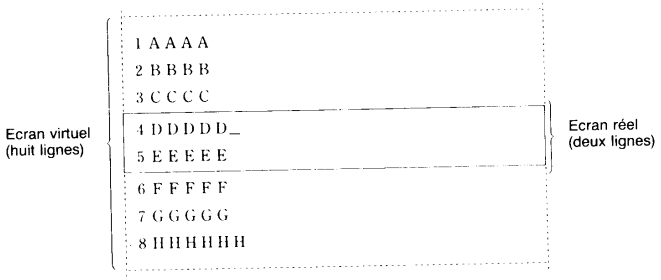
Une commande TRON affiche le numéro de la ligne de programme actuellement exécutée, permettant de facilement analyser l'exécution et de localiser les erreurs de programmation.

6. Puissant éditeur d'écran

Les programmes peuvent facilement être modifiés et corrigés sur l'écran.

7. Fonction d'écran virtuel

Bien que l'affichage physique réel de l'appareil possède une capacité de 32 colonnes × 2 lignes, l'écran virtuel est de 32 colonnes × 8 lignes. L'écran virtuel peut facilement être défilé en utilisant les touches de curseur.



8. Gestion de fichier agrandie

Les commandes BASIC standard telles que OPEN, CLOSE, INPUT# et PRINT# sont toutes disponibles pour la lecture et l'écriture des fichiers de données.

6-2 CONFIGURATION DE PROGRAMMES EN BASIC

6-2-1 Format de programmes en BASIC

L'exemple suivant est un programme en BASIC typique qui calcule le volume d'un cylindre.

EXEMPLE:

```
10 REM CYLINDER
20 R=15
30 INPUT "H=" ; H
40 V=PI*R^2*H        (PI indique π)
50 PRINT "V=" ; V
60 END
```

Comme on peut le voir, le programme en BASIC est en fait un ensemble de lignes (six lignes dans le programme ci-dessus). Une ligne peut être décomposée en un numéro de ligne et une instruction.

```
20  R=15
Numéro  Instruction
de ligne
```

Les ordinateurs exécutent les programmes dans l'ordre de leurs numéros de ligne. Dans le programme échantillon donné ci-dessus, l'ordre d'exécution est 10, 20, 30, 40, 50, 60. Les lignes de programme peuvent être entrées dans l'ordinateur dans un ordre quelconque et l'ordinateur arrange automatiquement le programme dans sa mémoire dans l'ordre du numéro de ligne le plus petit au plus élevé. Les lignes peuvent être numérotées en utilisant une valeur quelconque comprise entre 1 et 65535.

```
20 R=15
40 V=PI*R^2*H
60 END
10 REM CYLINDER
30 INPUT "H=" ; H
50 PRINT "V=" ; V
```

Ordre d'entrée

```
10 REM CYLINDER
20 R=15
30 INPUT "H=" ; H
40 V=PI*R^2*H
50 PRINT "V=" ; V
60 END
```

Contenu de la mémoire

Après le numéro de ligne se trouve une ou des instructions qui disent en fait à l'ordinateur quelle opération effectuer. L'exemple suivant revient au programme échantillon pour expliquer chaque instruction en détail.

10 REM CYLINDER.....REM signifie "remarques". Rien n'est exécuté dans cette ligne.
20 R=15.....Affecte la constante 15 (rayon) à la variable R.
30 INPUT "H=" ; H.....Affiche H ? pour demander une entrée de valeur pour la hauteur.
40 V=PI∗R^2∗H.....Calcule le volume (V) du cylindre.
50 PRINT "V=" ; V.....Imprime le résultat de la ligne 40.
60 END.....Termine le programme.

Comme on peut le voir, rien que six lignes de programmation traitent pas mal de données. Des lignes de programme multiples en BASIC peuvent aussi être liées dans une ligne unique en utilisant des deux-points.

EXEMPLE:
100 R=15:A=7:B=8

Une telle ligne de programme est appelée "instruction multiple".

Les détails concernant les autres opérations contenues dans le programme ci-dessus peuvent être trouvés dans d'autres sections de la PARTIE 6.

6-3 ENTREE DE PROGRAMMES EN BASIC

6-3-1 Préparation

Mettre tout d'abord l'interrupteur d'alimentation de l'ordinateur sur la position "ON". A ce moment, l'affichage doit apparaître de la manière illustrée ci-dessous.

```
CAPS   CAL   DEG

_
```

C'est le mode CAL, donc l'opération [MODE] [1] doit tout d'abord être effectuée pour permettre l'entrée de programmes en BASIC. L'affichage doit maintenant apparaître de la manière illustrée ci-dessous.

```
CAPS   BASIC DEG
P 0 1 2 3 4 5 6 7 8 9      3536B
Ready P0
```

Noter que le témoin CAL a été remplacé par BASIC pour indiquer le mode BASIC. C'est le mode utilisé pour l'entrée de programmes en BASIC. Dans le mode BASIC, les autres témoins situés sur l'affichage ont les significations suivantes.

P : Zone de programme
0 ~ 9 : Numéros de zone de programme. Les numéros des zones de programme contenant déjà des programmes sont remplacés par des astérisques.

EXEMPLE:
Programme sauvegardé dans la zone 3

```
CAPS   BASIC DEG
P 0 1 2 ∗ 4 5 6 7 8 9      3521B
Ready P0
```

3536B : Capacité (nombre d'octets) restante dans la zone pour l'écriture de programmes et de données (zone libre). La valeur est de 3536B lorsqu'il n'y a pas de programme ni de données sauvegardés dans la mémoire, avec cette valeur diminuant au fur et à mesure que l'espace de mise en mémoire est utilisé.
Ready P0 : Zone actuelle de programme = zone 0. La zone actuelle du programme peut être changée en appuyant sur [SHIFT] suivi de la zone de programme désirée.

EXEMPLE:
Changement à la zone de programme 5

[SHIFT] [P5]

```
CAPS   BASIC DEG
P 0 1 2 3 4 5 6 7 8 9      3536B
Ready P5
```

Des programmes déjà sauvegardés peuvent être effacés en utilisant l'une des deux différentes procédures.

NEW : Efface un programme sauvegardé uniquement dans la zone actuelle de programme.
NEW ALL : Efface tous les programmes sauvegardés dans la mémoire.

EXEMPLE:

NEW ALL [EXE]

```
CAPS   BASIC DEG
P 0 1 2 3 4 5 6 7 8 9      3536B
Ready P0
```

Cette opération efface tous les programmes sauvegardés dans la mémoire et ramène la zone actuelle du programme à 0.

6-3-2 Entrée de programmes

La procédure d'entrée suivante entre le programme échantillon pour le calcul du volume d'un cylindre (page 47).

[1] [0] [R] [E] [M] [SPC] [C] [Y] [L] [I] [N] [D] [E] [R] [EXE]
[2] [0] [R] [=] [1] [5] [EXE]
[3] [0] [I] [N] [P] [U] [T] [SHIFT] ["] [H] [=] [SHIFT] ["] [;] [H] [EXE]
[4] [0] [V] [=] [P] [I] [∗] [R] [^] [2] [∗] [H] [EXE]
[5] [0] [P] [R] [I] [N] [T] [SHIFT] ["] [V] [=] [SHIFT] ["] [;] [V] [EXE]
[6] [0] [E] [N] [D] [EXE]

Noter que la touche [EXE] est enfoncée à la fin de chaque ligne. Aucune ligne de programme n'est entrée dans la mémoire tant que la touche [EXE] n'est pas enfoncée.

ENTREE A UNE TOUCHE
Les commandes BASIC à une touche aident à rendre l'entrée de programmes encore plus facile.

EXEMPLE:
Entrée de la ligne 30.

[3] [0] [SHIFT] [INPUT] [SHIFT] ["] [H] [=] [SHIFT] ["] [;] [H] [EXE]

6-3-3 Edition de programmes

La procédure utilisée pour faire des corrections ou des changements à un programme dépend dans quel pas de programme d'entrée les changements sont à faire.

① Changements dans une ligne avant que [EXE] ne soit enfoncée
② Changements dans une ligne après que [EXE] soit enfoncée
③ Changements dans un programme déjà entré
④ Changements dans un programme après la commande EDIT

1. Changements dans une ligne avant que [EXE] ne soit enfoncée

EXEMPLE:
20 E = 15 entré par erreur pour 20 R = 15

```
10 REM CYLINDER
20 E=15_
```

[⇦] [⇦] [⇦] [⇦] — (Déplacer le curseur à E.)

```
10 REM CYLINDER
20 E=15
```

[R] — (Entrer le caractère correct.)

```
10 REM CYLINDER
20 R=15
```

[EXE] — (Edition terminée.)

```
20 R=15
_
```

Noter qu'une fois les changements désirés sont faits, la touche [EXE] doit être enfoncée pour sauvegarder la ligne entrée dans la mémoire.

2. Changements dans une ligne après que [EXE] soit enfoncée

EXEMPLE:
40 V = P1∗R^2∗H entré par erreur pour 40 V = PI∗R^2∗H

```
40 V=P1*R^2*H
_
```

[⇧] [⇨] [⇨] [⇨] [⇨] [⇨] [⇨]

```
40 V=P1*R^2*H
```

(Déplacer le curseur à 1.)

[I]

```
40 V=PI*R^2*H
```

(Entrer le caractère correct.)

[EXE]

```
40 V=PI*R^2*H
_
```

(Edition terminée.)

De nouveau, la touche [EXE] doit être enfoncée pour sauvegarder la ligne corrigée dans la mémoire après avoir effectué les changements.
Les procédures 1 et 2 indiquent des procédures pour de simples changements d'un caractère à un autre.

Les caractères peuvent aussi être insérés et effacés en utilisant les procédures suivantes.

i) Insertion
40 V = PI∗R2∗H entré par erreur pour 40 V = PI∗R^ 2∗H

```
40 V=PI*R2*H_
```

[⇦] [⇦] [⇦]

```
40 V=PI*R2*H
```

(Déplacer le curseur à l'emplacement de l'insertion.)

[INS]

```
40 V=PI*R_2*H
```

(Ouvrir un espace.)

[^] [EXE]

```
40 V=PI*R^2*H
_
```

(Entrer le caractère correct et appuyer sur [EXE])

ii) Effacement
40 V = PI∗RR ^ 2∗H entré par erreur pour 40 V = PI∗R ^ 2∗H

```
40 V=PI*RR^2*H_
```

[⇦] [⇦] [⇦] [⇦] [⇦]

```
40 V=PI*RR^2*H
```

(Déplacer le curseur au caractère à effacer.)

[SHIFT] [DEL]

```
40 V=PI*R^2*H
```

(Effacer le caractère.)

[EXE]

```
40 V=PI*R^2*H
_
```

(Edition terminée.)

La touche [BS] agit plutôt comme l'opération de [SHIFT] [DEL]. La différence entre les deux opérations est la suivante.

Différence entre [SHIFT] [DEL] et [BS]

• [SHIFT] [DEL]
Efface le caractère à la position actuelle du curseur et décale tout ce qui vient à la droite du curseur d'un espace vers la gauche.

A B C D E F G → [SHIFT] [DEL] → A B C E F G
↑ Curseur

• [BS]
Efface le caractère à la gauche de la position actuelle du curseur et décale tout ce qui vient à partir de la droite de la position du curseur d'un espace vers la gauche.

A B C D E F G → [BS] → A B D E F G

3. Changements dans un programme déjà entré

La commande LIST affiche le programme sauvegardé dans la zone actuelle de programme à partir du début jusqu'à la fin.

LIST [EXE]

```
10 REM CYLINDER
20 R=15
⋮
60 END
Ready P0
```

La dernière ligne du programme est affichée lorsque l'opération LIST est terminée.

[⇧] [⇧] [⇧] [⇧] [⇧] [⇧]

```
10 REM CYLINDER
20 R=15
```

L'écran virtuel de 8 lignes de l'ordinateur permet d'utiliser les touches de curseur pour défiler aux lignes précédentes non indiquées sur l'affichage (voir page 7).

