

LE COMPAGNON DE VOYAGE DE LA HP48 G/GX

Paul
Courbis
&
Cyrille
de Brébisson
(HPMâd)

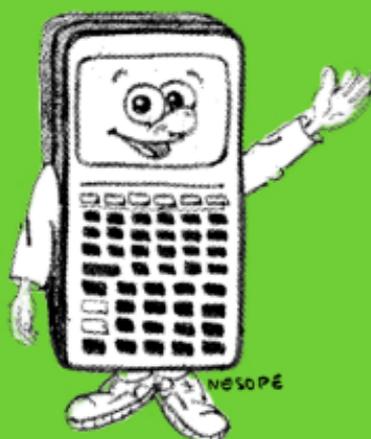


Table des Matières

Les objets de la HP48 :	Liste des objets	2
	System Binary	2
	Real	2
	Long Real	2
	Complex	3
	Long Complex	3
	Character	3
	Array	3
	Linked Array	4
	String	4
	Binary Integer	4
	List	4
	Directory	5
	Algebraic	6
	Unit	6
	Tagged	6
	Graphic	6
	Library	7
	Backup	8
	Library Data	9
	Extended Pointer	9
	Reserved 1, 2 et 3	9
	Program	9
	Code	9
	Global Name	9
	Local Name	10
	XLIB Name	10
Codage de KEYSTATE		10
La ram I/O		11
La mémoire vive :	Plan général de la ram	12
	La ram réservée	12
Le gestionnaire de bancs		15
Les drapeaux-système		16
Le SATURN :	Les champs des registres	17
	Les instructions du Saturn	17
Le clavier de la HP48 :	Touches α -shiftées	24
	Couples IN/OUT	24
	Codes buffer	24
Routines utiles :	Sauver et restaurer les registres	25
	Prendre des éléments sur la pile	25
	Mettre des éléments sur la pile	25
	Gestion du clavier	25
	Gestion de l'affichage	25
	Gestion de la mémoire	26
	Routines de calcul	26
	Routines diverses	26
Externals utiles :	Vérification des arguments	27
	Gestion de la pile	27
	Les entiers-système	27
	Les réels	27
	Les caractères	27
	Les chaînes	28
	Les entiers binaires	28
	Les variables globales	28
	Les variables locales	28
	Les objets composites	29
	Le graphisme	29
	Les XLIB Names	29
	Les drapeaux TRUE et FALSE	29
	Gestion du clavier	29
	Les structures conditionnelles	29
	Les boucles	30
	La génération d'erreurs	30
	Externals divers	30
Polices de caractères :	Codes ASCII (bin, hexa, décimal)	30
	Stockage en mémoire morte	30

Prologue	Objet	Type	Vérif.	Page
02911	System Binary	20	1F	2
02933	Real	0	1	2
02955	Long Real	21	3F	2
02977	Complex	1	2	3
0299D	Long Complex	22	4F	3
029BF	Character	24	6F	3
029E8	Array	3/4	4	3
02A0A	Linked Array	23	5F	4
02A2C	String	2	3	4
02A4E	Binary Integer	10	B	4
02A74	List	5	5	4
02A96	Directory	15	2F	5
02AB8	Algebraic	9	9	6
02ADA	Unit	13	E	6
02AFC	Tagged	12	D	6
02B1E	Graphic	11	C	6
02B40	Library	16	8F	7
02B62	Backup	17	9F	8
02B88	Library Data	26	AF	9
02BAA	Extended Pointer	27	BF	9
02BCC	Reserved 1	27	CF	9
02BEE	Reserved 2	27	DF	9
02C10	Reserved 3	27	EF	9
02D9D	Program	8	8	9
02DCC	Code	25	7F	9
02E48	Global Name	6	6	9
02E6D	Local Name	7	7	10
02E92	XLIB Name	14	0F	10

System Binary

@	Prologue (02911)	5 q
@+5h	Contenu	5 q
@+Ah		

Real

@	Prologue (02933)	5 q
@+5h	Exposant	3 q
@+8h	Mantisse	12 q
@+14h	Signe	1 q
@+15h		

Long Real

@	Prologue (02955)	5 q
@+5h	Exposant	5 q
@+Ah	Mantisse	15 q
@+19h	Signe	1 q
@+1Ah		

Complex

@	Prologue (02977)		5 q
@+5h	Exposant 1	Partie	3 q
@+8h	Mantisse 1		12 q
@+14h	Signe 1	Réelle	1 q
@+15h	Exposant 2	Partie	3 q
@+18h	Mantisse 2		12 q
@+24h	Signe 2	Imaginaire	1 q
@+25h			

Long Complex

@	Prologue (0299D)		5 q
@+5h	Exposant 1	Partie	5 q
@+Ah	Mantisse 1		15 q
@+19h	Signe 1	Réelle	1 q
@+1Ah	Exposant 2	Partie	5 q
@+1Fh	Mantisse 2		15 q
@+2Eh	Signe 2	Imaginaire	1 q
@+2Fh			

Character

@	Prologue (029BF)		5 q
@+5h	Caractère		2 q
@+7h			

Array

@	Prologue (029E8)		5 q
@+5h	Longueur totale hors-prologue l_t		5 q
@+Ah	Type des objets		5 q
@+Fh	Nombre d de dimensions		5 q
@+14h	Dimension 1 (d_1)		5 q
@+d*5+Fh	Dimension d (d_d)		5 q
@+d*5+14h	Élément 1 (objet sans prologue)		
	Élément d_2+1 (objet sans prologue)		
	Élément $d_1 \cdot \dots \cdot d_d$ (objet sans prologue)		
@+l _t +5h			

Linked Array

@	Prologue (02A0A)	5 q
@+5h	Longueur totale hors-prologue l_t	5 q
@+Ah	Type des objets	5 q
@+Fh	Nombre d de dimensions	5 q
@+14h	Dimension 1 (d_1)	5 q
@+d*5+Fh	Dimension d (d_d)	5 q
@+d*5+14h	Pointeur sur élément 1	5 q
@pn	Pointeur sur élément n ($@_{on} - @_{pn}$)	5 q
	Pointeur sur élément $d_1 * \dots * d_d$	5 q
	Élément 1 (objet sans prologue)	
@on	Élément n (objet sans prologue)	
	Élément $d_1 * \dots * d_d$ (objet sans prologue)	
@+l _t +5h		

String

@	Prologue (02A2C)	5 q
@+5h	Longueur totale hors-prologue l_t	5 q
@+Ah	Premier caractère	2 q
@+l _t -2h	Dernier caractère	2 q
@+l _t +5h		

Binary Integer

@	Prologue (02A4E)	5 q
@+5h	Longueur totale hors-prologue l_t	5 q
@+Ah	Valeur de l'entier	$l_t - 5$ q
@+l _t +5h		

List

@	Prologue (02A74)	5 q
@+5h	Premier objet	
	Dernier objet	
	Épilogue (0312B)	5 q

Directory

La structure du HOME-directory est la suivante :

@	Prologue (02A96)	5 q
@+5h	Nombre de librairies attachées n_l	3 q
@+8h	N° Librairie 1	3 q
@+Bh	Adresse Hash Table 1	5 q
@+10h	Adresse Liste Messages 1	5 q
	N° Librairie n_l	3 q
	Adresse Hash Table n_l	5 q
	Adresse Liste Messages n_l	5 q
@0	Offset sur dernier objet (@d-@0)	5 q
@0+5h	00000	5 q
@1=@0+7h	Nombre n_1 de caractères du nom 1	2 q
@1+2h	Caractère 1	2 q
	nom 1 (nom	
	de l'objet 1)	
@1+2*n ₁	Caractère n_1	2 q
@1+2*n ₁ +2h	Nombre n_1 de caractères du nom 1	2 q
@1+2*n ₁ +4h	Objet 1	
@t ₁ =@2-5h	Taille zone précédente (@t ₁ -@1)	5 q
@2	Nombre n_2 de caractères du nom 2	2 q
@d	Nombre n_d de caractères du nom d	2 q
@d+2h	Caractère 1	2 q
	nom d (nom	
	de l'objet d)	
@d+2*n _d	Caractère n_d	2 q
@d+2*n _d +2h	Nombre n_d de caractères du nom d	2 q
@d+2*n _d +4h	Objet d	

Pour un sous-directory :

@	Prologue (02A96)	5 q
@+5h	Numéro de la librairie attachée	3 q
@+8h	Offset sur dernier objet (@d-@1)	5 q
@+Dh	00000	5 q
@+12h	Nombre n_1 de caractères du nom 1	2 q

Algebraic

@	Prologue (02AB8)	5 q
@+5h	Premier objet	
	Dernier objet	
	Épilogue (0312B)	5 q

Unit

@	Prologue (02ADA)	5 q
@+5h	Objet impliqué	
	Desc 1	Description de
		l'unité
	Desc n	
	Épilogue (0312B)	5 q

Opération	Adresse
-	#10B86h
*	#10B5Eh
/	#10B68h
^	#10B72h

Tagged

@	Prologue (02AFC)	5 q
@+5h	Longueur l_t du tag	2 q
@+7h	Caractère 1	Caractères
		du tag
@+ l_t *2+5h	Caractère l_t	2 q
@+ l_t *2+7h	Objet "taggué"	

Graphic

@	Prologue (02B1E)	5 q
@+5h	Longueur totale hors-prologue l_t	5 q
@+Ah	Nombre n_l de lignes (en pixels)	5 q
@+Fh	Nombre n_c de colonnes (en pixels)	5 q
@+14h	Colonnes 1 à 8	Pixels de la
		Ligne 1
	Derniers pixels	
	Colonnes 1 à 8	Pixels de la
		Ligne n_l
	Derniers pixels	
@+ l_t +5h		