(Ecran virtuel de 8 lignes)

```
Ready P0
10  REM CYLINDER
20  R=15
30  INPUT "H=" ; H
40  V=PI*R^2*H
50  PRINT "V=" ; V
60  END
Ready P0
```

Affichage de 2 lignes
Les touches [⇧] [⇩] déplacent l'affichage.

Lorsqu'un programme supérieur à huit lignes est sauvegardé dans la mémoire, l'opération LIST doit être effectuée en spécifiant les numéros de lignes à afficher.

EXEMPLE:
Affichage de la ligne 110 à la ligne 160 sur l'écran virtuel.
LIST 110 — 160 [EXE]

(Ecran virtuel de 8 lignes)

```
LIST 110—160
110  A=1
120  FOR I=1 TO 100
130  B=LOG ( I )
140  PRINT B
150  NEXT I
160  E=A*B
Ready P0
```

Affichage de 2 lignes

Les changements peuvent être faits dans un programme grâce à l'opération LIST en utilisant les mêmes procédures décrites pour 1 et 2 ci-dessus.

* La touche [BRK] peut être utilisée pour terminer l'opération LIST. La touche [STOP] suspend l'opération et le listing peut être repris en appuyant sur [EXE] .

4. Changements dans un programme après la commande EDIT

La commande EDIT facilite l'édition ou la revue de programmes déjà sauvegardés dans la mémoire.

EDIT [EXE]

```
CAPS    BASIC DEG                EDIT
10 REM CYLINDER
20 R=15
```

De cet affichage, [⇩] (ou [EXE]) fait passer à la ligne suivante, alors que [⇧] (ou [SHIFT] [EXE]) fait revenir à la ligne précédente.

[⇩] [⇩]

```
30 INPUT "H=" ; H
40 V=PI+R^2*H
```

(Affiche les lignes 30 et 40.)

Ici, une correction sera faite à la ligne 40.

[⇩]

```
40 V=PI+R^2*H
50 PRINT "V=" ; V
```

(Affiche la ligne 40 à la ligne supérieure de l'affichage.)

[⇨]

```
40 V=PI+R^2*H
```

(Permet l'édition du programme.)

[⇨] [⇨] [⇨] [⇨] [⇨] [⇨] [⇨] [*] [EXE]

```
40 V=PI*R^2*H
50 PRINT "V=" ; V
```

(Correction)

[BRK]

```
Ready P0
```

(Touche [BRK] sort du mode EDIT.)

6-4 EXECUTION DE PROGRAMMES EN BASIC

6-4-1 Exécution de programmes

Une fois qu'un programme en BASIC est sauvegardé dans la mémoire, il peut être exécuté en utilisant l'une des deux procédures suivantes.

1. Utilisation de [SHIFT] (zone de programme) dans mode CAL

   EXEMPLE: [SHIFT] [9]
   Exécute le programme dans la zone de programme 9.

2. Entrée de la commande RUN dans le mode BASIC

   EXEMPLE: RUN [EXE]
   Exécute le programme dans la zone de programme actuelle.

Exécute l'entrée du programme de la section précédente pour déterminer le volume d'un cylindre avec une hauteur de 10 (rayon fixé à 15).

RUN [EXE]

```
RUN
H=?_
```

(Exécute le programme.)

10 [EXE]

```
H=?10
V= 7068.583471
```

(Volume correspondant affiché lorsque la hauteur est entrée.)

[EXE]

```
V= 7068.583471
Ready P0
```

Lorsque ce programme est exécuté, l'affichage du volume entraîne l'apparition du symbole STOP dans le coin supérieur droit de l'affichage. Ce symbole indique que l'exécution du programme a été suspendue à cause de l'exécution de la commande PRINT. L'exécution du programme peut être reprise à ce moment en appuyant à nouveau sur la touche EXE . La fin d'une instruction PRINT avec un point-virgule entraîne la continuation de l'exécution lorsque l'instruction PRINT est exécutée, provoquant l'apparition de l'affichage de l'instruction PRINT suivante immédiatement après l'affichage précédent.

EXEMPLE 1:

```
10 PRINT "BASIC  "
20 PRINT "PROGRAM"
30 END
```

L'exécution de ce programme entraîne l'affichage suivant.

RUN EXE	RUN BASIC
EXE	BASIC PROGRAM
EXE	PROGRAM Ready P0

EXEMPLE 2:

```
10 PRINT "BASIC  " ;
20 PRINT "PROGRAM"
30 END
```

L'inclusion d'un point-virgule à la fin de la première instruction PRINT donne l'affichage suivant.

RUN EXE	RUN BASIC PROGRAM
EXE	BASIC PROGRAM Ready P0

6-4-2 Erreurs

Parfois, les résultats produits par un programme ne sont pas ceux qui étaient prévus. De telles exécutions irrégulières peuvent être généralement divisées en deux classifications principales.

① Les exécutions qui produisent des erreurs
Erreurs de programmation simple
Erreurs logiques de programme
② L'exécution irrégulière qui ne produit pas d'erreurs
La plupart des erreurs logiques de programme

1. Exécutions qui produisent des erreurs

i) Erreurs de programmation simple

C'est le type le plus commun d'erreur de programme et il est généralement causé par des erreurs dans la syntaxe du programme. De telles erreurs entraînent l'affichage du message suivant:

SN error P0—10 (Erreur SN P0—10)

Ce message indique qu'une erreur de syntaxe a été détectée à la ligne 10 du programme sauvegardé dans la zone de programme 0. La ligne de programme indiquée doit être vérifiée et corrigée pour permettre une bonne exécution.

ii) Erreurs logiques de programme

Ce type d'erreur est généralement causé par des opérations illégales telles que la division par 0 ou LOG(0). De telles erreurs entraînent l'affichage du message suivant:

MA error P0—10 (Erreur MA P0—10)

Comme avant, ce message indique qu'une erreur mathématique a été détectée à la ligne 10 du programme sauvegardé dans la zone de programme 0. Toutefois, dans ce cas, les lignes de programme liées à la ligne de programme indiquée ainsi que la ligne de programme indiquée elle-même doivent être vérifiées et corrigées. Lorsqu'un message d'erreur est affiché, les deux opérations suivantes peuvent être utilisées pour afficher le numéro de la ligne dans laquelle l'erreur a été détectée.

LIST 10 EXE
EDIT 10 EXE

LIST . EXE
EDIT . EXE

* Les points contenus dans **LIST .** et **EDIT .** instruisent l'ordinateur pour automatiquement afficher la dernière ligne de programme exécutée.
* Changer au mode BASIC si un programme en BASIC a été exécuté dans le mode CAL.

2. Exécution irrégulière qui ne produit pas d'erreurs

De telles erreurs sont également causées par une imperfection dans le programme et doivent être corrigées en utilisant la commande LIST ou EDIT pour examiner le programme afin de détecter le problème. La commande TRON peut également être utilisée pour aider à analyser l'exécution du programme.
L'entrée de **TRON** EXE fait apparaître le symbole TR sur l'affichage pour indiquer que le mode d'analyse est activé. Maintenant, l'exécution d'un programme en BASIC permet d'afficher la zone de programme et le numéro de ligne au fur et à mesure que l'exécution est effectuée et de suspendre l'exécution jusqu'à ce que EXE soit enfoncée. Ceci permet la confirmation de chaque ligne du programme, rendant possible une rapide identification des lignes à problème. L'exécution de **TROFF** EXE permet d'annuler le mode d'analyse.

6-5 COMMANDES

Bien qu'il y ait une foule de commandes disponibles dans le BASIC pour être utilisées dans les programmes, les neuf commandes fondamentales énumérées ci-dessous sont les plus largement utilisées.

Le programme suivant lit les articles de données contenus dans le programme lui-même, avec le nombre d'articles de données lus déterminé par entrée à partir d'un opérateur. L'opérateur peut entrer toute valeur, mais noter que les valeurs supérieures à 5 sont traitées comme 5 (parce qu'il n'y a que 5 articles de données à la ligne 180). Le programme affiche alors la somme des données lues à partir de la ligne 180, suivi de la racine carrée et de la racine cubique de la somme. L'exécution du programme est terminée lorsqu'un 0 est entré par l'opérateur.

10 S=0	Efface le total actuel affecté à S.
20 RESTORE	Spécifie que l'opération de lecture doit commencer par le premier article de données.
30 INPUT N	Entrée du nombre d'articles de données à lire.
40 IF N>5 THEN N=5	L'entrée d'une valeur supérieure à 5 doit être traitée comme 5.

```
 50 IF N=0 THEN GOTO 130.............Saut à la ligne 130 lorsque l'entrée est 0.
 60 FOR I=1 TO N
 70 READ X   Lecture de données
 80 S=S+X    Calcul de somme de
 90 NEXT I   données
                                      Cette section répète le nombre de fois spécifié par
                                      l'entrée de l'opérateur à la ligne 30.
                                      Branchement au sous-programme en commençant
100 GOSUB 140.....................par la ligne 140.
110 PRINT S ; Y ; Z...............Affiche le contenu des variables S, Y, Z.
120 GOTO 10.......................Saut à la ligne 10
130 END...........................Fin de programme
140 REM SQUARE ROOT/CUBE ROOT...Remarques
150 Y=SQR S.......................Calcul de racine carrée
160 Z=CUR S.......................Calcul de racine cubique
170 RETURN........................Retour à l'instruction située après l'instruction qui a
                                  appelé le sous-programme.
180  DATA 9, 7, 20, 28, 36.......Lecture de données par l'instruction READ à la ligne
                                  70.
```

① **REM**

La commande REM (ligne 140) est en fait l'abréviation du mot "remarques". L'ordinateur néglige tout ce qui vient après une commande REM et elle est donc utilisée pour servir de label afin de rendre la liste de programmes plus facile à suivre. Noter qu'un guillemet unique(SHIFT ⌴) peut être utilisé à la place des lettres "REM".

② **INPUT**

La commande INPUT (ligne 30) est utilisée pour permettre l'entrée à partir du clavier de l'ordinateur pendant l'exécution du programme. Les données entrées sont affectées à une variable située immédiatement après la commande INPUT. Dans l'exemple ci-dessus, les données numériques entrées sont affectées à la variable N. Noter qu'une variable de chaîne de caractères doit être utilisée pour l'entrée de chaîne de caractères.

EXEMPLE:

10 INPUT A$ (chaîne de caractères)

③ **PRINT**

La commande PRINT (ligne 110) est utilisée pour afficher les données sur l'affichage de l'ordinateur. Dans cet exemple, cette commande est utilisée pour afficher les résultats des calculs de la somme, de la racine carrée et de la racine cubique. Lorsque les données sont affichées, le symbole STOP apparaît et l'exécution du programme est suspendue. L'exécution peut être reprise en appuyant sur la touche EXE.

④ **END**

La commande END (ligne 130) termine l'exécution du programme et peut être partout incluse dans un programme.

⑤ **IF ~ THEN ~**

La commande IF/THEN (lignes 40 et 50) est utilisée pour les comparaisons de certaines conditions, en basant l'opération suivante selon que la comparaison se révèle vraie ou fausse. La ligne 40 vérifie si oui ou non la valeur affectée à N est supérieure à 5 et affecte une valeur de 5 à N lorsque la valeur initiale est supérieure. Lorsqu'une valeur inférieure ou égale à 5 est initialement affectée à N, l'exécution passe à la ligne suivante, avec N retenant sa valeur initiale. La ligne 50 vérifie si oui ou non la valeur affectée à N est 0. Dans le cas de 0, l'exécution du programme saute à la ligne 130, alors que l'exécution passe à la ligne suivante (ligne 60) lorsque N est une autre valeur en dehors de zéro.

* La ligne 50 peut aussi être abrégée de la manière suivante:
  50 IF N=0 THEN 130

* Les zones de programme peuvent également être spécifiées comme destinations de saut:
  IF A=1 THEN GOTO #2 (Programme sauvegardé dans la zone de programme 2 exécuté lorsque A est égal à 1.)

⑥ **GOTO**

La commande GOTO (lignes 50 et 120) effectue un saut vers un numéro de ligne ou une zone de programme spécifiée. L'instruction GOTO de la ligne 120 est un saut inconditionnel, en cela que l'exécution retourne toujours à la ligne 10 du programme chaque fois que la ligne 120 est exécutée. D'un autre côté, l'instruction GOTO de la ligne 50 est un saut conditionnel, parce que la condition de l'instruction IF ~ THEN doit être remplie avant que le saut à la ligne 130 ne soit réalisé.

* Les sauts de zone de programme sont spécifiés par GOTO #2 (pour sauter à la zone de programme 2).

⑦ **FOR/NEXT**

La combinaison FOR/NEXT (lignes 60 et 90) forme une boucle. Toutes les instructions comprises à l'intérieur de la boucle sont répétées le nombre de fois spécifié par une valeur située après le mot "TO" dans l'instruction FOR. Dans l'exemple en question ici, la boucle est répétée N fois, avec la valeur de N étant entrée par l'opérateur à la ligne 30.

⑧ **READ/DATA/RESTORE**

Ces instructions (lignes 70, 180, 20) sont utilisées lorsque la quantité de données à traiter est trop large pour nécessiter l'entrée du clavier à chaque exécution. Dans ce cas, les données sont comprises dans le programme lui-même. La commande READ affecte des données aux variables, l'instruction DATA retient les données à lire et la commande RESTORE est utilisée pour spécifier à partir de quel point l'opération de lecture est à effectuer.
Dans le présent programme échantillon, la commande READ lit le nombre d'articles de données spécifiés par l'entrée pour la variable N. Bien que l'instruction DATA ne retienne que 5 articles de données, la commande RESTORE de la ligne 20 renvoie toujours la position de lecture suivante au premier article de données et l'instruction READ ne manque jamais de données à lire.

⑨ **GOSUB/RETURN**

Les commande GOSUB/RETURN (lignes 100 et 170) sont utilisées pour le branchement vers/de sous-programmes. Les sous-programmes (lignes 140 à 170) sont actuellement de miniprogrammes situés dans le programme principal et représentent d'habitude des programmes qui sont effectués de manière répétitive à divers endroits dans le programme principal. Ceci signifie que GOSUB/RETURN permet d'écrire une fois l'opération répétée comme un sous-programme, au lieu de l'écrire à chaque fois qu'elle est nécessaire dans le programme principal.

EXEMPLE:

```
120  GOSUB 1000
       ⋮
370  GOSUB 1000
       ⋮
```

L'exécution de l'instruction RETURN à la fin d'un sous-programme ramène l'exécution du programme à l'instruction située après la commande GOSUB. Dans ce programme échantillon, l'exécution revient à la ligne 110 après l'exécution de la commande RETURN à la ligne 170.

* Les programmes GOSUB peuvent également être utilisés pour brancher à d'autres zones de programme, comme dans GOSUB #3 (branche à la zone de programme 3). Noter cependant qu'un retour doit être fait à la zone de programme initiale en utilisant la commande RETURN avant qu'une commande END ne soit exécutée.
* Pour de plus amples détails sur les commandes en BASIC, voir la PARTIE 10 "REFERENCE DE COMMANDES".

6-6 OPERATEURS

Les opérateurs suivants sont les opérateurs utilisés pour des calculs qui comprennent des variables.

Opérateurs	Opérateurs arithmétiques	Signes	+ , −
		Addition	+
		Soustraction	−
		Multiplication	∗
		Division	/
		Puissance	^
		Division de nombre entier	¥
		Reste entier de division de nombre entier	MOD
	Opérateurs de relation	Egal à	=
		Différent de	< >, > <
		Plus petit que	<
		Plus grand que	>
		Plus petit ou égal à	= <, < =
		Plus grand ou égal à	= >, > =
	Opérateurs logiques	Négation	NOT
		Produit logique	AND
		Somme logique	OR
		OR exclusif	XOR
	Opérateur de chaîne de caractères		+

1. Opérateurs arithmétiques (+, −, ∗, /, ^, ¥, MOD)

- Les fractions sont tronquées dans les calculs ¥ et MOD, lorsque les opérandes situés des deux côtés de l'opérateur ne sont pas des nombres entiers.
- Dans les calculs ¥ et MOD, la division est effectuée avec les valeurs absolues des deux opérandes. Dans la division de nombre entier (¥), le quotient est tronqué à un nombre entier. Avec l'opérateur MOD, le reste reçoit le signe du dividende.

EXEMPLES:

−15¥7 = −2
−15 MOD7 = 1

$$-15 \div 7 = \underbrace{-2}_{-15¥7} \cdots\cdots \underbrace{-1}_{-15MOD7}$$

- La longueur d'une expression est limitée à 255 caractères.

2. Opérateurs de relation (=, < >, > <, <, >, = <, < =, = >, > =)

Les opérateurs de relation ne peuvent être effectués que lorsque les opérateurs sont tous les deux des chaînes de caractères ou des valeurs numériques.
Avec des chaînes, les codes de caractères sont comparés un par un à partir du début des chaînes. C'est à dire que la première position de la chaîne A est comparée avec la première position de la chaîne B, la deuxième position de la chaîne A avec la deuxième position de la chaîne B, etc. Le résultat de la comparaison se base sur les codes de caractères de la première différence détectée entre les chaînes de caractères, sans tenir compte de la longueur des chaînes de caractères comparées.

EXEMPLES:

CHAINE A	CHAINE B	RESULTAT	
ABC	ABC	A = B	
ABC	ABCDE	A < B	
ABC	XYZ	A < B	(code de caractères de A inférieur à celui de X)
XYZ	ABCDE	A > B	(code de caractères de X supérieur à celui de A)

Un résultat de − 1 est renvoyé lorsque le résultat d'une opération de relation est vraie (conditions remplies), alors que 0 est renvoyé lorsque le résultat est faux (conditions non remplies).

EXEMPLE:

```
10 PRINT 10>3.................................. − 1 renvoyé parce que 10>3 est vrai
20 PRINT 7<1...................................0 renvoyé parce que 7<1 est faux
30 PRINT "ABC" = "XYZ".........0 renvoyé parce que ABC = XYZ est faux
40 END
```

3. Opérateurs logiques

Les opérandes des opérateurs logiques sont tronqués aux nombres entiers et l'opération est effectuée bit par bit pour obtenir le résultat.

Négation

X	NOT X
0	1
1	0

Produit logique

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

Somme logique

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

OR exclusif

X	Y	X XOR Y
0	0	0
0	1	1
1	0	1
1	1	0

4. Opérateur de chaîne de caractères (+)

Les chaînes de caractères peuvent être enchaînées en utilisant un signe +.
Le résultat de l'opération (y compris les résultats intermédiaires) ne doivent pas dépasser 255 caractères.

EXEMPLE:

A$ = "AD" + "1990"

L'exemple ci-dessus résulte dans la chaîne "AD1990" étant affectée à la variable A$.

5. Ordre des opérations

Les opérations arithmétiques, de relation et logiques sont effectuées dans l'ordre de priorité suivant:

1. (,)
2. Fonction scientifique
3. Puissance
4. Signes (+, −)
5. ∗, /, ¥, MOD
6. Addition et soustraction
7. Opérateurs de relation
8. NOT
9. AND
10. OR, XOR

Les opérations sont effectués de gauche à droite lorsque l'ordre de priorité est identique.

6-7 CONSTANTES ET VARIABLES

6-7-1 Constantes

Les lignes suivantes indiquent les constantes comprises dans le programme échantillon de la page 47.

PROGRAMME	CONSTANTES
20 R=15	15
30 INPUT ''H='' ; H	''H=''
40 V=PI∗R^2∗H	2
50 PRINT ''V='' ; V	''V=''

Parmi elles, 15 et 2 sont des constantes numériques, alors que "H=" et "V=" sont des constantes de chaîne de caractères.

Constantes numériques

• Notation numérique

① Notation décimale
② Notation hexadécimale

• Précision de valeur numérique

① Opérations numériques internes
Mantisse de 12 chiffres, exposant de 2 chiffres (PI = 11 chiffres: 3,1415926536; affiché en 10 chiffres: 3,141592654)
② Résultats
Mantisse de 10 chiffres, exposant de 2 chiffres (exposant arrondi à 10 chiffres)
③ Nombre de caractères par ligne
255 caractères par ligne
④ Affichage de résultat

Nombres entiers inférieurs à 1×10^{10}	:	Affichage de nombre entier
Partie décimale inférieure à 11 chiffres	:	Affichage décimal
Autres	:	Affichage exponentiel
Arrondissage d'affichage	:	Les résultats sont arrondis au 10ème chiffre et affichés.

Constantes de chaîne de caractères

Chaînes de caractères entre guillemets (p.ex., "ABC", "H=")
Les guillemets de fermeture situés à la fin d'une ligne peuvent être omis (10 PRINT "TEST" peut être écrit 10 PRINT "TEST).
Des chaînes de caractères multiples peuvent être connectées avec un signe "+".

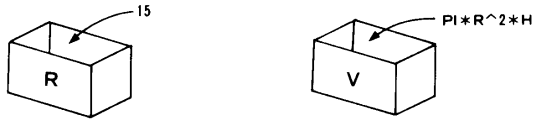
6-7-2 Variables

Variables numériques

Les lignes suivantes indiquent les variables numériques comprises dans le programme échantillon de la page 47.

PROGRAMME	VARIABLES NUMERIQUES
20 R=15	R
30 INPUT ''H='' ; H	H
40 V=PI∗R^2∗H	V

Les variables numériques sont ainsi nommées parce que leur contenu est traité comme des chiffres. Les noms de variables numériques peuvent avoir 15 caractères de long et sont utilisés dans des programmes pour sauvegarder des résultats de calculs ou des constantes dans la mémoire. Dans le programme échantillon, la valeur 15 est sauvegardée dans H, alors que V, qui est le résultat du calcul, contient la valeur qui représente le volume du cylindre. Comme on peut le voir, l'affectation à une variable est effectuée en utilisant le symbole "=". Ceci diffère du signe égal en cela qu'il déclare que ce qui se trouve à la droite doit être affecté à ce qui se trouve à la gauche. En fait, une variable peut être imaginée comme un genre de boîte de la manière illustrée ci-dessous:



Variables de chaîne de caractères

Un autre type de variable est connu sous le nom de variable de chaîne de caractères, qui est utilisée pour sauvegarder les données de chaîne de caractères. Les noms de variable de chaîne de caractères sont indiqués par "$" après le nom.

EXEMPLE:

```
10 A$=''AD''.................................Affecte "AD" à la variable de chaîne de caractères A$.
20 INPUT ''YEAR='' ; B$.............Affecte l'entrée de clavier à la variable B$.
30 C$=A$+B$..............................Affecte la combinaison de A$ et B$ à C$.
40 PRINT C$..................................Affiche le contenu de C$.
50 END
```

Dans le programme échantillon mentionné ci-dessus, l'entrée d'une année telle que 1990 à la ligne 20 entraîne un affichage de AD1990 à la ligne 40.

* Avec des variables de chaîne de caractères, "+" peut être utilisé pour connecter deux chaînes de caractères.
* Noter ici que les chaînes de caractères ne peuvent pas être affectées aux variables numériques telles que A et les valeurs numériques ne peuvent pas être affectées aux variables de chaînes de caractères telles que A$.

Variables de tableau

Les valeurs numériques et les variables de chaîne de caractères ne peuvent toutes les deux sauvegarder qu'un seul article de données par variable. Pour cette raison, de larges quantités de données sont mieux traitées en utilisant des variables de tableau (d'habitude simplement appelées "tableaux"). Avant de pouvoir utiliser une variable de tableau dans un programme, une instruction DIM doit apparaître au début d'un programme pour "déclarer" à l'ordinateur qu'une variable de tableau est à utiliser.

EXEMPLE:
Déclarer la variable de tableau A pour la sauvegarde de 21 articles de données.

```
10  DIM A (20)
```

* Le format mentionné ci-dessus est utilisé pour déclarer "NOM DE VARIABLE DE TABLEAU (NOMBRE D'ELEMENTS)".
* Une valeur déclarée de 20 permet de sauvegarder 21 articles de données (voir page 64 pour les détails).