Library

@	Prologue (02B40)	5 q
@+5h	Longueur totale hors-prologue l _t	5 q
@+Ah	Nombre n _c de caractères du nom	2 q
@+Ch	Caractère 1 _____ Caractères	2 q
	_____ du nom	
@+n _c *2+Ah	Caractère n _c _____	2 q
@+n _c *2+Ch	Nombre n _c de caractères du nom	2 q
@+n _c *2+Eh	Library number	3 q
@1	Offset to Hash Table (@h-@1)	5 q
@2	Offset to Message Array (@m-@2)	5 q
@3	Offset to Link Table (@l-@3)	5 q
@4	Offset to Config Object (@c-@4)	5 q
@h	Hash Table	
@m	Message Array	
@l	Link Table	
@o1-7/9h	Type XLIB ₁ (commande/fonction)	1 / 3 q
@o1-6h	Numéro de librairie de XLIB ₁	3 q
@o1-3h	Numéro de commande de XLIB ₁	3 q
@o1	Objet(s) XLIB ₁	
@on-7/9h	Type XLIB _n (commande/fonction)	1 / 3 q
@on-6h	Numéro de librairie de XLIB _n	3 q
@on-3h	Numéro de commande de XLIB _n	3 q
@on	Objet(s) XLIB _n	
@o(n+1)	Autre objet 1 _____ Autres objets	
	_____ (non visibles)	
@o(n+m)	Autre objet m _____	
@c	Config Object (non visible)	
@+l _t +1h	Checksum (CRC)	4 q
@+l _t +5h		

Message Array

Il s'agit d'un vecteur (objet "array" [voir page 3] ou "linked array" [voir page 4] à une dimension) de chaînes de caractères.

Link table

@l	Prologue (02A4E)	5 q
@l+5h	Longueur totale hors-prologue l _l	5 q
@l1	Offset pour l'objet 1 (@o1-@l1)	5 q
@ld	Offset pour l'objet d (@od-@ld)	5 q
@l+l+5h		

Hash Table

@h	Prologue (02A4E)	5 q
@h+5h	Longueur totale hors-prologue l _h	5 q
@c1	Offset pour classe 1 (@n1-@c1)	5 q
@c16	Offset classe 16 (@n16-@c16)	5 q
@+5Ah	Longueur l _n de la liste des noms	5 q
@n1	Nombre de caractères du nom 1	2 q
@n1+2h	Premier caractère _____ Caractères	2 q
	Dernier caractère _____ du nom 1	2 q
	Numéro de commande 1	3 q
@nx	Nombre de caractères du nom x	2 q
@nx+2h	Premier caractère _____ Caractères	2 q
	Dernier caractère _____ du nom x	2 q
	Numéro de commande x	3 q
@+l _n +5Ah	Offset nom de commande 1	5 q
@ox	Offset nom de commande x (@ox-@nx)	
	Offset nom dernière commande	
@h+l _h +5h		

Backup

@	Prologue (02B62)	5 q
@+5h	Longueur totale hors-prologue l _t	5 q
@+Ah	Nombre n _c de caractères	2 q
@+Ch	Caractère 1 _____ Nom de	2 q
	_____ l'objet	
@+n _c *2+8h	Caractère n _c _____	2 q
@+n _c *2+Ah	Nombre n _c de caractères	2 q
@+n _c *2+Ch	Premier objet sauvegardé	
	Dernier objet sauvegardé	
@+l _t +5h		

Backup standard

L'objet backup standard (utilisé par la HP48 pour les sauvegardes dans un port) ne contient que deux objets : l'objet sauvegardé suivi de son CRC stocké dans un objet "System Binary" (voir page 2).

Library Data

@	Prologue (02B88)	5 q
@+5h	Longueur totale hors-prologue l _t	5 q
@+Ah	Contenu	l _t -5 q
@+l _t +5h		

Extended Pointer

@	Prologue (02BAA)	5 q
@+5h	Adresse	5 q
@+Ah	Adresse routine d'accès	5 q
@+Fh		

Reserved 1, 2, et 3

@	Prologue (02BCC, 02BEE, 02C10)	5 q
@+5h	Longueur totale hors-prologue l _t	5 q
@+Ah	Contenu	l _t -5 q
@+l _t +5h		

Program

@	Prologue (02D9D)	5 q
@+5h	Premier objet	
	Dernier objet	
	Épilogue (0312B)	5 q

Code

@	Prologue (02DCC)	5 q
@+5h	Longueur totale hors-prologue l _t	5 q
@+Ah	Codes machine	l _t -5 q
@+l _t +5h		

Global Name

@	Prologue (02E48)	5 q
@+5h	Nombre n _C de caractères	2 q
@+7h	Caractère 1 _____ Caractères	2 q

@+n _C *2+5h	Caractère n _C _____ du nom	2 q
@+n _C *2+7h		

Local Name

@	Prologue (02E6D)	5 q
@+5h	Nombre n_C de caractères	2 q
@+7h	Caractère 1 Caractères	2 q
@+ n_C *2+5h	Caractère n_C du nom	2 q
@+ n_C *2+7h		

XLIB name

@	Prologue (02E92)	5 q
@+5h	Numéro de librairie	3 q
@+8h	Numéro de commande	3 q
@+Bh		

Codage de KEYSTATE

Adresse	Bit 3	Bit 2	Bit 1	Bit 0	Quartet
#8065Ch	.	SPC	+	ON	0d=0h
#8065Dh	3	-	'	Ø	1d=1h
#8065Eh	→	R	1	2	2d=2h
#8065Fh	4	5	6	×	3d=3h
#80660h	9	÷	↶	MTH	4d=5h
#80661h	α	SIN	7	8	5d=5h
#80662h	+/-	EEX	DEL	←	6d=6h
#80663h	√x	y ^x	1/x	ENTER	7d=7h
#80664h	↓	→	COS	TAN	8d=8h
#80665h	NXT	SIN	EVAL	+	9d=9h
#80666h	PRG	CST	VAR	↑	10d=Ah
#80667h	C	D	E	F	11d=Bh
#80668h				B	12d=Ch

La ram I/O

#00100h	Display	Marge gauche		
#00101h	Contraste écran			
#00102h				
#00103h	Balayage			
#00104h à	Calculateur de codes de redondance cyclique			
#00107h	Tests des batteries			
#00108h				
#00109h				
#0010Ah				
#0010Bh	Alert	Alpha	Right Shift	Left Shift
#0010Ch	Indicateurs		Transmitting	Busy
#0010Dh		Vitesse RS232c		
#0010Eh	Configuration de la détection de cartes			
#0010Fh	Informations sur les ports (HP48 GX)			
#00110h	Contrôle RS232c			
#00111h	Informations sur la réception RS232c			
#00112h	Informations sur l'émission RS232c			
#00113h	Annulation des erreurs de réception			
#00114h	Buffer d'entrée RS232c			
#00115h	Buffer de sortie RS232c			
#00116h	Buffer de sortie RS232c			
#00117h	Type d'interruption			
#00118h	Réception Infra-Rouge			
#00119h				
#0011Ah	Réception Infra-Rouge			
#0011Bh				
#0011Ch	Émission Infra-Rouge			
#0011Dh				Buffer IR
#0011Eh	Sauvegarde pour les interruptions			
#0011Fh	Adresse de base de la Ram interne			
#00120h à	Adresse du début de la bitmap écran			
#00124h	Adresse du début de la bitmap écran			
#00125h à	Marge à droite (en quartets)			
#00127h	Hauteur menu & Vsync			
#00128h				
#00129h	Cntrl Rom			
#0012Ah				
#0012Bh				
#0012Ch				
#0012Dh				
#0012Eh	Contrôle horloge 1			
#0012Fh	Contrôle horloge 2			
#00130h à	Adresse du début de la bitmap du menu			
#00134h				
#00135h				
#00136h				
#00137h	Horloge 1			
#00138h à	Horloge 2			
#0013Fh	Horloge 2			

Plan général de la ram

#8000h	Ram réservée	
(#806D0h)	GROBS écran	
(#806E9h)	Objets temporaires	
(#806EEh)	Return Stack	
B	Mémoire libre	≈ D*5 q
D1	Pile	(#8081Dh) q
(#806FDh)	Ligne de commande	48 q min.
(#80702h)	Pile d'undo et var. locales	
(#80707h)	Données boucles internes	
(#8070Ch)	Temporary environment	78 q
(#80711h)	Variables utilisateur	
(#80716h)	Sauvegardes en port 0	
(#807E8h)		

La ram réservée

#8000h	CMOS word		5 q
#80005h	Taille actuelle de la ram interne		5 q
#8000Ah	Taille normale de la ram interne		5 q
#8000Fh	Base de l'adresse de fin de la ram interne		1 q
#80010h	Type		1 q
#80011h	Date	WSLOG 1	13 q
#8001Eh	CRC		4 q
#80022h	Type		1 q
#80023h	Date	WSLOG 2	13 q
#80030h	CRC		4 q
#80034h	Type		1 q
#80035h	Date	WSLOG 3	13 q
#80042h	CRC		4 q
#80046h	Type		1 q
#80047h	Date	WSLOG 4	13 q
#80054h	CRC		4 q
#80058h	Valeur	Offset d'horloge	13 q
#80065h	CRC		4 q
#80069h	Prochaine extinction automatique		13 q
#80076h	Drapeau extinction automatique		1 q
#80077h	Drapeau pour le mode "coma"		1 q
#80078h	Auto-test start time		13 q
#80085h	Auto-test fail time		13 q
#80092h	Préparation affichage		44 q
#800BEh			35 q
#800E1h	Cartes enfichées (bits 0 et 1)		1 q
#800E2h			4 q
#800E6h	Sauvegarde #8065Ah (Disable On-x)		1 q
#800E7h	Sauvegarde ST(15)		1 q
#800E8h			13 q
#800F5h	Zone de transfert		532 q
#80309h	Transferts : quartets restants		5 q
#8030Eh			12 q