EXEMPLE:
Trouver la somme (X) et la somme des carrés (Y) pour les 10 articles de données suivants:

10, 12, 9, 11, 13, 14, 11, 12, 9, 10

Le programme suivant serait nécessaire pour effectuer le calcul si seulement des variables numériques étaient utilisées:

```
10  A1=10 : A2=12 : A3=9 : A4=11 : A5=13          } Affecte des valeurs à chaque
20  A6=14 : A7=11 : A8=12 : A9=9 : A10=10         } variable
30  X=A1+A2+A3+A4+A5+A6+A7+A8+A9+A10.....Calcule la somme
40  Y=A1^2+A2^2+A3^2+A4^2+A5^2+A6^2+
      A7^2+A8^2+A9^2+A10^2.............................Calcule la somme des carrés
```

Le programme devient beaucoup plus simple lorsqu'un tableau est utilisé.

```
10  DIM A (10)....Déclare un tableau
20  A (1) =10 : A (2) =12 : A (3) =9 : A (4)=11 : A (5)=13   } Affecte des valeurs au
30  A (6) =14 : A (7) =11 : A (8)=12 : A (9)=9 : A (10)=10   } tableau
40  X=0 : Y=0
50  FOR I=1 TO 10
60  X=X+A ( I ) : Y=Y+A ( I )^2   } Calcule la somme et la somme des carrés
70  NEXT I
```

Au premier coup d'oeil, le tableau peut apparaître plutôt difficile à utiliser, mais il rend en fait la programmation plus simple lorsque de larges volumes de données sont affectés.

EXEMPLE:
100 articles de données

Variables numériques

```
10  A1=61 : A2=38 : A3=90 : A4=37 : A5=99
20  A6=12 : A7=17 : A8=94 : A9=39 : A10=75
30  A11=24 : A12=84 : A13=46 : A14=18 : A15=55
[illegible]  A18=51 : A19=91 : A20=[illegible]
150 A71=31 : A77=69 : A[illegible]=30
160 A76=40 : A77=69 : A78=51 : A19=91 : A20=30   } Affecte des valeurs aux
170 A81=91 : A82=46 : A83=23 : A84=37 : A85=84   } variables
180 A86=65 : A87=23 : A88=98 : A89=51 : A90=30
190 A91=57 : A92=78 : A93=16 : A94=39 : A95=46
200 A96=59 : A97=24 : A98=32 : A99=74 : A100=47
210 X=A1+A2+A3+.............+A49+A50   } Calcule la somme
220 X=X+A51+A52+..........+A99+A100
230 Y=A1^2+A2^2+.........+A39^2+A40^2   } Calcule la somme des
240 Y=Y+A41^2+A42^2+..........+A79^2+A80^2   } carrés
250 Y=Y+A81^2+A82^2+..........+A99^2+A100^2
```

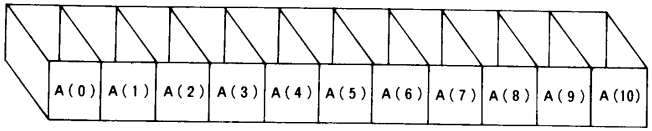
Tableau

```
10  DIM A (100) }   Déclare un tableau
20  FOR I=1 TO 100 : READ A (I) : NEXT I }   Affecte des valeurs au tableau
30  X=0 : Y=0
40  FOR I=1 TO 100                   } Calcule la somme et la somme des carrés
50  X=X+A (I) : Y=Y+A (I)^2          }
60  NEXT I
70  DATA 61, 38, 90, 37, 99          }
80  DATA 12, 17, 94, 39, 75          }
90  DATA 24, 84, 46, 18, 55          }
100 DATA 46, 65, 51, 91, 30          } Données
[illegible] 26, 11, 88, 78           }
220 DATA 47, [illegible]             }
230 DATA 91, 46, 28, 31, 64          }
240 DATA 65, 23, 98, 51, 30          }
250 DATA 57, 78, 16, 39, 46          }
260 DATA 59, 24, 32, 74, 47          }
```

Un regard à ces programmes révèle qu'une augmentation dans les données n'impose virtuellement aucun changement dans la portion qui calcule la somme et la somme des carrés. Les seuls changements seraient dans les lignes 10, 20 et 40 où la constante serait modifiée de 10 à 100.

A nouveau, le concept du tableau peut être mieux saisi en pensant à des boîtes. Auparavant, une simple variable était décrite comme une boîte unique. Les tableaux, d'un autre côté, seraient une série de boîtes numérotées qui forment un ensemble.

EXEMPLE:
Tableau A (10)



Le tableau A (10) contient en fait un total de onze boîtes, numérotées de A (0) à A (10), avec chaque boîte capable de contenir une valeur différente, de la manière illustrée ci-dessus. Le terme réel utilisé pour se référer à une boîte est "élément". Le rappel d'une valeur sauvegardée est effectué en spécifiant simplement le numéro de l'élément correspondant.

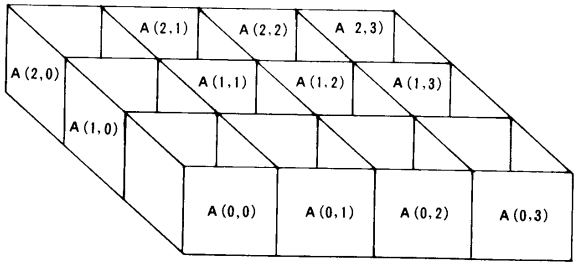
EXEMPLE:
Rappeler la valeur sauvegardée dans l'élément 4 du tableau A

Y = (4)
ou
X = 4 : Y = A (X)

La valeur qui spécifie un élément dans un tableau (4 ci-dessus) est appelée un indice inférieur.

Jusqu'à maintenant, les seuls tableaux couverts ont été ceux formés par une seule ligne d'éléments ou "boîtes". Ceux-ci sont connus sous le nom de tableaux "à une dimension". Les tableaux peuvent également contenir plus d'une dimension avec des éléments connectés verticalement et horizontalement en tableaux à deux dimensions et à trois dimensions.

EXEMPLE:
DIM A (2, 3)



La déclaration de cet exemple constitue un tableau de trois lignes et quatre colonnes, le rendant capable de sauvegarder 12 différentes valeurs.

Tableaux numériques et tableaux de chaîne de caractères
Comme avec les variables simples, les tableaux peuvent aussi être déclarés pour contenir des chaînes de caractères en utilisant le symbole "$" après le nom de variable de tableau. A nouveau se souvenir que les valeurs numériques ne peuvent pas être affectées aux tableaux de chaînes de caractères et que les chaînes de caractères ne peuvent pas être affectées aux tableaux numériques.

EXEMPLE:
La procédure suivante est utilisée pour déclarer un tableau et sauvegarder les données de cinq individus et leurs points marqués pendant un certain jeu.

Tableau de chaîne de caractères N$ (5) déclaré pour les noms
Tableau numérique P (5) déclaré pour les points

```
10  DIM N$ (5), P (5)..............Déclaration des tableaux pour sauvegarder les noms et les
20  FOR I = 1 TO 5                 points
30  READ A$, X
40  N$ ( I ) = A$.......................Sauvegarde les noms au tableau de chaîne de caractères
50  P ( I ) = X..........................Sauvegarde les points au tableau numérique
60  NEXT I
70  END
80  DATA SMITH, 70, BROWN, 68, JONES, 87, CARTER, 80, MILLS, 74
```

6-7-3 Sommaire

Types de variables

Les trois types de variables suivants sont disponibles pour être utilisés avec cet appareil.

1. Variables numériques (mantisses jusqu'à 12 chiffres)		A, a, CHIFFRE, POINTS
2. Variables de chaîne de caractères (jusqu'à 255 caractères)		A$, CHAINES DE CARACTERE$
3. Variables de tableau	Tableau numérique	A (10), XX (3, 3, 3)
	Tableau de chaîne de caractères	A$ (10), TABLEAU$ (2, 2)

Noms de variables

- Les noms de variables peuvent comprendre des caractères majuscules, minuscules ou numériques, mais un caractère numérique ne peut pas être utilisé à la première position du nom de la variable (p.ex. 1AE, 3BC$ sont interdits).
- Les mots réservés (voir page 426) ne peuvent pas être utilisés comme caractères de tête d'un nom de variable (p.ex. RUNON, LIST1$ sont interdits).
- La longueur maximale d'un nom de variable est de 15 caractères.

Tableaux

1. Les tableaux sont déclarés par des instructions DIM.
2. Les éléments sont décrits par des indices inférieurs qui sont des nombres entiers supérieurs à 0. Les fractions sont négligées.
3. Le nombre des dimensions est limité par la capacité de la pile.
4. La valeur maximum des indices inférieurs est limitée par la capacité de la mémoire.

Application de variable/tableau

1. Les variables et les tableaux peuvent être utilisés ensemble par toutes les zones de programme.
2. Les tableaux ne peuvent pas être utilisés à moins d'avoir au préalable été déclarés en utilisant l'instruction DIM.

Comptage d'octets utilisés par les variables

Les points suivants décrivent le nombre d'octets réservés lorsqu'une variable apparaît pour la première fois dans un programme.

• **Variables numériques**
(longueur de nom de variable + 12) octets dans zone de variables

• **Variables de chaîne de caractères**
(longueur de nom de variable + 4) octets dans la zone de variables et (longueur de chaîne de caractères + 1) octets dans la zone de chaîne de caractères.
Des zones sont réservées pour les variables de tableau lorsque le tableau est déclaré par l'instruction DIM.

• **Variables de tableaux numériques**
(longueur de nom de variable + 4) + (taille du tableau × 8) + (dimension × 2 + 1) octets dans la zone de variables

EXEMPLE:
DIM XYZ (3, 3, 5, 2)

Nom	:	3
Taille	:	$4 \times 4 \times 6 \times 3 = 288$
Dimension	:	4
Calcul	:	$(3+4)+(288 \times 8)+(4 \times 2+1)=2320$ octets

• **Variables de tableau de chaîne de caractères**
(longueur de nom de variable + 4) + (taille du tableau) + (dimension × 2) octets dans la zone de variables.
Les longueurs de chaînes de caractères individuelles sont nécessaires dans la zone de variables lorsque des chaînes de caractères sont affectées au tableau.

EXEMPLE:
10 DIM AB$ (3, 3)
20 AB$ (0, 0) = " * * * * * "

Nom	:	2
Taille	:	$4 \times 4 = 16$
Dimension	:	2
Calculs	:	$(2+4)+16+(2 \times 2)+5$ octets

Calcul de la longueur de programme

Les lignes suivantes indiquent les points dont on doit tenir compte lors du calcul des besoins de mémoire pour les programmes.

Numéros de lignes	:	2 octets par numéro de lignes, quelle que soit la longueur du numéro (1 ~ 65535)
Commandes	:	2 octets par commande
Fonctions	:	2 octets par fonction
Caractères numériques/ alphabétiques	:	1 octet par caractère (les espaces sont aussi comptés comme caractères)
Touche EXE	:	1 octet par opération de la touche EXE à la fin d'une ligne de programme (pour la sauvegarde de la ligne)

1 octet ajouté à la somme du total ci-dessus

EXEMPLE:
10 A = SIN X
2 (numéro de ligne) + 1 (espace après le numéro de la ligne) + 1 (A) + 1 (=) + 2 (SIN) + 1 (espace) + 1 (X) + 1 (EXE) + 1 = 11

Ce calcul indique qu'un total de 11 octets est nécessaire pour la sauvegarde du programme ci-dessus.
* L'espace situé après le numéro de ligne est automatiquement ajouté.

6-8 SAUVEGARDE ET CHARGEMENT DE PROGRAMMES

Les procédures suivantes de sauvegarde et de chargement ne peuvent être effectuées que lorsque l'unité d'interface FA-6 est utilisée.

6-8-1 Sauvegarde de programmes

Les programmes sauvegardés dans la mémoire de l'appareil sont protégés par une pile de protection de mémoire, même lorsque l'interrupteur d'alimentation de l'appareil est mis sur la position "OFF". Cependant, tout le contenu de la mémoire est effacé lorsque les piles de l'alimentation principale et de protection de mémoire sont enlevées en même temps de l'appareil ou lorsque la commande NEW ALL est exécutée. Le contenu de la zone de programme peut être sauvegardé sur des bandes cassettes standard pour se protéger contre des pertes de données importantes ou pour faire de la place pour une autre programmation lorsque toutes les zones de programme sont pleines. Les deux commandes suivantes sont disponibles pour ces opérations de sauvegarde.

SAVE	:	Sauvegarde le contenu de la zone actuelle de programme.
SAVE ALL	:	Sauvegarde tout le contenu de toutes les zones de programme.

EXEMPLE:
Dans ce cas, l'exécution de SAVE permet de sauvegarder le contenu de la zone de programme P0, alors que SAVE ALL sauvegarderait les contenus des zones de programme P0 à P9.

SAVE EXE

```
SAVE
```
(Sauvegarde le programme dans la zone de programme 0.)

```
SAVE
Ready P0
```
(La sauvegarde est terminée.)

Les noms de fichier d'un maximum de huit caractères de long peuvent également être affectés aux programmes sauvegardés sur bandes cassettes en utilisant les commandes SAVE et SAVE ALL.

SAVE "BASIC" EXE

```
SAVE"BASIC"
```
(Sauvegarde le programme sous le nom de fichier "BASIC".)

```
SAVE"BASIC"
Ready P0
```
(La sauvegarde est terminée.)

6-8-2 Vérification de programme

La commande VERIFY permet de vérifier si oui ou non le programme sauvegardé en utilisant SAVE ou SAVE ALL a été correctement copié sur la bande cassette.

EXEMPLE:
Vérifier la bonne sauvegarde du programme BASIC.

VERIFY "BASIC" EXE

VERIFY"BASIC"	(Vérification du programme sauvegardé)
BASIC B	(Trouve le programme spécifié et vérifie)
BASIC B Ready P0	(Vérification terminée)

Si le message "Ready" n'apparaît pas après quelque temps, vérifier si oui ou non le nom de fichier entré avec la commande VERIFY est correct. S'il est correct, régler le niveau du volume du magnétocassette utilisé et répéter la procédure de vérification.

```
P0 error      0
Ready P0
```

Le message d'erreur illustré ci-dessus indique que le programme n'a pas été correctement sauvegardé. Dans ce cas, vérifier les éléments suivants:

- Revérifier le programme, cette fois en ajoutant "CAS1:" avant le nom du fichier (VERIFY "CAS1 : BASIC" dans l'exemple ci-dessus).
- S'assurer que les connexions entre l'ordinateur et le magnétocassette sont correctes et solides.
- S'assurer que le niveau du volume du magnétocassette est réglé à proximité de son maximum.
- Vérifier si la bande cassette est endommagée.
- Vérifier si les têtes du magnétocassette sont sales.

Noter aussi qu'une erreur sera générée si un programme existe sur la bande avec le même nom que celui actuellement présent dans la mémoire de l'ordinateur, mais le contenu des deux programmes est différent.

* La commande VERIFY détermine automatiquement si le programme vérifié a été sauvegardé à l'aide de la commande SAVE ou SAVE ALL.

6-8-3 Chargement de programmes

Les programmes sauvegardés sur bandes cassettes en utilisant les commandes SAVE et SAVE ALL peuvent être chargés dans l'ordinateur en utilisant les commandes LOAD et LOAD ALL.

EXEMPLE:
Charger le programme "BASIC" de la bande cassette dans la mémoire.

LOAD EXE

LOAD	(Commande de chargement de programme)
BASIC B	(Nom de fichier de programme)
Ready P0	(Chargement terminé)

Noter que l'exécution des commandes LOAD et LOAD ALL alors que des programmes sont déjà sauvegardés dans la mémoire permet d'effacer le contenu actuel de la mémoire.

La commande LOAD ALL peut être utilisée pour charger des programmes vers toutes les zones de programme (P0 ~ P9). La spécification d'un nom de fichier dans les commandes LOAD et LOAD ALL entraîne la recherche par l'appareil du nom de fichier spécifié pour le chargement dans la mémoire. Le tableau suivant indique la relation entre les commandes LOAD, LOAD ALL, SAVE et SAVE ALL.

	LOAD	LOAD "nom de fichier"	LOAD ALL	LOAD ALL "nom de fichier"
SAVE	○	×	×	×
SAVE "nom de fichier"	○	○	×	×
SAVE ALL	×	×	○	×
SAVE ALL "nom de fichier"	×	×	○	○

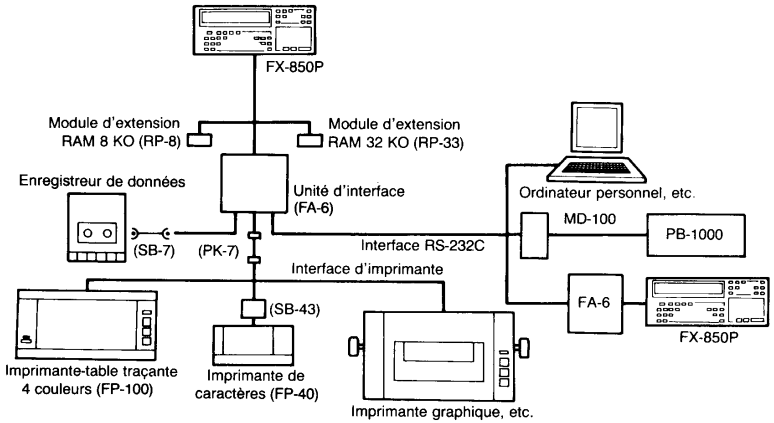
NOTE:
Voir la PARTIE 7 "DISPOSITIFS PERIPHERIQUES" pour les détails sur l'utilisation des commandes SAVE et LOAD.

PARTIE 7

DISPOSITIFS PERIPHERIQUES

Une foule de dispositifs périphériques sont disponibles pour la connexion à cet appareil afin de garantir encore plus de puissance de calcul.

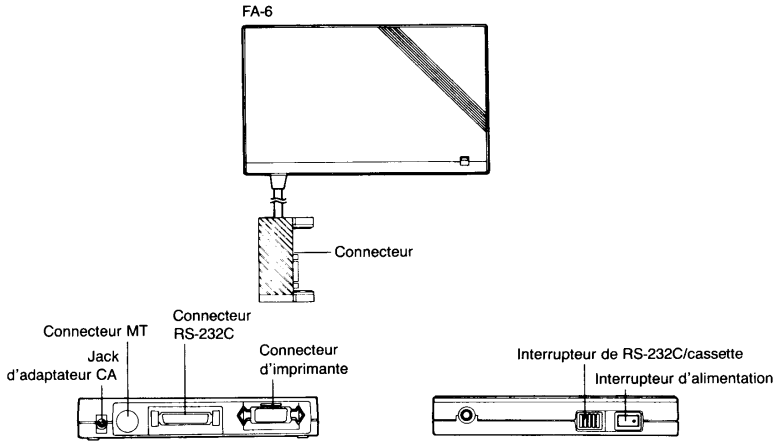
Configuration du système



7-1 UNITE D'INTERFACE POUR CASSETTE FA-6

7-1-1 Caractéristiques

La FA-6 est une unité d'interface qui rend possible la connexion d'un magnétocassette pour être utilisé comme dispositif externe de mise en mémoire de données. A part le fait d'être une interface pour cassette, la FA-6 est équipée d'une interface RS-232C et une interface d'imprimante norme Centronics.
Le connecteur d'interface RS-232C, l'interface d'imprimante, l'interface pour cassette et le jack d'adaptateur CA sont situés à l'arrière de la FA-6. Le compartiment des piles est situé au bas de l'appareil. Les piles sont chargées en retirant le couvercle du compartiment des piles et en insérant les piles tout en s'assurant que leurs polarités (⊕ ⊖) sont de la manière illustrée dans le compartiment.

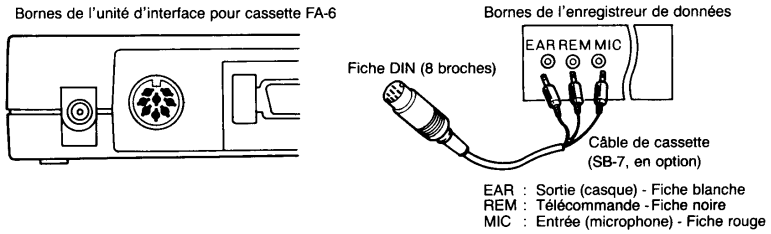


7-1-2 Connexions

S'assurer de mettre l'interrupteur d'alimentation de l'ordinateur et de l'unité d'interface sur la position "OFF" avant procéder aux connexions. Après les connexions, l'interrupteur d'alimentation de l'ordinateur doit être mis en premier sur la position "ON", puis celui de l'unité d'interface.

7-1-3 Interface pour cassette

L'interface pour cassette est utilisée pour la connexion d'un magnétocassette afin de permettre la sauvegarde de programmes sur bandes cassettes et la recharge ultérieure des programmes dans la mémoire de l'ordinateur. La connexion au magnétocassette est accomplie en utilisant le câble de connexion SB-7 en option. La fiche rouge est insérée dans le jack MIC ou LINE IN (la dénomination varie selon le type de magnétocassette utilisé) du magnétocassette, alors que la fiche blanche est insérée dans le jack EAR ou LINE OUT du magnétocassette. La fiche noire est insérée dans le jack REM de magnétocassettes munis d'une fonction de télécommande.



NOTE:
La fiche de la télécommande n'est pas utilisée lorsque le magnétocassette utilisé n'est pas muni d'une fonction de télécommande.
Le magnétocassette doit être réglé au mode d'enregistrement lors de l'exécution d'enregistrement de programmes ou de données. Pour le chargement de programmes, régler le magnétocassette dans le mode de lecture après avoir exécuté la commande LOAD.

Sauvegarde de programme unique

SAVE "descripteur de fichier" [EXE] (le descripteur de fichier peut être omis)
Le descripteur de fichier peut contenir tous les symboles, caractères ou chiffres (sauf les guillemets).

EXEMPLE:
SAVE "CAS0: AD1990" [EXE]
* "CAS0:" peut être omis.

Chargement de programme unique

LOAD "descripteur de fichier" [EXE] (le descripteur de fichier peut être omis)
Si un descripteur de fichier n'est pas spécifié, l'appareil charge le premier programme trouvé sur la bande.

EXEMPLE:
LOAD "CAS0: AD1990" [EXE]
* "CAS0:" peut être omis.

Chargement/sauvegarde de tous les programmes

SAVE ALL "descripteur de fichier" [EXE]
LOAD ALL "descripteur de fichier" [EXE]
(le descripteur de fichier peut être omis)

La commande SAVE ALL sauvegarde tous les programmes mis en mémoire dans les zones de programme P0 à P9 sur la bande cassette. D'un autre côté, LOAD ALL charge les programmes sauvegardés en utilisant la commande SAVE ALL. La commande LOAD ALL efface également tout contenu présent dans les zones de programme et le remplace par les programmes en provenance de la bande cassette.

Sauvegarde et chargement des données de la BANQUE DE DONNEES

Les commandes SAVE# et LOAD# sont utilisées pour la sauvegarde et le chargement des données en mémoire sauvegardées dans la BANQUE DE DONNEES. La procédure pour l'utilisation de ces commandes est identique à celle décrite pour SAVE et LOAD ci-dessus.

Vérification de fichier sauvegardé

La commande VERIFY vérifie si le programme ou les données sauvegardées sur la bande cassette correspondent exactement au contenu actuel de la mémoire.
VERIFY "descripteur de fichier" (le descripteur de fichier peut être omis)

Opération du magnétocassette

Le magnétocassette doit être réglé au mode d'enregistrement avant que la commande SAVE ne soit exécutée. Pour le chargement de programmes, régler le magnétocassette au mode de lecture et exécuter ensuite la commande LOAD.
Les commandes SAVE, SAVE ALL, LOAD, LOAD ALL, SAVE# et LOAD# ne peuvent pas être utilisées dans le mode CAL ou à l'intérieur de programmes. Toutefois, elles peuvent être utilisées pendant les opérations manuelles dans le mode BASIC.
* Voir la PARTIE 9 "ELEMENTS FONDAMENTAUX DE TRAITEMENT DE FICHIERS" pour les détails sur les descripteurs de fichier.