La mémoire vive de la HP48

#80319h	Données		512 q
#80519h	BufLen	Buffer	2 q
#8051Bh	BufFull		1 q
#8051Ch	BufStart	d'entrée	2 q
#8051Eh	État		1 q
#8051Fh	Demandé	de la prise	1 q
#80520h	Parité		1 q
#80521h		RS232c	2 q
#80523h	Tempo #110h		1 q
#80524h	Calcul conftab carte en port 1		2 q
#80526h	Calcul conftab carte en port 2		2 q
#80528h			3 q
#8052Bh	CRC de la table de configuration		4 q
#8052Fh	Drapeaux	Informations sur	1 q
#80530h	Taille	la carte en port 1	1 q
#80531h	Drapeaux	Informations	1 q
#80532h	Taille	sur la carte	1 q
#80533h	Nbr de bancs	en port 2	2 q
#80535h	Occupation #C0000h-#FFFFh		1 q
#80536h	Fin zone backup en port 0		5 q
#8053Bh	Fin zone backup en port 1		5 q
#80540h	Fin zone backup en port 2 / banc 1		5 q
#80545h	Fin zone backup en port 2 / banc 2		5 q
#805D6h	Fin zone backup en port 2 / banc 31		5 q
#805DBh	Sauvegardes pour les interruptions		103 q
#80642h	Masque de sortie pour test clavier		3 q
#80645h	Contenu de la bitmap menus (?)		5 q
#8064Ah	Adresse graphique PICT (?)		5 q
#8064Fh			6 q
#80655h	Vitesse machine		5 q
#8065Ah	Disable [ON]-[C/D/E/SPC]		1 q
#8065Bh	Disable keyboard		1 q
#8065Ch	Key state		13 q
#80669h	KeyStart	Buffer du	1 q
#8066Ah	KeyEnd	Clavier	1 q
#8066Bh	Codes Touches		32 q
#8068Bh	Sauvegarde touche (READKEY)		2 q
#8068Dh	Sauvegarde adr. bitmap écran (#00120h)		5 q
#80692h	Sauvegarde marge droite (#00125h)		3 q
#80695h	Sauvegarde adr. bitmap menu (#00130h)		5 q
#8069Ah	Sauvegarde hauteur menu (#00128h)		2 q
#8069Ch			52 q
#806D0h	Adresse GROB du menu		5 q
#806D5h	Adresse GROB de la pile		5 q
#806DAh	Adresse GROB courant		5 q
#806DFh	Adresse GROB non visible		5 q
#806E4h	Adresse GROB graphique (PICT)		5 q
#806E9h	Adresse début des objets temporaires		5 q
#806EEh	Adresse fin des objets temporaires		5 q
#806F3h	Adresse début de mémoire libre (B)		5 q
#806F8h	Adresse fin de mémoire libre (D1)		5 q
#806FDh	Adresse ligne de commande		5 q
#80702h	Adresse pile d'undo et variables locales		5 q
#80707h	Adresse données boucles internes		5 q
#8070Ch	Adresse temporary environment		5 q
#80711h	Adresse home directory		5 q
#80716h	Adresse zone de sauvegarde (port 0)		5 q

La mémoire vive de la HP48

#8071Bh	Adresse directory courant		5 q
#80720h	Adresse stockage tempo adresse directory		5 q
#80725h	Adresse userkeys		5 q
#8072Ah	Adresse alarmes		5 q
#8072Fh	Adresse objet à exécuter (D0)		5 q
#80734h	Zone de sauvegarde		5 q
#80739h	Adresse objet 1		5 q
#8073Eh	Adresse objet 2		5 q
#80743h	Adresse objet 3	Pile de LAST	5 q
#80748h	Adresse objet 4		5 q
#8074Dh	Adresse objet 5		5 q
#80752h	Adresse grand entier		5 q
#80757h			5 q
#8075Ch	Adresse commande 1	Pile des quatre	5 q
#80761h	Adresse commande 2	dernières	5 q
#80766h	Adresse commande 3	lignes de	5 q
#8076Bh	Adresse commande 4	commandes	5 q
#80770h			15 q
#8077Fh	Adresse error message		5 q
#80784h	Adresse application Display		5 q
#80789h	Adresse application Keys		5 q
#8078Eh	Adresse application Exit Condition		5 q
#80793h	Adresse application error		5 q
#80798h			5 q
#8079Dh	Menu courant		5 q
#807A2h	Last menu		5 q
#807A7h			5 q
#807ACh			5 q
#807B1h			5 q
#807B6h	Unshifted menu key routine		5 q
#807BBh	Left shifted menu key routine		5 q
#807C0h	Right shifted menu key routine		5 q
#807C5h	Review key		5 q
#807CAh	Adresse Last menu VAR		5 q
#807CFh			5 q
#807D4h	Exit action		5 q
#807D9h	Command number		5 q
#807DEh	Last RPL token		5 q
#807E3h			5 q
#807E8h	Adresse de fin de ram		5 q
#807EDh	Mémoire libre (en paquets de 5 quartets) (D)		5 q
#807F2h	Prochaine erreur à afficher		5 q
#807F7h	ATTN Counter		5 q
#807FCh	Adresse du noyau (#3858Eh)		5 q
#80801h	Drapeaux internes		12 q
#8080Dh			16 q
#8081Dh	StackSize		5 q
#80822h	Racine de génération aléatoire		16 q
#80832h			15 q
#80841h	Indicateurs		2 q
#80843h	Système	Drapeaux	16 q
#80853h	Utilisateur		16 q
#80863h			2 q
#80865h	Caractères à gauche (éditeur)		5 q
#8086Ah	Nombre de lignes (éditeur)		5 q
#8086Fh			5 q
#80874h	N° première ligne (éditeur)		5 q
#80879h			4 q
#8087Dh	Error number		5 q

La mémoire vive de la HP48

#80882h	Position du curseur dans la ligne (éditeur)		5 q
#80887h	Position verticale à l'écran (éditeur)		5 q
#8088Ch	Position horizontale à l'écran (éditeur)		2 q
#8088Eh	Curseur visible (éditeur)		1 q
#8088Fh	Code ascii caractère sous curseur (éditeur)		2 q
#80891h	Prologue	GROB du	5 q
#80896h	Longueur	caractère sous	5 q
#8089Bh	Taille (10 par 6)	le curseur	10 q
#808A5h	Pixels	(éditeur)	20 q
#808B9h	Position horiz. du curseur en pixels (éditeur)		5 q
#808BEh	Position vert. du curseur en pixels (éditeur)		5 q
#808C3h			12 q
#808CEh	Sauvegardes pour Garbage Collector		10 q
#808D8h			57 q
#80911h	Alpha-lock		1 q
#80912h	Program entry		1 q
#80913h			52 q
#80947h	Offset du menu courant		5 q
#8094Ch			5 q
#80951h	XLIB touche frappée		11 q
#8095Ch	Touche 1		11 q
#80967h	Touche 2	XLIBs	11 q
#80972h	Touche 3	d'exécution	11 q
#8097Dh	Touche 4	des touches	11 q
#80988h	Touche 5	de menu	11 q
#80993h	Touche 6		11 q
#8099Eh	Last menu offset		5 q
#809A3h	Nombre de librairies présentes		3 q
#809A6h	Numéro		3 q
#809A9h	Adresse infos	Info première	5 q
#809AEh	Adresse switcher	librairie	5 q
#809B3h			3 q
	Numéro		3 q
	Adresse infos	Info dernière	5 q
	Adresse switcher	librairie	5 q
			3 q
	Extended pointer (pour librairies attachées)		15 q
	Extended pointer (pour librairies attachées)		15 q

Le gestionnaire de bancs

#7F000h	Accès banc 0 du port 1 (?)		2 q
#7F002h	Accès banc 1 du port 1 (?)		2 q
#7F004h	Accès banc 2 du port 1 (?)		2 q
#7F03Eh	Accès banc 31 du port 1 (?)		2 q
#7F040h	Accès banc 0 du port 2		2 q
#7F042h	Accès banc 1 du port 2		2 q
#7F044h	Accès banc 2 du port 2		2 q
#7F07Eh	Accès banc 31 du port 2		2 q
#7F080h			