7-1-4 Interface RS-232C

Interrupteur du RS-232C

L'interface RS-232C peut être utilisée pour les communications de données après avoir réglé l'interrupteur du RS-232C/cassette sur la position "RS-232C".

Spécifications

Méthode de communication	:	Départ-arrêt (asynchrone) uniquement en mode duplex complet (full-duplex)
Vitesse de transmission	:	150, 300, 600, 1200, 2400, 4800 bauds
Bit de parité	:	Impair, Pair, Sans
Longueur du bit de caractère	:	7 ou 8 bits
Bits d'arrêt	:	1 ou 2 bits
Commande de signal CTS	:	Commande/pas de commande
Commande de signal DSR	:	Commande/pas de commande
Commande de signal CD	:	Commande/pas de commande
Commande occupée	:	Commande/pas de commande XON/XOFF
Système de code d'entrée/sortie	:	Commande/pas de commande de SI/SO

Configuration de broches

N° de borne	Nom de signal	Connexion de broche	E/S
1	FG		
2	TXD		SORTIE
3	RXD		ENTREE
4	RTS		SORTIE
5	CTS		ENTREE
6	DSR		ENTREE
7	GND		
8	DCD		ENTREE
9	NC		
10	NC	1 13	
11	NC		
12	NC		
13	NC		
14	NC		
15	NC	25 14	
16	NC		
17	NC		
18	NC		
19	NC		
20	DTR		SORTIE
21	NC		
22	NC		
23	NC		
24	NC		
25	NC		

* Voir la PARTIE 9 "ELEMENTS FONDAMENTAUX DE TRAITEMENT DE FICHIERS" pour les détails sur l'utilisation de l'interface RS-232C.

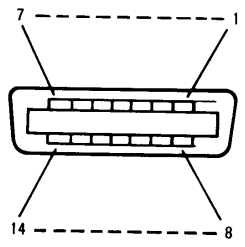
7-1-5 Interface Centronics (Interface d'imprimante)

Généralités

L'interface Centronics est utilisée pour sortir les résultats du traitement de données ou les listes de programmes sur une imprimante. Toutes les imprimantes Centronics peuvent être connectées à l'ordinateur par l'unité d'interface FA-6.

Configuration des broches

N° de borne	Nom de signal	Connexion de broche
1	$\overline{\text{PSTB}}$	
2	PDB0	
3	PDB1	
4	PDB2	
5	PDB3	
6	PDB4	
7	PDB5	
8	PDB6	
9	PDB7	
10	NC	
11	BUSY	
12	NC	
13	NC	
14	GND	



Commandes d'imprimante utilisées en BASIC

Commande	Fonction
LLIST	Sort le contenu du programme sur l'imprimante
LPRINT	Sort les caractères spécifiés sur l'imprimante
TAB	Sort des espaces jusqu'à une position spécifiée sur l'imprimante

Mode PRT ON (activation d'impression)

Le réglage de l'appareil au mode PRT ON (MODE 7), puis l'exécution des commandes PRINT, LIST ou VARLIST permet d'imprimer les résultats de l'exécution de commande et des données objet spécifiées par la commande TRON sur l'imprimante. L'inclusion de M O D E 7 dans un programme permet d'imprimer tout le contenu de commandes d'impression successives. Le mode PRT ON peut être annulé par MODE 8 ou par M O D E 8 (PRT OFF).

* Voir le manuel FA-6 pour les détails sur son fonctionnement correct.

7-2 IMPRIMANTE-TABLE TRACANTE (FP-100)

La FP-100 est une imprimante-table traçante 4 couleurs capable d'impression sur papier A-4. La FP-100 possède à la fois un mode de caractères et un mode de graphiques qui lui permet d'imprimer virtuellement tout type de sortie d'ordinateur.

Mode de caractères : Listes de programme, résultats de calculs
Mode de graphiques : Graphiques produits par commandes graphiques

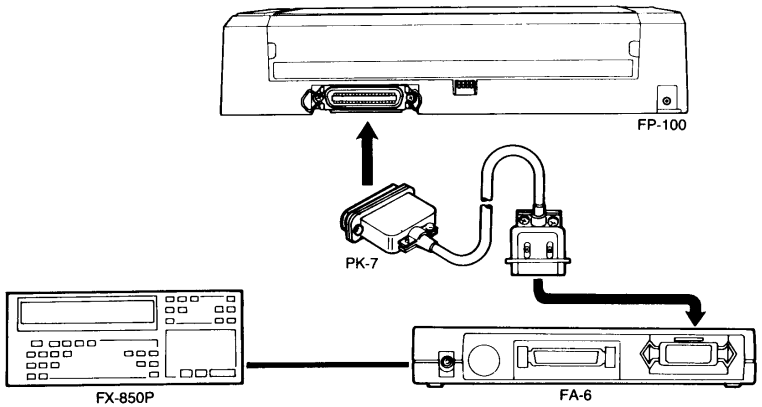
7-2-1 Caractéristiques

Impression en quatre couleurs: noir, rouge, bleu et vert
Effets de caractères : Italiques
Résolution d'impression : 0,1 mm/pas
Largeur de papier : Format carte postale au format lettre
Taille de caractères : 1,0mm × 1,2mm (S0, 0) à 16,0mm × 19,2mm (S15, 15); 256 types

7-2-2 Connexions

Cet appareil est connecté à la FP-100 par l'intermédiaire de l'unité d'interface FA-6 et du câble d'imprimante PK-7.

* N'utiliser que le câble d'imprimante CASIO PK-7.



7-2-3 Impression des données

Les listes de programmes sont sorties sur l'imprimante en utilisant la commande LLIST en BASIC. L'exécution de la commande LLIST imprime le programme actuellement accédé.
La commande LPRINT est utilisée pour imprimer des données à l'intérieur d'un programme, alors que l'exécution de la commande LLIST# dans le mode BASIC (MODE 1) imprime le contenu de la banque de données.

* Voir le manuel de la FP-100 pour les détails sur son fonctionnement correct.

7-3 IMPRIMANTE DE CARACTERES (FP-40)

La FP-40 peut être utilisée pour imprimer des données ou des listes de programmes écrits sur le FX-850P.

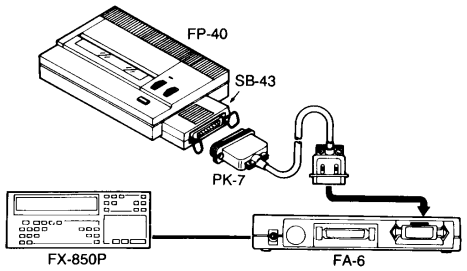
* L'addition d'un module d'interface en option permet aussi l'utilisation de la FP-40 avec les ordinateurs des séries PB-100 et PB-700.

7-3-1 Caractéristiques

Méthode d'impression	:	Système d'impression thermique (sans frappe)
Colonnes	:	Standard 40 (mode normal) Maximum 80 (mode 80CHR)
Vitesse d'impression	:	Environ 0,65 lignes/sec (mode normal)
Avance de papier	:	1/6 pouce ou 1/9 pouce
Rouleau de papier	:	Largeur 112mm, dia. ext. 30mm max.; papier thermique (TRP-112)

7-3-2 Connexion

Cet appareil est connecté à la FP-40 par l'intermédiaire de l'unité d'interface FA-6, du câble d'imprimante PK-7 et du module d'interface SB-43.

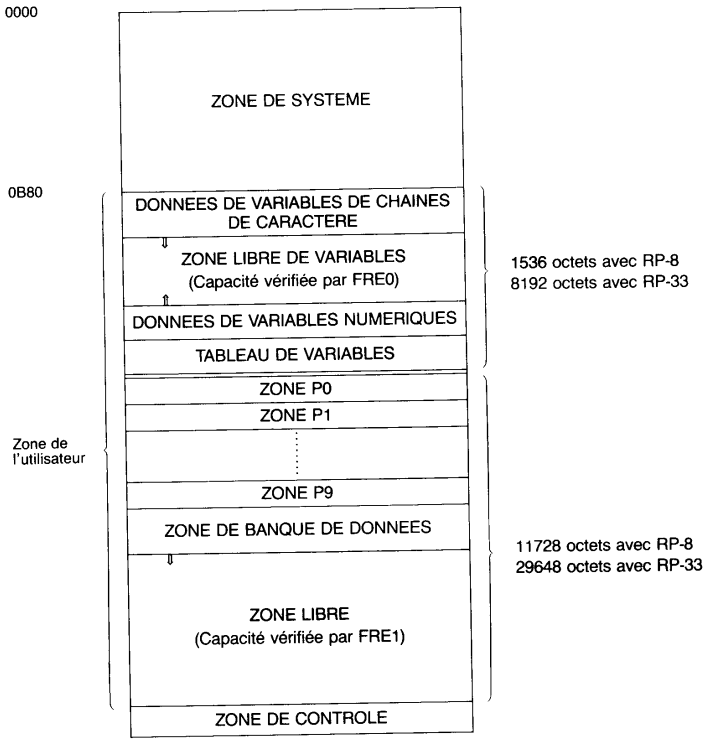


* Voir le manuel de la FP-40 pour les détails sur son fonctionnement correct.

7-4 MODULE D'EXTENSION RAM (mémoire vive) (RP-8 (8 KO)), (RP-33 (32 KO))

Cet appareil est livré équipé d'une RAM standard de 8 KO. Des modules d'extension RAM sont également disponibles en option pour des programmes plus importants et pour le traitement de larges quantités de données. L'addition du module RAM RP-8 étend la capacité de la mémoire à 16 KO, alors que le module RAM RP-33 l'étend à 40 KO.

7-4-1 Carte de mémoire d'extension



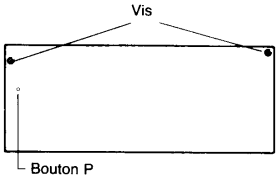
7-4-2 Manipulation des modules RAM

Préparation

Les charges d'électricité statique peuvent endommager les circuits internes des modules RAM. Toujours toucher une poignée de porte ou tout autre objet métallique pour décharger l'électricité statique avant de manipuler les modules RAM.

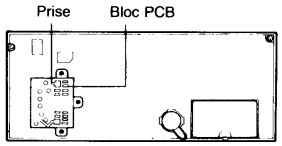
Procédure

① Mettre l'interrupteur d'alimentation de l'appareil sur la position "OFF".
② Enlever le couvercle arrière de l'appareil après avoir retiré les deux vis le maintenant en place.

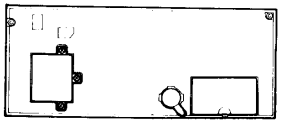


③ Insérer le module RAM dans la prise prévue dans l'appareil et le fixer en place en utilisant les trois vis prévues.

* Ne jamais toucher le connecteur du module RAM ni le bloc de PCB (circuit imprimé).



④ Remettre le couvercle arrière de l'appareil en place et le fixer à l'aide des deux vis.
⑤ Mettre l'interrupteur d'alimentation de l'appareil sur la position "ON" et appuyer sur le bouton P. Appuyer ensuite sur le bouton ALL RESET et mettre l'interrupteur d'alimentation de l'appareil sur la position "OFF" et puis à nouveau sur la position "ON".



* L'insertion ou le retrait du module RAM sans appuyer sur le bouton P et sur le bouton ALL RESET peut avoir pour résultat une altération du contenu de la mémoire de l'appareil.
* De la saleté, de la poussière ou des empreintes digitales sur le connecteur du module RAM ou le bloc de PCB peuvent entraîner une mauvaise connexion et un mauvais fonctionnement. Ne jamais toucher aux connecteurs.
* Toujours ranger les modules RAM dans leurs boîtes d'origine lorsqu'ils sont retirés de l'ordinateur et les mettre à l'abri de la poussière et de la saleté.

PARTIE 8

COMPATIBILITE DE LA SERIE PB-100

Cet appareil est capable de charger des données et des programmes écrits pour les ordinateurs de la série PB-100* et d'exécuter des programmes du PB-100. Toutefois, certaines commandes spéciales sont nécessaires pour permettre la compatibilité des programmes.

* La série PB comprend les modèles suivants:
PB-100, PB-100F, PB-110, PB-220, PB-240, PB-410, FX-720P, FX-730P, FX-770P, FX-785P, FX-790P

8-1 ENTREE/EDITION DES PROGRAMMES DE LA SERIE PB-100

Les conversions suivantes sont nécessaires pour permettre l'exécution des programmes de la série PB-100 sur cet appareil.

SERIE PB-100	FX-850P
STAT, STATLIST, STATCLEAR EOX, EOY	Utiliser la fonction de bibliothèque ou créer un programme.
PUT, GET	Changer en utilisant OPEN "CAS0:" Exemple 1: PUT$→OPEN "CAS0:" FOR OUTPUT AS#1: PRINT#1, A$:CLOSE Exemple 2: GET$→OPEN "CAS0:" FOR INPUT AS#1: INPUT #1, A$:CLOSE
≦	<=, =<
≠	>< , <>
≧	>=, =>
↑	^
π	PI

Il est recommandé d'effectuer également les conversions des commandes suivantes pour garantir la compatibilité entre les programmes de la série PB-100 et les programmes du FX-850P.

COMMANDES DE LA SERIE PB-100	COMMANDES DU FX-850P
VAC	CLEAR
IF ~ ; ~	IF ~ THEN ~
CSR	LOCATE ou TAB
KEY, KEY$	INKEY$
RND	ROUND
MID (emplacement, nombre de caractères)	MID$ ($, emplacement, nombre de caractères)
GOTO (expression numérique), GOSUB (expression numérique)	ON ~ GOTO, ON ~ GOSUB
MODE 4/5/6	ANGLE 0/1/2
MODE 7/8	LPRINT

Bien que l'entrée directe des caractères de la série PB-100 ne soit pas possible avec cet appareil, ils peuvent être affichés en utilisant la fonction CHR$.

EXEMPLE:

Affichage ≦
PRINT CHR$ (&HE1)

Voir le "TABLEAU DES CODES DE CARACTERES" à la page 421 pour des détails sur les codes de caractères.

NOTES

- Un espace doit être compris avant le THEN d'une instruction IF ~ THEN lorsque le caractère précédent le THEN est alphabétique.

  Exemple: IF3 = ATHEN → IF3 = A␣THEN

- La destination de saut d'une instruction GOTO ou GOSUB doit être entourée de parenthèses lorsqu'il s'agit d'une expression commençant par une valeur numérique.

  Exemple: GOTO 10∗A → GOTO (10∗A)

- Un espace doit être compris avant le TO dans la partie TO d'une boucle FOR ~ NEXT lorsque la valeur initiale est représentée par un caractère alphabétique.

  Exemple: FORA = BTOC → FORA = B␣TOC

- Une instruction INPUT suivie d'une virgule ne donnera pas de point d'interrogation comme question. Le point d'interrogation n'est affiché que lorsque l'instruction INPUT est suivie d'un point-virgule.

8-2 EXECUTION DES PROGRAMMES DE LA SERIE PB-100

8-2-1 Mode DEFM

La série PB-100 utilise les variables A ~ Z et A$ ~ Z$ comme tableaux. Le mode DEFM peut être utilisé lors de l'exécution des programmes de la série PB-100 sur cet appareil afin d'utiliser A ~ Z et A$ ~ Z$ de la même manière que l'ordinateur de la série PB-100. [D] [E] [F] [M] **nombre de variables à étendre** [EXE] effectuent l'extension de variables comme les ordinateurs de la série PB-100.

* Les programmes de la série PB-100 qui utilisent les tableaux DEFM doivent toujours être exécutés dans le mode DEFM.

Les variables de cet appareil deviennent ainsi lorsque les tableaux du mode DEFM sont utilisés.

		Variables de mode DIM
A (0)		→ A
A (1)	B (0)	→ B
⋮	⋮	⋮
A (25)	B (24)	→ Z
A (26)	B (25)	→ A (0)
A (27)	B (26)	→ A (1)
⋮	⋮	⋮
A$ (0)		→ A$
A$ (1)	B$ (0)	→ B$
⋮	⋮	
A$ (25)	B$ (24)	→ Z$
A$ (26)	B$ (25)	→ A$ (0)
A$ (27)	B$ (26)	→ A$ (1)
⋮	⋮	⋮

* A et A$ sont indépendants l'un de l'autre.

Le mode DEFM est annulé en déclarant un tableau à l'aide de l'instruction DIM ou par DIM [EXE] .

EXEMPLE:

DIM A (m, n, o) [EXE]	Déclaration du tableau A à 3 dimensions (m, n, o)
DIM [EXE]	Annule seulement le mode DEFM.
Mode DEFM → Mode DIM Mode DIM → Mode DEFM DEFM → DEFM n [EXE]	A ce moment, les variables A ( ) et A$ ( ) sont effacées. Ceci ne peut pas être utilisé à l'intérieur d'une boucle FOR ~ NEXT.

* n : nombre de variables à étendre

8-2-2 Tableaux en utilisant l'instruction DEFM

En cas de définition des tableaux en utilisant l'instruction DEFM, une zone de variables doit être réservée pour les variables utilisées, faute de quoi une erreur OM est générée (erreur de dépassement de mémoire) lorsque DEFM est spécifiée ou lorsque le programme est exécuté. Dans le cas d'une erreur OM, l'instruction CLEAR doit être utilisée pour réserver une zone de variables avant la prochaine spécification DEFM. Les lignes suivantes indiquent le calcul de la taille de la zone de variables nécessaire.

Variable $	:	62 + 3 (octets)
Variables A ~ Z	:	(8 + 4) × nombre de variables (octets)
Variables A$ ~ Z$	:	(8 + 5) × nombre de variables (octets)
Extension de variables	:	7 + 8 × nombre de variables à étendre + 7 + 9 × nombre de variables à étendre

8-2-3 Affichages du mode DEFM

L'utilisation de la spécification DEFM dans l'exécution manuelle (entrée de touche directe) permet d'afficher le nombre de variables compris dans le tableau DEFM. Cet affichage DEFM n'apparaît pas lorsque DEFM est spécifiée à l'intérieur d'un programme.

EXEMPLE:

[D] [E] [F] [M] [EXE]
```
A--Z:26   DEFM:0
```

[D] [E] [F] [M] **10** [EXE]
```
A--Z:26   DEFM:10
```

8-2-4 Commande CLEAR, commande DIM, commande DEFM dans mode DIM et mode DEFM

L'exécution de l'instruction CLEAR avec cet appareil permet d'effacer le contenu des variables et de réserver une zone de variables. Dans chaque mode, l'exécution de cette commande donne les résultats suivants.

Opération	Mode DIM	Mode DEFM
[SHIFT] [CLEAR] [EXE] ([V] [A] [C])	Efface le contenu des variables.	Efface le contenu des variables.
[SHIFT] [CLEAR] Zone de variables [EXE]	Efface le contenu des variables et réserve la zone de variables.	Efface le contenu des variables, annule le mode DEFM et réserve la zone de variables.
[D] [E] [F] [M] [EXE]	Passe dans le mode DEFM et exécute internement ERASE A, A$: DIM A ( ), A$ ( ).	Affiche le nombre de tableaux lorsqu'exécutée manuellement.
[D] [I] [M] [EXE]	Exécute DIM.	Annule DEFM et exécute ERASE A, A$.

* La spécification DEFM n'est pas nécessaire pour les programmes du FX-790P et du FX-730P qui utilisent les instructions DIM.

8-3 CHARGEMENT DES PROGRAMMES DE LA SERIE PB-100

Les commandes suivantes sont exécutées dans le mode BASIC pour charger les programmes de la série PB-100 à partir d'une bande cassette.

PBLOAD : Fichier de programme
PBLOAD ALL : Tous les fichiers

L'exécution de ces commandes permet de charger les programmes et de les modifier automatiquement pour permettre l'exécution sur le FX-850P.

EXEMPLE:
PBLOAD à partir d'une bande cassette contenant les programmes suivants:

TEST 1 (fichier de mémoire)
TEST 2 (fichier de données)
TEST 3 (tous les fichiers)
TEST 4 (fichier de programme)

[P] [B] [L] [O] [A] [D] [SHIFT] ["] [T] [E] [S] [T] **4** [SHIFT] ["] [EXE]

```
TEST1     MF
```
```
TEST2     VF
```
```
TEST3     AF
```
Recherche

```
TEST4     PF
```
Chargement

```
TEST4     PF
Converting...
```
Conversion de codes

```
Converting...
Ready P0
```
Fin

8-4 LECTURE DES DONNEES DE LA SERIE PB-100

Les commandes suivantes sont disponibles pour la lecture des fichiers de données de la série PB-100 et des fichiers de la BANQUE DE DONNEES.

PBLOAD # (fichier de la BANQUE DE DONNEES)
PBGET (fichier de données)

EXEMPLE:
Exécuter PBGET pour le nom de fichier TEST 5 contenant les données $, A, B, C, D.

[P] [B] [G] [E] [T] [SHIFT] ["] [T] [E] [S] [T] **5** [SHIFT] ["] [EXE]

```
TEST5     VF
```
```
GET $.4data
Converting...
```
```
Converting...
Ready P0
```

* Le programme, les données, les données de la BANQUE DE DONNEES et tous les fichiers sauvegardés sur la bande cassette en utilisant cet appareil ne peuvent pas être lus par les ordinateurs de la série PB-100.

8-5 FORMAT DE COMMANDES

PBLOAD { **0** / **1** } **["nom de fichier"]**

1. 0 : phase positive 1 : phase inverse (valeur par défaut = phase positive)
   Tentative d'utilisation de la phase opposée au réglage du courant si des problèmes sont rencontrés pendant les opérations PBLOAD.
2. Cette commande charge le fichier sous le nom de fichier spécifié dans la zone actuelle de programme.
3. Cette commande doit être exécutée dans le mode BASIC.
4. L'exécution de cette commande lit le programme spécifié de la bande et le convertit du format de la série PB-100 au format du FX-850P.
5. L'exécution de cette commande peut être terminée à tout moment par une pression sur la touche [BRK].

PBLOAD ALL $\left[\left\{\begin{matrix}0\\1\end{matrix}\right\}\right]$ **["nom de fichier"]**

1. Cette commande charge tous les fichiers sous le nom de fichier spécifié dans les zones de programme P0 à P9.

L'opération est identique à PBLOAD.

PBLOAD# $\left[\left\{\begin{matrix}0\\1\end{matrix}\right\}\right]$ **["nom de fichier"] [, M]**

1. Cette commande lit les données sous le nom de fichier spécifié dans la BANQUE DE DONNEES.
2. La spécification [, M] ajoute les données lues à la fin des données actuellement sauvegardées dans la BANQUE DE DONNEES.

L'opération est identique à PBLOAD.

PBGET $\left[\left\{\begin{matrix}0\\1\end{matrix}\right\}\right]$ **["nom de fichier"]**

1. Cette commande lit le contenu du fichier de données sous le nom de fichier spécifié et l'affecte à la variable $, au tableau A( ) ou au tableau A$( ).
2. Les données de chaînes de caractères et les données numériques sont automatiquement affectées au tableau correspondant adéquat.
3. Cette commande peut être exécutée dans le mode BASIC et le mode CAL, mais doit être précédée par une déclaration de tableau en utilisant l'instruction DIM ou l'instruction DEFM.
4. Les données sont automatiquement converties (Converting···affiché) avant d'être affectées aux variables.