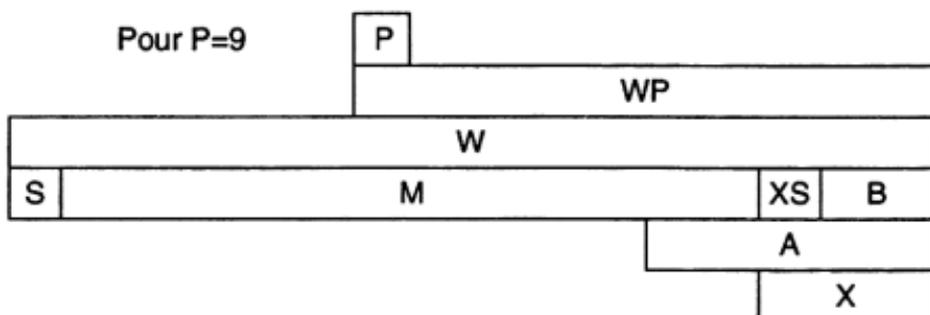
Les drapeaux-système de la HP48

N°	Signification	Désarmé	Armé
-1	Type de solution	Générale	Principale
-2	Constantes	Conservées	Evaluées
-3	Résultats	Symboliques	Numériques
-4	Non utilisé		
-5	Tailles des entiers (de 1 à 64, codée sur 4 bits). Le drapeau -5 est le bit de poids faible.		
-10			
-11	Base, stockée sur 2 bits : HEX (-11=1, -12=1), DEC (-11=0, -12=0), OCT (-11=1, -12=0), BIN (-11=0, -12=1)		
-12			
-13	Non utilisé		
-14	Paiements en fin/début	Fin	Début
-15	Type vecteurs 3D, stocké sur 2 bits : rectangulaire (-15=0, -16=0), cylindrique (-15=0, -16=1), sphérique (-15=1, -16=1)		
-16			
-17	Mesure des angles, stockée sur 2 bits : degrés (-17=0, -18=0), gradians (-17=1, -18=0), grades (-17=0, -18=1)		
-18			
-19	Résultat de $\rightarrow V2$	Vecteur	Complexe
-20	Underflow	0	Erreur
-21	Overflow	MAXR	Erreur
-22	Resultat infini	Erreur	MAXR
-23	Mis à un par un "underflow" négatif (si le flag -20 est armé)		
-24	Mis à un par un "underflow" positif (si le flag -20 est armé)		
-25	Mis à un par un "overflow" (si le flag -21 est armé)		
-26	Mis à un par un résultat infini (si le flag -22 est désarmé)		
-27	Nombres complexes	(x, y)	x+y*i
-28	Tracés multiples	En série	Simultanés
-29	Dessin des axes	Oui	Non
-30	Non utilisé		
-31	Type de tracé	Connecté	Non connecté
-32	Type de curseur graphique	Noir	Inversé
-33	Sortie I/O	Prise	IR
-34	Sortie imprimante	IR	Prise
-35	Transferts	ASCII	Binaires
-36	Réception	Nouveau nom	Écrase
-37	Impression	1 espace	2 espaces
-38	Saut de ligne ajouté ?	Oui	Non
-39	Messages I/O affichés ?	Oui	Non
-40	Affichage horloge ?	Non	Oui
-41	Heure	12 h	24 h
-42	Dates	mm/jj/aa	jj:mm:aa
-43	Alarme répétitive reprog. ?	Oui	Non
-44	Alarme passée	Annulée	Sauvegardée
-45	Nombre de décimales (modes FIX, SCI et ENG), de 0 à 11, le drapeau -45 est le bit de poids faible.		
-48			
-49	Affichage des nombres : STD (-49=0, -50=0), FIX (-49=1, -50=0), SCI (-49=0, -50=1), ENG (-49=1, -50=1).		
-50			
-51	Séparateur	Point	Virgule
-52	Affichage	Multiligne	Simple ligne
-53	Parenthèses	Minimum	Toutes
-54	Petit élément de matrice	0	Conservé
-55	Last arg.	Actif	Inactif
-56	Beep d'erreur	Actif	Inactif
-57	Beep d'alarme	Actif	Inactif
-58	INFO	Visible	Non visible
-59	Affichage contenu var.	Oui	Non
-60	Blocage du mode alpha	[α] [α]	[α]
-61	Blocage du mode user	[USER][USER]	[USER]
-62	Mode USER	Inactif	Actif
-63	ENTER revectorisé	Non	Oui
-64	GETI/PUTI cyclique ?	Non	Oui

Les instructions du SATURN

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

Pour P=9



Champ	a	f	b
P	0	0	8
WP	1	1	9
XS	2	2	A
X	3	3	B
S	4	4	C
M	5	5	D
B	6	6	E
W	7	7	F
A		F	

00	RTNSXM		11
01	RTN		11
02	* RTNSC		11
03	* RTNCC		11
04	SETHEX		4
05	SETDEC		4
06	RSTK=C		9
07	C=RSTK		9
08	CLRST		7
09	C=ST		7
0A	ST=C		7
0B	CSTEX		7
0C	* P=P+1		4
0D	* P=P-1		4
0Ef0	A=A&B	f	6+q
0Ef1	B=B&C	f	6+q
0Ef2	C=C&A	f	6+q
0Ef3	D=D&C	f	6+q
0Ef4	B=B&A	f	6+q
0Ef5	C=C&B	f	6+q
0Ef6	A=A&C	f	6+q
0Ef7	C=C&D	f	6+q
0Ef8	A=A!B	f	6+q
0Ef9	B=B!C	f	6+q
0EfA	C=C!A	f	6+q
0EfB	D=D!C	f	6+q
0EfC	B=B!A	f	6+q
0EfD	C=C!B	f	6+q
0EfE	A=A!C	f	6+q
0EfF	C=C!D	f	6+q
0F	RTI		11
100	R0=A		20.5
101	R1=A		20.5
102	R2=A		20.5
103	R3=A		20.5
104	R4=A		20.5
108	R0=C		20.5
109	R1=C		20.5

Les instructions du SATURN

10A	R2=C		20.5
10B	R3=C		20.5
10C	R4=C		20.5
110	A=R0		20.5
111	A=R1		20.5
112	A=R2		20.5
113	A=R3		20.5
114	A=R4		20.5
118	C=R0		20.5
119	C=R1		20.5
11A	C=R2		20.5
11B	C=R3		20.5
11C	C=R4		20.5
120	AR0EX		20.5
121	AR1EX		20.5
122	AR2EX		20.5
123	AR3EX		20.5
124	AR4EX		20.5
128	CR0EX		20.5
129	CR1EX		20.5
12A	CR2EX		20.5
12B	CR3EX		20.5
12C	CR4EX		20.5
130	D0=A		9.5
131	D1=A		9.5
132	AD0EX		9.5
133	AD1EX		9.5
134	D0=C		9.5
135	D1=C		9.5
136	CD0EX		9.5
137	CD1EX		9.5
138	D0=AS		8.5
139	D1=AS		8.5
13A	AD0XS		8.5
13B	AD1XS		8.5
13C	D0=CS		8.5
13D	D1=CS		8.5
13E	CD0XS		8.5
13F	CD1XS		8.5
140	DAT0=A	A	19.5
141	DAT1=A	A	19.5
142	A=DAT0	A	20.5,3.5
143	A=DAT1	A	20.5,3.5
144	DAT0=C	A	19.5
145	DAT1=C	A	19.5
146	C=DAT0	A	20.5,3.5
147	C=DAT1	A	20.5,3.5
148	DAT0=A	B	16.5
149	DAT1=A	B	16.5
14A	A=DAT0	B	19.5
14B	A=DAT1	B	19.5
14C	DAT0=C	B	16.5
14D	DAT1=C	B	16.5
14E	C=DAT0	B	19.5
14F	C=DAT1	B	19.5
150a	DAT0=A	a	19+q
151a	DAT1=A	a	19+q
152a	A=DAT0	a	20+q,1+q/2
153a	A=DAT1	a	20+q,1+q/2
154a	DAT0=C	a	19+q
155a	DAT1=C	a	19+q
156a	C=DAT0	a	20+q,1+q/2
157a	C=DAT1	a	20+q,1+q/2
158x	DAT0=A	x+1	18+q
159x	DAT1=A	x+1	18+q
15Ax	A=DAT0	x+1	19+q,1+q/2
15Bx	A=DAT1	x+1	19+q,1+q/2
15Cx	DAT0=C	x+1	18+q
15Dx	DAT1=C	x+1	18+q

Les instructions du SATURN

15Ex	C=DAT0	x+1	19+q, 1+q/2
15Fx	C=DAT1	x+1	19+q, 1+q/2
16n	* D0=D0+	n+1	8.5
17n	* D1=D1+	n+1	8.5
18n	* D0=D0-	n+1	8.5
19pq	D0=(2)	qp	6
1Apqrs	D0=(4)	srqp	9
1Bpqrst	D0=(5)	tsrqp	10.5
1Cn	* D1=D1-	n+1	8.5
1Dpq	D1=(2)	qp	6
1Epqrs	D1=(4)	srqp	9
1Fpqrst	D1=(5)	tsrqp	10.5
2n	P=	n	3
3xh0...hx	LCHEX	#hx...h0	3+3q/2
400	RTNC		12.5/4.5
420	NOP3		
4yz	GOC	zy	12.5/4.5
500	RTNNC		12.5/4.5
5yz	GONC	zy	12.5/4.5
6300	NOP4		14
64000	NOP5		14
6yzt	GOTO	tzy	14
7yzt	GOSUB	tzy	15
800	OUT=CS		5.5
801	OUT=C		7.5
802	A=IN		8.5
803	C=IN		8.5
804	UNCNFG		14.5
805	CONFIG		13.5
806	C=ID		13.5
807	SHUTDN		6.5
8080	INTON		7
80810	RSI		8.5
8082xh0...hx	LAHEX	#hx...h0	5+q+(5+q)/2
8083	BUSCB		10
8084d	ABIT=0	d	7.5
8085d	ABIT=1	d	7.5
8086d	* ?ABIT=0	d	20.5/12.5
8087d	* ?ABIT=1	d	20.5/12.5
8088d	CBIT=0	d	7.5
8089d	CBIT=1	d	7.5
808Ad	* ?CBIT=0	d	20.5/12.5
808Bd	* ?CBIT=1	d	20.5/12.5
808C	PC=(A)		26, 3.5
808D	BUSCD		10
808E	PC=(C)		26, 3.5
808F	INTOFF		7
809	* C+P+1		9.5
80A	RESET		7.5
80B	BUSCC		8.5
80Cx	C=P	x	8
80Dx	P=C	x	8
80E	SREQ?		9.5
80Fx	CPEX	x	8
810	ASLC		22.5
811	BSLC		22.5
812	CSLC		22.5
813	DSLC		22.5
814	ASRC		22.5
815	BSRC		22.5
816	CSRC		22.5
817	DSRC		22.5