NOTES

- La suspension de l'exécution de PBLOAD, PBLOAD ALL, PBLOAD# et PBGET à l'aide de la touche STOP et ensuite la reprise de l'opération doivent être évitées.
- Une erreur PO (erreur de programme) ou une erreur DA (erreur de données) est générée lorsque le chargement de programme ou de données n'est pas réussi en utilisant une commande PB.
- Les variables suivantes sont utilisées comme zones de travail lorsque l'une des commandes PB est utilisée:

  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
  a$, b$, c$, d$, e$, f$, g$, h$, i$, j$, k$, l$, m$, n$,
  o$, p$

- Les commandes STAT, EOX, EOY, PUT et GET sont converties au code ??? lorsque chargées vers un ordinateur de la série PB. L'exécution des programmes avec ce code génère une erreur SN (erreur de syntaxe). Voir la section 8-1 pour les détails sur la conversion manuelle du code ???.
- L'exécution d'une commande PB en étant dans le mode PRT (symbole PRT sur l'affichage) annule automatiquement le mode PRT.

PARTIE 9

ELEMENTS FONDAMENTAUX DE TRAITEMENT DE FICHIERS

9-1 CLASSEMENT DES DISPOSITIFS

A part l'exécution des programmes actuellement sauvegardés dans la mémoire, cet appareil peut également utiliser des bandes cassettes pour la sauvegarde des données et des programmes. Les données et les programmes peuvent aussi être échangés avec d'autres dispositifs par l'intérmédiaire d'un circuit de communications. Les commandes OPEN, CLOSE, PRINT#, INPUT#, SAVE et LOAD sont utilisées dans ce but.

• Noms des dispositifs

En cas d'utilisation des commandes notées ci-dessus pour des changements internes de fichiers, il faut tout d'abord spécifier si la bande cassette ou le circuit de communications doit être utilisé. Ceci est appelé spécification d'un "nom de dispositif". Le tableau suivant indique les noms de dispositif disponibles.

NOM DE DISPOSITIF	SIGNIFICATION
CAS0:	Magnétocassette (phase positive)
CAS1:	Magnétocassette (phase inverse)
COM0:	Circuit de communications (RS-232C)

• Noms de fichier

Une fois que le dispositif à utiliser est déterminé, l'étape suivante à faire consiste à affecter un "nom de fichier" au fichier. Un nom de fichier peut être une combinaison quelconque de caractères numériques et alphabétiques d'une longueur maximum de huit caractères.

L'appareil néglige automatiquement toute entrée de nom de fichier dépassant les huit premiers caractères.

* Un seul fichier peut être ouvert à la fois. Un programme ou sous-programme, qui ouvre et ferme successivement des fichiers au fur et à mesure qu'ils sont nécessaires, doit être préparé lorsque de multiples fichiers doivent être accédés.

• Descripteurs de fichier

Un "descripteur de fichier" est en fait une combinaison d'un nom de dispositif et d'un nom de fichier. En cas d'utilisation du circuit de communications, le descripteur de fichier spécifie également les divers paramètres de communications en plus du nom de dispositif et du nom de fichier.

9-2 DESCRIPTEURS DE FICHIER

9-2-1 Bandes cassettes

Les lignes suivantes donnent le descripteur de fichier lorsqu'une bande cassette est utilisée pour la sauvegarde de fichiers.

{ CAS0 : / CAS1 : } [(S) / (F)] Nom de fichier

Spécification de phase { CAS0 : Phase positive / CAS1 : Phase inverse }

Spécification de vitesse { S : 300 bauds (300 bits/seconde) / F : 1200 bauds (1200 bits/seconde) }

EXEMPLE 1:
Lecture des données d'un fichier nommé TEST, en phase positive, 300 bauds
OPEN "CAS0 : (S) TEST" FOR INPUT AS #1
Dans ce cas, le descripteur de fichier est CAS0 : (S) TEST.

EXEMPLE 2:
Ecriture de données vers un fichier nommé SAMPLE, en phase positive, 1200 bauds
SAVE "CAS0 : (F) SAMPLE"
Dans ce cas, le descripteur de fichier est CAS0 : (S) SAMPLE.

EXEMPLE 3:
Lecture d'un programme sous le nom de fichier AD1990
LOAD "AD1990"
Ceci est identique à LOAD "CAS0 : (F) AD1990"

Les commandes du descripteur de fichier pouvant être utilisées de la manière décrite ci-dessus sont:
OPEN, LOAD, SAVE, LOAD#, SAVE# et VERIFY

* CAS0 : et CAS1 : spécifient la phase de lecture (positive/inverse) des données enregistrées sur bande.
  CAS0 : est généralement utilisé, mais CAS1 : peut être utilisé pour certains magnétocassettes.
* Un attribut est automatiquement affecté lorsqu'un programme ou des données sont écrites sur la bande cassette. Les attributs peuvent être vérifiés lorsque la bande est lue.

ATTRIBUT	SIGNIFICATION
B	Fichier binaire (fichier de programme)
A	Tous les fichiers (fichier de programme sauvegardé en utilisant la commande SAVE ALL)
S	Fichier séquentiel (fichier de données, fichier de mémoire ou fichier de programme sauvegardé en format ASCII)

9-2-2 Circuit de communications

Le descripteur de fichier pour le changement interne des données du circuit de communications est le suivant:

COMO : [[Vitesse], [Parité], [Donnée], [Arrêt], [CS], [DS], [CD], [Occupé], [Code]]

① Vitesse (vitesse de transmission: vitesse de transfert de données)
- 6 : 4800 (4800 bits/sec.)
- 5 : 2400 (2400 bits/sec.)
- 4 : 1200 (1200 bits/sec.)
- 3 : 600 (600 bits/sec.)
- 2 : 300 (300 bits/sec.)
- 1 : 150 (150 bits/sec.)

NOTE: 2400 (5) ou moins doit être spécifié lorsque le circuit de communications est spécifié en utilisant la commande OPEN.

② Parité (bit de parité: bit de vérification pour l'envoi de données)
- N : Sans bit de parité
- E : Parité paire
- O : Parité impaire

③ Données (bit de données: nombre de bits représentant un caractère)
- 7 : 7 bits JIS (7 bits/seconde)
- 8 : 8 bits JIS (8 bits/seconde)

④ Arrêt (bit d'arrêt: bit(s) à la fin d'un caractère signifiant la fin du caractère)
- 1 : 1 bit
- 2 : 2 bits

⑤ CS (Prêt pour l'envoi (CTS) : fonction de commande pour informer le dispositif partenaire si oui ou non les données peuvent être envoyées)
- C : Utilisé — Le dispositif d'émission attend jusqu'à ce que CS soit activé.
- N : Inutilisé

⑥ DS (Ens. données prêt (DSR) : fonction pour informer le fonctionnement normal du dispositif partenaire)
- D : Utilisé — Une erreur NR est générée lorsque des données sont reçues alors que DS est désactivé. Lors de l'envoi, le dispositif attend jusqu'à ce que DS soit activé.
- N : Inutilisé

⑦ CD (Détection de porteuse (CD) : fonction pour informer que le dispositif partenaire est prêt à recevoir des données)
- C : Utilisé — Une erreur NR est générée lorsque des données sont reçues alors que CD est désactivé.
- N : Inutilisé

⑧ Occupé (Occupé, (XON/XOFF) : fonction pour suspendre provisoirement l'envoi de données)
- B : Utilisé — Une demande d'interruption d'envoi est envoyée au dispositif partenaire lorsque le nombre total de caractères restants à lire est inférieur à 64 pendant la réception de données. Pendant l'envoi de données, l'envoi est provisoirement suspendu lorsqu'une demande d'interruption d'envoi est reçue du dispositif partenaire. L'envoi est repris après une demande de début d'envoi.
- N : Inutilisé

⑨ Code (Système de code d'entrée/sortie (SI/SO) : Utilisé pour envoyer un caractère exprimé en huit bits lorsque les bits de données sont spécifiées en sept bits)

S : Utilisé — Cette fonction n'est applicable que lorsque le paramètre du bit de données (③ ci-dessus) est spécifié en sept bits. Dans ce cas, le code SO (0EH) est envoyé avant les codes 80H ou plus pour passer dans le mode SO. Les codes 7FH ou moins sont envoyés précédés du code SI (0FH) pour passer dans le mode SI.
La spécification S doit être utilisée chaque fois qu'il est nécessaire d'envoyer des données avec des codes 80H ou plus comme données à 7 bits.

N : Inutilisé

Valeurs par défaut des paramètres

COM0	:	2, E, 8, 1, N, N, N, B, N
Vitesse de transmission	:	300 bauds (300 bits/seconde)
Parité	:	Paire
Bits de données	:	8 bits
Bit d'arrêt	:	1 bit
CS	:	Non utilisé
DS	:	Non utilisé
CD	:	Non utilisé
Occupé	:	XON/XOFF
Code	:	SI/SO (Entrée/sortie) — Non utilisé

EXEMPLE 1:
Envoyer la chaîne de caractères notée ci-dessous vers RS-232C en utilisant les paramètres donnés.

Vitesse de transmission	:	300 bauds → 2
Parité	:	Paire → E
Bits de données	:	8 bits → 8
Bit d'arrêt	:	1 bit → 1
CS	:	Non utilisé → N
DS	:	Non utilisé → N
CD	:	Non utilisé → N
Occupé	:	Utilisé → B
SI/SO	:	Non utilisé → N

```
10 OPEN "COM0 : 2, E, 8, 1, N, N, B, N" AS #1
20 PRINT #1, "HELLO."
30 CLOSE
```

EXEMPLE 2:
Recevoir la chaîne de caractères de RS-232C de la manière mentionnée ci-dessus.

```
10 OPEN "COM0 : 2, E, 8, 1, N, N, N, B, N" AS #1
20 INPUT #1, A$
30 CLOSE
```

* Les communications sont effectuées par l'intermédiaire de la borne RS-232C.
* Le changement interne de données est effectué en utilisant un circuit full-duplex (les deux côtés peuvent communiquer simultanément, comme un téléphone standard) avec départ-arrêt (circuit typique de communications entre ordinateur, aussi connu sous le nom d'asynchrone).

PARTIE 10

REFERENCE DE COMMANDES

ELEMENTS DE FORMAT

La méthode d'entrée des instructions est expliquée ci-dessous.

- Les mots en caractères gras sont des commandes ou des fonctions et ils doivent être entrés de la manière indiquée.
- Des accolades indiquent qu'un des paramètres situé entre elles doit être spécifié.
- Des virgules contenus dans les accolades doivent être écrites dans la position indiquée.
- Des crochets indiquent que les paramètres compris entre eux peuvent être omis. Les crochets eux-mêmes ne sont pas entrés.
- Un astérisque indique que le terme qui le précède peut apparaître plus d'une fois.
- Expressions numériques—Constantes, expressions et valeurs numériques (p.ex., 10, 10 + 25, A, coût de l'unité ∗ quantité)
- Expressions de chaîne de caractères—Constantes de chaîne de caractères, variables de chaîne de caractères et expressions de chaîne de caractères (p.ex., "ABC", A$ et A$ + B$)
- Expressions—Terme général pour les expressions numériques et les expressions de chaîne de caractère
- Arguments—Eléments utilisés par les commandes et les fonctions
- Ⓟ ..... Ne peut être exécuté que dans un programme.
- Ⓜ ..... Ne peut être exécuté que manuellement.
- Ⓐ ..... Ne peut être exécuté que manuellement et dans un programme.
- Ⓕ ..... Instruction de fonction ne pouvant être exécutée que manuellement et dans un programme.

EXEMPLE: Fonction MID$

MID$ (tableau de chaîne de caractères, , position [, nombre de caractères])
Expression de chaîne de caractères — Expression numérique — Expression numérique

Le terme "expression de chaîne de caractères" en dessous de "tableau de chaîne de caractères" décrit ce tableau. De même, "expression numérique" en dessous de "position" et "expression numérique" en dessous de "nombre de caractères" sont des descripteurs. Aussi, étant donné que la virgule et le nombre de caractères sont entourés de crochets, ils peuvent être omis.

EXEMPLE: Instruction GOSUB

GOSUB { numéro de ligne de branchement / Numéro de ligne ; # numéro de zone de programme / Caractère unique; 0 ~ 9 }

Cet exemple illustre deux descripteurs pour GOSUB: le numéro de la ligne du sous-programme à laquelle le programme est raccordé et le nom de fichier auquel le programme est raccordé.

A suivre….