Les instructions du SATURN

818f0x	* A=A+x+1	f	8+q
818f1x	* B=B+x+1	f	8+q
818f2x	* C=C+x+1	f	8+q
818f3x	* D=D+x+1	f	8+q
818f8x	* A=A-x-1	f	8+q
818f9x	* B=B-x-1	f	8+q
818fAx	* C=C-x-1	f	8+q
818fBx	* D=D-x-1	f	8+q
819f0	ASRB	f	8.5+q
819f1	BSRB	f	8.5+q
819f2	CSRB	f	8.5+q
819f3	DSRB	f	8.5+q
81Af00	R0=A	f	9+q
81Af01	R1=A	f	9+q
81Af02	R2=A	f	9+q
81Af03	R3=A	f	9+q
81Af04	R4=A	f	9+q
81Af08	R0=C	f	9+q
81Af09	R1=C	f	9+q
81Af0A	R2=C	f	9+q
81Af0B	R3=C	f	9+q
81Af0C	R4=C	f	9+q
81Af10	A=R0	f	9+q
81Af11	A=R1	f	9+q
81Af12	A=R2	f	9+q
81Af13	A=R3	f	9+q
81Af14	A=R4	f	9+q
81Af18	C=R0	f	9+q
81Af19	C=R1	f	9+q
81Af1A	C=R2	f	9+q
81Af1B	C=R3	f	9+q
81Af1C	C=R4	f	9+q
81Af20	AR0EX	f	9+q
81Af21	AR1EX	f	9+q
81Af22	AR2EX	f	9+q
81Af23	AR3EX	f	9+q
81Af24	AR4EX	f	9+q
81Af28	CR0EX	f	9+q
81Af29	CR1EX	f	9+q
81Af2A	CR2EX	f	9+q
81Af2B	CR3EX	f	9+q
81Af2C	CR4EX	f	9+q
81B2	PC=A		19
81B3	PC=C		19
81B4	A=PC		11
81B5	C=PC		11
81B6	APCEX		19
81B7	CPCEX		19
81C	ASRB		21.5
81D	BSRB		21.5
81E	CSRB		21.5
81F	DSRB		21.5
821	XM=0		4.5
822	SB=0		4.5
824	SR=0		4.5
828	MP=0		4.5
82F	CLRST		4.5
831	* ?XM=0		15.5/7.5
832	* ?SB=0		15.5/7.5
834	* ?SR=0		15.5/7.5
838	* ?MP=0		15.5/7.5
84d	ST=0	d	5.5
85d	ST=1	d	5.5
86d	* ?ST=0	d	16.5/8.5
87d	* ?ST=1	d	16.5/8.5
88n	* ?P≠	n	15.5/7.5
89n	* ?P=	n	15.5/7.5
8A0	* ?B=A	A	21.5/13.5
8A1	* ?C=B	A	21.5/13.5

Les instructions du SATURN

8A2	*	?A=C	A	21.5/13.5
8A3	*	?C=D	A	21.5/13.5
8A4	*	?B≠A	A	21.5/13.5
8A5	*	?C≠B	A	21.5/13.5
8A6	*	?A≠C	A	21.5/13.5
8A7	*	?D≠C	A	21.5/13.5
8A8	*	?A=0	A	21.5/13.5
8A9	*	?B=0	A	21.5/13.5
8AA	*	?C=0	A	21.5/13.5
8AB	*	?D=0	A	21.5/13.5
8AC	*	?A≠0	A	21.5/13.5
8AD	*	?B≠0	A	21.5/13.5
8AE	*	?C≠0	A	21.5/13.5
8AF	*	?D≠0	A	21.5/13.5
8B0	*	?A>B	A	21.5/13.5
8B1	*	?B>C	A	21.5/13.5
8B2	*	?C>A	A	21.5/13.5
8B3	*	?D>C	A	21.5/13.5
8B4	*	?A<B	A	21.5/13.5
8B5	*	?B<C	A	21.5/13.5
8B6	*	?C<A	A	21.5/13.5
8B7	*	?D<C	A	21.5/13.5
8B8	*	?A≥B	A	21.5/13.5
8B9	*	?B≥C	A	21.5/13.5
8BA	*	?C≥A	A	21.5/13.5
8BB	*	?D≥C	A	21.5/13.5
8BC	*	?A≤B	A	21.5/13.5
8BD	*	?B≤C	A	21.5/13.5
8BE	*	?C≤A	A	21.5/13.5
8BF	*	?D≤C	A	21.5/13.5
8Cpqr		GOLONG	srqp	17
8Dpqrst		GOVLNG	tsrqp	18.5
8Epqr		GOSUBL	srqp	18
8Fpqrst		GOSBVL	tsrqp	19.5
<hr/>				
9a0	*	?A=B	a	16.5+q/8.5+q
9a1	*	?B=C	a	16.5+q/8.5+q
9a2	*	?C=A	a	16.5+q/8.5+q
9a3	*	?C=D	a	16.5+q/8.5+q
9a4	*	?A≠B	a	16.5+q/8.5+q
9a5	*	?B≠C	a	16.5+q/8.5+q
9a6	*	?C≠A	a	16.5+q/8.5+q
9a7	*	?D≠C	a	16.5+q/8.5+q
9a8	*	?A=0	a	16.5+q/8.5+q
9a9	*	?B=0	a	16.5+q/8.5+q
9aA	*	?C=0	a	16.5+q/8.5+q
9aB	*	?D=0	a	16.5+q/8.5+q
9aC	*	?A≠0	a	16.5+q/8.5+q
9aD	*	?B≠0	a	16.5+q/8.5+q
9aE	*	?C≠0	a	16.5+q/8.5+q
9aF	*	?D≠0	a	16.5+q/8.5+q
9b0	*	?A>B	b	16.5+q/8.5+q
9b1	*	?B>C	b	16.5+q/8.5+q
9b2	*	?C>A	b	16.5+q/8.5+q
9b3	*	?D>C	b	16.5+q/8.5+q
9b4	*	?A<B	b	16.5+q/8.5+q
9b5	*	?B<C	b	16.5+q/8.5+q
9b6	*	?C<A	b	16.5+q/8.5+q
9b7	*	?D<C	b	16.5+q/8.5+q
9b8	*	?A≥B	b	16.5+q/8.5+q
9b9	*	?B≥C	b	16.5+q/8.5+q
9bA	*	?C≥A	b	16.5+q/8.5+q
9bB	*	?D≥C	b	16.5+q/8.5+q
9bC	*	?A≤B	b	16.5+q/8.5+q
9bD	*	?B≤C	b	16.5+q/8.5+q
9bE	*	?C≤A	b	16.5+q/8.5+q
9bF	*	?D≤C	b	16.5+q/8.5+q

Les instructions du SATURN

Aa0	M*	A=A+B	a	4.5+q
Aa1	M*	B=B+C	a	4.5+q
Aa2	M*	C=C+A	a	4.5+q
Aa3	M*	D=D+C	a	4.5+q
Aa4	M*	A=A+A	a	4.5+q
Aa5	M*	B=B+B	a	4.5+q
Aa6	M*	C=C+C	a	4.5+q
Aa7	M*	D=D+D	a	4.5+q
Aa8	M*	B=B+A	a	4.5+q
Aa9	M*	C=C+B	a	4.5+q
AaA	M*	A=A+C	a	4.5+q
AaB	M*	C=C+D	a	4.5+q
AaC	M*	A=A-1	a	4.5+q
AaD	M*	B=B-1	a	4.5+q
AaE	M*	C=C-1	a	4.5+q
AaF	M*	D=D-1	a	4.5+q
Ab0		A=0	b	4.5+q
Ab1		B=0	b	4.5+q
Ab2		C=0	b	4.5+q
Ab3		D=0	b	4.5+q
Ab4		A=B	b	4.5+q
Ab5		B=C	b	4.5+q
Ab6		C=A	b	4.5+q
Ab7		D=C	b	4.5+q
Ab8		B=A	b	4.5+q
Ab9		C=B	b	4.5+q
AbA		A=C	b	4.5+q
AbB		C=D	b	4.5+q
AbC		ABEX	b	4.5+q
AbD		BCEX	b	4.5+q
AbE		ACEX	b	4.5+q
AbF		CDEX	b	4.5+q
Ba0	M*	A=A-B	a	4.5+q
Ba1	M*	B=B-C	a	4.5+q
Ba2	M*	C=C-A	a	4.5+q
Ba3	M*	D=D-C	a	4.5+q
Ba4	M*	A=A+1	a	4.5+q
Ba5	M*	B=B+1	a	4.5+q
Ba6	M*	C=C+1	a	4.5+q
Ba7	M*	D=D+1	a	4.5+q
Ba8	M*	B=B-A	a	4.5+q
Ba9	M*	C=C-B	a	4.5+q
BaA	M*	A=A-C	a	4.5+q
BaB	M*	C=C-D	a	4.5+q
BaC	M*	A=B-A	a	4.5+q
BaD	M*	B=C-B	a	4.5+q
BaE	M*	C=A-C	a	4.5+q
BaF	M*	D=C-D	a	4.5+q
Bb0		ASL	b	4.5+q
Bb1		BSL	b	4.5+q
Bb2		CSL	b	4.5+q
Bb3		DSL	b	4.5+q
Bb4		ASR	b	4.5+q
Bb5		BSR	b	4.5+q
Bb6		CSR	b	4.5+q
Bb7		DSR	b	4.5+q
Bb8	M*	A=-A	b	4.5+q
Bb9	M*	B=-B	b	4.5+q
BbA	M*	C=-C	b	4.5+q
BbB	M*	D=-D	b	4.5+q
BbC	M*	A=-A-1	b	4.5+q
BbD	M*	B=-B-1	b	4.5+q
BbE	M*	C=-C-1	b	4.5+q
BbF	M*	D=-D-1	b	4.5+q

Les instructions du SATURN

C0	M*	A=A+B	A	8
C1	M*	B=B+C	A	8
C2	M*	C=C+A	A	8
C3	M*	D=D+C	A	8
C4	M*	A=A+A	A	8
C5	M*	B=B+B	A	8
C6	M*	C=C+C	A	8
C7	M*	D=D+D	A	8
C8	M*	B=B+A	A	8
C9	M*	C=C+B	A	8
CA	M*	A=A+C	A	8
CB	M*	C=C+D	A	8
CC	M*	A=A-1	A	8
CD	M*	B=B-1	A	8
CE	M*	C=C-1	A	8
CF	M*	D=D-1	A	8
<hr/>				
D0		A=0	A	8
D1		B=0	A	8
D2		C=0	A	8
D3		D=0	A	8
D4		A=B	A	8
D5		B=C	A	8
D6		C=A	A	8
D7		D=C	A	8
D8		B=A	A	8
D9		C=B	A	8
DA		A=C	A	8
DB		C=D	A	8
DC		ABEX	A	8
DD		BCEX	A	8
DE		ACEX	A	8
DF		CDEX	A	8
<hr/>				
E0	M*	A=A-B	A	8
E1	M*	B=B-C	A	8
E2	M*	C=C-A	A	8
E3	M*	D=D-C	A	8
E4	M*	A=A+1	A	8
E5	M*	B=B+1	A	8
E6	M*	C=C+1	A	8
E7	M*	D=D+1	A	8
E8	M*	B=B-A	A	8
E9	M*	C=C-B	A	8
EA	M*	A=A-C	A	8
EB	M*	C=C-D	A	8
EC	M*	A=B-A	A	8
ED	M*	B=C-B	A	8
EE	M*	C=A-C	A	8
EF	M*	D=C-D	A	8
<hr/>				
F0		ASL	A	8
F1		BSL	A	8
F2		CSL	A	8
F3		DSL	A	8
F4		ASR	A	8
F5		BSR	A	8
F6		CSR	A	8
F7		DSR	A	8
F8	M*	A=-A	A	8
F9	M*	B=-B	A	8
FA	M*	C=-C	A	8
FB	M*	D=-D	A	8
FC	M*	A=-A-1	A	8
FD	M*	B=-B-1	A	8
FE	M*	C=-C-1	A	8
FF	M*	D=-D-1	A	8

Touche	
α - ←	α - →
Masque OUT	
Masque IN	
Code buffer	

A a α 002 0010 01	B b β 100 0010 02	C c γ 100 0008 03	D d δ 100 0004 04	E e ϵ 100 0002 05	F f θ 100 0001 06
MTH g γ 004 0010 07	PRG h η 080 0010 08	CST i θ 080 0008 09	VAR j ι 080 0004 0A	↑ k ↑ 080 0002 0B	NXT l λ 080 0001 0C
' ι n ι 001 0010 0D	STD n μ 040 0010 0E	EVRL o ρ 040 0008 0F	+ p + 040 0004 10	↓ q ↓ 040 0002 11	+ r + 040 0001 12
SIN s σ 008 0010 13	COS t τ 020 0010 14	TAN u ζ 020 0008 15	\sqrt{x} v \sim 020 0004 16	y^x u u 020 0002 17	$1/x$ x \bar{x} 020 0001 18
ENTER & ϵ 010 0010 19	+/- y \pm 010 0008 1A	EEK z Υ 010 0004 1B	DEL ! ι 010 0002 1C	← ? ζ 010 0001 1D	
α 008 0020 80	7 ' \cdot 008 0008 1F	8 ^ \sim 008 0004 20	9 - etc 008 0002 21	+ () # 008 0001 22	
← 004 0020 40	4 # ϵ 004 0008 24	5 £ \pounds 004 0004 25	6 ¢ \cdot 004 0002 26	x [] - 004 0001 27	
→ 002 0020 C0	1 == α 002 0008 29	2 < > 002 0004 2A	3 ≤ ≥ 002 0002 2B	- ↔ " '' 002 0001 2C	
ON 8000 2D	0 = \rightarrow 001 0008 2E	. \cdot , \cdot 001 0004 2F	SPC \forall \leftarrow 001 0002 30	+ { } : 001 0001 31	

Routines utiles

Sauver et restaurer les registres			
SAVE_REG	#0679Bh	Sauve les registres RPL	D0 Ca
LOAD_REG	#067D2h	Récupère les reg. RPL sauvés par SAVE_REG	D0 D1 Ba Ca Da
D0TOS	#6384Eh	D0=adresse niveau 1 pile	D0 Aa
D1TOS	#6385Dh	D1=adresse niveau 1 pile	D1 Ca
LREG_EXIT	#05143h	LOAD_REG + retour RPL	
LRT_EXIT	#25CE1h	LOAD_REG + push TRUE + ret. RPL	
LRF_EXIT	#26FAEh	LOAD_REG + push FALSE + retour RPL	
Prendre des éléments sur la pile			
POP_A	#06641h	Prend un SB dans Aa	D1 Aa Ca Da
POP_C A	#03F5Dh	SB dans Aa (1) et Aa (2)	D1 Aa Ca Da
DROP_EXIT	#03249h	Ote l'objet au niveau 1 et retour au RPL	
Mettre des éléments sur la pile			
PSHR0	#06537h	Pose R0a (SB niveau 1)	D0 D1 Aw Ba Ca Da
PSHR0R1	#06529h	Pose R0a (SB au niveau 2) et R1a (SB au niveau 1)	D0 D1 Aw Ba Ca Da R0w
PSHAEXIT	#0357Ch	Pose A et retour au RPL	
PSHR0EXIT	#0357Fh	Pose R0 et retour au RPL	
PSHR01EXIT	#03F14h	Pose R0a (2), R1a (SB 1) et retour RPL	
PSHR0TEXTIT	#036F7h	Pose R0a (SB 2), TRUE (1) et ret. RPL	
PSHR0FEXIT	#2E31Fh	Pose R0a (SB 2), FALSE (1) et ret. RPL	
PSH(A)EXIT	#03A86h	Pose l'objet à l'adresse Aa et retour RPL	
PSHTEXTIT	#620C3h	Pose TRUE et retour RPL	
PSHFEXIT	#620DCh	Pose FALSE et retour RPL	
PSHT/FEXIT	#620D9h	Selon Carry, pose TRUE (Carry=1) ou FALSE (Carry=0) et retour RPL	
Gestion du clavier			
AIN	#0115Ah	A=IN en adresse paire	A4
CIN	#01160h	C=IN en adresse paire	C4
COUTIN	#01EECh	OUT=C C=IN en adr. paire	C4
KSTATE	#008E6h	KEYSTATE courant dans Aw (voir page 10)	D0 P Aw Bw Cw
KSTATESTB	#009A5h	KEYSTATE stable dans Aw (voir page 10)	D0 P Aw Bw Cw
KINBUF?	#04999h	Carry à 1 si touche ds buf.	D1 P As Cas
READK	#04840h	Lit touche du buf. ds Cb	D1 P As Cas
EMPTKBUF	#00D57h	Vide le buffer clavier	D1 Ca
ONKEYSTB	#00C80h	Carry à 1 si ON enfoncé et stable, 0 sinon	D0 P Aw Bw Cw
ONKEY	#00C74h	Idem instantané	C4
EMPTATTN	#00D8Eh	Met à 0 ATTN Counter	D1 Ca
Gestion de l'affichage			
DISPOFF	#01BBDh	Eteint l'afficheur	D0 C0
DISPON	#01B8Fh	Rallume l'afficheur	D0 C0
AOFF	#01BD7h	Eteint les indicateurs	D0 Cb
AON	#01BA9h	Rallume les indicateurs	D0 Cb
ADISPOFF	#01BD3h	DISPOFF + AOFF	D0 Cb
ADISPON	#01BA5h	DISPON + AON	D0 Cb
BUSYON	#01BEBh	Active l'indicateur BUSY	D0 Cb
BUSYOFF	#42359h	Désactive l'indicateur BUSY	D0 Cb
D0PIXGROB	#01C31h	D0=adresse premier pixel du GROB courant	D0 Aa Ca
D0PIXMENU	#01C58h	Idem GROB de menu.	D0 Aa Ca
DISPINGROB	#11D8Fh	Ecrit en police 5x7. D1=adr. début texte, D0=où écrire, Ca=nbr de carac., Ba=marge à gauche (en caractères) Da=largeur GROB en quartets (IP[(largeur en pixels+7)/4])	XM P D0 D1 Aa Ba Cw
CONFGGRAPH	#01C7Fh	Recalcule les ptrs graph. après dépl. ram interne	D0 D1 Aa Ca

Routines utiles

Gestion de la mémoire			
TRDN	#0670Ch	Copie Ca qu. en D0 vers D1 (adresses de débuts) si D1 est inférieur à D0	P D0 D1 Aw Ca
TRUP	#066B9h	Copie Ca qu. en D0 vers D1 (adresses de fins) si D1 est supérieur à D0	P D0 D1 Aw Ca
ZEROM RES_ROOM	#0675Ch #039BEh	Met à zéro Ca qu. en D1 Réserve Ca qu. L'adresse zone dans D0 (GC si nécessaire et erreur si mem. insuffisante)	P D1 Aw Ca bits 0 1 10 de ST D0 D1 P Aw Bw Cw Dw
GARB_COLL	#0613Eh	Nettoie la mémoire de la HP48 (destruction des objets temporaires inutilisés)	bits 0 1 10 de ST D0 D1 P Aw Bw Cw Dw
RES_STR	#05B7Dh	Réserve une STRING de longueur Ca. Adr. dans R0a et adr. contenu dans D0 (GC / erreur)	bits 0 1 10 de ST D0 D1 P Aw Bw Cw Dw R0w
RERESSTR	#16671h	Redimensionne la chaîne à l'adresse R0a pour que D0 soit sa nvelle adr de fin	D0 D1 P Aw Ba Cw R1w
FREEMEM FREEMEMEQ SKIP	#069F7h #06806h #03019h	Recalcule (#807EDh) Nbr. qu. libres → dans Ca Saute l'objet pointé par D0	D0 A10 Cw D0 Aa Ca D0 bit 1 de ST Aa Ca D0 Ca
D0MAPC	#0C1A7h	Adresse de base ram dans le quartet 4 de D0	D0 Ca
D1MAPC	#0C198h	Idem dans D1	D1 Ca
D0MAPA	#0C1C5h	Idem D0	D0 Aa
D1MAPA	#0C1B6h	Idem D1	D1 Aa
D1MAP	#0C169h	Idem D1	qu. 4 de D1
Routines de calcul			
MULTA	#03991h	Ba=Aa * Ca	Aa Ba Ca
MULT	#53EE4h	Aw=Cw=Aw * Cw	Aw Cw Dw
DIV	#65807h	Aw=Cw=Aw / Cw, reste dans Bw	P Aw Bw Cw
DIV5	#06A8Eh	Ca=Ca/5	P A11 C D
ASLW5	#0D5F6h	5 fois ASLw	Aw
ASRW5	#0D5E5h	5 fois ASRw	Aw
CSLW5	#0D618h	5 fois CSLw	Cw
CSRW5	#0D607h	5 fois CSRw	Cw
Divers			
CRCC	#0597Eh	CRC de Aa quartets en Ca, résultat dans Aa	P D0 D1 Aa Ca
CRCD0	#05981h	CRC de Aa quartets en D0, résultat dans Aa	P D0 D1 Aa Ca
CNFTABCRC BEEP	#09B73h #017A6h	CRC table de conf. ds Ca Beep : fréquence dans Da, durée (en millisecondes) dans Ca	D0 D1 C7 D0 D1 P Aw Bw Cw Dw R1w R2w R3w
CHECK_BAT	#006EDh	Vérif. alim. (C0 : 1 piles, 2 très bas, 4 carte 1, 8 carte 2). Carry à 1 si faible	D0 Aa Ba Cx
CHECK_BATI	#325AAh	Piles : carry à 1 si faibles	D0 Aa Ba Ca
DISINTR	#01115h	Inhibe les interruptions	ST15 Ca
ALLINTR	#010E5h	Ré-autorise les int.	bit 15 de ST
EXHR	#026CAh	Exécute la routine en Rom cachée en Ca	P D0 D1 Aa Ca Da
SETFLAG	#53B31h	Met à 1 le flag def. par Cx	P D0 Aa Ca
CLEARFLAG	#53B0Ah	Met à 0 le flag def. par Cx	P D0 Aa Ca
EXIT	#2D564h	Retour au RPL	
ERRORA	#05023h	Affiche l'erreur Aa (faire LOAD_REG)	
ERRORC	#10F80h	Affiche l'erreur Ca (faire LOAD_REG)	
STOP	#10FDBh	Arrête le programme (faire LOAD_REG)	

Vérification des arguments

#1884Dh	Met à zéro le numéro de commande
#18A15h	Aucun argument
#18AB2h	Un argument
#18A8Dh	Deux arguments
#18A68h	Trois arguments
#18B9Fh	Quatre arguments
#18B7Ah	Cinq arguments
#18A1Eh	Aucun argument (objet de librairie)
#18AA5h	Un argument (objet de librairie)
#18A80h	Deux arguments (objet de librairie)
#18A5Bh	Trois arguments (objet de librairie)
#18B92h	Quatre arguments (objet de librairie)
#18B6Dh	Cinq arguments (objet de librairie)
#18C4Ah	n (réel) args. Renvoie le nombre d'arguments (SB)
#18C34h	Idem (objet de librairie)
#18FB2h	Vérifie le type des arguments (voir "Vérif", page 2)

Gestion de la pile

#0314Ch	DEPTH (donne un SB)
#03188h	DUP
#031ACh	DUP2
#031D9h	DUPN (prend un SB comme argument)
#5E370h	Renvoie n copies (SB 1) d'un objet (2)
#03244h	DROP
#03258h	DROP2
#0326Eh	DROPN prenant un SB dans la pile
#03223h	SWAP
#032C2h	OVER
#03295h	ROT
#032E2h	PICK (prend un SB)
#03325h	ROLL (prend un SB)
#0339Eh	ROLLD (prend un SB)

System Binary (SB)

#03DBCCh	+ (SB)
#03DE0h	- (SB)
#03EC2h	* (SB)
#03EF7h	Renvoie le reste (2) et le quotient (1) de 2 SB
#03DEFh	Incrémente un SB
#03E0Eh	Décrémente un SB
#624C6h	MAX (SB)
#624BAh	MIN (SB)
#18DBFh	Convertit un SB en réel
#07E50h	Convertit 2 SB en XLIB Name
#6212Fh	Renvoie TRUE si l'objet est un SB
#03D19h	= (comparaison de SB)
#03D4Eh	≠ (comparaison de SB)
#03CE4h	< (comparaison de SB)
#03D83h	> (comparaison de SB)

Real

#2A974h	+ (Réels)
#2A981h	- (Réels)
#2A9BCh	* (Réels)
#2A9FEh	/ (Réels)
#2AF4Dh	FP (Réel)
#2AF60h	IP (Réel)
#2A70Eh	MIN (Réels)
#2A6F5h	MAX (Réel)
#62D81h	Ordonne deux réels : le plus grand en (2)
#18CEAh	Convertit un réel en SB
#6216Eh	Renvoie TRUE si l'objet est un réel
#2A8C1h	= (comparaison de réels)
#2A8CCh	≠ (comparaison de réels)
#2A871h	<(comparaison de réels)
#2A88Ah	> (comparaison de réels)

Character

#05A51h	Transforme un caractère en SB
#62025h	Renvoie TRUE si l'objet est un caractère

String

#05193h	Concatène deux chaînes
#052EEh	Ajoute le caractère (1) en fin de chaîne
#0525Bh	Ajoute le caractère (1) en début de chaîne
#050EDh	Renvoie le premier caractère d'une chaîne
#0516Ch	Ote le premier caractère de la chaîne
#6326Dh	Renvoie les n (SB 1) derniers carac. de la chaîne (2)
#05636h	Taille d'une chaîne (SB)
#127A7h	Coupe une chaîne au premier RC
#05733h	SUB : les arguments sont une chaîne (3) et deux SB
#645B1h	Chaîne (3), sous-chaîne (2), SB (1) (n° du caractère de début) → position de la sous-chaîne (SB)
#645BDh	Idem mais en partant de la fin
#238A4h	STR→ et TRUE ou la chaîne (4), pos. de l'erreur (SB 3), reste de la chaîne (2) et FALSE
#15B13h	Convertit un objet en chaîne
#11CF3h	Chaîne en GROB (police 5x9)
#11D00h	Chaîne en GROB (police 5x7)
#11F80h	Chaîne en GROB (police 3x7)
#0556Fh	La chaîne est-elle vide ? (→ TRUE ou FALSE)
#62159h	Renvoie TRUE si l'objet est une chaîne

Binary Integer (BI)

#53EA0h	+ (Binary Integers)
#53EB0h	- (Binary Integers)
#53ED3h	* (Binary Integers)
#53F05h	/ (Binary Integers)
#53D04h	AND (Binary Integers)
#53D15h	OR (Binary Integers)
#53D4Eh	NOT (Binary Integer)
#53D26h	XOR (Binary Integers)
#5435Dh	Convertit un BI en réel
#05A03h	Convertit un BI en SB
#62144h	Renvoie TRUE si l'objet est un BI

Les variables globales

#05B15h	Convertit une chaîne en Global Name
#6203Ah	Renvoie TRUE si l'objet est un nom global
#07D27h	STO Variable Globale [et locale]
#08C27h	PURGE (Variable Globale)
#20B81h	RCL et TRUE ou FALSE si elle n'existe pas
#184E1h	CRDIR
#64078h	STO dans le répertoire caché
#64023h	RCL dans le rep. caché (FALSE ou l'objet et TRUE)
#6408Ch	Détruit une variable du répertoire caché
#08696h	STO, nvelle var. même si une var. de ce nom existe

Les variables locales

#6211Ah	Renvoie TRUE si l'objet est un nom local
#074D0h	Prend n objets et une liste de n noms locaux et crée les variables locales correspondantes
#61CE9h	Prend n objets sur la pile, un SB (n) et un nom local et crée un bloc de variables locales nommé
#07497h	Détruit le dernier ensemble de variables locales
#07D1Bh	Comme STO mais pour les variables locales
#07943h	RCL local et et TRUE ou FALSE
#613E7h	Rappelle le contenu de la variable locale N° 1
#6140Eh	Rappelle le contenu de la variable locale N° 2
#61438h	Rappelle le contenu de la variable locale N° 3
#6145Ch	Rappelle le contenu de la variable locale N° 4
#6146Ch	Rappelle le contenu de la variable locale N° 5
#6147Ch	Rappelle le contenu de la variable locale N° 6
#075A5h	Prend un SB n et renvoie le contenu de la var. loc. n
#615F0h	Stocke un objet dans la variable locale N° 1
#61600h	Stocke un objet dans la variable locale N° 2
#61615h	Stocke un objet dans la variable locale N° 3
#61625h	Stocke un objet dans la variable locale N° 4
#61635h	Stocke un objet dans la variable locale N° 5
#61645h	Stocke un objet dans la variable locale N° 6
#075E9h	Objet (2) et SB (1) → stocke l'objet dans la var. loc. n

Objets composites : List Program et Algebraic	
#05089h	Retourne le premier objet d'un composite
#056B6h	Composite (2) et n (SB 1) → GET et TRUE ou FALSE
#05153h	Ote le premier objet d'un composite
#054AFh	Donne les obj. d'un compo, sa taille (SB)
#0567Bh	Taille (SB) d'un composite
#052FAh	Rajoute l'objet (1) à la fin du composite (2)
#62216h	Renvoie TRUE si l'objet est une liste
#621ECh	Renvoie TRUE si c'est un programme
#621D7h	Renvoie TRUE si c'est un algebraic
Graphisme	
#50578h	Renvoie les dim. d'un GROB (2 SB)
#1158Fh	Prend 2 SB (dim.), donne un GROB blanc
#11679h	Prend 2 GROB (3 4) et 2 SB (coord.), place (3) dans (4)
#62201h	Renvoie TRUE si l'objet est un GROB
#12F94h	Stocke le GROB comme page graphique
#3A297h	Stocke le GROB comme menu
#12655h	Rappelle le GROB texte
#12665h	Rappelle le GROB graphique
#12645h	Rappelle le GROB de menu
#1314Dh	Passage en page texte
#13135h	Passage en page graphique
#4E2CFh	Enlève la barre de menu
#4E347h	Affiche la barre de menu
#134AEh	Efface l'écran
#126DFh	Deux n° de lignes (SB) → efface les lignes d'écr. entre
#12429h	Prend un SB n et affiche une chaîne en ligne de texte n
#1384Ah	Allume un pixel de la page texte
#1383Bh	Efface un pixel de la page texte
#13992h	TRUE si le pixel (texte) est allumé
#13825h	Allume un pixel de la page graphique
#1380Fh	Efface un pixel de la page graphique
#13986h	Renvoie TRUE si le pixel est allumé
#4D132h	Scroll d'un pixel vers le haut
#4D16Eh	Scroll d'un pixel vers le bas
#4D150h	Scroll d'un pixel vers la gauche
#4D18Ch	Scroll d'un pixel vers la droite
#51893h	Ordonne quadruplet de coord (pour les ext. suivants)
#1192Fh	Prend un GROB, 4 SB (2 coord.) → le sous-GROB
#50B17h	Prend 4 SB (coord.) → trace une ligne (page texte)
#50B08h	Prend 4 SB (coord.) → efface une ligne (page texte)
#50AF9h	Prend 4 SB (coord.) → trace une ligne (page PICT)
#50AEAh	Prend 4 SB (coord.) → efface une ligne (page PICT)
XLIB Name (XN)	
#07E50h	Deux SB -> XLIB Name correspondant
#07E99h	Rappelle le contenu d'un XN et TRUE ou FALSE
Drapeaux TRUE et FALSE	
#03A81h	Drapeau logique TRUE
#03AC0h	Drapeau logique FALSE
#03B46h	AND (Drapeaux TRUE et FALSE)
#03B75h	OR (Drapeaux TRUE et FALSE)
#03AF2h	NOT (Drapeaux TRUE et FALSE)
#5380Eh	Converti un drapeau en réel (TRUE=1)
Gestion du clavier	
#00D71h	Vide le buffer clavier
#42402h	FALSE si buffer vide, TRUE sinon
#04714h	Renvoie un code de touche et TRUE ou FALSE
#41F65h	Attente touche -> code (touche [SB 2], mode [SB 1])
#05068h	Met le compteur d'appuis sur ON à zéro
#42262h	TRUE si ON a été pressé, ou FALSE
Les structures d'exécution conditionnelle	
#070FDh	IFT sur la pile
#070C3h	IFTE sur la pile
#61993h	Structure CASE
#619BCh	Structure IF... THEN... END
#61AD8h	Structure IF... THEN... ELSE... END
#0712Ah	Structure IF... ELSE

Externals utiles

Les boucles	
#073F7h	Boucle : valeur max (SB 2) et initiale (SB 1)
#073DBh	Ne prend que la valeur max (valeur initiale 1)
#07334h	Équivalent du NEXT RPL
#07221h	Renvoie la valeur courante du compteur
#07270h	Stocke un SB dans le compteur
#07249h	Rappelle la valeur de fin de la boucle
#07295h	Stocke un SB comme borne maximale
#071A2h	Début de boucle WHILE ou UNTIL
#071EEh	Fin du corps de WHILE
#071E5h	Fin de boucle WHILE
#071C8h	Fin de boucle UNTIL
La génération d'erreurs	
#141E5h	Produit un beep d'erreur
#04ED1h	Appelle le gestionnaire d'erreur (erreur courante)
#04CE6h	Rappelle le numéro d'erreur courante
#04D0Eh	Stocke un SB comme erreur courante
#04D87h	Rappelle le message de numéro SB (1)
#04D33h	Met à zéro le numéro d'erreur courante
Commandes diverses	
#06E97h	Met l'objet suivant dans la pile sans évaluation
#05F42h	Déclenche le Garbage Collector
#05F61h	Mémoire libre (sans Garbage Collector)
#05944h	Donne le CRC d'un objet (BI 1) et sa taille en qu. (SB 2)
#141B2h	BEEP (prend deux SB : fréquence, durée en ms)

Jeu de caractères de la HP48

Hexad	décimal		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	Binaire	Déc	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
			0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1
			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	0000	0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
1	0001	16	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	...	
2	0010	32		!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
3	0011	48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
4	0100	64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
5	0101	80	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_	
6	0110	96	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
7	0111	112	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘	
8	1000	128	◀	⌘	▽	∫	∫	Σ	▶	π	ð	≤	≥	≠	α	+	←	↓	
9	1001	144	↑	γ	δ	ε	¶	θ	λ	ρ	σ	τ	υ	◄	∏	Ω	•	∞	
A	1010	160		i	¢	£	¤	¥	¦	§	¨	©	ª	«	-	-	®	¯	
B	1011	176	•	±	²	³	´	µ	¶	•	¸	¹	º	»	¼	½	¾	¿	
C	1100	192	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï	
D	1101	208	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß	
E	1110	224	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï	
F	1111	240	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ	

Police	Adresse	Description
5x10	#7A2B3h	20 quartets par caractère (10 lignes de 8 points). Caractères 31 à 255
5x8	#7B447h	16 quartets par caractère (8 lignes de 8 points). Caractères 31 à 255
menu	#7C257h	11 quartets par caractère (largeur sur 1 quartet, 5 lignes de 8 pixels). Caractères 31 à 96, 123 à 159, 161 à 165, 168, 171, 176, 177, 180, 181, 183, 187, 191, 215, 222, 223, 247 et 248

Ce livret est l'indispensable aide-mémoire du programmeur en assembleur sur HP48 G/GX (adresses utiles, schémas de la mémoire, des objets...).

Vous y retrouverez l'essentiel des informations détaillées dans le livre "Voyage au centre de la HP48 G/GX".

Pour des explications complètes, se référer à cet ouvrage.

Chez le même éditeur

Voyage au centre de la HP48 G/GX, par Paul COURBIS, le livre de référence du langage machine et de l'assembleur sur HP48, ISBN 2-87892-006-6

Initiation à l'assembleur sur HP48 S/SX et G/GX, l'invitation au voyage, par Olivier NACHBA et Cédric LECOURT (à paraître) ISBN 2-87892-008-2

Éditions Angkor
6, Rue Maître Albert
75005 Paris
Tel. : (1)-43-25-45-20

Tous les efforts ont été faits pour que les informations présentées dans ce livret soient aussi exactes et complètes que possible. Ni les auteurs, ni l'éditeur ne pourront en aucun cas être tenus pour responsables des préjudices de quelque nature que ce soit, pouvant résulter de leur utilisation, tant dans un cadre privé, que commercial ou professionnel.

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les "copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective", et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, "toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit, ou ayants cause, est illicite" (alinéa premier de l'article 40). Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles 425 et suivants du Code Pénal.

© Éditions Angkor et les auteurs - Janvier 1994

ISBN 2-87892-007-4

Impression AIRE
23, quai d'Austerlitz - 75013 Paris - FRANCE